HAL open science

Etude et programmation de problèmes relatifs à la présentation d'un groupe

Ishac Betour-Zoghbi

To cite this version:

Ishac Betour-Zoghbi. Etude et programmation de problèmes relatifs à la présentation d'un groupe. Modélisation et simulation. Université Joseph-Fourier - Grenoble I, 1971. Français. NNT : . tel-00282880

HAL Id: tel-00282880

https://theses.hal.science/tel-00282880v1

Submitted on 28 May 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

THESE

présentée à

L'UNIVERSITE SCIENTIFIQUE ET MEDICALE DE GRENOBLE

pour obtenir

LE GRADE DE DOCTEUR DE TROISIEME CYCLE

" INFORMATIQUE "

par

Ishac BETOUR-ZOGHBI

ETUDE ET PROGRAMMATION DE PROBLEMES RELATIFS A LA PRESENTATION D'UN GROUPE

Thèse soutenue le 30 Juin 1971 devant la Commission d'Examen:

Monsieur	J. KUNTZMANN	Président
Messieurs	C. BENZAKEN	Examinateurs
	P. JULLIEN	

UNIVERSITE SCIENTIFIQUE
ET MEDICALE DE GRENOBLE

LISTE DES PROFESSEURS

Président : Monsieur Michel SOUTIF
Vice-Président : Monsieur Gabriel CAU

PROFESSEURS TITULAIRES

MM.	ANGLES D'AURIAC Paul	Mécanique des fluides
	ARNAUD Georges	Clinique des maladies infectieuses
	ARNAUD Paul	Chimie
	AYANT Yves	Physique approfondie
	BARBIER Jean-Claude	Physique expérimentale
	BARBIER Reynold	Géologie appliquée
	BARJON Robert	Physique nucléaire
	BARNOUD Fernand	Biosynthèse de la cellulose
	BARRIE Joseph	Clinique chirurgicale
	BENOIT Jean	Radioélectricité
	BESSON Jean	Electrochimie
	BEZES Henri	Chirurgie générale
	BLAMBERT Maurice	Mathématiques pures
	BOLLIET Louis	Informatique (IUT B)
	BONNET Georges	Electrotechnique
	BONNET Jean-Louis	Clinique ophtalmologique
	BONNET-EYMARD Joseph	Pathologie médicale
	BONNIER Etienne	Electrochimie Electrométallurgie
	BOUCHERLE André	Chimie et Toxicologie
	BOUCHEZ Robert	Physique nucléaire
	BRAVARD Yves	Géographie
	BRISSONNEAU Pierre	Physique du Solide
	BUYLE-BODIN Maurice	Electronique
	CABANAC Jean	Pathologie chirurgicale
	CABANEL Guy	Clinique rhumatologique et hydrologique
	CALAS François	Anatomie
	CARRAZ Gilbert	Biologie animale et pharmacodynamie
	CAU Gabriel	Médecine légale et Toxicologie
	CAUQUIS Georges	Chimie organique
	CHABAUTY Claude	Mathématiques pures
	CHATEAU Robert	Thérapeutique
	CHENE Marcel	Chimie papetière
	COEUR André	Pharmacie chimique
	CONTAMIN Robert	Clinique gynécologique
	COUDERC Pierre	Anatomie Pathologique
	CRAYA Antoine	Mécanique
Mme	DEBELMAS Anne-Marie	Matière médicale
MM.	DEBELMAS Jacques	Géologie générale
	DEGRANGE Charles	Zoologie
	DESSAUX Georges	Physiologie animale
	DODU Jacques	Mécanique appliquée
	DREYFUS Bernard	Thermodynamique
	DUCROS Pierre	Cristallographie
	DUGOIS Pierre	Clinique de Dermatologie et Syphiligraphie

PROFESSEURS ASSOCIES

MM.	BULLEMER Bernhard	Physique
	RADHAKRISHNA Pidatala	Thermodynamique

PROFESSEURS SANS CHAIRE

MM.	AUBERT Guy	Physique
Mme	BARBIER Marie-Jeanne	Electrochimie
MM.	BARRA Jean	Mathématiques appliquées
	BEAUDOING André	Pédiatrie
	BERTRANDIAS Jean-Paul	Mathématiques appliquées
	BIAREZ Jean-Pierre	Mécanique
	BONNETAIN Lucien	Chimie minérale
Mme	BONNIER Jane	Chimie générale
MM.	CARLIER Georges	Biologie végétale
	COHEN Joseph	Electrotechnique
	COUMES André	Radioélectricité
	DEPASSEL Roger	Mécanique des Fluides
	DEPORTES Charles	Chimie minérale
	DESRE Pierre	Métallurgie
	DOLIQUE Jean-Michel	Physique des plasmas
	GAUTHIER Yves	Sciences biologiques
	GEINDRE Michel	Electroradiologie
	GIDON Paul	Géologie et Minéralogie
	GLENAT René	Chimie organique
	HACQUES Gérard	Calcul numérique
	JANIN Bernard	Géographie
Mme	KAHANE Josette	Physique
MM.	LATREILLE René	Chirurgie générale
	LAURENT Pierre	Mathématiques appliquées
	MULLER Jean-Michel	Thérapeutique
	PERRIAUX Jean-Jacques	Géologie et minéralogie
	POULOUJADOFF Michel	Electrotechnique
	REBECQ Jacques	Biologie (CUS)
	REVOL Michel	Urologie
	REYMOND Jean-Charles	Chirurgie générale
	ROBERT André	Chimie papetière
	SARRAZIN Roger	Anatomie et chirurgie
	SARROT-REYNAULD Jean	Géologie
	SIBILLE Robert	Construction Mécanique
	SIROT Louis	Chirurgie générale
Mme	SOUTIF Jeanne	Physique générale
M.	VALENTIN Jacques	Physique nucléaire

MAITRES DE CONFERENCES ET MAITRES DE CONFERENCES AGREGES

Mle	AGNIUS-DELORD Claudine	Physique pharmaceutique
	ALARY Josette	Chimie analytique
MM.	AMBLARD Pierre	Dermatologie
	AMBROISE-THOMAS Pierre	Parasitologie
	ARMAND Yves	Chimie

	FAU René	Clinique neuro-psychiatrique
	FELICI Noël	Electrostatique
	GAGNAIRE Didier	Chimie physique
	GALLISSOT François	Mathématiques pures
	GALVANI Octave	Mathématiques pures
	GASTINEL Noël	Analyse numérique
	GERBER Robert	Mathématiques pures
	GIRAUD Pierre	Géologie
	KLEIN Joseph	Mathématiques pures
Mme	KOFLER Lucie	Botanique et physiologie végétale
MM.	KOSZUL Jean-Louis	Mathématiques pures
	KRAVTCHENKO Julien	Mécanique
	KUNTZMANN Jean	Mathématiques appliquées
	LACAZE Albert	Thermodynamique
	LACHARME Jean	Biologie végétale
	LATURAZE Jean	Biochimie pharmaceutique
	LEDRU Jean	Clinique médicale B
	LLIBOUTRY Louis	Géophysique
	LOUP Jean	Géographie
Mle	LUTZ Elisabeth	Mathématiques pures
MM.	MALGRANGE Bernard	Mathématiques pures
	MALINAS Yves	Clinique obstétricale
	MARTIN-NOEL Pierre	Séméiologie médicale
	MASSEPORT Jean	Géographie
	MAZARE Yves	Clinique médicale A
	MICHEL Robert	Minéralogie et Pétrographie
	MOURIQUAND Claude	Histologie
	MOUSSA André	Chimie nucléaire
	NEEL Louis	Physique du Solide
	OZENDA Paul	Botanique
	PAUTHENET René	Electrotechnique
	PAYAN Jean-Jacques	Mathématiques pures
	PEBAY-PEYROULA Jean-Claude	Physique
	PERRET René	Servomécanismes
	PILLET Emile	Physique industrielle
	RASSAT André	Chimie systématique
	RENARD Michel	Thermodynamique
	REULOS René	Physique industrielle
	RINALDI Renaud	Physique
	ROGET Jean	Clinique de pédiatrie et de puériculture
	SANTON Lucien	Mécanique
	SEIGNEURIN Raymond	Microbiologie et Hygiène
	SENGEL Philippe	Zoologie
	SILBERT Robert	Mécanique des fluides
	SOUTIF Michel	Physique générale
	TANCHE Maurice	Physiologie
	TRAYNARD Philippe	Chimie générale
	VAILLAND François	Zoologie
	VAUQUOIS Bernard	Calcul électronique
Mme	VERAIN Alice	Pharmacie galénique
	VERAIN André	Physique
Mme	VEYRET Germaine	Géographie
MM.	VEYRET Paul	Géographie
	VIGNAIS Pierre	Biochimie médicale
	YOCCOZ Jean	Physique nucléaire théorique

	BEGUIN Claude	Chimie organique
	BELORIZKY Elie	Physique
	BENZAKEN Claude	Mathématiques appliquées
Mme	BERTRANDIAS Françoise	Mathématiques pures
MM.	BLIMAN Samuel	Electronique (EIE)
	BLOCH Daniel	Electrotechnique
Mme	BOUCHE Liane	Mathématiques (CUS)
MM.	BOUCHET Yves	Anatomie
	BOUSSARD Jean-Claude	Mathématiques appliquées
	BOUVARD Maurice	Mécanique des Fluides
	BRIERE Georges	Physique expérimentale
	BRODEAU François	Mathématiques (IUT B)
	BRUGEL Lucien	Energétique
	BUISSON Roger	Physique
	BUTEL Jean	Orthopédie
	CHAMBAZ Edmond	Biochimie médicale
	CHAMPETIER Jean	Anatomie et organogénèse
	CHARACHON Robert	Oto-Rhino-Laryngologie
	CHIAVERINA Jean	Biologie appliquée (EFP)
	CHIBON Pierre	Biologie animale
	COHEN-ADDAD Jean-Pierre	Spectrométrie physique
	COLOMB Maurice	Biochimie médicale
	CONTE René	Physique
	CROUZET Guy	Radiologie
	DURAND Francis	Métallurgie
	DUSSAUD René	Mathématiques (CUS)
Mme	ETERRADOSSI Jacqueline	Physiologie
MM.	FAURE Jacques	Médecine légale
	GAVEND Michel	Pharmacologie
	GENSAC Pierre	Botanique
	GERMAIN Jean-Pierre	Mécanique
	GIDON Maurice	Géologie
	GRIFFITHS Michael	Mathématiques appliquées
	GROULADE Joseph	Biochimie médicale
	HOLLARD Daniel	Hématologie
	HUGONOT Robert	Hygiène et médecine préventive
	IDELMAN Simon	Physiologie animale
	IVANES Marcel	Electricité
	JALBERT Pierre	Histologie
	JOLY Jean-René	Mathématiques pures
	JOUBERT Jean-Claude	Physique du Solide
	JULLIEN Pierre	Mathématiques pures
	KAHANE André	Physique générale
	KUHN Gérard	Physique
Mme	LAJZEROWICZ Jeannine	Physique
MM.	LAJZEROWICZ Joseph	Physique
	LANCIA Roland	Physique atomique
	LE JUNTER Noël	Electronique
	LEROY Philippe	Mathématiques
	LOISEAUX Jean-Marie	Physique nucléaire
	LONGEQUEUE Jean-Pierre	Physique nucléaire
	LUU DUC Cuong	Chimie organique
	MACHE Régis	Physiologie végétale
	MAGNIN Robert	Hygiène et Médecine préventive
	MARECHAL Jean	Mécanique
	MARTIN-BOUYER Michel	Chimie (CUS)
	MAYNARD Roger	Physique du Solide
	MICOUD Max	Maladies infectieuses
	MOREAU René	Hydraulique (INP)

	NEGRE Robert	Mécanique
	PARAMELLE Bernard	Pneumologie
	PECCOUD François	Analyse (IUT B)
	PEFFEN René	Métallurgie
	PELMONT Jean	Physiologie animale
	PERRET Jean	Neurologie
	PERRIN Louis	Pathologie expérimentale
	PFISTER Jean-Claude	Physique du Solide
	PHELIP Xavier	Rhumatologie
Mle	PIERY Yvette	Biologie animale
	RACHAIL Michel	Médecine interne
	RACINET Claude	Gynécologie et obstétrique
	RICHARD Lucien	Botanique
Mme	RINAUDO Marguerite	Chimie macromoléculaire
MM.	ROMIER Guy	Mathématiques (IUT B)
	ROUGEMONT (DE) Jacques	Neuro-chirurgie
	STIEGLITZ Paul	Anesthésiologie
	STOEBNER Pierre	Anatomie pathologique
	VAN CUTSEM Bernard	Mathématiques appliquées
	VEILLON Gérard	Mathématiques appliquées (INP)
	VIALON Pierre	Géologie
	VOOG Robert	Médecine interne
	VROUSSOS Constantin	Radiologie
	ZADWORNY François	Electronique

MAITRES DE CONFERENCES ASSOCIES

MM.	BOUDOURIS Georges	Radioélectricité
	CHEEKE John	Thermodynamique
	GOLDSCHMIDT Hubert	Mathématiques
	YACOUD Mahmoud	Médecine légale

CHARGES DE FONCTIONS DE MAITRES DE CONFERENCES

Mme	BERIEL Hélène	Physiologie
Mme	RENAUDET Jacqueline	Microbiologie

Je tiens :

A remercier Monsieur le Professeur J. KUNTZMANN de m'avoir fait l'honneur de présider le Jury de cette thèse et de l'accueil qu'il m'a réservé lors de mon arrivée au Laboratoire de Calcul de Mathématiques Appliquées de Grenoble.

A exprimer ma profonde reconnaissance à Monsieur C. BENZAKEN pour la façon dont il a bien voulu diriger ce travail, de son aide efficace et de ses conseils fructueux.

A remercier Monsieur P. JULLIEN qui a bien voulu être membre du Jury de cette thèse.

A remercier le C.N.R.S. Libanais de son aide matérielle.

Je remercie les services de dactylographie et de tirage qui ont assuré la réalisation matérielle de cette thèse.

TABLE DES MATIERES

INTRODUCTION I

CHAPITRE I 1

1. Etude de la présentation d'un groupe 1
   1.1. Présentation d'un groupe donné 1
   1.2. Groupe associé à une présentation formelle $< M ; S >$ 2
2. Les problèmes fondamentaux 5
   2.1. Enoncé des trois problèmes fondamentaux 5
   2.2. Rappels 7
   2.3. Groupe libre à n générateurs 8
3. Graphe d'un groupe finiment engendré 11
   3.1. Définition 11
   3.2. Caractérisation des graphes de groupe 12
   3.3. Lien entre graphe et présentation d'un groupe fini 13
4. Présentation d'un sous-groupe H d'index fini d'un groupe G 15
   4.1. Résultat de Reidemeister-Schreier 15
   4.2. A propos de la détermination de l'index fini 18

CHAPITRE II - ETUDE DES GROUPES DONT LA PRESENTATION EST UN PRODUIT LIBRE OU UN PRODUIT LIBRE AMALGAME 22

1. Produit libre 22
   1.1. Présentation qui est un produit libre 22
   1.2. Manipulation explicite d'un produit libre de deux groupes 23
   1.3. Problème du mot dans un produit libre $A * B$ 24
   1.4. Produit libre de plusieurs groupes 26

2. Produit libre avec sous-groupes amalgamés 27
2.1. Produit libre amalgamé 27
2.2. Manipulation explicite du produit libre amalgamé 29
2.3. Aperçu du problème du mot dans un produit libre amalgamé..... 29

CHAPITRE III - GROUPES AVEC UNE SEULE RELATION DEFINIE 33

Lemme 33
Théorème d'indépendance 34
Application du théorème d'indépendance pour résoudre le problème du mot 44
L'extension du problème du mot 45

CHAPITRE IV - ALGORITHMES 46

1. Présentation d'un groupe fini 46
2. Résolution du problème du mot 49
2.1. Représentation utilisée 49
2.2. Présentation de l'algorithme pratique du problème du mot 49
3. Langage de Dyck 52
4. Organigrammes et programmes ..... a

APPENDICE A1

BIBLIOGRAPHIE A4

INTRODUCTION

Les problèmes qui exigent une réponse par "oui" ou "non", "vrai" ou "faux" sont les problèmes fondamentaux de la logique. Ce genre de problèmes s'appelle "problème de décision". Un tel problème peut être résolu par l'introduction de la notion d'algorithme (ou de Machine de Turing).

L'algorithme est une suite de règles (ou d'instructions) calculables qui, par une simple manipulation mécanique de ses symboles sans se soucier de leur signification, résoud certains problèmes. Par exemple : chacun connaît un répertoire d'algorithmes pour les opérations arithmétiques usuelles dont l'exécution est réalisée par des machines simples ; la recherche d'un mot dans un dictionnaire est un autre exemple d'algorithme.

En fait, il est montré que l'existence d'un algorithme pour un "problème de décision" n'est pas toujours assuré, car tout problème de décision se ramène à celui de l'appartenance d'un entier à une partie donnée P de N ; or, une condition nécessaire et suffisante pour que ce dernier problème de décision soit soluble est que P soit un ensemble récursif. Malheureusement, on montre l'existence d'ensembles récursivement énumérables non récursifs [24].

On rencontre de nombreux problèmes de décision dans la théorie des langages. Soit V un alphabet et soit V^* le monoïde libre engendré par V en lui adjoignant la chaîne vide. Soit $x \in V^*$ et soit un langage $L \subset V^*$ défini par un processus formatif (Grammaire, etc...) ; ($x \in L$ ou non ?) est un "problème de décision" qui a pour certaines classes de langage une solution positive et pour d'autres une solution négative.

Nous nous sommes préoccupés dans notre travail d'un problème assez voisin qui est celui de l'équivalence des mots dans la présentation des groupes. Le Mathématicien Norvégien THUE a été le premier qui a posé ce problème fondamental.

Soit un alphabet A et soit un système de relations de THUE $\mathcal{R} = \{R_1 \sim R'_1 , \ldots, R_m \sim R'_m ; R_i , R'_i \in A^* \ \forall i = 1, \ldots, m \}$.

On dit que deux mots U et U' sont <u>contigus</u> lorsqu'on peut écrire $U = XR_iY$ et $U' = XR'_iY$ ou $U = XR'_iY$ et $U' = XR_iY$. La fermeture transitive de cette relation de contiguïté entre mots définit une équivalence dans A^*. Existe-t-il une méthode universelle propre à résoudre le problème de l'équivalence de deux mots appelé encore plus simplement "problème du mot" ?

De nombreux Mathématiciens ont étudié ce problème et la plupart du temps sont arrivés à des conclusions plutôt négatives ([8] , [22], ...). Il nous a semblé pour cela intéressant étant donné qu'on se trouve à la frontière de l'"indécidable" d'étudier et programmer les algorithmes lorsqu'il a été prouvé que le problème du mot a une solution positive.

Notre travail a été donc :

- 1 De rappeler les points de vue généraux relatifs à la présentation des groupes, sous-groupes (groupes libres - graphes de groupes, etc...) C'est l'objet du chapitre I.

- 2 D'exposer l'essentiel de la démonstration du théorème de Magnus, justifiant que le problème du mot est résoluble dans le cas où l'on a une seule relation de Thue non triviale.
C'est l'objet des chapitres II et III.

- 3 De bâtir et programmer des algorithmes.
Nous avons choisi les deux thèmes suivants :

  a) Présenter un groupe fini donné ;
  b) Résoudre automatiquement le problème du mot pour une présentation donnée d'un groupe avec une seule relation.

CHAPITRE I

1. ETUDE DE LA PRESENTATION D'UN GROUPE.

1.1. Présentation d'un groupe donné.

Soit $M = \{a, b, c, \ldots\}$ une partie génératrice d'un groupe G ; il peut y avoir deux générations d'un même élément x de G,

$$\left.\begin{array}{l} x = a_1 a_2 \ldots a_q \\ x = b_1 b_2 \ldots b_r \end{array}\right| \Longrightarrow a_1 a_2 \ldots a_q = b_1 b_2 \ldots b_r \Longleftrightarrow$$

$$a_1 a_2 \ldots a_q \, b_r^{-1} \, b_{r-1}^{-1} \ldots b_2^{-1} \, b_1^{-1} = 1.$$

Le premier membre de la dernière égalité s'appelle une relation satisfaite par le générateur M. C'est un produit dont les facteurs soit appartiennent à M, soit sont tels que leurs opposés appartiennent à M. En particulier

$$a\,a^{-1},\ a^{-1}\,a,\ b^{-1}\,b,\ b\,b^{-1},\ \ldots$$

sont des relations satisfaites dans tout groupe. Elles sont dites pour cela <u>triviales</u>.

En résumé : on pourra définir G à partir de M et de l'ensemble de toutes les relations $\mathcal{R}$ ce qu'on note par

$$G = \langle M \,;\, \mathcal{R} \rangle$$

$\mathcal{R}$ est l'ensemble de toutes les relations satisfaites. $\mathcal{R}$ en général est très redondant ; ainsi si $\alpha\,\beta \in \mathcal{R}$ alors $\beta\,\alpha \in \mathcal{R}$.

Peut-on "axiomatiser" $\mathcal{R}$, c'est-à-dire définir un sous-ensemble S dont $\mathcal{R}$ se "déduirait". Cet ensemble peut-il être "indépendant" ? Dans ce cas, on définira G par

$$\langle M \,;\, S \rangle$$

<u>Définition</u> :

Soit S une partie de $\mathcal{R}$, on dit que S est <u>complet</u> si :

a) S contient les relations triviales

b) Si toute relation W de $\mathcal{R}$, en tant que mot, peut être transformée en un mot vide en un nombre fini d'étapes du type suivant :

1 - Insertion avant, entre ou après les symboles du mot W de relations de S ou de leurs inverses

2 - Suppression de facteurs consécutifs dans W quand ces facteurs consécutifs forment une relation de S ou son inverse.

<u>Exemple</u> :

Le groupe cyclique

$$G = (1, a, a^2, a^3, a^4, a^5, a^6 = 1, a^7 = a, \ldots)$$

$$G = \langle a \; ; \; a^6, a^{12}, a^{18}, \ldots \rangle$$

$$M = \{a\} \text{ et } \mathcal{R} = \{a^6, a^{12}, a^{18}, \ldots\}$$

Dans ce cas $S = \{a^6\}$ est complet entraîne

$$G = \langle a \; ; \; a^6 \rangle.$$

1.2. <u>Groupe associé à une présentation formelle $\langle M \; ; \; S \rangle$</u> .

On va partir d'un ensemble $A = \{a, b, \ldots\}$, fini ou non, qu'on appelle <u>vocabulaire</u>. On prend en correspondances bijectives

$$\bar{A} = \{\bar{a}, \bar{b}, \ldots\}, \; (\bar{A} \cap A = \emptyset).$$

Soit le monoïde libre $M = (A \cup \bar{A})^*$ où la concaténation est la loi de composition interne. On identifie A, $\bar{A}$ à une partie de M. Soit $\mathcal{S}$ une partie de M. Un mot $m \in M$ est une chaîne

$$m = x_1 x_2 \ldots x_n \text{ avec } x_i \in A \cup \bar{A} \quad \forall \, i = 1, 2, \ldots, n.$$

Le mot vide est représenté par 1. On définit le mot inverse $\bar{m}$ d'un mot donné m par récurrence comme il suit :

Si $m = 1$, $\bar{m} = 1$

Si $m = \mu \, a$, où $\mu \in M$ et $a \in A$, $\bar{m} = \bar{a} \, . \, \bar{\mu}$

Si $m = \mu \, \bar{a}$, où $\mu \in M$ et $\bar{a} \in \bar{A}$, $\bar{m} = a \, . \, \bar{\mu}$

On peut facilement vérifier que si on a deux mots m et m' $\in$ M, on a :

$$\overline{(m \cdot m')} = \bar{m}' \cdot \bar{m} .$$

Soit un groupe G et soit φ une application de A dans G,

$$\varphi : A \longrightarrow G.$$

On peut prolonger φ en $\tilde{\varphi} : M \longrightarrow G$ de telle manière que si $m = x_1 x_2 \ldots x_n$

$\tilde{\varphi}(m) = 1$ si $m = 0$ c'est-à-dire si $m = 1$

$\tilde{\varphi}(m) = \tilde{\varphi}(x_1)\, \tilde{\varphi}(x_2) \ldots \tilde{\varphi}(x_n)$ dans le cas contraire c'est-à-dire $m \neq 1$ avec

$$\begin{cases} \tilde{\varphi}(a) = \varphi(a) & \text{si } a \in A \\ \tilde{\varphi}(a) = (\varphi(b))^{-1} & \text{si } a = \bar{b} \in \bar{A} \end{cases}$$

<u>Théorème [[19] pp. 13]</u> :

<u>Soit A un alphabet, $\mathcal{S}$ une partie du monoïde $M = (A \cup \bar{A})^*$. Il existe un groupe G unique, à un isomorphisme près, et une application</u>

$\theta : A \longrightarrow G$ <u>tels que</u> :

$$\begin{tikzcd} A \arrow[rr, "\theta"] \arrow[dr, "\varphi"'] & & G \arrow[dl, "\Psi"] \\ & G' & \end{tikzcd}$$

a) $\theta(A)$ <u>engendre</u> G

b) $\tilde{\theta}(\mu) = 1$ <u>si μ</u> $\in \mathcal{S}$

c) <u>pour tout groupe G' et toute application</u> $\varphi : A \longrightarrow G'$ <u>vérifiant</u>

$$\tilde{\varphi}(\mu) = 1 \quad \underline{\text{si } \mu} \in \mathcal{S},$$

<u>alors il existe un homomorphisme unique</u> $\Psi : G \longrightarrow G'$ <u>tel que</u>

$$\Psi \circ \theta = \varphi.$$

Le groupe G défini par ce théorème se notera

$$G = \langle A ; \mathcal{S} \rangle$$

et $\langle A ; \mathcal{S} \rangle$ est une présentation de G. On dit qu'une présentation $\langle A ; \mathcal{S} \rangle$ est <u>finie</u> si A est fini et $\mathcal{S}$ est fini en tant que sous ensemble de mots.

Nous ne démontrons pas ce théorème. Nous donnerons le moyen de définir G. On va rajouter à $\mathcal{S}$ les mots triviaux $a\,\bar{a}$, $\bar{a}\,a$ $\forall\, a \in A$ s'ils n'existent pas déjà.

Définition :

Soient deux mots μ et ν de M. μ et ν sont dits équivalents si l'on obtient ν à partir de μ en un nombre fini d'étapes du type suivant :

α) Insertion de mots de $\mathcal{S}$ ou de leurs inverses dans μ.

β) Suppression de lettres consécutives dans μ quand le mot qu'elles forment ou son image appartient à $\mathcal{S}$.

Remarque :

Il est facile de vérifier que c'est une relation d'équivalence et on la note $\mu \sim \nu$ (et quelquefois $\mu = \nu$). C'est même une congruence dans M. Notons que si $\mu \in \mathcal{S}$ alors $\mu \sim 1$. Le groupe G est alors défini comme le quotient de M par cette congruence.

Corollaire [[19] pp. 15] :

Supposons que A étant donné, $\mathcal{S}$ et $\mathcal{S}'$ sont deux parties du monoïde $M = (A \cup \bar{A})^*$ telles que $\mathcal{S} \subseteq \mathcal{S}'$; alors le groupe $< A \; ; \mathcal{S}' >$ est une image homomorphe du groupe $< A \; ; \mathcal{S} >$.

Théorème [19] :

Soit $G = < A \; ; \mathcal{R} >$ et soit $\mathcal{S}$ un ensemble de mots définissant un ensemble d'éléments de G et engendrant le sous-groupe normal N de G, alors le groupe quotient $G/_N$ a la présentation :

$$G/_N = < A \; ; \mathcal{R} \cup \mathcal{S} > .$$

D'après ce théorème, on peut considérer un groupe $G = < A \; ; \mathcal{R} >$ comme le quotient du groupe $G' = < A \; ; \emptyset >$ par le sous-groupe normal N de G' engendré par $[\mathcal{R}]_{G'}$.

$$G = < A \; ; \mathcal{R} > = < A \; ; \emptyset > /_N .$$

Chaque mot $W \in M$ est une <u>relation</u> si et seulement si $W \underset{G}{\sim} 1$ ce qui est équivalent à dire $[W]_{G'} \in N$. On peut en déduire que chaque relation W est <u>librement équivalente</u> (voir 2.3.) au produit

$$T_1^{-1} R_{i_1}^{\varepsilon_{i_1}} T_1 \cdot T_2^{-1} R_{i_2}^{\varepsilon_{i_2}} T_2 \cdot \ldots \cdot T_n^{-1} R_{i_n}^{\varepsilon_{i_n}} T_n$$

$(\varepsilon_{i_j} = \pm 1 \quad \forall\, j = 1, 2, \ldots, n)$;

où les T_j sont des mots quelconques par rapport aux a_i ($T_j \in M$) et $i_1, i_2, \ldots, i_n \in \{1, 2, \ldots, m\}$ ensemble d'indices de l'ensemble de relations définies $\mathcal{R}$.

2. LES PROBLEMES FONDAMENTAUX.

2.1. Enoncé des trois problèmes fondamentaux.

Dans un groupe présenté par un ensemble de générateurs A et un sous-ensemble $\mathcal{S} \subseteq M = (A \cup \bar{A})^*$ on pose trois problèmes fondamentaux qui sont les suivants :

<u>Problème I</u> :

C'est le <u>problème du mot</u>. Etant donné deux mots μ et $\nu \in M$, étudier si ces deux mots sont équivalents ou non (au sens de la relation $\sim$). Pour ceci, tout revient à étudier si $\mu\,\nu^{-1} \sim 1$. Le problème du mot revient donc à étudier si un mot donné $\gamma \in M$ est équivalent ou non à 1. C'est le problème fondamental qui permet de "manipuler" le groupe G comme quotient de M par $\sim$.

<u>Problème II</u> :

C'est le <u>problème de conjugaison</u>.
μ et ν étant deux mots de M définissant par leur classe $[\mu]$ et $[\nu] \in G$.
Etudier si $[\mu]$ et $[\nu]$ sont <u>conjugués</u> ou non. (on rappelle que dans un groupe G, a, b $\in$ G sont dits conjugués s'il existe un $c \in G$ tel que $b = c^{-1}\, a\, c$).

<u>Problème III</u> :

C'est le problème <u>d'isomorphisme entre deux présentations</u>. Soient deux présentations $G = \langle A \,;\, \mathcal{S} \rangle$ et $G' = \langle A' \,;\, \mathcal{S}' \rangle$. G et G' sont-ils isomorphes ?

Il a été démontré [8] et [22] que le problème I est indécidable même si on se limite aux présentations finies. Par ailleurs, il est clair que le problème II [13] généralise le problème I. Concernant le problème III il est probable qu'en général le problème est indécidable. Nous citerons toujours un théorème (ne donnant toutefois pas d'algorithme "effectif") donnant une C.N.S. pour que deux présentations <u>finies</u> définissent un même groupe. On introduit les transformations de Tietze qui sont les suivantes :

1) T1 . Si μ est dérivable de $\mathcal{R}$, on le rajoute à $\mathcal{R}$.

1)* T1* . Si dans $\mathcal{R}$ μ est dérivable des autres, on l'enlève de $\mathcal{R}$.

2) T2 . Si μ est un mot, auquel on attache une lettre $x \notin A$, on rajoute x à A et la relation $x = \mu$ à $\mathcal{R}$.

2)* T2* . Si une relation de $\mathcal{R}$ est résoluble par rapport à une lettre x, de la forme $x = \mu$ où x ne figure plus dans μ, on supprime x de A et $x = \mu$ de $\mathcal{R}$ et on remplace dans les autres relations de $\mathcal{R}$ x par μ et $\bar{x}$ par $\bar{\mu}$.

Ainsi à partir de ces 4 T on peut énoncer le théorème suivant :

<u>Théorème [19]</u> :

<u>Une condition nécessaire et suffisante pour que deux présentations finies définissent un même groupe, à un isomorphisme près, est que l'on passe de l'une à l'autre par un nombre fini de transformations T1, T1*, T2, T2*.</u>

<u>Exemple</u> :

Le groupe $< a, b, c ; (ab)^2 ab^2 >$ est un groupe isomorphe au groupe $< c, y ; \emptyset >$ car :

$$< a, b, c ; (ab)^2 ab^2 > \xrightarrow{i \quad T2} < a, b, c, y ; y = ab, (ab)^2 ab^2 >$$

$$= < a, b, c, y ; y = ab, y^3 b > \xrightarrow{i \quad T2^*} < a, c, y ; y = ay^{-3} >$$

$$\xrightarrow{i \quad T2^*} < c, y ; \emptyset > .$$

Remarque :

$\mathcal{R}$ est un ensemble non vide contenant des relations non résolubles. Si quelques unes de ces relations peuvent être résolues (au sens de $T2^*$) on obtient le groupe

$$G' = \langle A' \; ; \; \mathcal{R}' \rangle \xrightarrow{i} \langle A \; ; \; R \rangle$$

et tout revient à résoudre le problème du mot dans G'.

2.2. Rappels.

Nombreux sont ceux qui ont travaillé sur le problème du mot dans un groupe et surtout après avoir été formulé et décrit par DEHN en 1912 pour les divers groupes fondamentaux de dimension deux ; ceci en se basant sur le diagramme de Cayley associé à la présentation d'un groupe G.

Greendlinger [12] a décrit une méthode plus algébrique :

Soit le groupe $G = \langle A \; ; \; \mathcal{R} \rangle$. On peut supposer que $\mathcal{R}$ est un ensemble symmétrisé, c'est-à-dire à chaque élément $R \in \mathcal{R}$ on peut trouver R^{-1} et toutes les permutations circulaires de R et R^{-1}, sinon on peut ajouter ceux qui manquent et ceci ne change rien à la généralité du problème.

Le résultat principal de Greendlinger est le suivant :

Soit G un groupe présenté par $G = \langle A \; ; \; \mathcal{R} \rangle$ où $\mathcal{R}$ est un ensemble symmétrisé. Supposons que la fraction λ, d'un R_i quelconque de $\mathcal{R}$, qui peut réagir librement avec un R_j quelconque de $\mathcal{R}$ est $< 1/_6$, à moins que R_i et R_j sont inverses l'un de l'autre. Soit W un mot librement réduit tel que W = 1 dans G ; dans ce cas W contiendra plus que la moitié d'une des relations R_i.

Conséquence :

D'après cette propriété, on peut résoudre le problème du mot dans un cas particulier qui est difficile à être réalisé, et ceci en réduisant chaque fois la longueur du mot W comme il suit :

Si W contient une séquence α des a_i qui figure dans un $R_i = \alpha\,\beta$ tel que $l(\alpha) > \frac{1}{2}\; l(R_i)$ c'est-à-dire $l(\alpha) > l(\beta)$, (longueur signifie nombre de générateurs qui figure dans une séquence). On remplace α dans W par β^{-1} puis on réduit librement et on obtient le mot W' ($l(W') < l(W)$ puis on recommence de la même manière avec le mot W'.

Si on arrive à un mot W" tel que W" est non vide et ne contient pas plus que la moitié d'une des $R_i \in \mathcal{R}$; on peut conclure directement que le mot W" $\not\sim$ 1 ainsi W $\not\sim$ 1 car W $\sim$ W' $\sim$ W" $\sim$ ... Par contre, si on arrive à un certain $W^{(n)}$ vide, on peut conclure que le mot W $\sim$ 1.

La théorie combinatoire de Greendlinger [12],[13] a été précédée par une profonde étude faite par Tartakovskû en 1949 qui a résolu le problème du mot pour des diverses classes de groupes. Britton [3], ..., [6] est arrivé aux mêmes résultats que Tartakovskû par une autre méthode.

2.3. Groupe libre à n générateurs.

Soit $A = \{a_1, a_2, \ldots, a_n\}$; le groupe libre engendré par A est le groupe

$$G = \langle A \; ; \; \emptyset \rangle.$$

Définition :

Un mot $\mu \in M$ est dit réduit (ou librement réduit) si deux symboles consécutifs ne sont jamais inverses l'un de l'autre. Le même mot est dit cycliquement réduit s'il est réduit et si la première lettre de μ (notée $[\mu)$ et la dernière lettre (notée $\mu]$) sont tels que $\mu] \neq [\mu$. Pour trouver la réduction libre d'un mot $\mu \in M$, on définit récursivement une fonction

$$\rho : M \longrightarrow M$$

de la façon suivante :

$$\rho(1) = 1$$

$$\rho(\mu.x) = \begin{cases} \rho(\mu).x \;, \text{ (avec } \mu \in M \text{ et } x \in A \cup \bar{A}), \text{ si } \rho(\mu)] \neq \bar{x} \\ \nu \text{ si } \rho(\mu) = \nu\,\bar{x} \text{ où } \nu \in M. \end{cases}$$

La fonction ρ a les propriétés suivantes démontrées dans [19] :

1) $\rho(\mu)$ est un mot réduit

2) $\rho(\mu) \approx \mu$ ($\approx$ signifie réduction libre)

3) si μ est réduit, alors $\rho(\mu) = \mu$

4) $\rho(\mu.\nu) = \rho(\rho(\mu).\nu)$

5) $\rho(\mu \times \bar{x}) = \rho(\mu)$

6) $\rho(\mu \times \bar{x}\ \nu) = \rho(\mu\ \nu)$

<u>Il en résulte que</u>

$$\underline{\mu \approx \nu \Leftrightarrow \rho(\mu) = \rho(\nu)}$$

<u>En effet</u> :

* Soient deux mots μ et $\nu \in M$ tels que $\mu \approx \nu$.
$\mathcal{R} = \emptyset$ entraîne $\mu = \mu_0 \to \mu_1 \to \mu_2 \to \dots \mu_m = \nu$ en un nombre <u>fini</u> d'étapes en insérant ou en supprimant chaque fois une relation triviale ($a_i\ \overline{a_i}$ ou $\overline{a_i}\ a_i$ avec $a_i \in A$). D'après la 6ème propriété de ρ on peut écrire :

$$\rho(\mu) = \rho(\mu_o) = \rho(\mu_1) = \dots = \rho(\mu_m) = \rho(\nu) \Longrightarrow \rho(\mu) = \rho(\nu).$$

Donc $\mu \approx \nu \Longrightarrow \rho(\mu) = \rho(\nu)$.

* <u>Réciproquement</u> : Soient μ et ν deux mots de M tels que $\rho(\mu) = \rho(\nu)$.

D'après la 2ème propriété de ρ

$$\rho(\mu) \approx \mu$$

et $\rho(\nu) \approx \nu$

plus l'hypothèse $\rho(\mu) = \rho(\nu)$ on aura : $\mu \approx \nu$.

ρ étant une fonction récursive $\Longrightarrow$ le problème du mot est résoluble dans le cas où $\mathcal{R}$ est vide. L'ensemble des mots réduits constitue un système de représentants de G.

<u>Exemple</u> : $G = \langle a, b, c, d, e\ ;\ \emptyset \rangle$ est un groupe libre à cinq générateurs.

Le problème du mot dans le groupe libre est équivalent à l'analyse syntaxique d'un langage C.F.

* Soit S l'axiome
* La grammaire est définie par $V_T = A \cup \bar{A}$, $V_N = S$ et les règles de production

$$\left.\begin{array}{l} S \longrightarrow SS \\ S \longrightarrow a_i\ S\ \bar{a}_i \\ S \longrightarrow \bar{a}_i\ S\ a_i \\ S \longrightarrow a_i\ \bar{a}_i \\ S \longrightarrow \bar{a}_i\ a_i \end{array}\right| \qquad (1 \leqslant i \leqslant n)$$

Cette grammaire définit l'ensemble des mots équivalents à un dans le groupe libre. On peut en déduire que cet ensemble de mots forme un langage C.F. (appelé langage de Dyck à n générateurs). On peut trouver une grammaire C.F. non ambiguë d'après [[11] pp. 145].

De la même manière, pour trouver le mot réduit cycliquement, on peut définir une fonction

$$\sigma : \{ \text{des mots réduits } \{\longrightarrow\} \text{ des mots réduits } \}$$

d'une manière récursive, de la façon suivante :

$$\sigma(1) = 1$$

$$\sigma(x) = x \quad \text{où } x \in A \cup \bar{A}$$

$$\sigma(x\ \mu\ y) = \begin{cases} x\ \mu\ y & \text{si } x \neq \bar{y} \\ \sigma(\mu) & \text{si } x = \bar{y} \end{cases}$$

Aussi la fonction σ a les propriétés suivantes démontrées dans [19] :

1) $\sigma(\mu)$ est un mot cycliquement réduit
2) $\sigma(\mu)$ est conjugué de μ
3) si μ est cycliquement réduit, $\sigma(\mu) = \mu$.

On peut remarquer que c'est une propriété de fermeture. Dans le groupe libre, le problème II peut être résolu par le théorème suivant :

Théorème [19] :

Deux mots μ et ν (supposés réduits) sont conjugués dans le groupe libre si et seulement si $\sigma(\mu)$ et $\sigma(\nu)$ sont dans la même classe de permutation circulaire.

Le problème III est trivial dans le cas particulier suivant :

$< A \; ; \; \emptyset >$ est isomorphe à $< B \; ; \; \emptyset >$ si et seulement si

$$\text{card } A = \text{card } B$$

N.B. Quelquefois on note $\mathcal{S}$ par $\mathcal{R}$.

3. GRAPHE D'UN GROUPE FINIMENT ENGENDRE.

3.1. Définition.

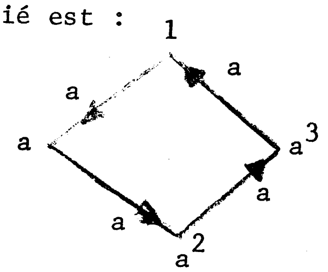
A un groupe G finiment engendré de générateurs $a_1, a_2, \ldots, a_n$ on associe un graphe orienté dont les sommets sont les éléments de G ; l'on joint un sommet s à un sommet s' par une arête d'étiquette a_i si $s' = s\, a_i$.

Remarque :

Si l'on change de système de générateurs du même groupe G, le graphe associé change en général.

Exemples :

1) Soit le groupe $G = < a \; ; \; a^4 > = \{1, a, a^2, a^3\}$ groupe cyclique d'ordre 4. $A = \{a\}$ et le graphe associé est :



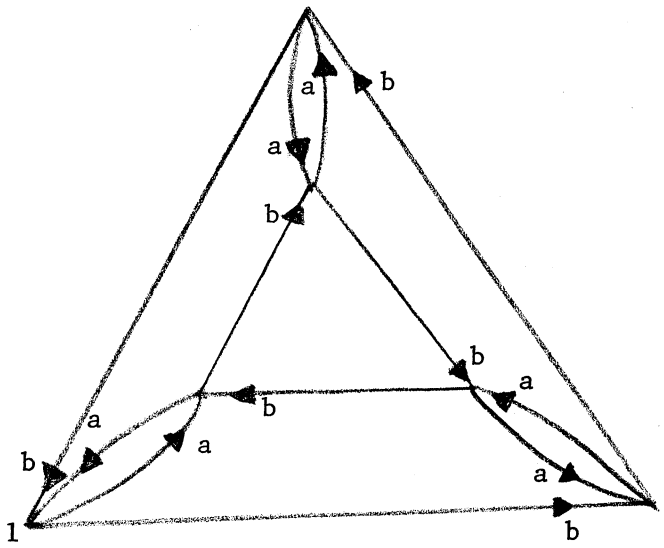
où chaque arête est étiquetée a.

2) Voici un exemple qui est plus complexe.

Le groupe symétrique de $\{1, 2, 3\}$ a la présentation

$$G = \langle a, b \; ; \; a^2, b^3, (ab)^2 \rangle$$

Le graphe du groupe sera représenté comme suit :



Le graphe obtenu n'est pas quelconque ; il est caractérisé par le théorème suivant :

3.2. Caractérisation des graphes de groupe.

Théorème [19] :

La condition nécessaire et suffisante pour qu'un graphe (S, A) puisse être le graphe associé à un groupe engendré par n éléments, il faut que :

1) Ce graphe soit connexe homogène de degré n (en degré intérieur comme extérieur)

2) Qu'il existe une application

$$\varphi : A \longrightarrow L,$$

où L est un ensemble à n éléments, telle que :

a) pour chaque sommet s de S la restriction de φ sur les arêtes d'origine s soit surjective ainsi que la restriction sur les arêtes d'extrémité s.

b) désignant par $\tilde{L}$ l'ensemble des éléments de L muni d'un signe (L^+, L^-), par m l'application de $B \overset{\text{déf}}{=} (A \cup \bar{A})^*$ dans $\tilde{L}$

$$m : B \longrightarrow \tilde{L}$$

définie par :

- $m(a) = \varphi(a)^+$ si $a \in A$ et $m(\bar{a}) = \varphi(a)^-$ si $\bar{a} \in \bar{A}$
  que l'on étend aux chaînes
- $m(b_1 b_2 \ldots b_n) = m(b_1)\, m(b_2) \ldots m(b_n) \in \tilde{L}^*$.

Si π et $\pi' \in B$ vérifient $m(\pi) = m(\pi')$ alors π est un cycle $\Leftrightarrow$ π' est un cycle.

3.3. Lien entre graphe et présentation d'un groupe fini.

Soit un groupe $G = \langle A \,;\, \mathcal{R} \rangle$. Si, dans le graphe du groupe G, s_o est le sommet correspondant à l'élément identité 1 de G, alors tout mot W est uniquement représenté par un chemin issu de s_o. A un chemin correspond un mot et inversement à un mot correspond un chemin. Les relations satisfaites par la $\langle A \,;\, \mathcal{R} \rangle$ définissent des chemins qui sont des cycles du graphe et inversement à un chemin fermé du graphe associé correspond une relation satisfaite.

Ainsi le problème de la construction du graphe d'une présentation finie $G = \langle A \,;\, \mathcal{R} \rangle$ est évidemment équivalent au problème du mot pour la présentation.

Théorème :

Soit G un groupe fini donné et soit $A = \{a_1, a_2, \ldots, a_n\}$ un système générateur de G. Soient Γ le graphe étiqueté associé à G, A et Γ^o le graphe symétrique qu'il engendre. Soit C l'espace vectoriel des cycles de Γ^o.

1) <u>A toute base des cycles de Γ^o on peut définir un système complet de relations</u> [voir [19] pp. 56-67].

2) <u>Il existe au moins une base de cycles telle que le système soit indépendant</u>.

On va démontrer 2) :

Soit G un groupe fini engendré par A ayant une présentation $< A ; R_1, R_2, \ldots, R_q >$ où le système $R_1, \ldots, R_q$ est irrédondant. Prenons tous les cycles du graphe dont le mot associé est (à une permutation circulaire près) $R_1, R_2, \ldots, R_q$.

<u>Ces cycles engendrent l'espace des cycles</u>. En effet, soit C un cycle quelconque non trivial c'est-à-dire il n'y a pas une arête qui soit parcourue deux fois, et soit R la relation associée.

- ou bien R est l'un des R_i, rien à démontrer
- ou bien R est différent de tous les R_i, $i = 1, 2, \ldots, q$.

Dans ce cas, R est déduit des R_i et on a :

$$R = T_1 R_{i_1}^{\varepsilon_{i_1}} T_1^{-1} . T_2 R_{i_2}^{\varepsilon_{i_2}} T_2^{-1} . \ldots . T_p R_{i_p}^{\varepsilon_{i_p}} T_p^{-1} \text{ où } \varepsilon_{i_j} = \pm 1.$$

Tous les T_j sont vides, $j = 1, \ldots, p$, car on a exclu les cycles triviaux. Ainsi,

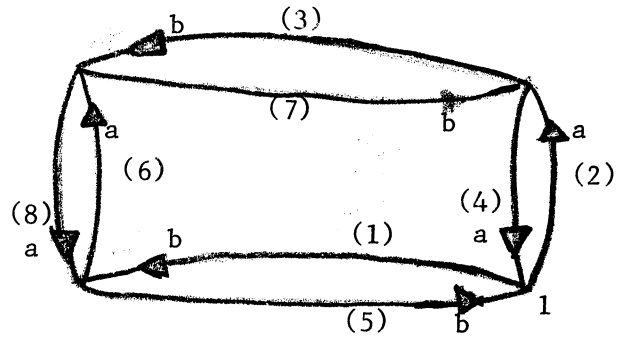
$$R = R_{i_1}^{\varepsilon_{i_1}} . R_{i_2}^{\varepsilon_{i_2}} . \ldots . R_{i_p}^{\varepsilon_{i_p}}$$

donc $C = C_{i_1} \oplus C_{i_2} \oplus \ldots \oplus C_{i_q}$.

où les C_{i_j} sont les cycles associés.

Exemple :

Soit le groupe fini $G = < a, b ; a^2, b^2, (ab)^2 >$. $a^2, b^2, (ab)^2$ est un système irrédondant. Le graphe associé est :



Les cycles du graphe, dont le mot associé est $a^2, b^2, (ab)^2$ à une permutation circulaire près, sont :

(2).(4) , (6).(8) , (3).(7) , (1).(5) , (2).(3).(8).(5) et (1).(6).(7).(4).
Ces cycles forment une base de cycles telle que le système associé soit indépendant.

4. PRESENTATION D'UN SOUS-GROUPE H D'INDEX FINI D'UN GROUPE G.

4.1. Résultat de Reidemeister-Schreier.

Soit un groupe de présentation finie

$$G = < a_1, a_2, \ldots, a_n ; R_1, R_2, \ldots, R_q, \ldots, R_m >$$

et soit H un sous-groupe de G d'index fini et engendré par $\mu_1, \mu_2, \ldots, \mu_k$.
La question qui se pose est la suivante :

Peut-on avoir une présentation de H ?

A chaque μ_i on définit une lettre s_i. Soient

$$S = \{s_1, s_2, \text{———}, s_k\} \text{ et } \mathcal{G} = (A \cup \bar{A})^* \text{ et soit } \mathcal{B} \subseteq \mathcal{G}$$

l'ensemble des mots dont l'élément associé dans G appartient à H et

$\mathcal{V} = (S \cup \bar{S})^*$. Le sous-groupe H sera présenté par

$$H = < s_1, s_2, \ldots, s_k ; \sigma_1, \sigma_2, \ldots >$$

Le problème est de trouver les σ_i .

<u>Définition</u> :

On appelle codage de H par S une application

$$\tau : \mathcal{B} \longrightarrow \mathcal{V}$$

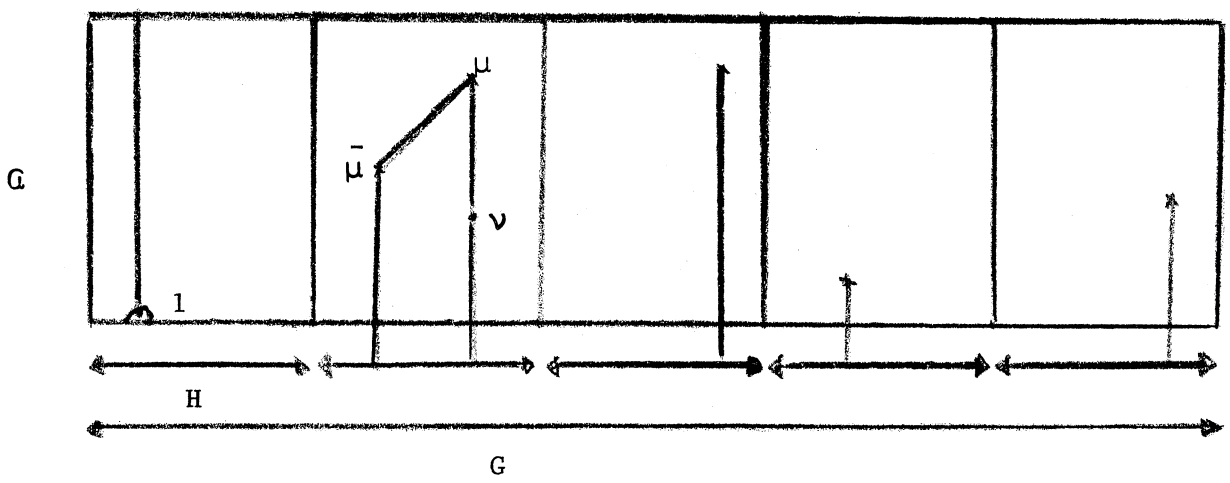
$$\text{où } \nu \in \mathcal{B} \longrightarrow \tau\, \nu\, (s_1, \ldots, s_k)$$

telle que $\tau\, \nu\, (\mu_1, \ldots, \mu_k) \approx \nu$.

Supposons que H soit d'index fini dans G, ce qui veut dire que l'ensemble $\{H\, x\ ;\ x \in G\}$ des classes à droite est fini. Dans chaque classe à droite, on définit un représentant par un mot de $\mathcal{G}$ (ce mot étant le mot vide 1 pour la classe H). Soit T l'ensemble des mots ainsi choisis. On définit ainsi une fonction représentative de $\mathcal{G}$ dans T

$$\mathcal{G} \longrightarrow T$$

$$\mu \longrightarrow \bar{\mu} \quad \text{avec } H\, \bar{\mu} = H\, \mu.$$



Le grand rectangle représente $\mathcal{G}$ et chacun des petits rectangles représente les mots associés d'une classe à droite H x. Le grand segment représente G et un petit segment représente un H x. Les mots situés sur une même hauteur sont équivalents.

<u>Théorème [[19] pp. 91]</u> :

<u>Le sous-groupe H est engendré par les mots</u>

$\xi\, a_i\, .\, ((\overline{\xi\, a_i}))^{-1}$ où $\xi \in T$ et les $a_i \in A$.

Avec ce théorème, on peut dire qu'il existe un système de codage τ. $\xi\, a_i$ est un mot μ. $(\overline{\xi\, a_i}) = \bar{\mu} \in T$. μ et $\bar{\mu}$ définissent la même classe à droite $\Longrightarrow \mu\, \bar{\mu}^{-1} \in H$. On représente

$$\mu_i = \xi\, a_i \,.\, ((\overline{\xi\, a_i}))^{-1} \text{ par } s_{\xi, a_i} \quad \text{et } \nu_i = \xi\, a_i^{-1} .\, ((\overline{\xi\, a_i^{-1}}))^{-1}$$

par $s_{\overline{\xi\, a_i^{-1}}^{-1},\, a_i}$ car $\mu\, a(\overline{\mu\, a})^{-1}$, avec $\mu = (\overline{\xi\, a^{-1}})$, a pour inverse

$$\xi\, a^{-1}\, (\overline{\xi\, a^{-1}})^{-1}$$

D'où le codage $\tau : \mathcal{B} \longrightarrow \mathcal{V}$

Soit $u = G_{i_1}^{\varepsilon_1}\, G_{i_2}^{\varepsilon_2} \ldots G_{i_r}^{\varepsilon_r} \in \mathcal{B}$; $\tau(u) = s_{k_1, a_{i_1}}^{\varepsilon_1}\, s_{k_2, a_{i_2}}^{\varepsilon_2} \ldots s_{k_r, a_{i_r}}^{\varepsilon_r}$

quand k_j est le représentant du segment initial de u précédant a_{i_j} si $\varepsilon_j = +1$ et k_j est le représentant du segment initial de u et $a_{i_j}^{-1}$ inclu si $\varepsilon_j = -1$. Alors τ est un procédé de réécriture pour H.

d'où $H = \langle s_{\xi, a} \;/\; \xi \in T,\ a \in A\ ;\ \Sigma \rangle$

$$\text{avec } \Sigma = \begin{cases} (1) : s_{\xi, a} \approx \tau\, [\xi\, a\, (\overline{\xi\, a})^{-1}] \\ (2) : \tau(\xi\, R\, \xi^{-1}) \approx 1 \text{ seulement pour } \xi \in T \text{ et } R \in \mathcal{R}. \end{cases}$$

C'est le théorème de Reidemeister.

Le système représentatif T est dit système de Schreier, si toute section commençante d'un $\xi \in T$ est élément de T ; dans ce cas, on peut remplacer (1) dans Σ (ci-dessus) par

$$(1)' : s_{\xi, a}$$

quand ξ est un élément représentatif de Schreier arbitraire et $a \in A$; ce qui revient à prendre moins de générateur.

Exemple :

Soit le groupe $G = \langle x, y \; ; \; y\,x\,y\,x^{-1} \rangle$ et soit N le sous-groupe normal de G engendré par y. Comme un système T représentant les classes à droite de G (modulo N), on choisit les x^k, où k est un entier quelconque. Un mot W(x,y) a pour représentant $\bar{W} = x^k$ où k est la somme des puissances de x dans W. Ainsi N est engendré par les y_k définis par

$$y_k = x^k\, y\, x^{-k} = x^k . y . \overline{x^k y}^{-1} = s_{x^k,\, y}.$$

Soit un mot $W(x,y) = x^2 y x^3 y^{-1} x y x^{-1} y^4$ alors $\bar{W} = x^5$.

En outre, soit $u = x y x^2 y^{-1} x^{-2} y^2 x^{-1} y$ un mot de $\mathcal{B}$ (le $\mathcal{B}$ relatif à N) alors $\tau(u) = s_{1,x} . s_{x,y} . s_{x,x} . s_{x^2,x} . s_{x^3,y^{-1}} . s_{x,y} . s_{x,y} . s_{1,x} . s_{1,y}$

en appliquant la définition de y_k, on trouve

$$\tau(u) = y_1 y_3^{-1} y_1^2 y_o.$$

4.2. A propos de la détermination de l'index fini.

Todd et Coxeter ont décrit une méthode systématique pour déterminer l'index de H dans G, en énumérant les classes de G (modulo H). Mais cette méthode n'était pas suffisante pour l'utilisation par l'ordinateur ; c'est grâce à de nombreuses recherches effectuées par Mendelsohn [20] et d'autres que la méthode est devenue exécutable et dont voici une notion générale de cet algorithme que nous exposerons sur un exemple.

Soit une présentation finie d'un groupe G

$$G = \langle a_1, \ldots, a_n \; ; \; R_1, R_2, \ldots, R_m \rangle$$

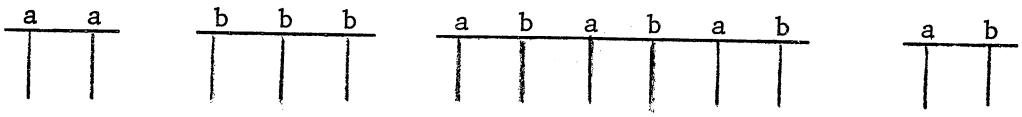
et soit H un sous-groupe de G engendré par les mots $W_1, \ldots, W_k$.

On commence à écrire les relations R_i et les mots W_j, côte à côte respectivement, à la tête des tables (une table pour chaque R_i ou W_j) dont chacune a ℓ_{i+1} colonnes avec ℓ_i = longueur de R_i ou longueur de W_j et tel que chaque générateur a_i ou $\bar{a}_i$, dans R_i ou W_j se trouve au sommet d'une colonne.

Par exemple : soit le groupe

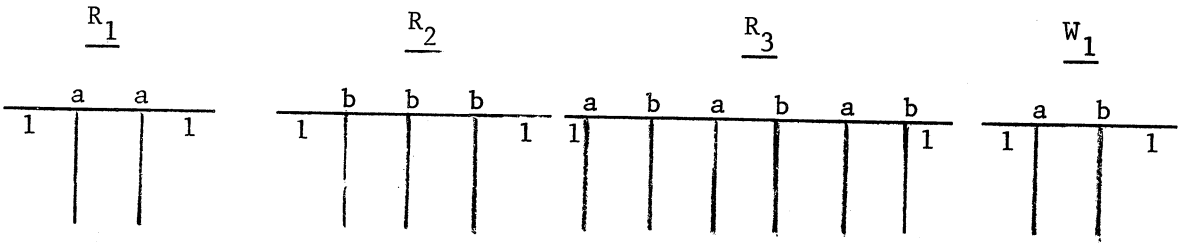
$$< a, b \; ; \; a^2, \; b^3, \; (ab)^3 >$$

et soit H le sous-groupe de G engendré par le mot $W_1 = ab$. On écrit :



On remplit ces tables de la manière suivante :

- Le premier et le dernier élément de chaque $k^{ème}$ ligne est l'entier k.
- Chaque table W_j est formée <u>d'une seule ligne</u> qui débute et se termine par l'entier 1
- On désigne chaque classe H_{g_i} par un entier k tel que 1 désigne la classe H
- Si j et k sont des entiers désignant respectivement les classes Hg_j et Hg_k et si a_i est un générateur, on fait rentrer les relations de la forme $j\, a_i = k$ ou $j\, \bar{a}_i = k$ dans la table en écrivant j et k côte à côte dans une paire de colonnes séparées au sommet par a_i ou $\bar{a}_i$.



D'où le procédé d'énumération :

<u>Si</u> un mot W_j est formé d'un seul générateur a_i, alors sa table représentera l'équation $1\, a_i = 1$ $\begin{array}{c} a_i \\ \hline 1 \,|\, 1 \end{array}$. Si ce chiffre 1 se trouve à droite ou à gauche de a_i ou $\bar{a}_i$ dans les autres tables, un chiffre 1 sera placé de l'autre côté de a_i dans la même ligne de telle manière que les relations $1\, a_i = 1$ et $1\, \bar{a}_i = 1$ soient toujours vérifiées. Quand toutes ces possibilités ont fini, le chiffre 2 est placé dans le premier espace vide en partant de la gauche vers la droite et en parcourant toute la ligne pour toutes les tables afin de définir une relation de la forme $1\, a_\ell = 2$ ou $2\, \bar{a}_\ell = 1$ où a_ℓ est un générateur se trouvant juste au sommet de la colonne de cet espace.

a	a	
1	2	1
2	1	2

b	b	b	
1		2	1
2	1		2

a	b	a	b	a	b	
1	2	1	2	1	2	1
2	1					2

a	b	
1	2	1

Ainsi, on fait rentrer le chiffre 2 partout où c'est possible, de la même manière que tout à l'heure, et on continuera le procédé. Une nouvelle relation $j\ a_i = k$ ou $j\ \bar{a}_i = k$ peut être induite pour un a_i, quand une ligne sera complète, puis on introduira cette information pour définir une nouvelle classe.

a	a	
1	2	1
2	1	2
3		3

b	b	b	
1	3	2	1
2	1	3	2
3	2	1	3

a	b	a	b	a	b	
1	2	1	2	1	2	1
2	1	3			3	2
3			3	2	1	3

a	b	
1	2	1

Le procédé s'arrête quand toutes les tables seront complètement remplies.

Ainsi, on a :

a	a	
1	2	1
2	1	2
3	4	3
4	3	4

b	b	b	
1	3	2	1
2	1	3	2
3	2	1	3
4	4	4	4

a	b	a	b	a	b	
1	2	1	2	1	2	1
2	1	3	4	4	3	2
3	4	4	3	2	1	3
4	3	2	1	3	4	4

a	b	
1	2	1

Il arrive quelquefois de trouver deux entiers différents qui désignent une même classe (c'est-à-dire $j\ a_i = k$ et $\ell\ a_i = k$), dans ce cas, toutes les occurrences du plus grand entier doivent être remplacées par le plus petit entier. C'est le phénomène de redondance.

Puisque pour chaque classe j, il existe un générateur a_i tel que ou bien $j\ a_i = k$, ou bien $j\ \bar{a}_i = k$ implique $j > k$, il est facile de lire les éléments représentant h_i des classes et on a :

$$G = H\,h_1 \cup H\,h_2 \cup \ldots \cup H\,h_r$$

où r est l'index de H dans G.

Dans le cas de notre exemple $r = 4$ et on a :

$h_1 = 1$, $h_2 = a$, $h_3 = b$ et $h_4 = b\,a$, alors

$G = H \cup Ha \cup Hb \cup Hba$

$H = \{ 1, ab, (ab)^2 \}$ d'où l'ordre de H est 3 ainsi l'ordre de G est 1 2 et on a :

$$G = \{ 1, ab, (ab)^2, a, aba, (ab)^2a, b, ab^2, (ab)^2b, ba, ab^2a, (ab)^2\,ba \}$$

CHAPITRE II

ETUDE DES GROUPES DONT LA PRESENTATION EST UN PRODUIT LIBRE OU UN PRODUIT LIBRE AMALGAME

En vue d'exposer la méthode de résolution du problème du mot dans le cas où $\mathcal{R}$ est réduit à une seule relation $\mathcal{R} = \{R\}$, on va rappeler la notion de produit libre et celle de produit libre amalgamé.

1. PRODUIT LIBRE.

1.1. Présentation qui est un produit libre.

Définition :

$G = \langle X \; ; \; \mathcal{R} \rangle$ est la présentation d'un produit libre lorsque on peut partitionner $X = X_1 \cup X_2$ $(X_1 \cap X_2 = \emptyset)$ et $\mathcal{R} = \mathcal{R}_1 \cup \mathcal{R}_2$ $(\mathcal{R}_1 \cap \mathcal{R}_2 = \emptyset)$ les relations de $\mathcal{R}_1$ sont des mots en $X_1 \cup \bar{X}_1$ et les relations de $\mathcal{R}_2$ sont des mots en $X_2 \cup \bar{X}_2$.

Propriété [19] :

Désignant par A et B les groupes

$$A = \langle X_1 \; ; \; \mathcal{R}_1 \rangle \text{ et } B = \langle X_2 \; ; \; \mathcal{R}_2 \rangle .$$

Le groupe G est uniquement déterminé par A et B. Plus précisément, G contient deux sous-groupes $\bar{A}$ et $\bar{B}$ respectivement isomorphes à A et B tels que $\bar{A} \cap \bar{B} = \{1\}$. On identifie $\bar{A}$ à A et $\bar{B}$ à B et le groupe G se note :

$$G = A * B.$$

Exemple :

La présentation $\langle a, b \; ; \; a^5, b^3 \rangle$ est celle d'un produit libre des groupes cycliques respectifs

$$A = \langle a \; ; \; a^5 \rangle \quad \text{et} \quad B = \langle b \; ; \; b^3 \rangle$$

1.2. Manipulation explicite d'un produit libre de deux groupes A,B tels que $A \cap B = 1$.

Définition 1 :

On appelle séquence toute suite finie $(x_1 \ldots x_n)$ (éventuellement vide $n = 0$) d'éléments de $A \cup B$. Si $x_i \neq 1$ et $x_i \in A$ (resp. B) on notera $F(x_i) = A$ (resp. B).

On dit qu'une séquence $(x_1, x_2, \ldots, x_n)$ est alternée lorsque

ou bien $n = 0$ (séquence vide)

ou bien $\forall i \quad x_i \neq 1$ et $F(x_i) \neq F(x_{i+1})$.

Définition 2 :

On définit l'opération de réduction π de l'ensemble des séquences dans lui-même récursivement par :

- $\pi(\ ) = (\ )$ pour la séquence vide

- $\pi(x_1) = \begin{cases} x_1 \text{ si } x_1 \neq 1 \\ (\ ) \text{ si } x_1 = 1 \text{ pour la séquence à un élément} \end{cases}$

- et si $\pi(x_1, \ldots, x_{n-1}) = (y_1, \ldots, y_r)$

$$\text{alors } \pi(x_1, \ldots, x_{n-1}, x_n) = \begin{cases} (y_1, \ldots, y_r) \text{ si } x_n = 1 \\ (y_1, \ldots, y_{r-1}) \text{ si } x_n = \bar{y}_r \\ (y_1, \ldots, y_r . x_n) \text{ si } x_n \neq \bar{y}_r \text{ et } F(x_n) = F(y_r) \\ (y_1, \ldots, y_r, x_n) \text{ si } F(x_n) \neq F(y_r). \end{cases}$$

Propriétés :

Soit ξ une séquence $(x_1, x_2, \ldots, x_n)$.

L'opération de réduction π a les propriétés suivantes :

- $\pi(\xi)$ est alternée

- si ξ est alternée, alors $\pi(\xi) = \xi$

- $\pi(\xi, \xi') = \pi(\pi(\xi), \xi')$ où ξ' est une autre séquence

- $\pi(\xi, x, \bar{x}, \xi') = \pi(\xi, \xi')$.

Théorème [19] :

Le produit libre $G = A * B$ est isomorphe à l'ensemble des séquences alternées muni du produit.

$\xi . \xi' = \pi (\xi, \xi')$ où (ξ, ξ') est la mise bout à bout des séquences ξ et ξ'.

Exemple :

Dans l'exemple 1.1., les séquences suivantes sont alternées

$$\xi = (a, b^2, a^3) \qquad \xi' = (a^2, b, a, b^2)$$

alors $\xi . \xi' = \pi (a, b^2, a^3, a^2, b, a, b^2) = (a^2, b^2)$

Remarque :

Si l'on assimile chaque séquence alternée (x) de longueur 1 à l'élément $x \neq 1$ (appartenant à A ou B), il est clair que tout élément g (défini par sa séquence alternée $(x_1, \ldots, x_n)$) s'exprime

$$g = (x_1) . (x_2) . \ldots . (x_n) = x_1 . x_2 . \ldots . x_n$$

de manière unique.

1.3. Problème du mot dans un produit libre $A * B$.

Soit $G = \langle X, Y ; \mathcal{S}, \mathcal{T} \rangle$ une présentation du produit libre des groupes

$$A = \langle X ; \mathcal{S} \rangle \text{ et } B = \langle Y ; \mathcal{T} \rangle .$$

Nous allons justifier très rapidement que lorsque le problème du mot dans les présentations de A et B est résoluble, alors il l'est dans la présentation de G.

Soit W un mot par rapport aux lettres de $X \cup Y \cup \bar{X} \cup \bar{Y}$. On découpe le mot en syllabes

$$W = W_1 W_2 \ldots W_q$$

où chaque W_i (appelé syllabe) est un mot par rapport à $X \cup \bar{X}$ (resp. $Y \cup \bar{Y}$), l'élément W_{i+1} étant un mot par rapport à $Y \cup \bar{Y}$ (resp. $X \cup \bar{X}$).

Cette décomposition est d'ailleurs unique. Chaque mot W_i définit un élément $[W_i]$ du sous-groupe A (resp. B). On appellera q la <u>longueur syllabique</u> de W.

<u>Exemple</u> :

Dans $G = \langle a, b \; ; \; a^5, b^3 \rangle$

le mot $a^2 \, b^4 \, \bar{b}^4 \, \bar{a}^3 \, b^3 \, \bar{a}^2$

a pour longueur syllabique 5 et ses syllabes sont :

$a^2, \; b^4 \, \bar{b}^4, \; \bar{a}^3, \; b^3, \; \bar{a}^2.$

Pour savoir si $W \sim 1$, on peut supposer au départ que W est <u>librement réduit</u>, car dans tout groupe, chaque mot est équivalent au mot librement réduit associé.

Partons alors de la décomposition syllabique de

$$W = W_1 \, W_2 \, \ldots \, W_q$$

On définit récursivement α (W) de la manière suivante :

- si $q = 0$ $\quad \alpha(\,) = (\,)$
- si $q = 1$ $\quad \alpha(W_1) = \begin{cases} (\,) \text{ si } W_1 \sim 1 \\ (W_1) \text{ sinon} \end{cases}$
- si $q = r - 1 \geqslant 1$. Supposons que

$$\alpha(W_1, W_2, \ldots, W_{r-1}) = (u_1, u_2, \ldots, u_s) \quad (s \geqslant 0)$$

alors

$$\alpha(W_1, W_2, \ldots, W_r) = \begin{cases} \text{si } W_r \sim 1 \text{ alors } (u_1, \ldots, u_s) \\ \text{sinon si } F[u_s] \neq F[W_r] \text{ alors } (u_1, \ldots, u_s, W_r) \\ \text{sinon si } u_s \, W_r \sim 1 \text{ alors } (u_1, \ldots, u_{s-1}) \\ \text{sinon } (u_1, \ldots, u_s \, W_r) \end{cases}$$

Il est clair alors que

$$W \sim 1 \Leftrightarrow \alpha(W) = (\,)$$

Par ailleurs, le calcul de α ne conduit qu'à la recherche de l'équivalence à 1 de mots appartenant à A ou B.

Exemple :

Dans l'exemple 1.1., le mot $W = a\, b^2\, a^{-5}\, b\, a^4$ est équivalent à 1.

En effet : par lecture successive de ses syllables, on obtient par α

a

$a,\ b^2$

$a,\ b^2$

a

$(\ )$

Remarque :

Dans la recherche de la réponse à la question $W \sim 1$, on peut supposer que les syllabes W_1, W_q sont tels que

$$F\,[W_1] \neq F\,[W_q].$$

1.4. Produit libre de plusieurs groupes.

Ayant défini le produit libre de deux groupes, on peut l'étendre à plusieurs groupes $G_1, G_2, \ldots, G_n$. Désignons par $\mathcal{G}_i$ et $\mathcal{R}_i$ respectivement l'ensemble des générateurs et l'ensemble des relations de G_i

$$\forall i = 1, 2, \ldots, n.$$

Posons $\mathcal{G} = \bigcup_{i=1}^{n} \mathcal{G}_i$ et

$\mathcal{R} = \bigcup_{i=1}^{n} \mathcal{R}_i$, alors le groupe

$$G = \langle \mathcal{G} \,;\, \mathcal{R} \rangle$$

est le produit libre de $G_1, \ldots, G_n$.

Le produit libre est commutatif et associatif

$$A * B = B * A$$

et $$(A * B) * C = A * (B * C)$$

où A, B et C sont trois présentations des groupes.

Remarque :

Un groupe libre $< x, y, \ldots, t ; \emptyset >$ peut être considéré comme le produit libre de $< x ; \emptyset > * < y ; \emptyset > * \ldots * < t ; \emptyset >$

2. PRODUIT LIBRE AVEC SOUS-GROUPES AMALGAMES.

Le groupe $< a , b ; a^4, b^6 >$ est le produit libre de deux groupes cycliques $< a ; a^4 >$ et $< b ; b^6 >$. Le groupe $< a , b ; a^4, b^6, a^2 b^{-3} >$ ($< a, b ; a^4, b^6, a^2 = b^3 >$) n'a pas la présentation d'un produit libre de deux groupes mais s'en rapproche d'une certaine manière.

2.1. Produit libre amalgamé.

Définition :

Soit le groupe G présenté par

$$G = < a_1, \ldots, a_n, b_1, \ldots, b_m ; R(a_i), \ldots, S(b_j), \ldots, U_1(a_i) = V_1(b_j), \ldots, U_q(a_i) = V_q(b_j) >$$

et soient

A' le sous-groupe de G engendré par $[a_1], \ldots, [a_n]$,

B' le sous-groupe de G engendré par $[b_1], \ldots, [b_m]$,

H' le sous-groupe de A' engendré par $[U_1(a_i)], \ldots, [U_q(a_i)]$,

K' le sous-groupe de B' engendré par $[V_1(b_j)], \ldots, [V_q(b_j)]$,

G est appelé le produit libre de A' et B' avec les sous-groupes H' et K' amalgamés par l'application $[U_i(a_\ell)] \longrightarrow [V_i(b_j)]$.

Exemples :

- Si $G = < a, b ; a^4, b^6, a^2 = b^3 >$, alors

$A' = < a ; a^4 > = \{ 1, [a], [a]^2, [a]^3 \}$

$B' = < b ; b^6 > = \{ 1, [b], [b]^2, [b]^3, [b]^4, [b]^5 \}$

$H' = \{ 1, [a]^2 \}$

$K' = \{ 1, [b]^3 \}$

H' et K' sont amalgamés par l'application

$$\{ 1, [a]^2 \} \longrightarrow \{ 1, [b]^3 \}$$
$$1 \longrightarrow 1$$
$$[a]^2 \longrightarrow [b]^3$$

- Autre exemple

$$G = \langle x , y ; x^3 \bar{y}^3 \rangle , \text{ alors}$$
$$A' = \langle x ; \emptyset \rangle$$
$$B' = \langle y ; \emptyset \rangle$$
$$H' = \langle x^3 ; \emptyset \rangle$$
$$K' = \langle y^3 ; \emptyset \rangle$$

H' et K' sont amalgamés par l'application $x^3 \longrightarrow y^3$.

La question qui se pose est la suivante :

Quand, dans le groupe G, les sous-groupes A' et B' auront-ils les présentations naturelles

$$\langle a_1, \ldots, a_n ; R(a_i), \ldots \rangle$$

et $\langle b_1, \ldots, b_m ; S(b_j), \ldots \rangle$ respectivement ?

La réponse est donnée par le théorème suivant démontré dans [19].

<u>Théorème [19]</u> :

<u>Appelons</u>

$$A = \langle a_1, \ldots, a_n ; R(a_i), \ldots \rangle$$
$$B = \langle b_1, \ldots, b_m ; S(b_j), \ldots \rangle$$

et $G = \langle a_1, \ldots, a_n, b_1, \ldots, b_m ; R(a_i), \ldots, S(b_j), \ldots, U_1(a_i) = V_1(b_j), \ldots$

$$U_q(a_i) = V_q(b_j) \rangle .$$

<u>Si A' est le sous-groupe de G engendré par</u> $[a_1], \ldots, [a_n]$,

<u>et B' le sous-groupe de G engendré par</u> $[b_1], \ldots, [b_m]$,

<u>alors A est isomorphe à A' par l'application</u> $[a_i]_A \longrightarrow [a_i]_G$

<u>et B est isomorphe à B' par l'application</u> $[b_j]_B \longrightarrow [b_j]_G$

si et seulement si l'application

$$[U_i(a_\ell)]_A \longrightarrow [V_i(b_j)]_B$$

induit un isomorphisme φ entre le sous-groupe H de A engendré par les $[U_i(a_\ell)]_A$ et le sous-groupe K de B engendré par les $[V_i(b_j)]_B$.

Remarques :

- Si on a $A \cap B = 1$, on appelle G produit libre de A et B avec les sous-groupes amalgamés H et K par φ et on note G par :

$$* (A, B, H, K, \varphi).$$

comme φ est un isomorphisme ; on appelle G le produit libre de A et B avec un sous-groupe amalgamé H. A et B sont appelés les facteurs de l'amalgamation et K sera déterminé par les générateurs V_i qui seront déterminés par les générateurs U_i et l'isomorphisme φ.

$$G = * (A, B, H, K, \varphi).$$

Le produit libre des groupes avec un sous-groupe amalgamé est une généralisation du produit libre, ce dernier en effet n'est autre que le produit libre amalgamé avec le sous-groupe trivial $\{1\}$.

2.2. Manipulation explicite du produit libre amalgamé.

Elle est nettement plus délicate et même dans le cadre du théorème précédent. Nous renvoyons à [[19] pp. 201].

2.3. Aperçu du problème du mot dans un produit libre amalgamé.

Supposons que nous sommes dans les conditions du théorème précédent et que nous avons choisi des éléments représentant les classes à droite de (A mod. H) et (B mod. K). On désigne par $\{C_1, C_2, \ldots\}$ l'ensemble de ces représentants et par $\bar{g}$ le représentant de g dans (A mod. H) ou (B mod. K).

Pour savoir si un mot $W \sim 1$, on peut supposer au départ que W est librement réduit.

Partons alors de la décomposition syllabique de

$$W = W_1 \, W_2 \, \ldots \, W_q .$$

On définit récursivement $\beta(W)$ de la manière suivante :

- $\beta(\ ) = 1$ l'élément identité de H

$$- \beta(W_r) = \begin{cases} W_r, \text{ si } W_r \in H ; \\ \varphi^{-1}(W_r), \text{ si } W_r \in K ; \\ (h, \overline{W}_r) \text{ où } h = W_r \, \overline{W}_r^{-1}, \text{ si } W_r \in A, \ W_r \notin H ; \\ (h, \overline{W}_r) \text{ où } h = \varphi^{-1}(W_r \, \overline{W}_r^{-1}), \text{ si } W_r \in B, \ W_r \notin K ; \end{cases}$$

Supposons que

$$\beta(W_2, \ldots, W_r) = (h, C_2, C_3, \ldots, C_q) \text{ où } h \in H \text{ ; alors}$$

$$\beta(W_1, W_2, \ldots, W_r) = \begin{cases} * \ (W_1 . h, C_2, C_3, \ldots, C_q), \text{ si } W_1 \in H ; \\ * \ (\varphi^{-1}(W_1) . h, C_2, \ldots, C_q), \text{ si } W_1 \in K ; \\ * \ (h', \overline{W_1 h}, C_2, C_3, \ldots, C_q) \text{ où } h' = (W_1 h) . (\overline{W_1 h})^{-1}, \\ \quad \text{si } W_1 \in A, \ W_1 \notin H, \ C_2 \notin A ; \\ * \ (h', \overline{W_1 \varphi(h)}, C_2, C_3, \ldots, C_q) \text{ où } h' = \varphi^{-1}[W_1 \varphi(h) . \overline{W_1 \varphi(h)}^{-1}], \\ \quad \text{si } W_1 \in B, \ W_1 \notin K, \ C_2 \notin B ; \\ * \ (W_1 h C_2, C_3, \ldots, C_q), \text{ si } W_1 \in A, \ W_1 \notin H, \ C_2 \in A, \\ \quad W_1 h C_2 \in H ; \\ * \ (\varphi^{-1}(W_1 \varphi(h) C_2), C_3, \ldots, C_q), \text{ si } W_1 \in B, \ W_1 \notin K, \\ \quad C_2 \in B, \ W_1 \varphi(h) C_2 \in K ; \\ * \ (h', W_1 h C_2, C_3, \ldots, C_q) \text{ où } h' = (W_1 h C_2) . (\overline{W_1 h C_2})^{-1}, \\ \quad \text{si } W_1 \in A, \ W_1 \notin H, \ C_2 \in A, \ W_1 h C_2 \notin H ; \\ * \ (h', \overline{W_1 \varphi(h)} C_2, C_3, \ldots, C_q) \text{ où} \\ \quad h' = \varphi^{-1}[(W_1 \varphi(h) C_2) . (\overline{W_1 \varphi(h) C_2})^{-1}], \\ \quad \text{si } W_1 \in B, \ W_1 \notin K, \ C_2 \in B, \ W_1 \varphi(h) C_2 \notin K. \end{cases}$$

<u>Remarque</u> : Le point (.) signifie l'élément associé dans H, c'est-à-dire $x.y = [xy]$.

En utilisant cette fonction récursive β Magnus [[19] pp. 201] a démontré que :

A chaque élément $[W]$ de G, on peut associer une séquence <u>unique</u>

$$(h, C_1, C_2, \ldots, C_r).$$

telle que

- h est un élément, éventuellement 1, de H
- C_i est un représentant de (A mod. H) ou de (B mod. K)
- $C_i \neq 1$
- C_i et C_{i+1} ne sont pas à la fois dans A ou B
- si h', C_i' sont les éléments de G qui correspondent à h, C_i par les homomorphismes $[a_j] \longrightarrow [a_j]$ et $[b_\ell] \longrightarrow [b_\ell]$ de A dans G et de B dans G respectivement ; alors

$$[W] = h' \, C_1' \, C_2' \ldots C_r'$$

Soit un mot W de syllabes $W_1, W_2, \ldots, W_r$; on applique α à $W_1, W_2, \ldots, W_r$, en tenant compte des groupes A et B seulement, c'est-à-dire du groupe $A * B$, on aura la séquence $u_1, u_2, \ldots, u_\ell$. Ensuite, on applique β à la séquence $u_1, u_2, \ldots, u_\ell$. On obtient la séquence <u>unique</u>

$$(h, C_1, C_2, \ldots, C_q).$$

Il en résulte que :

$$W \sim 1 \Leftrightarrow \beta(\alpha(W)) = 1.$$

Ainsi, β et α sont bien définis d'une manière récursive et si φ est <u>constructif</u>, on pourra décider si W est équivalent ou non à 1.

<u>Exemple</u> :

Soit le groupe $G = \langle a, b \; ; \; a^9, \; b^{12}, \; a^3 = b^4 \rangle$

On a successivement :

$$A = \langle a \; ; \; a^9 \rangle$$

$$B = \langle b \; ; \; b^{12} \rangle$$

$$H = \{ 1, \; a^3, \; a^6 \}$$

$$K = \{ 1, \; b^4, \; b^8 \}$$

H et K sont deux groupes cycliques d'ordre trois et l'application $a^3 \longrightarrow b^4$ induit un isomorphisme φ entre H et K. En utilisant la méthode de Mendelsohn, on peut écrire :

$$A = H \cup H\,a \cup H\,a^2$$

et $$B = K \cup K\,b \cup K\,b^2 \cup K\,b^3.$$

Le mot $W = a^6 \, b^8 \, a^6 \, b^{12}$ est équivalent à 1.

En effet : par lecture successive de ses syllabes, on obtient par α

$$a^6, \; b^8, \; a^6$$

puis on obtient par β

$$a^6$$

$$\varphi^{-1}(b^8).a^6 = a^6.a^6 = a^3$$

$$a^6.a^3 = a^9 = 1.$$

CHAPITRE III

GROUPES AVEC UNE SEULE RELATION DEFINIE.

On rappelle que le problème du mot en général est indécidable. Pour étudier les groupes qui possèdent une présentation ayant une seule relation définie, on applique la théorie du produit libre, et du produit libre amalgamé. On démontre que le problème du mot dans ce cas est résoluble grâce au théorème d'indépendance de Freiheitssatz. [19].

Lemme :

Soit $S_1 \subset S_2 \subset \ldots \subset S_n \subset \ldots$ une séquence d'ensembles de symboles.

Soit $T_1 \subset T_2 \subset \ldots \subset T_n \subset \ldots$ une séquence d'ensembles de mots, où T_i est un ensemble de mots par rapport aux symboles de S_i. Soit $G_i = < S_i \; ; T_i >$ et supposons que pour chaque i, le sous-groupe de G_i engendré par les symboles de S_{i-1} est isomorphe à G_{i-1} sous l'application identique. Démontrons qu'en posant $S = \bigcup_{i \geqslant 1} S_i$, $T = \bigcup_{i \geqslant 1} T_i$ et $G = \bigcup_{i \geqslant 1} G_i$, G est isomorphe à $< S \; ; T >$.

1) - G_n peut être assimilé à un ensemble W_n de mots en S_n. En effet :
G_1 peut être assimilé à un ensemble $W_1 \subset (S_1 \cup \bar{S}_1)^*$. On peut représenter $G_2 \setminus G_1$ par des mots W_2 en S_2
d'où $W_1 \subset W_2 \subset \ldots \subset W_n \subset \ldots$

$W = \bigcup_{n \geqslant 1} W_n$ représente alors par (une bijection appropriée) les éléments de $G = \bigcup_{n \geqslant 1} G_n$. On peut y définir un produit induit (qui n'est plus la concaténation) de telle sorte que W soit un groupe isomorphe à G. Montrons que $W \simeq < S \; ; T >$.

2) - Nous pouvons définir $\pi : S \longrightarrow W$ par :

$s \in S$ donc $s \in S_n - S_{n-1}$ alors $\pi(s)$ est l'élément de W_n tel que $s \underset{n}{\sim} \pi(s)$. Il est clair que $\pi(S)$ engendre W pour le produit <u>induit</u> dans W.

Par suite, il suffit de démontrer (étant donnée la propriété universelle d'une présentation) que $\begin{cases} \sigma \in (S \cup \bar{S})^* \\ \pi(\sigma) = 1 \text{ \underline{dans} } W \Leftrightarrow \sigma \sim 1 \text{ dans } < S \ ; \ T > , \end{cases}$

où $\pi(\sigma) = \pi(s_1)^{\varepsilon_1} . \pi(s_2)^{\varepsilon_2} . \ldots . \pi(s_k)^{\varepsilon_k}$ (avec $\sigma = s_1^{\varepsilon_1} s_2^{\varepsilon_2} \ldots s_k^{\varepsilon_k}$).

* Pour cela : si $\sigma \in T$ donc $\sigma \in T_n - T_{n-1}$ donc $\pi(\sigma) \in W_n$ et comme

$$\pi(\sigma) \underset{n}{\sim} \sigma \sim 1 \quad \text{d'où} \quad \pi(\sigma) = 1.$$

* De même : si $\pi(\sigma) = 1$, or $\sigma \in (S_n \cup \bar{S}_n)^*$ pour un certain n.

Donc $\sigma \underset{n}{\sim} \pi(\sigma) = 1$ dans W_n. Par suite σ est déduit de T_n et $\sigma \sim 1$ dans

$$< S \ ; \ T >$$

THEOREME D'INDEPENDANCE

<u>Théorème</u> :

<u>Soit</u> $R(a_1, a_2, \ldots, a_n)$ $(n \geqslant 2)$ <u>un mot cycliquement réduit par rapport à</u> $a_1, a_2, \ldots, a_n$ <u>qui contient</u> a_n <u>; le sous-groupe de</u>

$$G = < a_1, a_2, \ldots, a_n \ ; \ R(a_1, a_2, \ldots, a_n) >$$

<u>engendré par</u> $A1 = \{a_1, a_2, \ldots, a_{n-1}\}$ <u>est librement engendré par A1</u>.

<u>Donc chaque relation non triviale de G doit contenir</u> a_n.

<u>Remarque</u> :

G peut avoir un nombre infini de générateurs. La relation définie R engendre seulement un nombre fini de générateurs. Les générateurs qui ne figurent pas dans R engendrent <u>librement</u> un sous-groupe H de G, alors G sera considéré comme le produit libre de H et le sous-groupe G' de G défini par les générateurs qui figurent dans R et la relation définie R.

Par exemple si :

$$G = \langle a, b, c, d, e \; ; \; a^2 \; b^2 \; e^2 \rangle \text{, alors}$$

$$H = \langle c, d \; ; \; \emptyset \rangle \text{,} \quad G' = \langle a, b, e \; ; \; a^2 \; b^2 \; e^2 \rangle$$

$$\text{et} \quad G = H * G' = \langle c, d \; ; \; \emptyset \rangle * \langle a, b, e \; ; \; a^2 \; b^2 \; e^2 \rangle .$$

<u>Démonstration</u> :

Compte tenu de la remarque précédente et du fait que le produit libre de deux groupes libres est un groupe libre, on peut supposer que tous les générateurs de G figurent dans R et pour faciliter l'expression de la solution, on note : a_n par t

et $a_1, a_2, \ldots$ par b, c, ...

Ainsi, $G = \langle b, c, \ldots, t \; ; \; R(b, c, \ldots, t) \rangle$.

Pour démontrer ce théorème, on va utiliser <u>le raisonnement par induction sur la longueur de la relation définie R, (notée ℓ (R))</u>.

* Le théorème est vrai si $\ell(R) = 1$ (cela est trivial).

* Supposons que le théorème reste vrai pour tous les groupes avec une relation définie R, où $\ell(R) \leqslant r - 1$ et démontrons-le pour $\ell(R) = r$.

Nous étudierons deux cas :

A) : R contient deux générateurs.

B) : R contient trois générateurs ou plus.

<u>Notation</u> :

On désigne par $d_a(R)$ la somme des exposants du générateur a dans R.

- Si R contient un seul générateur, cela veut dire $R = t^r$, alors $G = \langle t \; ; \; t^r \rangle$ est manifestement le groupe cyclique d'ordre r.

<u>A) :</u> R contient deux générateurs d'où

$$G = \langle b, t \; ; \; R(b, t) \rangle .$$

<u>A1) :</u> Supposons que $d_t(R) \neq 0$.

Dans ce cas, si b est d'ordre fini, cela revient à dire qu'il existe un entier k tel que $b^k \simeq 1$, donc b^k appartient au sous-groupe normal N de G engendré par R, d'où

$$b^k = \prod_i S_i R^{\varepsilon_i} S_i^{-1} \quad \text{où } S_i \in (A \cup \bar{A})^* \text{ , } A = \{b, t\} \text{ et } \varepsilon_i = \pm 1 \ \forall\, i.$$

Alors, $k = d_b (b^k) = d_b (\prod_i S_i R^{\varepsilon_i} S_i^{-1}) = \sum_i d_b (R^{\varepsilon_i}) = (\sum_i \varepsilon_i)\, d_b (R)$;

$$0 = d_t (b^k) = \sum_i d_t (R^{\varepsilon_i}) = (\sum_i \varepsilon_i)\, d_t (R).$$

Comme $d_t (R) \neq 0$, on doit avoir $(\sum_i \varepsilon_i) = 0$, alors $k = 0$, ainsi b a un ordre infini dans $< b, t ; R(b, t) >$ et engendre <u>librement</u> un groupe.

<u>A2) :</u> $d_t (R) = 0$.

Dans ce cas, on va examiner le comportement de b dans le sous-groupe <u>normal</u> N de G engendré par b. Pour cela, il faut essayer de présenter N en utilisant le procédé de réécriture de Reidemeister-Schreier. On prend comme système représentatif de Schreier pour G mod. N l'ensemble de t^k, où k parcourt tous les entiers. N est engendré par b et les conjugués de b ($t^k b^{\ell} t^{-k}$ ou $t^k b^{-\ell} t^{-k}$). Ainsi, un mot W (b,t) définit un élément de N si et seulement si $d_t (W) = 0$; d'où le représentant $\overline{W}$ d'un mot W mod. N sera t^k, où $k = d_t (W)$. On peut conclure que N est engendré par les éléments b_k définis par $b_k = t^k b t^{-k} = t^k . b . \overline{(t^k b)}^{-1}$, et pour chaque mot W qui définit un élément de N, nous obtenons une chaîne τ (W) de W par rapport aux termes b_k ; ceci de la manière suivante :

Chaque terme b^{ε}, ($\varepsilon = \pm 1$), dans W est remplacée par b_k^{ε} où k est la somme des exposants des symboles t qui précèdent cet élément particulier b^{ε} dans W. Ainsi, le mot

$$W = t^2 b t^3 b^{-1} t^{-4} b^2 t^{-1} b$$

appartient à N car $d_t (W) = 0$ d'où

$$\tau (W) = b_2 b_5^{-1} b_1^2 b_0 .$$

Il est clair que ce procédé de codage $R \longrightarrow \tau(R)$ est tel que $\ell(\tau(R)) < \ell(R)$ car t figure dans R au moins une fois et pas dans $\tau(R)$. Si on adopte la notation

$$\tau(R) = P(\ldots, b_k, \ldots) = P_0 \text{ alors}$$

$$\tau(t^m R t^{-m}) = P(\ldots, b_{k+m}, \ldots) = P_m ,$$

N sera engendré par les b_k et aura comme relations définies les P_i.

$$N = < \ldots, b_{-1}, b_0, b_1, \ldots ; \ldots, P_{-1}, P_0, P_1, \ldots > .$$

Montrons que b est d'ordre infini dans

$$G = < b , t ; R (b , t) > .$$

Or, $b = t^0 b t^{-0} = b_0$; <u>R est cycliquement réduit</u> et $d_t (R) = 0$, ainsi $R \neq t^{\alpha} b^{\beta} t^{\gamma}$ où $\alpha, \gamma \neq 0$, et $\alpha + \gamma = 0$ car dans ce cas R n'est pas cycliquement réduit et on peut le remplacer par $R = b^{\beta}$, cas déjà étudié. Donc R contient un <u>sous-mot</u> de la forme $b^{\alpha} t^{\beta} b^{\gamma}$ où α, β et $\gamma \neq 0$. Si la somme des exposants de t dans le segment initial de R précédant b^{α} est k, en appliquant τ à $b^{\alpha} t^{\beta} b^{\gamma}$, on aura : $\ldots b_k^{\alpha} b_{k+\beta}^{\gamma} \ldots$.

Comme R est cycliquement réduit, par conséquent $\tau(R) = P_0 \Longrightarrow k + \beta \neq k$ (notons $q = k + \beta$). Ainsi, P_0 doit contenir au moins deux b_i différents (b_k et b_q).

Soient m et M les indices minimum et maximum respectivement de b dans P_0 , $m < M$. Pour chaque entier i, on va définir

$$N_i = < b_{m+i} , b_{m+i+1} , \ldots, b_{M+i} ; P_i >$$

où $$N_0 = < b_m , b_{m+1} , \ldots, b_M ; P_0 > .$$

Comme la longueur de P_i est inférieure à celle de R pour tout i on a :

(1) :: $\{ b_{m+i} , b_{m+i+1} , \ldots, b_{M+i-1} \}$

engendre un sous-groupe <u>libre de N_i</u>, de même pour

(2) :: $\{ b_{m+i+1} \ldots, b_{M+i} \}$

car b_{M+i} et b_{m+i} figurent dans P_i, et ceci reste vrai pour n'importe quel sous-ensemble de (1) et (2).

On désigne par

$$N_{0,1} = < b_m, b_{m+1}, \ldots, b_M, b_{M+1} ; P_0, P_1 >$$

le produit libre de N_0 et N_1 avec le sous-groupe libre dans chacun librement engendré par $b_{m+1}, b_{m+2}, \ldots, b_M$ amalgamé par l'application identique $\varphi = id$.

Comme $N_0 \subset N_{0,1}$ et

$$N_0 = < b_m, b_{m+1}, \ldots, b_M ; P_0 > ,$$

il suit que $b_m, b_{m+1}, \ldots, b_{M-1}$ engendrent librement un sous-groupe libre dans $N_{0,1}$, aussi dans

$$N_{-1} = < b_{m-1}, b_m, \ldots, b_{M-1} ; P_{-1} > .$$

$N_{-1,1}$ est le produit libre de N_{-1} et $N_{0,1}$ amalgamé par le sous-groupe librement engendré par $b_m, b_{m+1}, \ldots, b_{M-1}$ dans N_{-1} et $N_{0,1}$ (φ sera l'application identique $\varphi = id$.) ;

$$N_{-1,1} = < b_{m-1}, b_m, b_{m+1}, \ldots, b_{M-1}, b_M, b_{M+1} ; P_{-1}, P_0, P_1 > .$$

De même, puisque $N_1 \subset N_{0,1} \subset N_{-1,1}$,

$(b_{m+2}, \ldots, b_{M+1})$ engendrent librement un sous-groupe libre de $N_{-1,1}$. On considère maintenant

$$N_{-1,2} = N_{-1,1} * N_2 \text{ , puis } N_{-2,2} = N_{-2} * N_{-1,2}, \ldots .$$

En continuant ainsi, on obtient la chaîne des groupes

(3) :: $N_0 \subset N_{0,1} \subset N_{-1,1} \subset \ldots \subset N_{-i+1,i} \subset N_{-i,i} \subset N_{-i,i+1} \subset \ldots,$

où $N_{-i,i}$ est le produit libre de N_{-i} et $N_{-i+1,i}$ amalgamé par un sous-groupe libre, et $N_{-i,i+1}$ est le produit libre de $N_{-i,i}$ et N_{i+1} avec un sous-groupe libre amalgamé. D'après le lemme démontré au début de ce chapitre, on peut conclure que N est l'union de cette chaîne (3) de groupes. Puisque chaque N_i est un sous-groupe d'un certain groupe de (3), alors N_i est un sous-groupe de N. Etant donné que $m < M$ entraîne n'importe quel générateur de N_i engendre un groupe libre dans N_i donc dans N. Mais il existe un certain entier i_0 tel que $b_0 \in N_{i_0}$, ainsi b_0 engendre un groupe libre dans N_{i_0} donc dans N, ainsi b engendre un groupe libre dans G car N est un sous-groupe de G.

<u>Exemple</u> :

Soit le groupe

$$G = \langle b, t \; ; \; t^{-1} b t b^2 t b^{-3} t^{-1} \rangle$$

$d_t(R) = 0$, $\tau(R) = b_{-1} b_0^2 b_1^{-3}$ d'où $m = -1$ et $M = 1$.

$$N = \langle \ldots, b_{-1}, b_0, b_1, \ldots \; ; \; \ldots, b_{-1} b_0^2 b_1^{-3}, \ldots \rangle$$

$$N_i = \langle b_{i-1}, b_i, b_{i+1} \; ; \; b_{i-1} b_i^2 b_{i+1}^{-3} \rangle$$

$$N_{0,1} = \langle b_{-1}, b_0, b_1, b_2 \; ; \; b_{-1} b_0^2 b_1^{-3}, b_0 b_1^2 b_2^{-3} \rangle = N_0 * N_1$$

$$N_{-1,1} = \langle b_{-2}, b_{-1}, b_0, b_1, b_2 \; ; \; b_{-2} b_{-1}^2 b_0^{-3}, b_{-1} b_0^2 b_1^{-3}, b_0 b_1^2 b_2^{-3} \rangle = N_{-1} * N_{0,1} .$$

Ici, on peut choisir $N_{i_0} = N_0$.

<u>B) :</u> R contient <u>au moins</u> trois générateurs b, c, ..., t.

<u>B1) :</u> La somme des exposants d'un générateur de R, <u>autre que t</u>, est nulle. Par exemple $d_b(R) = 0$.

On va démontrer qu'il n'existe pas de relations $\mathcal{S}(b, c, \ldots)$ de G qui ne contiennent pas t ; pour montrer ceci, on va supposer le contraire c'est-à-dire on suppose qu'il existe une telle relation $\mathcal{S}(b, c, \ldots)$.

Comme R (b, c, ..., t) est la relation définie de G et $d_b(R) = 0$, on doit avoir $d_b(\mathcal{S}) = 0$ pour la même raison que précédemment. De même, on va utiliser le même raisonnement que pour -A2- sous une forme plus détaillée. Comme $d_b(R) = 0$, il est naturel de considérer le sous-groupe normal N de G engendré par c, ..., t c'està-dire par tous les générateurs de G en excluant b. Un mot W (b, c, ..., t) définit un élément de N si et seulement si $d_b(W) = 0$. Pour obtenir une présentation de N, nous utilisons un procédé de réécriture τ de Reidemeister-Schreier qui a comme système représentatif de Schreier $\{b^k$, k un entier quelconque $\}$.

Nous notons les éléments $b^k\, c\, b^{-k}, \ldots, b^k\, t\, b^{-k}$ par

$c_k, \ldots, t_k$ respectivement.

Le représentant de W (b, c, ..., t) est b^k où $k = d_b(W)$. Pour représenter un élément V(b,c, ..., t) de N par rapport aux $c_k, \ldots, t_k$, on remplacera chaque symbole $c^{\varepsilon}, \ldots, t^{\varepsilon}$ de V, ($\varepsilon = \pm 1$), à l'aide de τ, par $c_k^{\varepsilon}, \ldots, t_k^{\varepsilon}$, où k est la somme des exposants de b du segment initial de V précédant $c^{\varepsilon}, \ldots, t^{\varepsilon}$, respectivement.

Par exemple : $\tau(b\, c\, t\, d\, c^2\, t^2\, b^{-2}\, t^{-2}\, d\, b\, t\, c) = c_1\, t_1\, d_1\, c_1^2\, t_1^2\, t_{-1}^{-2}\, d_{-1}\, t_0\, c_0$.

On pose
$$\begin{cases} \tau(R) = P(\ldots, c_k, \ldots, t_k, \ldots) = P_0 \\ \tau(b^m R b^{-m}) = P(\ldots, c_{k+m}, \ldots, t_{k+m}, \ldots) = P_m . \end{cases}$$

Ainsi, N sera présenté par

$$(4) :: \quad N = \langle \ldots, c_{-1}, c_0, c_1, \ldots, t_{-1}, t_0, t_1, \ldots ; \ldots, P_{-1}, P_0, P_1, \ldots \rangle .$$

On va démontrer que les générateurs de N autres que les t_i, ($i = 0, \pm 1, \pm 2, \ldots$), engendrent librement un groupe libre.

R étant cycliquement réduit, $\tau(R) = P_0$ l'est aussi par rapport aux $(c_k, \ldots, t_k)$. En outre, t figure dans R, il suit que P_0 contient quelques t_k. Soient m et M respectivement les indices minimum et maximum de t dans P_0 et soit N_i engendré par tous les générateurs de (4) à l'exception des t_j où $j < m + i$ et $j > M + i$, et soit

$P_i = \tau(b^i R b^{-i}) = P(\ldots, c_{k+i}, \ldots, t_{k+i}, \ldots)$ la seule relation définie de N_i. On a :

(5) :: $N_i = < \ldots, c_{-1}, c_0, c_1, \ldots, t_{m+i}, \ldots, t_{M+i} ; P_i >$.

R contient b, donc $\tau(R) = P_0$ a une longueur plus petite que celle de R. On obtient les P_i par une simple translation des indices, donc on peut appliquer l'hypothèse d'induction sur les N_i. Les générateurs de (5), sauf t_{m+i} (ou t_{M+i}) engendrent librement un sous-groupe libre de N_i.

Soit $N_{0,1} = < \ldots, c_{-1}, c_0, c_1, \ldots, t_m, t_{m+1}, \ldots, t_{M+1} ; P_0, P_1 >$, (on trouve ici tous les générateurs de (4) sauf les t_k pour

$k < m$ et $k > M + 1$),

le produit libre de N_0 et N_1 amalgamé par le sous-groupe librement engendré par $\ldots, c_{-1}, c_0, c_1, \ldots, t_{m+1}, \ldots, t_M$ et par l'application identique (φ = id.).

Comme $N_0 \subset N_{0,1}$, alors les éléments $\ldots, c_{-1}, c_0, c_1, \ldots, t_m, t_{m+1}, \ldots, t_{M-1}$ engendrent librement un sous-groupe libre de $N_{0,1}$ et ainsi on peut définir

$N_{-1,1} = < \ldots, c_{-1}, c_0, c_1, \ldots, t_{m-1}, t_m, \ldots, t_M, t_{M+1} ; P_1, P_0, P_1 >$

comme étant le produit libre de N_{-1} et $N_{0,1}$ amalgamé par le sous-groupe librement engendré par $\ldots, c_{-1}, c_0, c_1, \ldots, t_m, \ldots, t_{M-1}$ et par l'application identique (φ = id.).

$$N_1 \subset N_{0,1} \subset N_{-1,1}$$

d'où les éléments $\ldots, c_{-1}, c_0, c_1, \ldots, t_{m+2}, \ldots, t_{M+1}$ engendrent librement un sous-groupe libre de $N_{-1,1}$, donc on peut définir $N_{-1,2}$ et ainsi de suite, on construit une chaîne de groupes

$$N_0 \subset N_{0,1} \subset N_{-1,1} \subset \ldots \subset N_{-i+1,i} \subset N_{-i,i} \subset N_{-i,i+1} \subset \ldots$$

où $N_{-i,i}$ est le produit libre de N_{-i} et $N_{-i+1,i}$ amalgamé par le sous-groupe librement engendré par

$$\ldots, c_{-1}, c_0, c_1, \ldots, t_{m-i+1}, \ldots, t_{M-i}$$

et par l'application identique (φ = id.).

De même $N_{-i,i+1}$ est le produit libre de $N_{-i,i}$ et N_{i+1} amalgamé par le sous-groupe librement engendré par

$$\ldots, c_{-1}, c_0, c_1, \ldots, t_{m+i+1}, \ldots, t_{M+i}$$

et par l'application identique (φ = id.).

En utilisant le résultat du lemme, on peut en déduire que N est l'union de cette chaîne. Puisque les générateurs de N_i autres que les t_k engendrent librement un groupe libre (hypothèse d'induction) en particulier ceux de N_0 ; et comme $N_0 \subset N$ alors, les générateurs de N autres que les t_k engendrent librement un groupe libre dans N.

Reconsidérons la relation $\mathcal{S}(b, c, \ldots)$. On remarque que $d_b(\mathcal{S}) = 0$; donc si l'on suppose que $\mathcal{S}$ est cycliquement réduit et différent du mot vide, alors $\tau(\mathcal{S})$ est cycliquement réduit et il ne contient aucun t_k car t ne figure pas dans $\mathcal{S}(b, c, \ldots)$. Il contient du même coup d'autres générateurs que b ; par conséquent $\tau(\mathcal{S})$ est une relation non vide de N où ne contient aucun t_k, ce qui est contraire à ce qu'on vient de démontrer. Ainsi chaque relation non triviale de G contient t, et par suite les générateurs de G autres que t engendrent librement un sous-groupe libre.

<u>B2) :</u> La somme des exposants de chaque générateur dans R, autre que t, est non nulle.

On va essayer de ramener ce cas au cas B1.

Comme R contient au moins trois générateurs b, c, t, on pose $\beta = d_b(R)$ et $\gamma = d_c(R)$ où β et γ sont différents de zéro. On va appliquer homomorphiquement

$$G = \langle b, c, \ldots, t \; ; \; R(b, c, \ldots, t) \rangle$$

dans le groupe

$$E = \langle x, c, \ldots, t \; ; \; R(x^{\gamma}, c, \ldots, t) \rangle$$

en appliquant $b \longrightarrow x^{\gamma}$, $c \longrightarrow c$, ..., $t \longrightarrow t$.

Pour démontrer que b, c, ... sans t engendrent librement un sous-groupe libre de G, il suffit de démontrer que x, c, ... sans t engendrent librement un sous-groupe libre de E, car on a ici une application homomorphe et on remarque que si l'image homomorphe des générateurs est un ensemble libre de générateurs, l'ensemble original des générateurs l'est aussi.

On pose $y = c\, x^{\beta}$ et en appliquant la transformation de Tietze à E on obtient :

$$E = \langle x, c, \ldots, t \; ; \; R(x^{\gamma}, c, \ldots, t) \rangle$$

$$\simeq \langle x, c, \ldots, t, y \; ; \; R(x^{\gamma}, c, \ldots, t), \; y = c\, x^{\beta} \rangle$$

$$\simeq \langle x, y, \ldots, t \; ; \; R(x^{\gamma}, yx^{-\beta}, \ldots, t) \rangle .$$

$$d_x\,(R\,(x^{\gamma}, yx^{-\beta}, \ldots, t)) = \beta\gamma - \gamma\beta = 0.$$

Si l'on réduit cycliquement $R\,(x^{\gamma}, yx^{-\beta}, \ldots, t)$ t y figure encore ; sinon, $R\,(x^{\gamma}, (c\,x^{\beta})\,x^{-\beta}, \ldots, t)$ cycliquement réduit ne contient pas t, d'où $R\,(x^{\gamma}, c, \ldots, t)$ et $R\,(b, c, \ldots, t)$ cycliquement réduits ne contiennent pas t ; or, ce résultat est inexact car $R\,(b, c, \ldots, t)$ contient t.

$R(x^{\gamma}, y\,\bar{x}^{\beta}, \ldots, t)$ peut aussi avoir par rapport aux x, y, ..., t une longueur supérieure à celle de $R(b, c, \ldots, t)$ par rapport à b, c, ..., t ; mais cette augmentation de la longueur n'est due qu'à des symboles x. Quand on réécrit en utilisant le symbole x pour définir les indices des nouveaux générateurs la longueur de $\tau(R(x^{\gamma}, y\,\bar{x}^{\beta}, \ldots, t))$, tout compte fait, aura diminué et par suite

$$\ell(\tau(R(x^{\gamma}, y\,\bar{x}^{\beta}, \ldots, t))) < \ell(R(b, c, \ldots, t)).$$

Ainsi, le théorème est vérifié.

<u>APPLICATION DU THEOREME D'INDEPENDANCE POUR RESOUDRE LE PROBLEME DU MOT</u> :

Soit $G = \langle a_1, a_2, \ldots, a_n ; R(a_1, \ldots, a_n) \rangle$ une présentation d'un groupe avec une seule relation définie $R(a_1, \ldots, a_n)$ supposée cycliquement réduite, et soit N le sous-groupe normal du groupe libre.

$$F = \langle a_1, \ldots, a_n ; \emptyset \rangle$$

engendré par $R(a_1, \ldots, a_n)$. N est engendré par R et les conjugués de R. G peut s'écrire

$$G = F/N .$$

Soit un mot W de $(A \cup \bar{A})^*$, où $A = \{a_1, \ldots, a_n\}$, et on suppose que W, cycliquement réduit, est <u>différent du vide</u>.

<u>Théorème</u> :

<u>La condition nécessaire pour que</u> W <u>soit équivalent à 1 est que</u> $d_{a_i}(W) / d_{a_i}(R)$ <u>soit un entier</u> $\forall\, i = 1, \ldots, n$.

Si W est équivalent à 1 dans G et pas dans F, c'est-à-dire $W \in N$, alors W est <u>librement</u> équivalent à un produit fini

$$T_1 R^{\varepsilon_1} T_1^{-1} \cdot \ldots \cdot T_q R^{\varepsilon_q} T_q^{-1} \text{ où } \varepsilon_i = \pm 1 \text{ pour } i = 1, 2, \ldots, q ;$$

$$W \approx \prod_{i=1}^{q} T_i R^{\varepsilon_i} T_i^{-1}$$

où les T_i appartiennent à F, sont librement réduits et ne contiennent pas un sous-mot de la forme $R^{\pm 1}$. D'où la condition nécessaire

$d_{a_i}(W)$ est divisible par $d_{a_i}(R)$ $\forall$ le générateur a_i.

Remarque :

Soit un mot V qui s'écrit $V = s_1 s_2 s_3$ où s_1, s_2 et s_3 sont aussi des mots ; alors s_1, s_2 ou s_3 sont dits des sous-mots de V.

Etant donné un mot quelconque W où on veut étudier son équivalence à 1. On commence par le test de la divisibilité. Si $d_{a_i}(W) / d_{a_i}(R)$ est vrai pour tous les a_i, alors le mot W peut être équivalent à 1 ; par contre, s'il existe au moins un a_{i_0} tel que $d_{a_{i_0}}(R)$ ne divise pas $d_{a_{i_0}}(W)$, alors W n'est pas équivalent à 1.

Dans le cas où le mot W peut être équivalent à 1, on va résoudre un problème plus général :

l'extension du problème du mot.

Définition :

Soit $A\,1 \subsetneq A$, A 1 éventuellement vide, et soit un élément $W \in (A \cup \bar{A})^*$. L'extension du problème du mot, c'est d'étudier la possibilité d'exprimer l'élément W par rapport aux seuls générateurs de A 1, et dans le cas positif de l'exprimer au moins d'une manière. A 1 est appelé un sous-ensemble propre de A.

En utilisant le résultat du théorème d'indépendance et sa méthode de démonstration, on peut en déduire que :

Résoudre l'extension du problème du mot par rapport à un sous-ensemble propre quelconque A 1, revient à le résoudre par rapport à un sous-ensemble propre maximal B qui contient A 1, puis on réduit librement l'élément exprimé en fonction des termes de B.

CHAPITRE IV

ALGORITHMES

Nous expliquons d'abord brièvement chacun des algorithmes que nous avons programmés, la plupart avec un exemple au moins pour faciliter leur compréhension, puis nous donnons un organigramme de chaque algorithme, cet organigramme étant suivi du programme correspondant.

1. PRESENTATION D'UN GROUPE FINI.

Chaque groupe dénombrable est isomorphe à un groupe de transformations, lequel est isomorphe à un sous-groupe d'un groupe à deux générateurs [[19] pp. 259] ; ceci reste vrai pour un groupe fini. Nous nous bornerons à trouver une présentation d'un groupe fini de transformations à deux générateurs. En utilisant la représentation graphique d'un groupe et le résultat de [2], on peut trouver une présentation d'un groupe de transformations.

A partir de deux permutations quelconques de σ_n , on définit la table de Cayley du groupe et d'après cette table on construit la matrice d'incidence (sommets-branches) du graphe associé. Comme c'est un groupe à deux générateurs, la matrice est de catégorie deux [2] et sa dernière ligne est la somme des autres lignes restantes (modulo 2) car dans chaque colonne, il y a seulement deux composantes de valeur 1. Soit p l'ordre du groupe. La matrice sera d'ordre (p,2p). Comme la dernière ligne est une combinaison linéaire des autres (modulo 2), on peut l'éliminer et on aura une matrice d'ordre (p-1,2p). Par la technique de Jordan, on peut l'écrire sous la forme $M = [I,K]$ où I et K sont respectivement la matrice unité d'ordre (p-1,p-1) et une matrice d'ordre (p-1,p+1). En partant de la matrice $A = [T_{K,I}]$ où I est la matrice unité d'ordre (p+1,p+1), on aura une base de cycles du graphe associé (chaque ligne de A forme un élément de cette base). On associe les relations algébriques à cette base et on aura un système de relations (en général redondantes même après avoir enlevé les relations algébriques qui sont égales à une autre ou qui s'obtiennent par permutation circulaire d'une autre).

Ainsi, on a une présentation d'un groupe fini.

<u>Exemple</u> : Soient les deux permutations de σ_4

a (notée 2) $\quad (1\ 2\ 3\ 4) \xmapsto{2} (1\ 2\ 4\ 3)$

et b (notée 3) $\quad (1\ 2\ 3\ 4) \xmapsto{3} (2\ 1\ 4\ 3)$

La table de Cayley est la suivante :

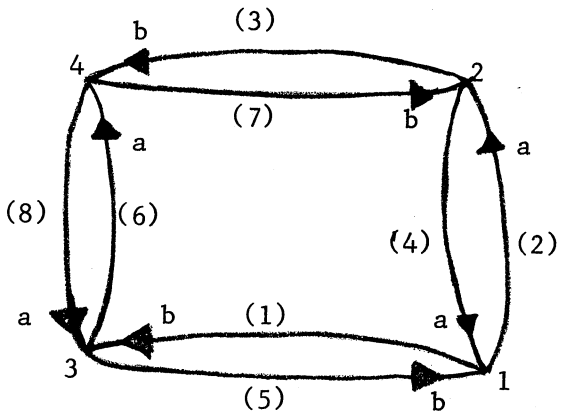
G	1	2	3	4
1	1	2	3	4
2	2	1	4	3
3	3	4	1	2
4	4	3	2	1

où 1 et 4 sont respectivement la permutation identique et la permutation

$$(1\ 2\ 3\ 4) \xmapsto{4} (2\ 1\ 3\ 4)$$

La matrice d'incidence (sommets-branches) est d'ordre (4,8).

M	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
1	1	1		1	1			
2		1	1	1			1	
3	1				1	1		1
4			1			1	1	1



On remarque que la 4ème ligne est égale à

$M(1,*) \oplus M(2,*) \oplus M(3,*)$.

Ainsi on peut simplifier M en M'

M'	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
1	1	1		1	1			
2		1	1	1			1	
3	1				1	1		1

Puis on aura par la technique de Jordan

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
1	1				1	1		1
2		1		1		1		1
3			1			1	1	1

$= (I,K)$

d'où la matrice $A = [T_K, I_5] =$

	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
1		1		1				
2	1				1			
3	1	1	1			1		
4			1				1	
5	1	1	1					1

Les cycles de base sont :

(2).(4) , (1).(5) , (1).(2).(3).(6) , (3).(7) et (1).(2).(3).(8).

Les relations algébriques associées sont :

aa, bb, $\bar{b}ab\bar{a}$, bb et $\bar{b}aba$.

En simplifiant, on obtient : aa, bb, $\bar{b}ab\bar{a}$ et $\bar{b}aba$.

On remarque que $\bar{b}ab\bar{a}.aa \approx \bar{b}aba$.

2. RESOLUTION DU PROBLEME DU MOT.

2.1. Représentation utilisée.

On représente un mot $a_1^{\varepsilon_1} a_2^{\varepsilon_2} \ldots a_n^{\varepsilon_n}$, par une suite de (couples d'entiers) séparés par un blanc, les couples étant séparés par une virgule. Pour indiquer la fin d'un mot, on met 0 0 ,. Les résultats sont représentés par une suite de la forme A i j A i' j' ... où (i , j ; i' , j' , ...) sont des entiers et un couple (i , j) représente respectivement l'indice et l'exposant d'un générateur.

Présentation de l'algorithme pratique du problème du mot.

Soit le groupe

$$G = \langle a_1, \ldots, a_n \; ; \; R\,(a_1, \ldots, a_n) \rangle$$

et soit W un mot par rapport aux générateurs $a_1, \ldots, a_n$.

W est-il équivalent à un ?

Pour répondre à cette question, on va appliquer les deux phases essentielles (d) et (e) suivantes, successivement plusieurs fois si cela est nécessaire.

(d) : On effectue le test de divisibilité (cf. Ch. III) ; si ce test est vérifié, alors W peut être équivalent à 1, sinon il n'est pas équivalent à 1 et le problème de décision a une réponse négative.

(e) : Dans le cas où la divisibilité est vérifiée, on va essayer de réduire le problème du mot à un problème où la relation définie R' (déduite de R) est résoluble ou vide et ceci en réduisant chaque fois la longueur de la relation définie correspondante, c'est-à-dire en suivant le même principe de la démonstration du théorème d'indépendance.

On suppose qu'il existe un générateur a_{i_0} qui figure dans R tel que $d_{a_{i_0}}(R) = 0$, sinon le programme prévoit de faire une transformation sur R. On calcule $^T(W) = W'$, W' appartient au sous-groupe normal de G engendré par $A - a_{i_0}$ (où $A = \{a_1, \ldots, a_n\}$).

On décompose W' en ses formes canoniques par rapport aux sous-groupes N_i de N qui ont les relations correspondantes P_i dont la longueur est inférieure à la longueur de R et où φ est l'application identique. Pour chaque syllabe W'_j de W', on recommence (d) et (e) et puis on appliquera les propriétés du produit libre et du produit libre amalgamé pour les sous-groupes N_i. Toutes les autres techniques de la résolution pratique du problème du mot, c'est-à-dire de la programmation, se trouvent plus détaillées dans le programme.

<u>Exemples</u> :

-1- Soit le groupe $G = \langle a , b ; aba^2b \rangle$ et soit le mot $W = aba^{-1}b^{-1}$ dont on va étudier son équivalence à un.

(d) : On remarque facilement que :

$d_a(W)/d_a(R) = 0/3 = 0$ et $d_b(W)/d_b(W) = 0/2 = 0$, d'où W peut être équivalent à 1.

(e) : $R = aba^2b$ n'est pas résoluble ; $d_a(R)$ et $d_b(R)$ sont différents de zéro, ainsi on doit faire la transformation

$$a \longrightarrow x^2 \text{ et } b \longrightarrow yx^{-3} \text{ ce qui nous donne}$$

$$\langle a,b ; aba^2b \rangle \xrightarrow{\text{hom}} \langle x,b ; x^2 b x^4 b \rangle \xrightarrow{i} \langle x,b,y ; x^2bx^4b,y = bx^3 \rangle$$

$$\xrightarrow{i} \langle x,y ; x^2 y x^{-3} x^4 y x^{-3} \rangle$$

En réduisant cycliquement $x^2 y x^{-3} x^4 y x^{-3}$, on aura :

$$R'(x,y) = x^{-1} y x y.$$

En appliquant la même transformation au mot W, on obtiendra le mot

$$W'(x,y) = x^2 y x^{-3} x^{-2} x^3 y^{-1} \approx x^2 y x^{-2} y^{-1} \text{ (cycliquement).}$$

Ainsi, tout revient à étudier l'équivalence à 1 de W' dans le groupe

$$G' = \langle x,y ; x^{-1} y x y \rangle$$

(On remarque que $\ell(R) = 5 > \ell(R') = 4$).

(d) : Comme on a fait un changement de générateurs, on recommence le test de divisibilité par rapport à x et y et on trouvera :

$d_x(W') = d_x(R') = 0$ et $d_y(W')/d_y(R') = 0/2 = 0$, d'où W' peut être équivalent à 1.

(Le deuxième exemple montrera qu'à ce stade le travail s'arrête, ce qui montre chaque fois qu'on fait un changement sur les générateurs, il faut recommencer le test).

(e) : $R' = x^{-1}\, y\, x\, y$ n'est pas résoluble, mais $d_x(R') = 0$; ainsi, soit N le sous-groupe normal de G' engendré par y.

$$P'_0 = \tau_x\,(x^{-1}\, y\, x\, y) = y_{-1}\, y_0$$

$$P'_i = \tau_x\,(x^i.\, x^{-1}\, y\, x\, y.x^{-i}) = y_{i-1}\, y_i \qquad \text{et}$$

$$N = \langle \ldots, y_{-1}, y_0, y_1 \ldots ; \ldots, P'_{-1}, P'_0, P'_1, \ldots \rangle$$

$$W'_0 = \tau_x\,(x^2\, y\, x^{-2}\, y^{-1}) = y_2\, y_0^{-1}\,.$$

Etudions l'équivalence à 1 de W'_0 dans N en tenant compte du produit libre amalgamé des sous-groupes

$N_i = \langle y_{i-1}, y_i ; y_{i-1}\, y_i \rangle$ de N avec $\varphi = id$.

On remarque que chaque relation P'_i est résoluble par rapport à un quelconque de ses générateurs.

$P'_2 = y_1\, y_2$, d'où $y_2 = y_1^{-1}$ et $W'_0 = y_2\, y_0^{-1} \simeq y_1^{-1}\, y_0^{-1}$.

En plus, $P'_1 = y_0\, y_1$, d'où $y_1 = y_0^{-1}$ et $W' \simeq y_1^{-1}\, y_0^{-1} \simeq y_0\, y_0^{-1} \approx 1$.

Ainsi, le mot $W = aba^{-1}b^{-1} \simeq 1$ dans le groupe $G = \langle a,b ; aba^2b \rangle$ c'est-à-dire ab = ba ; alors on peut conclure que ce groupe est commutatif, ce qui facilite le traitement du problème du mot pour un autre mot V de ce groupe.

- 2 - Soit le groupe

$G = < a,b \; ; \; a^{-2} \, b^{2} \, a^{-2} \, b^{2} >$ et soit le mot

$W = b^{4} \, a^{2} \, b^{-2} \, a^{-2} \, b^{4} \, a^{-2} \, b^{-2} \, a^{2}$ dont on veut étudier l'équivalence à 1.

On suit la même marche que tout à l'heure ; on aura successivement :

(d) : $d_a(W)/d_a(R) = 0/4 = 0$ et $d_b(W)/d_b(R) = 4/4 = 1$, alors le mot donné W peut être équivalent à 1.

(e) : R n'est pas résoluble ; $d_a(R)$ et $d_b(R)$ sont différents de zéro.

On fait la transformation $a \longrightarrow x^{4}$ et $b \longrightarrow yx^{4}$ dans G et on aura :

$$G' = < x,y \; ; \; x^{-4} \, y \, x^{4} \, y \, x^{-4} \, y \, x^{4} \, y >$$

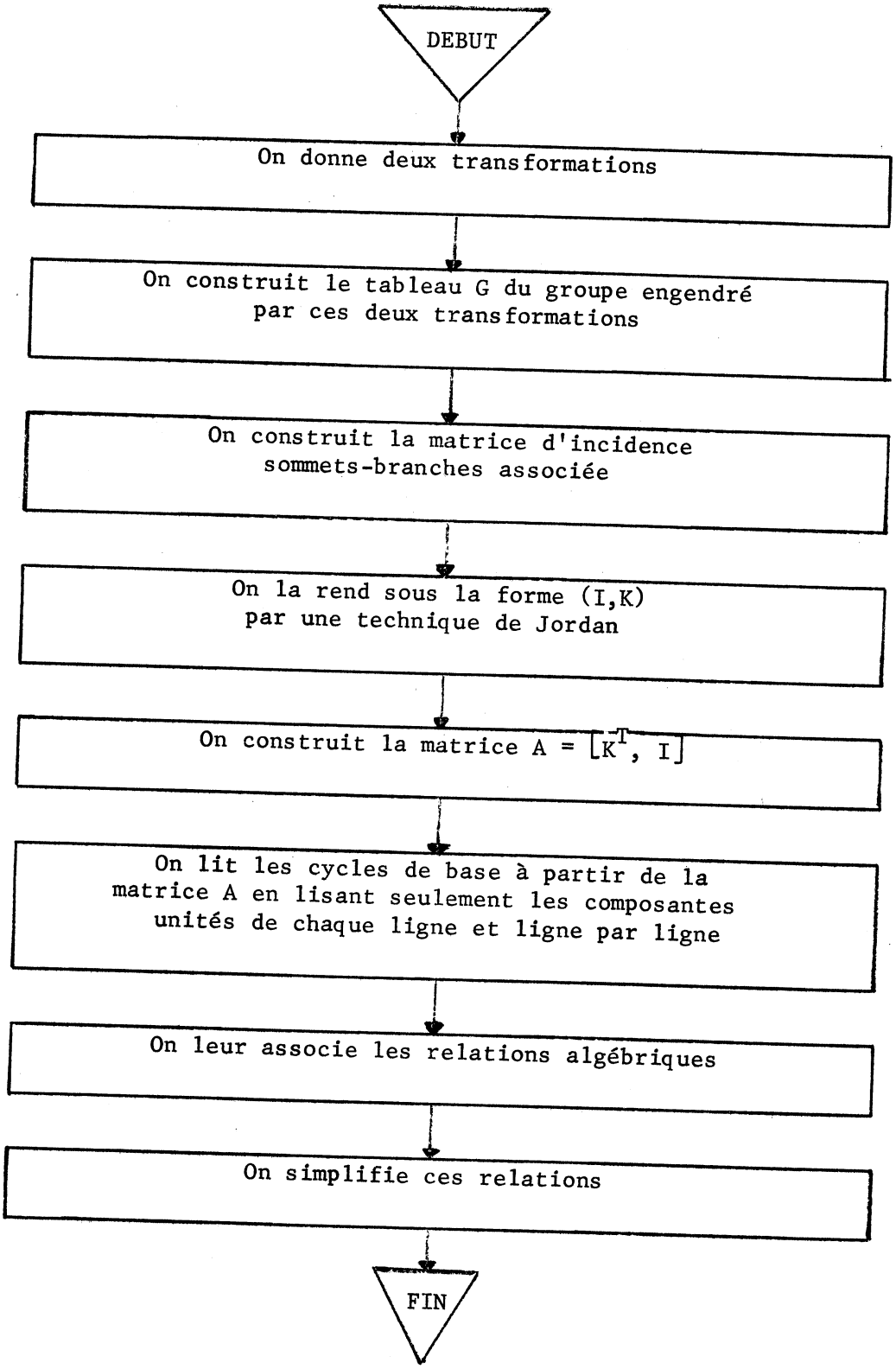
et $$W' = y \, x^{4} \, y \, x^{4} \, y \, x^{4} \, y \, x^{8} \, y^{-1} \, x^{-4} \, y^{-1} \, x^{-8} \, y \, x^{4} \, y \, x^{4} \, y \, x^{4} \, y \, x^{-8} \, y^{-1} \, x^{-4} \, y^{-1} \, x^{8}$$

(d) : $d_x(W')/d_x(R') = 16/0$ n'est pas divisible et par suite on conclut que le mot W' (donc W) n'est pas équivalent à 1.

3. LANGAGE DE DYCK.

Pour cet algorithme trivial, nous renvoyons à l'organigramme qui précède le programme.

ORGANIGRAMME DE LA PRESENTATION D'UN GROUPE FINI



```
PRESGRO: PROC OPTIONS(MAIN) ;
  DCL(N,N1,N2,I,J,K,L,C1,C2) DEC FIXED(3) ;
  DCL(ALPHA,G)(N2,N2) DEC FIXED(3) CTL ;
  DCL BETA(N) DEC FIXED(3) CTL ;
  GET DATA(N) ; N2=1 ;
  DO I=2 TO N ; N2=N2*I ; END ;
  ALLOCATE ALPHA,G,BETA ;
  ALPHA,G=0 ;
  DO I=1 TO N ; ALPHA(1,I)=I ; END ;
  DO K=2 TO 3 ;
  GET DATA(N1) ; DO L=1 TO N1 ;
  GET LIST(I,J) ; ALPHA(K,I)=J ;
  END ; END ;
  /* C'EST LA CONSTRUCTION DU PREMIER GENERATEUR ALPHA(K,*) */
  DO I=1 TO N ;
  IF ALPHA(2,I)=0 THEN ALPHA(2,I)=I ;
  IF ALPHA(3,I)=0 THEN ALPHA(3,I)=I ;
  END; G(1,1)=1; N1=2;
  C1,C2=3 ; /* 3 EST LE NOMBRE DES ELEMENTS DU GROUPE AU DEPART */
RECOM: PUT LIST('LE GROUPE N''EST PAS ENCORE DEFINI'); PUT SKIP(3);
  DO I=1 TO C1;
  DO J=1 TO C1;
  IF (I<N1)&(J<N1) THEN GOTO EXTER;
  DO L=1 TO N ;
  BETA(L)=ALPHA(I,ALPHA(J,L)) ;
  END ;
  DO K=1 TO C2;
  DO L=1 TO N;
    IF BETA(L)¬=ALPHA(K,L) THEN GOTO NOUVEAU;
  END;
  G(I,J)=K; GOTO EXTER;
NOUVEAU: ; /* TESTER L'EGALITE POUR UN AUTRE ELEMENT */
  END;
  C2=C2+1;
  DO L=1 TO N; ALPHA(C2,L)=BETA(L); END;
  G(I,J)=C2 ;
EXTER: ;
  END ; END ;
  IF C1 ¬= C2 THEN DO ; N1=C1+1;
  C1=C2 ; GOTO RECOM ; END ;
  /* LE GROUPE G EST DEFINI MAINTENANT PAR UN TABLEAU */
  PUT PAGE ;
  DO K=1 TO C2 ;
  DO I=1 TO N ;
  PUT EDIT('ALPHA(',K,',',I,')=',ALPHA(K,I))(A(6),
  F(2),A(1),F(2),A(2),F(2)) ;
  PUT SKIP ;
```

```
    END ; PUT SKIP(3) ;
    END ; PUT PAGE ;
    /* IMPRESSION DU TABLEAU DU GROUPE */
    PUT EDIT('VOICI LE GROUPE ENGENDRE PAR LES GENERATEURS 2 ET 3 DEFINI
  CI_DESSUS')(A) ; PUT SKIP ;
    PUT EDIT('ET REPRESENTE PAR LE TABLEAU G1')(A) ; PUT SKIP(2) ;
    PUT EDIT('  |')(A(3)) ;
    DO I=1 TO C2 ;
    PUT EDIT('    |')(A(4)) ;
    END ; PUT SKIP ;
    PUT EDIT('G1|')(A(3)) ;
    DO I=1 TO C2 ;
    PUT EDIT(I,'|')(F(3),A(1)) ;
    END ; PUT SKIP ;
    DO I=1 TO C2 ;
    PUT EDIT('__|')(A(3)) ;
    DO J=1 TO C2 ;
    PUT EDIT('___|')(A(4)) ;
    END ; PUT SKIP ; PUT EDIT(I,'|')(F(2),A(1)) ;
    DO J=1 TO C2 ;
     PUT EDIT(G(I,J),'|')(F(3),A(1));
    END ; PUT SKIP ;
    END ;
    PUT EDIT('__|')(A(3));
    DO J=1 TO C2;
      PUT EDIT('___|')(A(4));
    END; PUT SKIP;
  MATINCID: BEGIN; /* CONSTRUCTION DE LA BASE DE CYCLES */
    /* PUIS DEDUCTION DES RELATIONS DEFINIES DU GROUPE ASSOCIE */
    DCL CH1 CHAR(4);
    DCL (L1,L3,L4,I2,J2) DEC FIXED(3) ;
    DCL LIG(2*C2) DEC FIXED(3) ;
    DCL(C3,C4,K1,K2,K3,REF,REP,REP1,J1) DEC FIXED(3) ;
    DCL C5 DEC FIXED(3,2) ;
    DCL MISB(C2,2*C2) DEC FIXED(1) ;
    DCL SOMISB(C2-1,2*C2) DEC FIXED(1) ;
    DCL VECTL(2*C2) DEC FIXED(1) ;
    DCL VECTC(C2-1) DEC FIXED(1) ;
    DCL MATTR(C2+1,2*C2) DEC FIXED(1) ;
    /* MISB EST LA MATRICE D'INCIDENCE SOMMETS BRANCHES */
    /* VECTL RESP. VECTC SONT DEUX MATRICES A UNE DIMENSION */
                  /* POUR MANIPULER MISB PAR UNE TECHNIQUE DU JOURDAN */
    /* MATTR EST LA TRANSPOSEE DE MISB APRES MANIPULATION */
    DCL CA1 CHAR(6) ;
    DCL (CA,CAR1,CAR2) CHAR(240) VARYING ;
   DCL TABCHA(C2+1) CHAR(120) ;
    /* LIG EST UNE MATRICE LIGNE QUI PORTE LES NOMS DES ARCS DU GRAPHE */
```

```
/* ASSOCIE AU GROUPE DEFINI CI_DESSUS DANS L'ORDRE DE LEURS          */
/* APPARITIONS DANS MISB,SOMISB ET MATTR                              */
/* LES ARCS DU GRAPHE ASSOCIE SONT NOMMES DE LA FACON SUIVANTE :      */
/* SOIT UN ARC D'ORIGINE I ET D'EXTREMITE J,SI J=IA ALORS ON LUI      */
/* ATTRIBUERA LE NOM (2*I) SINON,C_A_D. SI J=IB,ON LUI ATTRIBUERA     */
                    /* LE NOM (2*I-1)  */
 C3=2*C2 ; DO I=1 TO C3 ; LIG(I)=I ; END ;
 PUT EDIT('MATRICE LIGNE DES BRANCHES')(A) ;
 PUT SKIP ; /* MATRICE DE L'ORDRE DES ARCS */
 DO I=1 TO C3 ;
 PUT EDIT('LIG(',I,')=',LIG(I))(A(4),F(3),A(2),F(3));
 PUT SKIP ; END ;
 /* RECHERCHE DES CYCLES DE BASE POUR EN TIRER */
       /* LES RELATIONS DEFINIES DU GROUPE */
 MISB=0 ; DO I=1 TO C2 ;
 MISB(I,2*I),MISB(G(I,2),2*I)=1 ;
 MISB(I,2*I-1),MISB(G(I,3),2*I-1)=1;   END; PUT PAGE;
  PUT EDIT('MATRICE D''INCIDENCE SOMMETS BRANCHES MISB')(A);
 CH1='MISB'; L3=C2; L4=C3;
 CALL IMPRIM(CH1,L3,L4,MISB);
 PUT EDIT('ON REMARQUE QUE LA DERNIERE LIGNE EST LA SOMME DES AUTRES
MODULO 2')(A) ; PUT SKIP ;
 PUT EDIT('DONC ON PEUT L''ELIMINER')(A) ;  PUT SKIP ;
 C4=C2-1; I=1;
 DO WHILE(I <= C4) ;   K1=I+1;
RET: IF MISB(I,I)=0 THEN DO ; I2=I+2 ; C5=LIG(I)/2 ;
       IF(CEIL(C5)=FLOOR(C5)) THEN J1=C3 ;
            ELSE J1=C3-1 ;
      DO K=J1 TO I2 BY -2 ;
     IF MISB(I,K)=1 THEN DO ; VECTC=0 ;
        DO L=1 TO C4 ; VECTC(L)=MISB(L,K) ;
        MISB(L,K)=MISB(L,I) ;
        MISB(L,I)=VECTC(L) ;
        END ; L=LIG(I) ; LIG(I)=LIG(K) ; LIG(K)=L ;
        GO TO RET; END;
     ELSE IF K=I2 THEN DO ;
     /* ON PERMUTE LES LIGNES I ET C4 */
    DO J2=1 TO C3 ;
    MISB(I,J2)=MISB(C4,J2) ; END; C4=C4-1 ; GO TO RET ; END ;
            END ;    END ;
        DO J=1 TO C4 ;
        IF (J=I | MISB(J,I)=0) THEN GO TO AUGM ;
 CALL SOMMOD2(MISB(I,*),MISB(J,*),VECTL);
     DO K=1 TO C3 ;
        MISB(J,K)=VECTL(K) ; END ;
AUGM:        ; END ;
     I=I+1 ; END ;        PUT PAGE;
```

```
   PUT EDIT('MATRICE D''INCIDENCE,SOMMETS BRANCHES MISB,MISE SOUS LA FO
ME SUIVANTE')(A) ; PUT SKIP ;
   PUT EDIT('PAR UNE TECHNIQUE DE JORDAN')(A) ; L3=L3-1 ;
   CH1='SOMI'; L3=C2-1;
   DO I=1 TO L3;            DO J=1 TO L4;
         SOMISB(I,J)=MISB(I,J);  END;  END;
   CALL IMPRIM(CH1,L3,L4,SOMISB);
        PUT SKIP ;   DO J1=1 TO C3 ;
     PUT EDIT('LIG(',J1,')=',LIG(J1))(A(4),F(3),A(2),F(3)) ;
     PUT SKIP ; END ;
      MATTR=0 ;
      /* ON VA TRANSPOSER LA MATRICE MISB */
   DO J=1 TO C4 ;
   DO L=C2 TO C3 ;
   MATTR(L-C4,J)=MISB(J,L) ;
   END ; END ;
   DO L=C2 TO C3 ;
      MATTR(L-C4,L)=1 ; END ; PUT PAGE ;
   PUT EDIT('MATRICE TRANSPOSEE DE MISB OU CHACUNE DE SES LIGNES EST UN
ELEMENT')(A) ; PUT SKIP ;
   PUT EDIT('DE LA BASE DE CYCLES')(A) ;
   CH1='MATT'; L3=C2+1; L4=C3;
    CALL IMPRIM(CH1,L3,L4,MATTR);  C4=C2+1;
   /* RECHERCHE DES CYCLES DE BASE */
   PUT EDIT('ON A',C4,'CYCLES DE BASE')(A(4),F(3),X(3),A(14)) ;
     PUT SKIP;
   DO I=1 TO C4 ; CA='' ;          TABCHA(I)=' ' ; I2=1 ;
   DO J=1 TO C3 ;
   IF MATTR(I,J)=0 THEN GOTO AUTRE ;
   CA1=LIG(J) ; K=1 ;
   DO WHILE(SUBSTR(CA1,K,1)=' ') ; K=K+1 ; END ;
   CA=CA || '.' || SUBSTR(CA1,K) ;
AUTRE: ; END ;
   CAR1,CAR2='' ; J,N=2 ; L1=0 ;
   DO WHILE(SUBSTR(CA,N,1)¬='.') ; N=N+1 ; L1=L1+1 ; END ;
      CAR1=CAR1 || SUBSTR(CA,J,L1) ; N2=LENGTH(CAR1) ;
      CAR2=CAR2 || SUBSTR(CA,N) ; N1=SUBSTR(CA,J,L1) ;
      L=0 ; C5=N1/2 ; L3=LENGTH(CAR2) ;
   IF(CEIL(C5)=FLOOR(C5)) THEN DO ; K1=N1/2 ;
   K2=G(K1,2) ; REP=1 ; END ;
   ELSE DO ; K1=(N1+1)/2 ;     K2=G(K1,3) ; REP=0 ;  END ;
   CAR2=CAR2 || '.' ;
   DO WHILE(L<L3) ; J1=K2 ; J2=K1 ; N=J ; L1=0 ;
   DO WHILE(SUBSTR(CAR2,N,1)¬='.') ; N=N+1 ; L1=L1+1 ; END ;
      N1=SUBSTR(CAR2,J,L1) ; C5=N1/2 ;
      IF(CEIL(C5)=FLOOR(C5)) THEN DO ;
   K=N1/2 ;     K3=G(K,2) ; REP1=1 ; END ;
```

```
      ELSE DO ;          K=(N1+1)/2 ;      K3=G(K,3) ; REP1=0 ; END ;
   IF(K2=K | K2=K3 | K1=K | K1=K3) THEN DO ;
     CAR1=CAR1 || SUBSTR(CAR2,1,L1+1) ; L=LENGTH(CAR1)-N2 ;
     CAR2=SUBSTR(CAR2,N) ;
   IF(K=K2 | K2=K3) THEN DO ;
      IF REP=0 THEN TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'BN' ;
               ELSE TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'AN' ;
         END ;
         ELSE DO ;
      IF REP=0 THEN TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'BI' ;
               ELSE TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'AI' ;
       END ; I2=I2+2 ; REP=REP1 ; K2=K3 ; K1= K ; END ;
   ELSE DO ; CA=SUBSTR(CAR2,J,L1) ;
            CAR2=SUBSTR(CAR2,N) || CA || '.' ; END ;
        END ; CA=CAR1 ;
/* K EST L'ORIGINE DE L'ARC QU'ON PARCOURT A CET INSTANT                */
/* K1 EST L'ORIGINE DE L'ARC PRECEDEMMENT PARCOURU                      */
/* K2 EST L'EXTREMITE DE L'ARC PRECEDEMMENT PARCOURU                    */
/* K3 EST L'EXTREMITE DE L'ARC QU'ON PARCOURT A CET INSTANT             */
/* J1 ET J2 SONT COMME K2 ET K1 RESP. L'EXTREMITE ET L'ORIGINE RESP.*/
/*                  DE L'ARC PRECEDEMMENT PARCOURU                      */
/* REP ET REP1 SONT DEUX REPERES QUI ONT LA VALEUR 0 SI L'ARC           */
/* PARCOURU CORRESPOND AU DEUXIEME GENERATEUR ET LA VALEUR 1 SI         */
/* C'EST LE CONTRAIRE . REP REPERE L'ARC PRECEDEMMENT PARCOURU          */
/* PAR CONTRE REP1 REPERE L'ARC PARCOURU A CET INSTANT                  */
      IF(K1=J1 | K1=J2) THEN DO ;
         IF REP=1 THEN TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'AN' ;
         ELSE TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'BN' ;   END;
         ELSE DO ;
         IF REP=1 THEN TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'AI' ;
         ELSE TABCHA(I)=SUBSTR(TABCHA(I),1,I2) || 'BI' ;   END;
   PUT EDIT('LE CYCLE DE BASE',I,'EST LE SUIVANT')
   (A(16),F(3),X(2),A(14)) ;
   PUT SKIP ; PUT EDIT(CA)(A) ; PUT SKIP ;
      END ; PUT SKIP(2) ;
  PUT LIST('LES RELATIONS ALGEBRIQUES ASSOCIEES SONT LES SUIVANTES:');
   DO I=1 TO C4;      PUT SKIP;
       PUT EDIT(I,TABCHA(I))(F(3),X(3),A); END; PUT PAGE;
   PUT EDIT('RELATIONS CORRESPONDANTES DE BASE')(SKIP,A) ;
   PUT SKIP ;     REF=1 ;
   PUT EDIT('LA',REF,
   'RELATION DE L''ENSEMBLE GENERATEUR EST LA SUIVANTE:')
 (A(2),F(3),X(2),A(51)) ; PUT SKIP ;
   PUT EDIT(TABCHA(I))(A) ; PUT SKIP ;
   DO I=2 TO C4 ;     DO J=1 TO REF ;
   IF TABCHA(I)=TABCHA(J) THEN GO TO AUTRECH ;
   END ; REF=REF+1 ;
```

```
   PUT EDIT('LA',REF,
   'RELATION DE L''ENSEMBLE GENERATEUR EST LA SUIVANTE:')
 (A(2),F(3),X(2),A(51)) ; PUT SKIP ;
   PUT EDIT(TABCHA(I))(A) ; PUT SKIP ;
AUTRECH:  ; END;
 PUT EDIT('LE GROUPE EST REPRESENTE PAR LES GENERATEURS A ET B C-A-D. 2
 ET 3')(A) ; PUT SKIP ;
   PUT EDIT('DEFINIS CI_DESSUS ET LES RELATIONS DEFINIES ET ECRITES CI_
DESSUS')(A) ;
            /* PROCEDURE POUR CALCULER LA SOMME MODULO 2 */
 SOMMOD2: PROC(VECT1,VECT2,VECTL);
          DCL(VECT1,VECT2,VECTL)(C3) DEC FIXED(1);
          DCL I1 DEC FIXED(3);
          DO I1=1 TO C3 ;
          VECTL(I1)=VECT2(I1)+VECT1(I1);
          IF VECTL(I1) > 1 THEN VECTL(I1)=0 ;
          END ;
      END SOMMOD2 ;
     /* PROCEDURE POUR IMPRIMER LES TABLEAUX */
IMPRIM: PROC(CH1,L3,L4,TAB);
   DCL CH1 CHAR(4);
   DCL (L3,L4) DEC FIXED(3);
   DCL TAB(L3,L4) DEC FIXED(1);
   PUT SKIP(2);
   PUT EDIT('    |')(A) ;
   DO I=1 TO L4 ; PUT EDIT('  |')(A) ; END ; PUT SKIP ;
   PUT EDIT(CH1||'|')(A);
   DO I=1 TO L4 ;
   PUT EDIT(LIG(I),'|')(F(2),A(1)) ; END ; PUT SKIP ;
   PUT EDIT('----|')(A) ; DO I=1 TO L4 ;
   PUT EDIT('--|')(A) ; END ; PUT SKIP ;
            DO J=1 TO L3 ;
   PUT EDIT(J,'|')(F(4),A(1));
   DO I=1 TO L4 ;
   PUT EDIT(TAB(J,I),'|')(F(2),A(1)) ;
   END ; PUT SKIP ;
   PUT EDIT('----|')(A) ; DO I=1 TO L4 ;
   PUT EDIT('--|')(A) ; END ; PUT SKIP ;     END ;
   END IMPRIM;
   PUT SKIP ;
   PUT LIST('FIN DU TRAITEMENT DU PROBLEME DE LA PRESENTATION D''UN GR
PE FINI');      PUT SKIP;
   END MATINCID;
   END PRESGRO ;
```

LE GROUPE N'EST PAS ENCORE DEFINI

LE GROUPE N'EST PAS ENCORE DEFINI

```
ALPHA( 1, 1)= 1
ALPHA( 1, 2)= 2
ALPHA( 1, 3)= 3

ALPHA( 2, 1)= 2
ALPHA( 2, 2)= 1
ALPHA( 2, 3)= 3

ALPHA( 3, 1)= 3
ALPHA( 3, 2)= 1
ALPHA( 3, 3)= 2

ALPHA( 4, 1)= 3
ALPHA( 4, 2)= 2
ALPHA( 4, 3)= 1

ALPHA( 5, 1)= 1
ALPHA( 5, 2)= 3
ALPHA( 5, 3)= 2

ALPHA( 6, 1)= 2
ALPHA( 6, 2)= 3
ALPHA( 6, 3)= 1
```

VOICI LE GROUPE ENGENDRE PAR LES GENERATEURS 2 ET 3 DEFINIS CI_DESSUS ET REPRESENTE PAR LE TABLEAU G1

G1	1	2	3	4	5	6
1	1	2	3	4	5	6
2	2	1	4	3	6	5
3	3	5	6	2	4	1
4	4	6	5	1	3	2
5	5	3	2	6	1	4
6	6	4	1	5	2	3

MATRICE LIGNE DES BRANCHES

LIG(1)= 1
LIG(2)= 2
LIG(3)= 3
LIG(4)= 4
LIG(5)= 5
LIG(6)= 6
LIG(7)= 7
LIG(8)= 8
LIG(9)= 9
LIG(10)= 10
LIG(11)= 11
LIG(12)= 12

MATRICE D'INCIDENCE SOMMETS BRANCHES MISB

MISB	1	2	3	4	5	6	7	8	9	10	11	12
1	1	1	0	1	0	0	0	0	0	0	1	0
2	0	1	1	1	0	0	0	0	1	0	0	0
3	1	0	0	0	1	1	0	0	0	1	0	0
4	0	0	1	0	0	0	1	1	0	0	0	1
5	0	0	0	0	0	1	1	0	1	1	0	0
6	0	0	0	0	1	0	0	1	0	0	1	1

ON REMARQUE QUE LA DERNIERE LIGNE EST LA SOMME DES AUTRES MODULO 2 DONC ON PEUT L'ELIMINER

MATRICE D'INCIDENCE, SOMMETS BRANCHES MISB, MISE SOUS LA FORME SUIVANTE PAR UNE TECHNIQUE DE JORDAN

SOMI	1	2	3	12	9	6	7	8	5	10	11	4
1	1	0	0	0	0	1	0	0	1	1	0	0
2	0	1	0	0	0	1	0	0	1	1	1	1
3	0	0	1	0	0	0	1	0	1	0	1	0
4	0	0	0	1	0	0	0	1	1	0	1	0
5	0	0	0	0	1	1	1	0	0	1	0	0

LIG(1)= 1
LIG(2)= 2
LIG(3)= 3
LIG(4)= 12
LIG(5)= 9
LIG(6)= 6
LIG(7)= 7
LIG(8)= 8
LIG(9)= 5
LIG(10)= 10
LIG(11)= 11
LIG(12)= 4

MATRICE TRANSPOSEE DE MISB OU CHACUNE DE SES LIGNES EST UN ELEMENT DE LA BASE DE CYCLES

MATT	1	2	3	12	9	6	7	8	5	10	11	4
1	1	1	0	0	1	1	0	0	0	0	0	0
2	0	0	1	0	1	0	1	0	0	0	0	0
3	0	0	0	1	0	0	0	1	0	0	0	0
4	1	1	1	1	0	0	0	0	1	0	0	0
5	1	1	0	0	1	0	0	0	0	1	0	0
6	0	1	1	1	0	0	0	0	0	0	1	0
7	0	1	0	0	0	0	0	0	0	0	0	1

ON A 7 CYCLES DE BASE
LE CYCLE DE BASE 1 EST LE SUIVANT
1.2.9.6
LE CYCLE DE BASE 2 EST LE SUIVANT
3.9.7
LE CYCLE DE BASE 3 EST LE SUIVANT
12.8
LE CYCLE DE BASE 4 EST LE SUIVANT
1.2.3.12.5
LE CYCLE DE BASE 5 EST LE SUIVANT
1.2.9.10
LE CYCLE DE BASE 6 EST LE SUIVANT
2.3.12.11
LE CYCLE DE BASE 7 EST LE SUIVANT

LES RELATIONS ALGEBRIQUES ASSOCIEES SONT LES SUIVANTES:

1 BIANBIAI

2 BIBIBI

3 ANAN

4 BIANBNAIBI

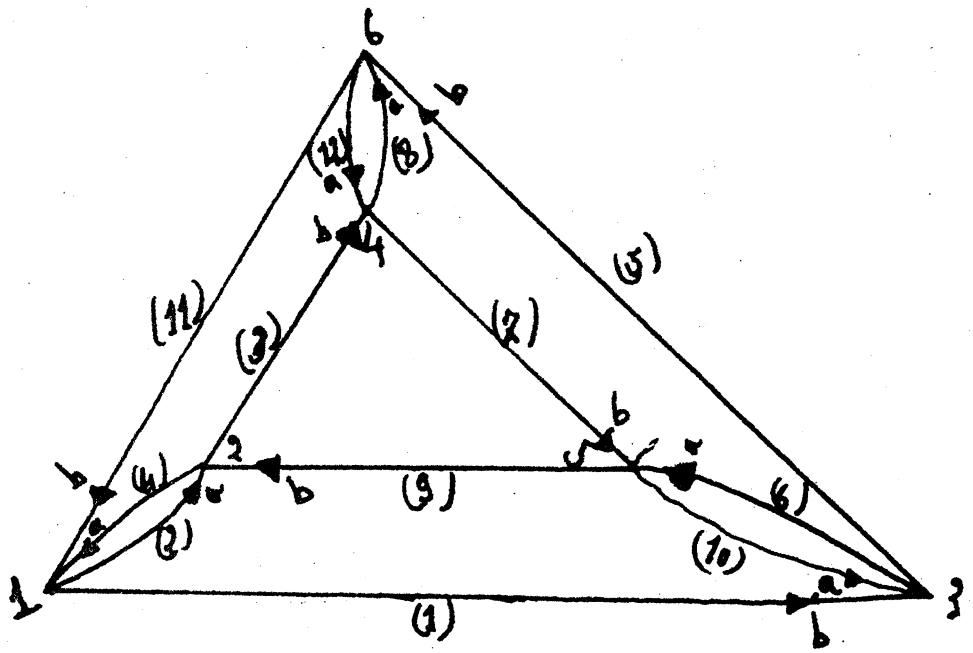
5 BIANBIAN

6 ANBNAIBN

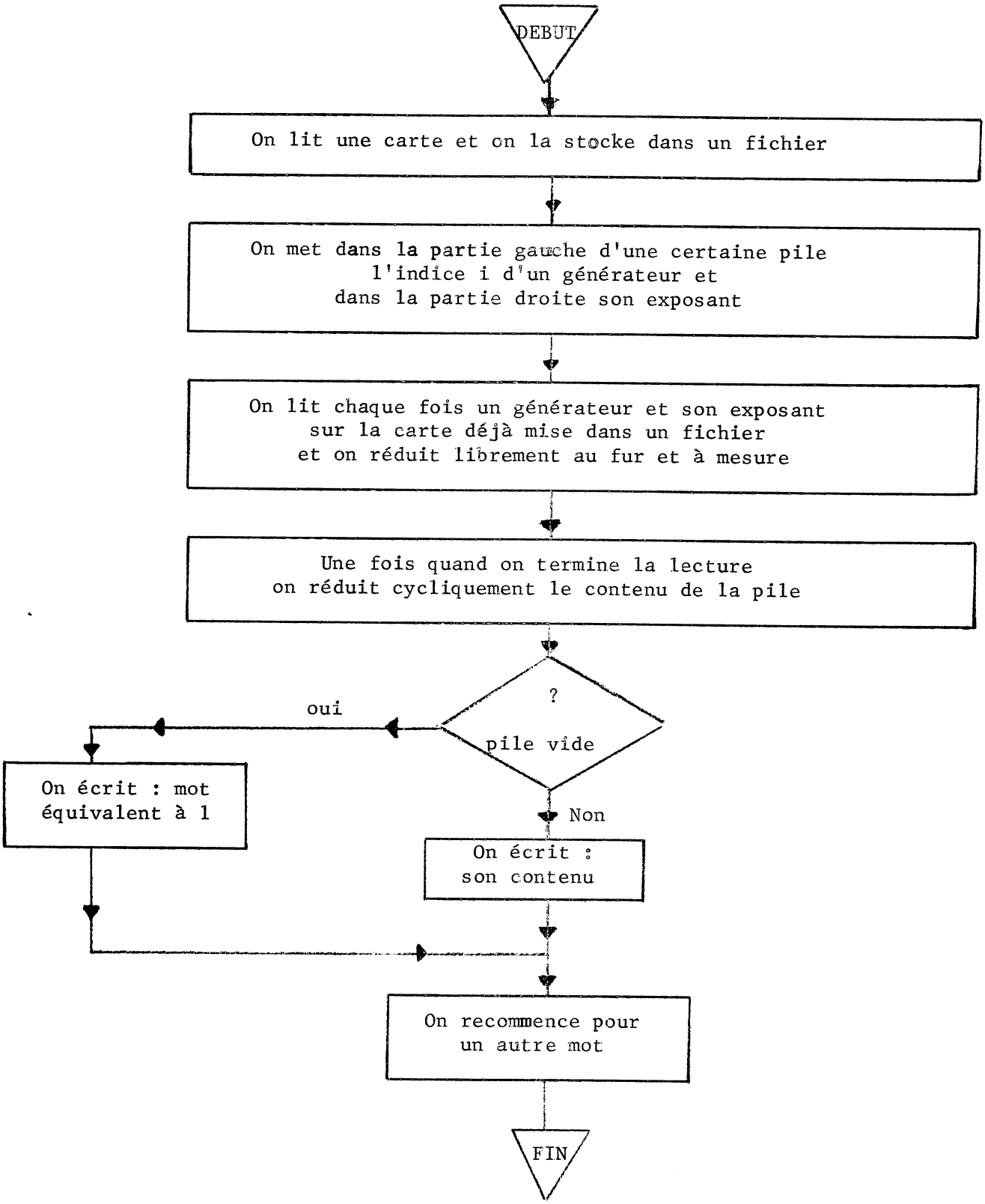
7 ANAN

RELATIONS CORRESPONDANTES DE BASE
LA 1 RELATION DE L'ENSEMBLE GENERATEUR EST LA SUIVANTE:
BIANBIAI
LA 2 RELATION DE L'ENSEMBLE GENERATEUR EST LA SUIVANTE:
BIBIBI
LA 3 RELATION DE L'ENSEMBLE GENERATEUR EST LA SUIVANTE:
ANAN
LA 4 RELATION DE L'ENSEMBLE GENERATEUR EST LA SUIVANTE:
BIANBNAIBI
LA 5 RELATION DE L'ENSEMBLE GENERATEUR EST LA SUIVANTE:
BIANBIAN
LA 6 RELATION DE L'ENSEMBLE GENERATEUR EST LA SUIVANTE:
ANBNAIBN
LE GROUPE EST REPRESENTE PAR LES GENERATEURS A ET B C-A-D. 2 ET 3
DEFINIS CI_DESSUS ET LES RELATIONS DEFINIES ET ECRITES CI_ DESSUS
FIN DU TRAITEMENT DU PROBLEME DE LA PRESENTATION D'UN GROUPE FINI

Graphe associé à cet exemple



ORGANIGRAMME DE LA REDUCTION LIBRE (LANGAGE DE DYCK)



```
00R2  STMT   SOURCE STATEMENT

        3 ANALYZ   OSECT
        4          USING ANALYZ,12
000C    5          STM   14,12,12(13)
        6          LR    2,13
        7          LR    12,15
04FC    8          LA    13,SAVE
0008    9          ST    13,8(0,2)
0004   10          ST    2,4(0,13)
       11          OPEN  (LECTUR,INPUT,ECRIT,OUTPUT)
0750   12 NCH      L     4,=A(PILE)
       20          LR    5,4
0778   21          SH    5,=H'2'
0754   22          L     6,=A(FINPILE)
       23          GET   LECTUR,CARTE
0105   28          LA    2,CARTE
       29          MVI   ERRSW,X'0'
       30          MVI   LIGNE,C' '
025B   31          MVC   LIGNE+1(79),LIGNE
       32 NOUVEL   CLI   0(2),C' '
0060   33          BNE   NVIRGL
0186   34          BAL   11,AVANC2
0050   35          B     NOUVEL
       36 NVIRGL   CLI   0(2),C','
C38C   37          BE    IMP
0104   38          BAL   11,DECBIN
0758   39          C     1,=F'1'
00FC   40          BL    ERRCH
075C   41          C     1,=F'256'
00FC   42          BH    ERRCH
       43          BCTR  1,0
0610   44          STC   1,NUMSYMB
       45          CLI   0(2),C','
00EC   46          BE    EXPOS1
       47          CLI   0(2),C' '
00EC   48          BE    EXPOS1
       49          CLI   0(2),C'+'
00AA   50          BE    EXPOPLUS
       51          CLI   0(2),C'-'
00CA   52          BE    EXPOMOIN
0174   53          BAL   11,ERRPRT
0164   54          B     FBLANC

0186   56 EXPOPLUS BAL   11,AVANC2
0104   57          BAL   11,DECBIN
0758   58          C     1,=F'1'
00F4   59          BL    ERREXP
0760   60          C     1,=F'127'
00F4   61          BH    ERREXP
0611   62          STC   1,EXPVAL
02E0   63          B     EXPOS
0186   64 EXPOMOIN BAL   11,AVANC2
0104   65          BAL   11,DECBIN
0758   66          C     1,=F'1'
00F4   67          BL    ERREXP
```

```
ERR2  STMT   SOURCE STATEMENT

0764    63           C     1,=F'128'
00F4    69           BH    ERREXP
        70           LCR   1,1
0611    71           STC   1,EXPVAL
02E0    72           B     EXPCS
        73 EXPCS1    MVI   EXPVAL,X'1'
02E0    74           B     EXPCS
0174    75 ERREXP    BAL   11,ERRPRT
0164    76           B     FBLANC
0174    77 ERRCH     BAL   11,ERRPRT
0164    78           B     FBLANC
0154    79 DECBIN    ST    11,DECBRETA
        80           SR    3,3
        81           CLI   0(2),C'0'
015C    82           BL    ERCHIF
        83           CLI   0(2),C'9'
015C    84           BH    ERCHIF
0000    85           IC    3,0(2)
0158    86           S     3,F0

0186    88 DECBOUC   BAL   11,AVANC2
        89           CLI   0(2),C'0'
014A    90           BL    DECBRET
        91           CLI   0(2),C'9'
014A    92           BH    DECBRET
077A    93           MH    3,=H'10'
        94           SR    1,1
0000    95           IC    1,0(2)
0158    96           S     1,F0
        97           AR    3,1
0122    98           B     DECBOUC


       100 DECBRET   LR    1,3
0154   101           L     11,DECBRETA
       102           BR    11
       103 DECBRETA  DS    F
       104 F0        DC    A(X'F0')
0174   105 ERCHIF    BAL   11,ERRPRT
0164   106           B     FBLANC
       107 FBLANC    CLI   0(2),C' '
0050   108           BE    NOUVEL
0186   109           BAL   11,AVANC2
0164   110           B     FBLANC
0768   111 ERRPRT    L     3,=A(LIGNE-CARTE)
       112           AR    3,2
       113           MVI   0(3),C'$'
       114           OI    ERRSW,X'FF'
       115           BR    11
       116 ERRSW     DS    C

0001   117 AVANC2    LA    2,1(2)
076C   118           C     2,=A(CARTE+72)
       119           BLR   11
       120           PUT   ECRIT,LIGNE1
```

```
CDR2  STMT   SCURCE STATEMENT

C700   125           CLC   LIGNE(80),=CL80' '
01C0   126           BE    AVANCG
       127           PUT   ECRIT,LIGNE2
       132           MVI   LIGNE,C' '
C25B   133           MVC   LIGNE+1(79),LIGNE
       134 AVANCG    GET   LECTUR,CARTE
C1D5   139           LA    2,CARTE
       140           BR    11
       141 LIGNE1    DC    C' '
       142 CARTE     DS    CL80
       143           DC    53C' '
       144 LIGNE2    DC    C' '
       145 LIGNE     DS    CL80
       146           DC    53C' '
       147 EXPCS     TM    ERRSW,X'FF'

C050   149           BNZ   NOUVEL
       150           CR    5,4
036E   151           BL    MISENPIL
C000   152           CLC   NUMSYMB,0(5)
0352   153           BNE   TMISEN
C611   154           IC    1,EXPVAL
0018   155           SLL   1,32-8
0018   156           SRA   1,32-8
0001   157           IC    3,1(5)
0018   158           SLL   3,32-8
0018   159           SRA   3,32-8
       160           AR    1,3
0760   161           C     1,=F'127'
0330   162           BH    EXPOVER
0770   163           C     1,=F'-128'
C330   164           BL    EXPOVER
       165           LTR   1,1
C34A   166           BNZ   REMPLAC
C778   167           SH    5,=H'2'
C384   168           B     IMPRET
       169 EXPOVER   PUT   ECRIT,=CL133' DEBORDE EXP'
C174   174           BAL   11,ERRPRT
       175           OI    ERRSW,X'FF'
C050   176           B     NOUVEL
0001   177 REMPLAC   STC   1,1(5)
C384   178           B     IMPRET
       179 TMISEN    CR    5,6
C36E   180           BL    MISENPIL
       181 PILDEB    PUT   ECRIT,=CL133' PILE DEBORDEE'
       186           OI    ERRSW,X'FF'
C050   187           B     NOUVEL
C002   188 MISENPIL  LA    5,2(,5)
       189           CR    5,6
C358   190           BNL   PILDEB
C610   191           MVC   0(1,5),NUMSYMB
C611   192           MVC   1(1,5),EXPVAL
C43E   193 IMPRET    BAL   11,IMPILE
C050   194           B     NOUVEL
```

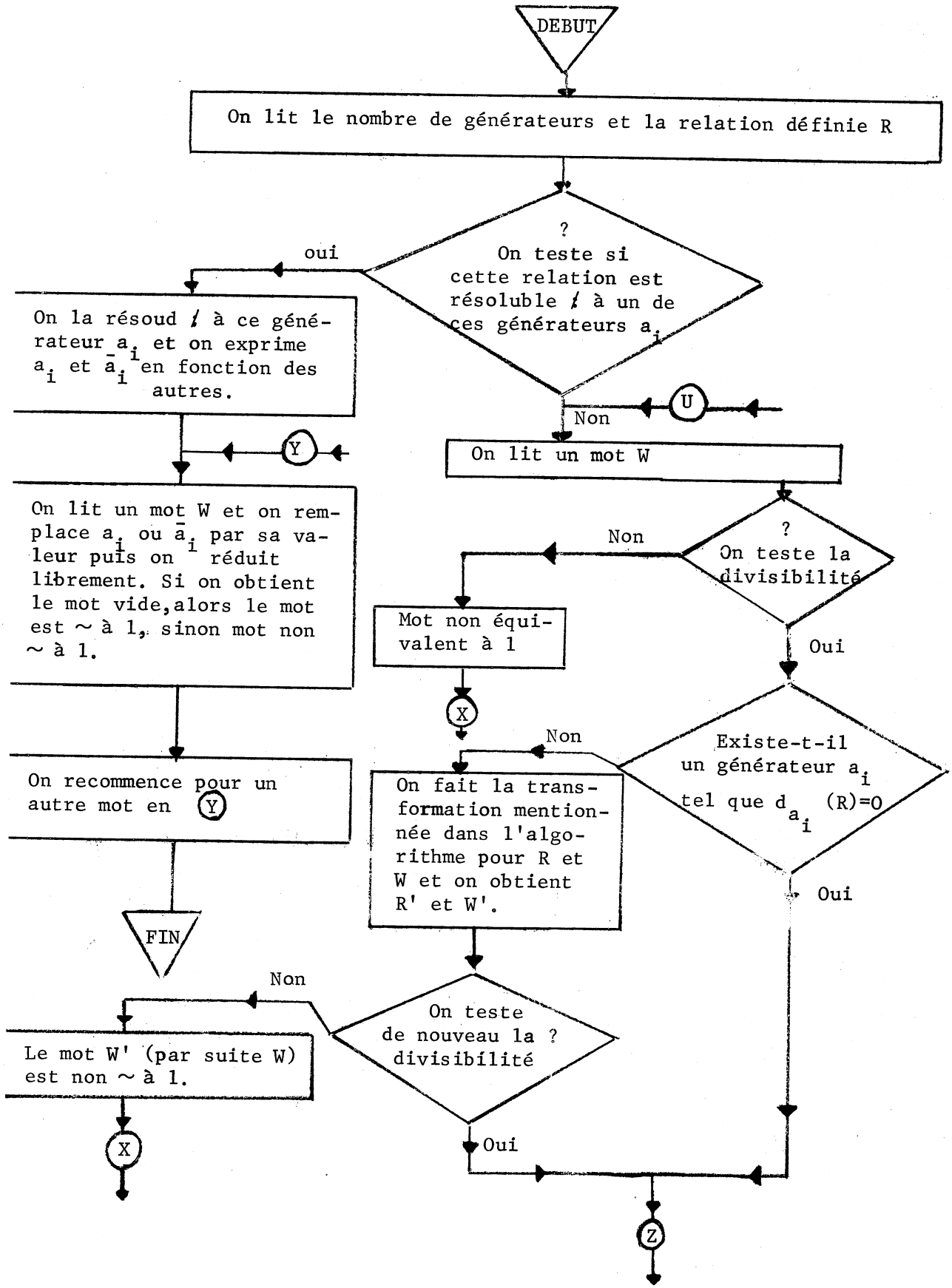
```
DDRS  STMT   SOURCE STATEMENT

        195 IMP       TM    ERRSW,X'FF'
041A    196           BNZ   ERRCHAN
        197 IMPRED    LR    3,5

0778    199           SH    3,=H'2'
        200           CR    3,4
0404    201           BNH   IMPEND
0000    202           CLC   0(1,4),0(5)
0404    203           BNE   IMPEND
        204           SR    1,1
        205           SR    3,3
0001    206           IC    1,1(4)
0018    207           SLL   1,32-8
0018    208           SRA   1,32-8
0001    209           IC    3,1(5)
0018    210           SLL   3,32-8
0018    211           SRA   3,32-8
        212           AR    1,3
0760    213           C     1,=F'127'
03EE    214           BH    IMPDEB
0770    215           C     1,=F'-128'
03FE    216           BL    IMPDEB
0001    217           STC   1,1(4)
0778    218           SH    5,=H'2'
        219           LTR   1,1
0404    220           BNZ   IMPEND
0002    221           LA    4,2(,4)
0394    222           B     IMPRED
        223 IMPDEB    PUT   ECRIT,=CL133' DEB EXP'
        228           OI    ERRSW,X'FF'
0022    229           B     NCH
        230 IMPEND    PUT   ECRIT,LIGNE1
043E    235           BAL   11,IMPILE
0022    236           B     NCH
        237 ERRCHAN   PUT   ECRIT,LIGNE2
        242           PUT   ECRIT,=CL133' ERREUR DANS CHAINE '
        247           OI    ERRSW,X'FF'
0022    248           B     NCH
05E0    249 IMPILE    STM   0,11,IMPSAVE
        250           CR    5,4
04E6    251           BL    IMPVIDE
        252           LR    7,4
        253           MVI   LIGNOUT,C' '
0544    254           MVC   LIGNOUT+1(132),LIGNOUT
0545    255           LA    9,LIGNOUT+1
        256 IMPN      SR    1,1
0000    257           IC    1,0(7)
0001    258           LA    1,1(0,1)
05D0    259           CVD   1,SYMBD
        260           SR    1,1
0001    261           IC    1,1(7)
0018    262           SLL   1,32-8
0018    263           SRA   1,32-8
05D8    264           CVD   1,EXPD
```

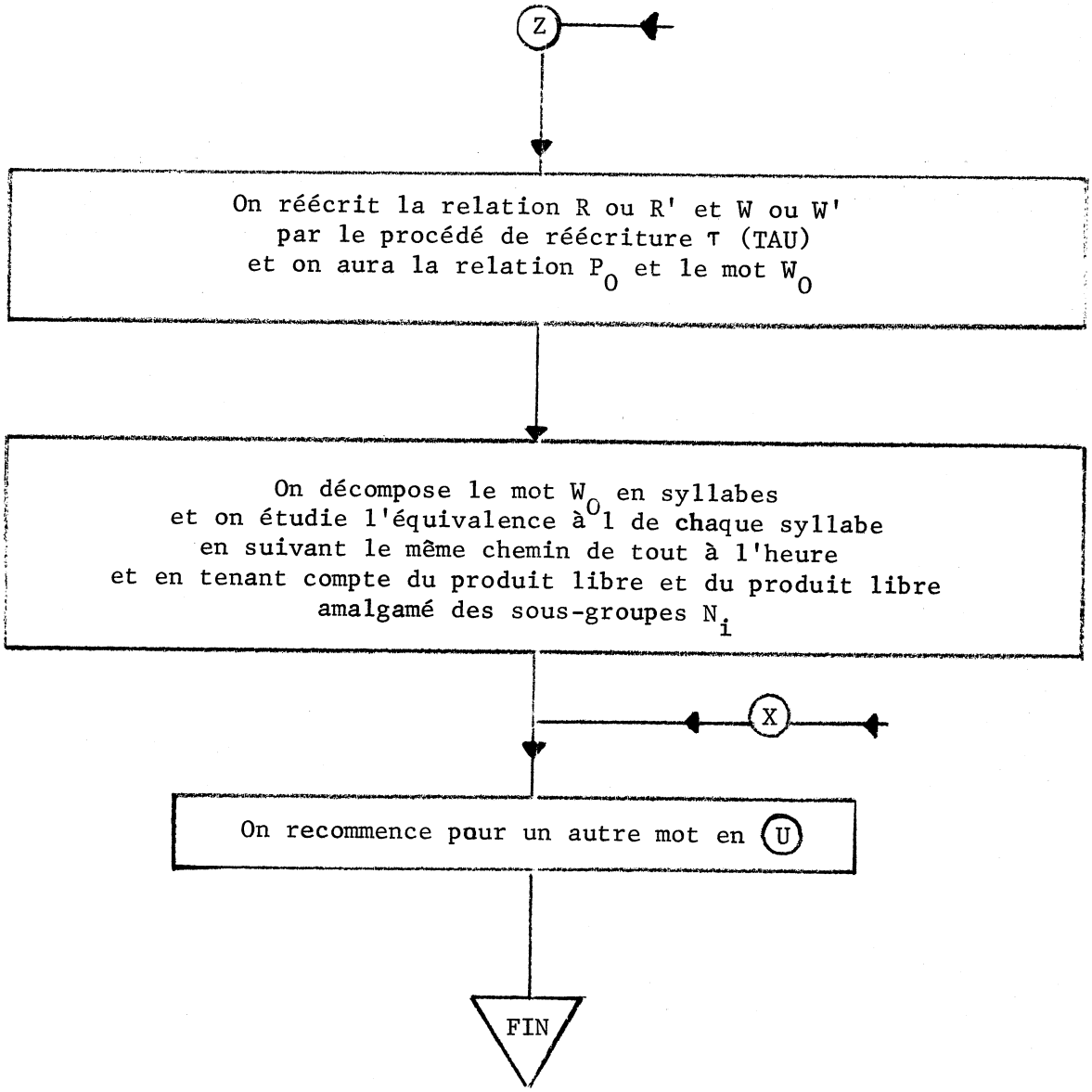
```
DR2  STMT   SOURCE STATEMENT

      265           MVI   0(9),C'A'
5B6   266           UNPK  1(3,9),SYMBD+6(2)
      267           OI    3(9),X'F0'
      268           MVI   4(9),C'+'
      269           LTR   1,1
4A4   270           BNM   *+8
      271           MVI   4(9),C'-'
5DE   272           UNPK  5(3,9),EXPD+6(2)
      273           OI    7(9),X'F0'
      274           MVI   8(9),C' '
C09   275           LA    9,9(0,9)
774   276           C     9,=A(LIGNOUT+133-9)
4CA   277           BL    IMPNXT
      278           PUT   ECRIT,LIGNOUT
545   283           LA    9,LIGNOUT+1
      284           MVI   LIGNOUT,C' '
544   285           MVC   LIGNOUT+1(132),LIGNOUT
002   286 IMPNXT    LA    7,2(,7)
      287           CR    7,5
458   288           BNM   IMPN
      289           PUT   ECRIT,LIGNOUT
4F4   294           B     IMPPRET
      295 IMPVIDE   PUT   ECRIT,=CL133' MOT EQUIVALENT A 1'
5E0   300 IMPPRET   LM    0,11,IMPSAVE
      301           BR    11
      302 SAVE      DS    18F
      303 LIGNOUT   DS    CL133
      304 SYMBD     DS    D
      305 EXPD      DS    D
      306 IMPSAVE   DS    12F
      307 NUMSYMB   DS    C
      308 EXPVAL    DS    C
      309 LECTUR    DCB   DDNAME=LECT,LRECL=80,EODAD=FININ,DSORG=PS,
                          BLKSIZE=80,MACRF=GM,RECFM=F
      365 ECRIT     DCB   DDNAME=IMPRI,LRECL=133,BLKSIZE=133,RECFM=FA,DSORG=PS
                          MACRF=PM
      421 FININ     PUT   ECRIT,=CL133' FIN DE FICHIER EN ENTREE'
      426           CLOSE (LECTUR,,ECRIT)
004   434           L     13,4(,13)
01C   435           LM    2,12,28(13)
00C   436           L     14,12(,13)
      437           BR    14


      439           LTORG
      440                 =CL80' '
      441                 =A(PILE)
      442                 =A(FINPILE)
      443                 =F'1'
      444                 =F'256'
      445                 =F'127'
      446                 =F'128'
      447                 =A(LIGNE-CARTE)
      448                 =A(CARTE+72)
      449                 =F'-128'
```

```
450                     =A(LIGNOUT+133-9)
451                     =H'2'
452                     =H'10'
453                     =CL133' DEBORDE EXP'
454                     =CL133' PILE DEBORDEE'
455                     =CL133' DEB EXP'
456                     =CL133' ERREUR DANS CHAINE '
457                     =CL133' MOT EQUIVALENT A 1'
458                     =CL133' FIN DE FICHIER EN ENTREE'

460 PILE        DC      1000F'0'
461 FINPILE     EQU     *
462             END
```

```
AC05+CC1
MOT EQUIVALENT A 1
5 5-1,
MOT EQUIVALENT A 1
A001+CC1
1 ,
A001+CC1
A004+CC1
AC04+CC3
AC04+CC2
4 4+2 4-1,
AC04+CC2
AC04+CC1
AC04+CC1 AC05-001
AC04+CC1 AC05-001 A004-CC1
4 5-1 4-1,
AC05-CC1
A001+CC1
AC01+CC1 AC02+001
A001+CC1 AC02+001 A003+001
1 2 3 ,
A001+CC1 A002+001 A003+001
AC12-CC1
MOT EQUIVALENT A 1
12-1 12 ,
MOT EQUIVALENT A 1
A012+CC1
MOT EQUIVALENT A 1
12 12-1,
MOT EQUIVALENT A 1
A012+CC1
AC12+CC1 A001+002
AC12+CC1 A001+002 A012-CC1
12 1+2 12-1,
A001+CC2
A001+CC1
A001+CC1 AC11-001
A001+CC1 AC11+001
AC01+CC1 AC11+001 A001-001
1 11-1 11+2 1-1,
A011+CC1
A012+CC1
A012+CC1 A013+002
A012+CC1 AC13-003
A012+CC1 AC13-003 A001-CC2
A012+CC1 AC13-003 A001-005
A012+CC1 AC13-003 A001-005 A012-CC4
12 13+2 13-5 1-2 1-3 12-4,
A012-CC3 A013-003 A001-005
FIN DE FICHIER EN ENTREE
```

ORGANIGRAMME DE LA RESOLUTION DU PROBLEME DU MOT D'UNE PRESENTATION D'UN GROUPE AVEC UNE SEULE RELATION DEFINIE




Z
On réécrit la relation R ou R' et W ou W'
par le procédé de réécriture τ (TAU)
et on aura la relation P_0 et le mot W_0
On décompose le mot W_0 en syllabes
et on étudie l'équivalence à 1 de chaque syllabe
en suivant le même chemin de tout à l'heure
et en tenant compte du produit libre et du produit libre
amalgamé des sous-groupes N_i
X
On recommence pour un autre mot en U
FIN

```
PROEQUI: PROC OPTIONS(MAIN);
 /* ************************************************************ */
 /* *                                                          * */
 /* *    CE PROGRAMME PERMET D'ANALYSER LE PROBLEME DU MOT.    * */
 /* *                                                          * */
 /* ************************************************************ */
      DCL (TA(N),PR(N),PW(N)) DEC FIXED(3) CTL;
      DCL (CAR,CUI,CUN,CARR) CHAR(240) VARYING;
      DCL (CA,RO) CHAR(240) VARYING;
      DCL (C1,C2) CHAR(6);
      DCL (CC1,CC2) CHAR(3);
      DCL CU CHAR(8);
      DCL (R,I,J,K,L,M,N,IN,EX) DEC FIXED(3);
      DCL (MAXO,MINO,KO,U,I1,RP1) DEC FIXED(3);
      DCL (MAX2,MAX1,MIN2,MIN1,K2,K1) DEC FIXED(3);
      DCL (TI(MINO:MAXO),TN(MINO:MAXO)) DEC FIXED(3) CTL;
      DCL (TAO(MIN2:MAX2),PRO(MIN2:MAX2)) DEC FIXED(3) CTL;
      DCL REMP ENTRY RETURNS(CHAR(240) VARYING);
      DCL LIBRED ENTRY RETURNS(CHAR(240) VARYING);
      DCL APP ENTRY RETURNS(CHAR(240) VARYING);
      DCL REMPEQUI ENTRY RETURNS(CHAR(240) VARYING);
      DCL (NR,L3) DEC FIXED(3);
 /* ************************************************************ */
 /* *                                                          * */
 /* *                LECTURE DE LA RELATION R.                 * */
 /* *                                                          * */
 /* ************************************************************ */
      GET DATA(NR) COPY; PUT SKIP;
      DO L3=1 TO NR;
      CAR='';
      GET DATA(N) COPY;/*LECTURE DU NOMBRES**** */
      PUT SKIP;        /*DE GENERATEURS*****   */
      ALLOCATE TA,PR,PW;
      PR,TA=0;
LECTR: GET LIST(IN,EX) COPY;/*LECTURE DE LA */
      PUT SKIP;            /*RELATION R */
      DO WHILE(EX¬=0);
      C1=IN;C2=EX;CC1=SUBSTR(C1,4,3);
      CC2=SUBSTR(C2,4,3);
      CAR=CAR||' '||'A'||CC1||CC2;
      PR(IN)=PR(IN)+EX;
      TA(IN)=TA(IN)+1; GO TO LECTR;
      END;
      CARR=CAR; /*CONSERVATION DE LA RELATION R*/
      PUT LIST('*FIN DE LECTURE DE LA RELATION R***');
      PUT SKIP;PUT LIST(CAR);PUT SKIP;
      DO I=1 TO N;
```

```
        PUT EDIT('PR(',I,') =',PR(I))(R(ETIQ2));
ETIQ2:   FORMAT(X(2),A(3),F(2),A(3),F(2));
        PUT EDIT('TA(',I,') =',TA(I))(R(ETIQ2));
        PUT SKIP;
        END;
        CALL MAXMIN(CAR,MAXO,MINO,KO);
        PUT LIST(MAXO,MINO,KO); PUT SKIP;
        CALL EXIUG(TA,PR,MAXO,MINO,K,R);/*TEST SUR L UNICITE */
        PUT EDIT('K=',K,'R=',R)(A(2),F(2),X(2),A(2),F(2));  PUT SKIP
        IF ABS(K) ¬= 1 THEN GO TO TEST2;
        ELSE DO; /*EXPRIMER LE GEN.UNIQ.EN FCT.DAU*/
        U=0; ALLOCATE TI,TN;
        TI,TN=0;
        CALL EXPR(CAR,CUI,CUN,CU);
        PUT EDIT('CUI=',CUI)(A(4),A);   PUT SKIP;
        PUT EDIT('CUN=',CUN)(A(4),A);
        END;
        GET DATA(M) COPY;/*LECTURE DU NOMBRE */
        PUT SKIP;          /*  DES MOTS DONNES */
        DO J=1 TO M;
        PW=0;  /*INITIALISATION DU TABLEAU PW A 0 */
        CAR='';/*INITIALISATION DE CAR A VIDE     */
        PUT EDIT('LECTURE DU MOT W',J)(A(16),X(2),
        F(1));
 /* ************************************************************ */
 /* *                                                          * */
 /* *                    LECTURE DU MOT W.                     * */
 /* *                                                          * */
 /* ************************************************************ */
LECTW: GET LIST(IN,EX) COPY;/*LECTURE D UN MOT W */
        PUT SKIP;
        DO WHILE(EX¬=0);
        IF IN=R THEN DO;
        IF EX<0 THEN DO;
        EX=ABS(EX);
        DO I=1 TO EX;
        CAR=CAR||CUI;
        END;
        DO L=1 TO N-1;
        PW(L)=PW(L)+EX*TI(L);
        END;      END;
        ELSE DO;
        DO I=1 TO EX;
        CAR=CAR||CUN;
        END;
        DO L=1 TO N-1;
        PW(L)=PW(L)+EX*TN(L);
```

```
        END; END;
        GO TO LECTW; END;
        ELSE DO;
        C1=IN;C2=EX;/*CONVERSION EN UNE CHAINE*/
        CC1=SUBSTR(C1,4,3);/*RACCOURCISSEMENT*/
        CC2=SUBSTR(C2,4,3);/*DE CES CHAINES  */
        CAR=CAR||' '||'A'||CC1||CC2;/*CONCATEN*/
        PW(IN)=PW(IN)*EX;GO TO LECTW;
        END;
        END;
        PUT EDIT('***FIN DE LECTURE DU MOT W',J)(A(26),
        F(2)); PUT SKIP;
        PUT LIST(CAR);/*IMPRESSION DU CAR POUR */
        PUT SKIP;      /*TESTER LA MARCHE DU PROGRA*/
        DO I=1 TO N-1;
        IF PW(I)=0 THEN GO TO CONT3;
        ELSE GO TO DEH;
CONT3:  ;
        PUT EDIT('PW(',I,') =',PW(I))(R(ETIQ1));
ETIQ1:    FORMAT(X(2),A(3),F(2),A(3),X(2),F(2));
        PUT SKIP;
        END;
        PUT LIST('MOT PEUT ETRE EQUIVALENT A 1 ');
        PUT SKIP;
        RO=LIBREC(CAR);
        PUT LIST(RO);PUT SKIP;GO TO CONT1;
DEH:    PUT LIST('MOT N,EST PAS EQUIVALENT A 1 MAIS EQUIVALENT A');
        PUT SKIP; RO=LIBREC(CAR);
        PUT SKIP;
CONT1:  ;
        END;
        GO TO REDLI;
 /* ************************************************************** */
 /* *                                                            * */
 /* *    CAS OU LA RELATION R NE CONTIENT PAS UN GENERATEUR      * */
 /* *              A LA PUISSANCE PLUS OU MOINS UN.              * */
 /* *                                                            * */
 /* ************************************************************** */
TEST2: BEGIN;
    DCL (CAR1,CAR2,W) CHAR(240) VARYING AUTO;
        DCL REP DEC FIXED(3) AUTO;
        DCL PWO(MIN1:MAX1) DEC FIXED(3) CTL;
        DCL (L6,R1) DEC FIXED(3);
        DCL DIVISIB ENTRY RETURNS(BIT(1));
        DCL VALLOGI BIT(1);
 /* ************************************************************** */
 /* *                                                            * */
```

```
/* *                          LECTURE DU MOT W.                          * */
/* *                                                                     * */
/* ******************************************************************** */
      U=1;
CALCUL:GET DATA(M) COPY;/*LECTURE DU NOMBRE */
      PUT SKIP;          /*DES MOTS DONNES    */
      DO L6=1 TO M; /* POUR LIRE LES MOTS SUCCESSIFS */
      PW=0;  /*INITIALISATION DU TABLEAU PW A0*/
      CAR='';/*INITIALISATION DE CAR A VIDE    */
      PUT EDIT('LECTURE DU MOT W',L6)(A(16),F(2));
LECTW: GET LIST(IN,EX) COPY;/*LECTURE D UN MOT W */
      PUT SKIP;
      DO WHILE(EX¬=0);
      C1=IN;C2=EX;/*CONVERSION EN UNE CHAINE     */
      CC1=SUBSTR(C1,4,3);/*REDUCTION DE LA LONGU*/
      CC2=SUBSTR(C2,4,3);/*EUR DE CES CHAINES    */
      CAR=CAR||' '||'A'||CC1||CC2;
      PW(IN)=PW(IN)+EX;GO TO LECTW;
      END;
      PUT SKIP;/*FIN DE LECTURE D UN MOT DONNE */
      PUT EDIT('*** FIN DE LECTURE DU MOT W',L6)(A(27),F(2));
      PUT SKIP;
      PUT LIST(CAR); PUT SKIP;
      DO I=1 TO N;
      PUT EDIT('PW(',I,') =',PW(I))(R(ETIQ1));
ETIQ1: FORMAT(X(2),A(3),F(2),A(3),X(2),F(2));
      END;
      PUT SKIP;
      CA=CARR;
      MAX2,MAX1=MAXO;
      MIN2,MIN1=MINO;
      ALLOCATE PRO,PWO;
      VALLOGI=DIVISIB(PR,PW);
      IF VALLOGI='0'B THEN GO TO FIN;
      FREE PRO,PWO;
      PUT SKIP;PUT LIST('LE MOT');PUT SKIP;
      PUT LIST(CAR);PUT SKIP;
      PUT LIST('PEUT ETRE EQUIVALENT A 1');
      PUT SKIP;/*TRAVAIL CONSIDERE A REDUIRE R */
/* ******************************************************************** */
/* *                                                                   * */
/* *                REDUCTION DE LA LONGUEUR DE R.                     * */
/* *                                                                   * */
/* ******************************************************************** */
      IN=MIN2;
TEST3: IF PR(IN)¬=0 THEN DO;
      IF IN<N THEN DO;
```

```
ROEQUI: PROC OPTIONS(MAIN);

          IN=IN+1;GC TC TEST3;
          END;
          REP=1; RO=REMP(CA,PR);
          PUT LIST(RO);PUT SKIP;CA=RO;
          RO=LIBRED(CA);
          CA=RO; I1=MIN2; /* ON MET DANS I1 MIN2 QUAND S(AI) EST¬=0 */
          W=REMP(CAR,PR); CAR=W;
          PUT EDIT('W=',W)(A(2),A); PUT SKIP;
          W=LIBRED(CAR); CAR=W;
          CALL MAXMIN(CA,MAX2,MIN2,K2);
          CALL MAXMIN(CAR,MAX1,MIN1,K1);
           ALLOCATE TAO,PRO,PWO;
          CALL TAPCAL(CA,MAX2,MIN2,TAO,PRO);
          I=3; L=LENGTH(CAR)-5; PWO=0;
          DO WHILE(I<=L);
            IN=SUBSTR(CAR,I,3); EX=SUBSTR(CAR,I+3,3);
            PWO(IN)=PWO(IN)+EX; I=I+8;
          END;
          VALLOGI=DIVISIB(PRO,PWO);
          IF VALLOGI='0'B THEN DO; PUT SKIP;
           PUT LIST('MOT N''EST PAS EQUIVALENT A 1');
          PUT SKIP; GO TC IMPRI; END;
          FREE TAO,PRO,PWO;
          PUT LIST('MOT PEUT_ETRE EQUIVALENT A 1'); PUT SKIP;
          END;
          ELSE DO;I1=IN; /*CN MET DANS I1 IN QD UN S(AI)=0 */
          REP=0;  /* CONSERVATION D UNE TEST POUR CONTINUER LEPROG. */
          END;
          RO=APP(CA);
          PUT EDIT('RO=',RO)(A(3),A);PUT SKIP;CA=RO;
          CALL MAXMIN(CA,MAX2,MIN2,K2);
          PUT EDIT('MAX2=',MAX2)(A(5),F(3));  PUT SKIP;
          PUT EDIT('MIN2=',MIN2)(A(5),F(3));     PUT SKIP;
          PUT EDIT('K2=',K2)(A(3),F(3)); PUT SKIP;
          ALLOCATE TAO,PRO;
          CALL TAPCAL(RO,MIN2,MAX2,TAO,PRO);
          CALL EXIUG(TAO,PRO,MAX2,MIN2,K,R);
          R1=R; /* CONSERVATION DE LA VALEUR INITIALE DE R */
          FREE TAO,PRO;
          PUT EDIT('K=',K,'R=',R)(A(2),F(2),X(2),
          A(2),F(2)); PUT SKIP;
          PUT LIST('PAS DE REMPLACEMENT DANS CE CAS POUR LE MOT W ');
          PUT SKIP;
          W=APP(CAR);
          PUT EDIT('W=',W)(A(2),A); PUT SKIP;
          IF ABS(K)¬=1 THEN GO TO CONT4;
          W=REMPEQUI(W,RO,R1); GO TO IMPRI;
```

```
/* ************************************************************** */
/* *                                                            * */
/* *      ETUDE DE L'EQUIVALENCE A UN, DES SOUS_CHAINES.        * */
/* *                                                            * */
/* ************************************************************** */
CONT4: BEGIN;
       DCL (C,RI,W4) CHAR(240) VARYING;
       DCL (DIFABS,NI,I4,NG,I3,J1,REP1) DEC FIXED(3);
       DCL (I2,MAX,LM,I5,LI) DEC FIXED(3);
       DCL (RMAX2,RMIN2,REFERENC) DEC FIXED(3);
       CALL MAXMIN(RO,MAX2,MIN2,K2);
       RMAX2=MAX2; RMIN2=MIN2;
        NG=1; L=LENGTH(RO)-5;
        DO I3=11 TO L BY 8;
        IN=SUBSTR(RO,I3,3);
        I4=I3-8;
        DO J1=3 TO I4 BY 8;
       IF IN=SUBSTR(RO,J1,3) THEN GO TO SORTIE;
       END;
        NG=NG+1;
       PUT EDIT('NG=',NG)(A(3),F(3)); PUT SKIP;
SORTIE: ;
        END;      DIFABS=ABS(MAX2-MIN2);
REETUDE: L=LENGTH(W)-5; REFERENC=0; /* UNE REFERENCE POUR VOIR SI
ON PEUT TROUVER UNE TRANCHE EQUIVALENTE A 1 */
        W4='';    I3=3;
AUTREMOT: DO WHILE(I3<=L);
        C=''; NI,REP1=0; I2=I3;
         IF REP1=0 THEN DO;
         REP1=1; GO TO MOTINITI;
                    END;
       ELSE DO;
SEPARMOT: IN=SUBSTR(W,I3,3);
         IF ABS(IN-SUBSTR(W,I2,3))<=DIFABS THEN DO;
         I4=I3-8;
        DO J1=I2 TO I4 BY 8;
        IF IN=SUBSTR(W,J1,3) THEN GO TO PASAUGIN;
       END;                                  END;
       ELSE GO TO TESTEQUI;
MOTINITI: NI=NI+1;
PASAUGIN: C=C||SUBSTR(W,I3-2,8); I3=I3+8;
       PUT EDIT('C=',C)(A(2),A); PUT SKIP;
         IF I3<=L THEN GO TO SEPARMOT;
TESTEQUI: IF NI<NG THEN DO;
       PUT EDIT('LA TRANCHE',C,'DU MOT',W,'N,EST PAS EQUIVALENT A 1'
(10),X(2),A,X(2),A(6),X(2),A,X(2),A(24)); PUT SKIP;
       PUT LIST('*****************************************************
```

```
          PUT SKIP;
                          END;
          ELSE DO;
          PUT EDIT('ETUDIER L,EQUIVALENCE A 1 DE LA TRANCHE',C,
     'DU MOT INITIAL',W)(A(39),X(2),A,X(2),A(14),X(2),A); PUT SKIP;
           MAX=SUBSTR(C,3,3); LM=LENGTH(C)-5; J1=11;
           DO WHILE(J1<=LM);
          IN=SUBSTR(C,J1,3);
          IF MAX<IN THEN MAX=IN;
           J1=J1+8;
          END;
           J1=3; LM=LENGTH(RO)-5; RI='';
           DO WHILE(J1<=LM);
           CC2=SUBSTR(RO,J1+3,3);
           IN=SUBSTR(RO,J1,3); IN=IN+(MAX-RMAX2);
          C1=IN; CC1=SUBSTR(C1,4,3);
           RI=RI||' '||'A'||CC1||CC2; J1=J1+8;
          END;
          PUT EDIT('LA RELATION CORRESPONDANTE EST RI=',RI)(A(34),A);
          PUT SKIP;
          PUT LIST('*****************************************************'
          PUT SKIP;
          CALL MAXMIN(RI,MAX2,MIN2,K2);
          I1=MIN2; ALLOCATE TAO,PRO;
          CALL TAPCAL(RI,MIN2,MAX2,TAO,PRO);
          CALL MAXMIN(C,MAX1,MIN1,K1);
          ALLOCATE PWO;
           I5=3; LI=LENGTH(C)-5; PWO=0;
          DO WHILE(I5<=LI);
          IN=SUBSTR(C,I5,3); EX=SUBSTR(C,I5+3,3);
          PWO(IN)=PWO(IN)+EX; I5=I5+8;
          END;
          VALLOGI=DIVISIB(PRO,PWO);
          IF VALLOGI='0'B THEN GO TO SORTIR;
               PUT LIST('MOT PEUT_ETRE EQUIVALENT A 1'); PUT SKIP;
          RI=REMP(RI,PRO); C=REMP(C,PRO);
          RI=LIBRED(RI); C=LIBRED(C);
          RI=APP(RI); C=APP(C);
          FREE TAO,PRO,PWO;
          CALL MAXMIN(RI,MAX2,MIN2,K2);
          CALL MAXMIN(C,MAX1,MIN1,K1);
          ALLOCATE TAO,PRO;
          CALL TAPCAL(RI,MIN2,MAX2,TAO,PRO);
          CALL EXIUG(TAO,PRO,MAX2,MIN2,K,R);  RP1=R;
          IF ABS(K)¬=1 THEN DO;
          PUT LIST('MOT N''EST PAS EQUIVALENT A 1');  PUT SKIP;   END;
          ELSE DO; ALLOCATE PWO;
```

```
(OEQUI: PROC OPTIONS(MAIN);

        C=REMPEQUI(C,RI,PR1);
        IF C='' THEN DO;
        PUT EDIT('LA TRANCHE',C,'EST EQUIVALENTE A 1')(A(12),
A,A(20)); PUT SKIP;
        REFERENC=1;  END;
             END;
        END; GO TO SAUTER;
SORTIR: PUT EDIT('LE MOT',C,' N''EST PAS EQUIVALENT A 1')(A(7),A,
A(26)); PUT SKIP;
        FREE TAO,PRO,PWO;
SAUTER: W4=W4||C;
        PUT EDIT('W4=',W4)(A(3),A); PUT SKIP; L=LENGTH(W)-5; END;
        END AUTREMOT;
        IF REFERENC¬=0 THEN DO;
         W=W4; GO TO REETUDE; END;
        ELSE DO;
                IF W4='' THEN DO;
                PUT LIST('MOT EQUIVALENT A 1'); PUT SKIP; END;
                ELSE DO; PUT LIST('MOT NON EQUIVALENT A 1'); PUT SKIP
             END;    END;
        END CONT4;   GO TO IMPRI;
 /* ************************************************************ */
 /* *                                                          * */
 /* *                  IMPRESSION DE LA CHAINE.                * */
 /* *                                                          * */
 /* ************************************************************ */
FIN:    PUT LIST('LE MOT');PUT SKIP;
        PUT LIST(CAR); PUT SKIP;
        PUT LIST('N EST PAS EQUIVALENT A 1');
         PUT SKIP(2);
IMPRI:  FREE TI,TN,TAO,PRO,PWO;
        END;
 /* ************************************************************ */
 /* *                                                          * */
 /* *     FONCTION PROCEDURE POUR TESTER LA DIVISIBILITE.      * */
 /* *                                                          * */
 /* ************************************************************ */
DIVISIB: PROC(PRO,PWO) RETURNS(BIT(1));
         DCL D1 DEC FIXED(4,2);
         DCL (PGE,PPE) DEC FIXED(3);
         DCL (PRO(MIN2:MAX2),PWO(MIN1:MAX1)) DEC FIXED(3);
         DO I=MIN2 TO MAX2;
            IF PRO(I)=0 THEN DO;
              IF PWO(I)=0 THEN GO TO RETESTER;
              ELSE RETURN('0'B);
                              END;
            ELSE DO;
```

```
:QUI: PROC OPTIONS(MAIN);

                        D1=PWO(I)/PRO(I);
                        PGE=CEIL(D1); /* PLUS GRAND ENTIER */
                        PPE=FLOOR(D1); /* PLUS PETIT ENTIER */
                        IF PGE=PPE THEN GO TO RETESTER;
                        ELSE RETURN('0'B);
                    END;
 RETESTER: ;
          END;
          RETURN('1'B);
          END DIVISIB;
        END TEST2;
 /* ************************************************************** */
 /* *                                                            * */
 /* *                  CHANGEMENT DES DONNEES.                   * */
 /* *                                                            * */
 /* ************************************************************** */
REDLI: PUT SKIP(4);
        PUT LIST('TRAITEMENT DU DEUXIEME PROBLEME,S''IL Y EN A,EN CHANG
ANT LA RELATION'); PUT SKIP;
        END;
 /* ************************************************************** */
 /* *                                                            * */
 /* *    FONCTION PROCEDURE POUR REMPLACER UNE SOUS_CHAINE       * */
 /* *                  PAR UNE AUTRE EQUIVALENTE.                * */
 /* *                                                            * */
 /* ************************************************************** */
REMPEQUI: PROC(CA,RO,RP1) RETURNS(CHAR(240) VARYING);
       DCL (CA,RR,RO,CAR1,CAR2) CHAR(240) VARYING;
       DCL (RP1,L5) DEC FIXED(3);
       DCL PWO(MIN1:MAX1) DEC FIXED(3) CTL;
TEST6: PUT LIST('*****************************************');
        PUT SKIP;
       CALL MAXMIN(CA,MAX1,MIN1,K1);
       PUT LIST(MAX1,MIN1,K1); PUT SKIP;
       ALLOCATE PWO;
       IF ABS(K1)>=ABS(K2) THEN DO;
       J=MAX1-RP1; RR=''; I=3; L=LENGTH(RO)-5;
       DO WHILE (I<=L);
       CC2=SUBSTR(RO,I+3,3);
       IN =SUBSTR(RO,I,3);
       IN=IN+J;C1=IN;CC1=SUBSTR(C1,4,3);
       RR=RR||' '||'A'||CC1||CC2; I=I+8;
       END;
       CALL MAXMIN(RR,MAX2,MIN2,K2);
       MINO=MIN2; MAXO=MAX2;
       ALLOCATE TI,TN;
       ALLOCATE TAO,PRO;
```

```
CALL TAPCAL(RR,MIN2,MAX2,TAO,PRO);
TI,TN=0;
R=RP1+J;
CALL EXPR (RR,CUI,CUN,CU);
PUT EDIT('CUI=',CUI)(A(4),A);PUT SKIP;
PUT EDIT('CUN=',CUN)(A(4),A);PUT SKIP;
PUT EDIT('CU =',CU)(A(4),A); PUT SKIP;
 CAR=CA; L=LENGTH(CAR)-5; I=3;
DO WHILE(I<=L);
IN=SUBSTR(CA,I,3);
IF IN=R THEN DO;
EX=SUBSTR(CA,I+3,3);
CAR1=SUBSTR(CA,1,I-3);
CAR2=SUBSTR(CA,I+6);
IF EX<0 THEN DO;
EX=ABS(EX); CA=CAR1;
L5=LENGTH(CUI);
DO J=1 TO EX;
CA=CA||CUI; I=I+L5;
END;
CA=CA||CAR2; EX=-EX;
END;
ELSE DO;
CA=CAR1;
L5=LENGTH(CUN);
DO J=1 TO EX;
CA=CA||CUN; I=I+L5;
END;
CA=CA||CAR2;
END;
END;
I=I+8; L=LENGTH(CA)-5;
END;
CAR=CA; PUT EDIT('CA=',CA)(A(3),A); PUT SKIP;
I=3;L=LENGTH(CA)-5; PWO=0;
DO WHILE(I<=L);
IN=SUBSTR(CA,I,3);
EX=SUBSTR(CA,I+3,3);
PWO(IN)=PWO(IN)+EX; I=I+8;
END;
CA=LIBRED(CAR);
PUT EDIT('CA=',CA)(A(3),A); PUT SKIP;
L=LENGTH(CA);
IF L¬=0 THEN DO;
FREE PWO,TAO,PRO,TI,TN;
GO TO TEST6; END;
ELSE DO; PUT LIST('MOT EQUIVALENT A 1'); PUT SKIP;
```

```
        FREE TI,TN,TAO,PRO,PWO; RETURN(CA); END;
        END;
        ELSE DO;
        CAR=CA; CA=LIBRED(CAR);
        L=LENGTH(CA);
        IF L=0 THEN DO;
        PUT LIST('MOT EQUIVALENT A 1'); PUT SKIP;
        FREE TI,TN,TAO,PRO,PWO; RETURN(CA); END;
        ELSE DO;
        PUT LIST('MOT N,EST PAS EQUIVALENT A 1 ');
        PUT SKIP(2);
        FREE TI,TN,TAO,PRO,PWO; RETURN(CA);
        END;
        END; RETURN(CA);
        END REMPEQUI;
 /* ************************************************************ */
 /* *                                                          * */
 /* *     PROCEDURE POUR CALCULER LES TABLEAUX TAO ET PRO.     * */
 /* *                                                          * */
 /* ************************************************************ */
TAPCAL: PROC(CA,MIN2,MAX2,TAO,PRO);
        DCL CA CHAR(240) VARYING;
        DCL (MIN2,MAX2) DEC FIXED(3);
        DCL (TAO(MIN2:MAX2),PRO(MIN2:MAX2))  DEC FIXED(3);
        I=3;L=LENGTH(CA)-5;
        TAO,PRO=0;
        DO WHILE(I<=L);
        IN=SUBSTR(CA,I,3);
        EX=SUBSTR(CA,I+3,3);
        PRO(IN)=PRO(IN)+EX;
        TAO(IN)=TAO(IN)+1;
        I=I+8;
        END;
        END TAPCAL;
 /* ************************************************************ */
 /* *                                                          * */
 /* * PROCEDURE POUR TESTER SI ON PEUT EXPRIMER UN GENERATEUR  * */
 /* *          DE LA RELATION EN FONCTION DES AUTRES.          * */
 /* *                                                          * */
 /* ************************************************************ */
EXIUG: PROC(TA4,PR4,MAXO,MINO,K4,R4);
        DCL (MAXO,MINO) DEC FIXED(3);
        DCL (K4,R4) DEC FIXED(3);
        DCL (TA4(MINO:MAXO),PR4(MINO:MAXO)) DEC FIXED(3);
        L,K4,R4=0;
        DO I=MINO TO MAXO;
        IF (TA4(I)=1)&(ABS(PR4(I))=1) THEN DO;
```

```
        IF L=1 THEN DO;
        IF R4>=I THEN PUT LIST('PAS DE CHANGEMENT');
        ELSE DO;
        R4=I;K4=PR4(I);
        END;END;
        ELSE DO;
        L=1;R4=I;K4=PR4(I);
        END; END;
        END;
        END EXIUG;
/* ************************************************************ */
/* *                                                          * */
/* *    CALCUL D'UN GENERATEUR DE R PAR RAPPORT AUX AUTRES.   * */
/* *                                                          * */
/* ************************************************************ */
EXPR:   PROC(CA,CUI,CUN,CU);
        DCL (CA,CUI,CUN) CHAR(240) VARYING;
        DCL CU CHAR(8);
        DCL (C1I,C2I,CAR1,CAR2) CHAR(240) VARYING;
        DCL (L1,L2) DEC FIXED(3) AUTO;
        C1I='';C2I=''; C2=K;C1=R;
        CC1=SUBSTR(C1,4,3);CC2=SUBSTR(C2,4,3);
        CU=' '||'A'||CC1||CC2;
        PUT LIST(CU); PUT SKIP;
        L=INDEX(CA,CU);
        CAR1=SUBSTR(CA,1,L-1);
        CAR2=SUBSTR(CA,L+8);
        L2=LENGTH(CAR2);L1=LENGTH(CAR1);
        IF K=-1 THEN DO;
        CUN=CAR2||CAR1;
        DO I=1 TO L2 BY 8;
        IN=SUBSTR(CAR2,I+2,3);
        EX=-SUBSTR(CAR2,I+5,3);
        C1=IN;C2=EX;
        CC1=SUBSTR(C1,4,3);CC2=SUBSTR(C2,4,3);
        C2I=' '||'A'||CC1||CC2||C2I;
        END;
        DO J=1 TO L1 BY 8;
        IN=SUBSTR(CAR1,J+2,3);
        EX=-SUBSTR(CAR1,J+5,3);
        C1=IN;C2=EX;
        CC1=SUBSTR(C1,4,3);CC2=SUBSTR(C2,4,3);
        C1I=' '||'A'||CC1||CC2||C1I;
        END;
        CUI=C1I||C2I;
        IF U=0 THEN DO I=1 TO N;
        TN(I)=PR(I); TI(I)=-PR(I); END;
```

```
          ELSE DO I=MINO TO MAXO;
          TN(I)=PRO(I); TI(I)=-PRO(I); END;
          END;
          ELSE DO;
          CUI=CAR2||CAR1;
          DO I=1 TO L2 BY 8;
          IN=SUBSTR(CAR2,I+2,3);
          EX=-SUBSTR(CAR2,I+5,3);
          C1=IN;C2=EX;
          CC1=SUBSTR(C1,4,3);CC2=SUBSTR(C2,4,3);
          C2I=' '||'A'||CC1||CC2||C2I;
          END;
          DO J=1 TO L1 BY 8;
          IN=SUBSTR(CAR1,J+2,3);
          EX=-SUBSTR(CAR1,J+5,3);
          C1=IN;C2=EX;
          CC1=SUBSTR(C1,4,3);CC2=SUBSTR(C2,4,3);
          C1I=' '||'A'||CC1||CC2||C1I;
          END;
          CUN=C1I||C2I;
          IF U=0 THEN DO I=1 TO N;
          TN(I)=-PR(I); TI(I)=PR(I); END;
          ELSE DO I=MINO TO MAXO;
          TN(I)=-PRO(I); TI(I)=PRO(I); END;
          END;
          PUT LIST('LE PROBLEME EST EQUIVALENT A UN PROBLEME DE REDUCTION
LIBRE'); PUT SKIP;
          END EXPR;
 /* ************************************************************ */
 /* *                                                          * */
 /* *       FONCTION PROCEDURE POUR METTRE UN GENERATEUR       * */
 /* *                    A LA PUISSANCE ZERO.                  * */
 /* *                                                          * */
 /* ************************************************************ */
REMP:     PROC(CA,PRO) RETURNS(CHAR(240) VARYING);
          DCL CA CHAR(240) VARYING;
          DCL PRO(MIN2:MAX2) DEC FIXED(3);
          DCL (X,Y) DEC FIXED(3);
          DCL L4 DEC FIXED(3);
          J=3;L=LENGTH(CA);
          X,Y=0;DCL (CAR1,CAR2) CHAR(240) VARYING;
          DO WHILE(J<=L);
          IN=SUBSTR(CA,J,3);
          IF IN=MIN2 THEN DO;
          CAR1=SUBSTR(CA,1,J-3);
          CAR2=SUBSTR(CA,J+6);
          EX=SUBSTR(CA,J+3,3);
```

```
          L4=PRO(MAX2)*EX; C2=L4; CC2=SUBSTR(C2,4,3);
           C1=MIN2; CC1=SUBSTR(C1,4,3);
          CA=CAR1||' '||'A'||CC1||CC2||CAR2;
          PUT LIST(CA); PUT SKIP;
          X=X+PRO(MAX2)*EX;
          END;
          ELSE DO;
          IF IN¬=MAX2 THEN GO TO ITER;
          ELSE DO;
          CAR1=SUBSTR(CA,1,J-3);
          CAR2=SUBSTR(CA,J+6);
          EX=SUBSTR(CA,J+3,3);
          IF EX<0 THEN DO;
          C2=PRO(MIN2); CC2=SUBSTR(C2,4,3);
          CA=CAR1;EX=ABS(EX);
          DO I=1 TO EX;
          C1=MIN2; CC1=SUBSTR(C1,4,3);
          CA=CA||' '||'A'||CC1||CC2;
          C1=MAX2; CC1=SUBSTR(C1,4,3);
           CA=CA||' '||'A'||CC1||' -1';
          END;
          CA=CA||CAR2; EX=-EX;
          END;
          ELSE DO;
          C2=-PRO(MIN2); CC2=SUBSTR(C2,4,3);
          CA=CAR1;
          DO I=1 TO EX;
          C1=MAX2; CC1=SUBSTR(C1,4,3);
           CA=CA||' '||'A'||CC1||'  1';
          C1=MIN2; CC1=SUBSTR(C1,4,3);
          CA=CA||' '||'A'||CC1||CC2;
          END;
          CA=CA||CAR2;
          END;
          Y=Y+EX; X=X-PRO(MIN2)*EX; J=J-8+16*ABS(EX);
          END; END;
ITER:     J=J+8; L=LENGTH(CA);
          END;
          I=1; RETURN(CA);
          END REMP;
 /* ***************************************************************
 /* *                                                             *
 /* *       FONCTION PROCEDURE POUR LA REDUCTION LIBRE.           *
 /* *                                                             *
 /* ***************************************************************
LIBRED:PROC(CA) RETURNS(CHAR(240) VARYING);
          DCL CA CHAR(240) VARYING;
```

```
          DCL (IN2,EX2) DEC FIXED(3);
          DCL (CAR1,CAR2) CHAR(240) VARYING AUTO;
          L=LENGTH(CA)-13; I=3;
          DO WHILE(I<=L);
          IN=SUBSTR(CA,I,3);
          IN2=SUBSTR(CA,I+8,3);
          IF IN=IN2 THEN DO;
          EX=SUBSTR(CA,I+3,3);
          EX2=SUBSTR(CA,I+11,3); EX=EX+EX2;
          IF EX=0 THEN DO;
          CAR1=SUBSTR(CA,1,I-3);
          CAR2=SUBSTR(CA,I+14);
          CA=CAR1||CAR2;
          GO TO COM; END;
          C2=EX; CC2=SUBSTR(C2,4,3);
          SUBSTR(CA,I+3,3)=CC2;
          CAR1=SUBSTR(CA,1,I+5);
          CAR2=SUBSTR(CA,I+14);
          CA=CAR1||CAR2;
COM:      L=LENGTH(CA)-13;
          IF I>8 THEN I=I-8;
          ELSE I=3; END;
          ELSE I=I+8;
          END;
          L=L+13;
          DO WHILE(L>3);
          IN=SUBSTR(CA,3,3);
          IN2=SUBSTR(CA,L-5,3);
          IF IN=IN2 THEN DO;
          EX=SUBSTR(CA,6,3);
          EX2=SUBSTR(CA,L-2,3); EX=EX+EX2;
          IF EX=0 THEN DO;
          CA=SUBSTR(CA,9,L-16); GO TO TES;
          END;
          C2=EX; CC2=SUBSTR(C2,4,3);
          SUBSTR(CA,6,3)=CC2;
          CA=SUBSTR(CA,1,L-8);
TES:      L=LENGTH(CA);END;
          ELSE GO TO IMP;
          END;
IMP:      IF L=0 THEN DO;
          PUT LIST('MOT EQUIVALENT A 1'); PUT SKIP;
          END;
          ELSE DO;
          PUT LIST('UNE CHAINE EQUIVALENTE A'); PUT SKIP;
          PUT LIST(CA); PUT SKIP;
          END;
```

```
        RETURN(CA);
        END LIBRED;
  /* ************************************************************ */
  /* *                                                          * */
  /* *     FONCTION PROCEDURE POUR CALCULER LA FONCTION TAU.    * */
  /* *                                                          * */
  /* ************************************************************ */
APP:    PROC(R5) RETURNS(CHAR(240) VARYING);
        DCL (RPO,R5) CHAR(240) VARYING;
        DCL (CH1,CH2) CHAR(3) VARYING;
        DCL CH3 CHAR(3);
        DCL (Z1,Z2) CHAR(240) VARYING;
        DCL (S,INDICE,PUMAXPI1) DEC FIXED(3);
          J=3; L=LENGTH(R5)-5;
        DO WHILE(J<=L);
          IF I1=SUBSTR(R5,J,3) THEN DO;
            PUMAXPI1=SUBSTR(R5,J+3,3);
            GO TO PERMUTA;          END;
        J=J+8;
        END;
PERMUTA: INDICE=J-3; J=J+8;
        DO WHILE(J<=L);
          EX=SUBSTR(R5,J+3,3);
            IF ((I1=SUBSTR(R5,J,3))&(PUMAXPI1<=EX)) THEN DO;
            PUMAXPI1=EX; INDICE=J-3;                    END;
            J=J+8;
        END;
        Z1=SUBSTR(R5,1,INDICE);
        Z2=SUBSTR(R5,INDICE+1);
        R5=Z2||Z1;
        RPO=''; S=0; J=3;
        DO WHILE(J<=L);
        IN=SUBSTR(R5,J,3);
        IF I1=IN THEN DO;
        EX=SUBSTR(R5,J+3,3);
        S=S+EX; END;
            ELSE DO;
            C1=S; CC1=SUBSTR(C1,4,3);CC2=SUBSTR(R5,J+3,3);
            C1=IN; CH3=SUBSTR(C1,4,3); I=1;
        DO WHILE((I<=3)&(SUBSTR(CC1,I,1)=' '));
          I=I+1;
        END; CH2=SUBSTR(CC1,I); I=1;
        DO WHILE((I<=3)&(SUBSTR(CH3,I,1)=' '));
          I=I+1;
        END; CH1=SUBSTR(CH3,I);  CH3=CH1||CH2;
          IN=CH3; C1=IN; CC1=SUBSTR(C1,4,3);
          RPO=RPO||' '||'A'||CC1||CC2;
```

```
DEQUI: PROC OPTIONS(MAIN);

            PUT EDIT('RPO=',RPO)(A(4),A); PUT SKIP;
                END;
            J=J+8;
          END; RETURN(RPO);
          END APP;
 /* ************************************************************** */
 /* *                                                            * */
 /* *  PROCEDURE POUR CALCULER LES INDICES MAXIMUM ET MINIMUM    * */
 /* *                  DE LA CHAINE DONNEE CA.                   * */
 /* *                                                            * */
 /* ************************************************************** */
MAXMIN:PROC(CA,MAX3,MIN3,K3);
          DCL CA CHAR(240) VARYING;
          DCL (MAX3,MIN3,K3) DEC FIXED(3);
          IN=SUBSTR(CA,3,3);
          MAX3,MIN3=IN;
          L=LENGTH(CA)-5; I=11;
          DO WHILE(I<=L);
          IN=SUBSTR(CA,I,3);
          IF MIN3<=IN THEN DO;
          IF MAX3>=IN THEN GO TO CROIS;
          ELSE MAX3=IN;  END;
          ELSE MIN3=IN;
CROIS:    I=I+8;
          END;  K3=MAX3-MIN3+1;
          END MAXMIN;
 /* ************************************************************** */
 /* *                                                            * */
 /* *      FIN DU PROGRAMME ANALYSANT LE PROBLEME DU MOT.        * */
 /* *                                                            * */
 /* ************************************************************** */
          PUT LIST('FIN DU TRAITEMENT DE PROBLEME');
          END PROCEQUI;
```

```
NR=5;

N=2;

1
1,

2
1,

1
2,

2
1,

0
0,

*FIN DE LECTURE DE LA RELATICN R***
 A  1  1 A  2  1 A  1  2 A  2  1
  PR( 1) = 3  TA( 1) = 2
  PR( 2) = 2  TA( 2) = 2
     2                          1                          2
K= 0  R= 0
M=1;

LECTURE DU MOT W 1
1
1,

2
1,

1
-1,

2
-1,

0
0,


*** FIN DE LECTURE DU MOT W 1
 A  1  1 A  2  1 A  1 -1 A  2 -1
  PW( 1) =   C  PW( 2) =   0

LE MOT
 A  1  1 A  2  1 A  1 -1 A  2 -1
PEUT ETRE EQUIVALENT A 1
 A  1  2 A  2  1 A  1  2 A  2  1
 A  1  2 A  2  1 A  1 -3 A  1  4 A  2  1
 A  1  2 A  2  1 A  1 -3 A  1  4 A  2  1 A  1 -3
UNE CHAINE EQUIVALENTE A
 A  1 -1 A  2  1 A  1  1 A  2  1
 A  1  2 A  2  1 A  1 -1 A  2 -1
 A  1  2 A  2  1 A  1 -3 A  1 -2 A  2 -1
W= A  1  2 A  2  1 A  1 -3 A  1 -2 A  1  3 A  2 -1
UNE CHAINE EQUIVALENTE A
```

```
 A  1  2 A  2  1 A  1 -2 A  2 -1
MOT PEUT_ETRE EQUIVALENT A 1
RPO= A 21  1
RPO= A 21  1 A 20  1
RO= A 21  1 A 20  1
MAX2= 21
MIN2= 20
K2=  2
K= 1  R=21
PAS DE REMPLACEMENT DANS CE CAS POUR LE MOT W
RPO= A 22  1
RPO= A 22  1 A 20 -1
W= A 22  1 A 20 -1
*********************************************
    22                          20                          3
 A 22  1
LE PROBLEME EST EQUIVALENT A UN PROBLEME DE REDUCTION  LIBRE
CUI= A 21  1
CUN= A 21 -1
CU = A 22  1
CA= A 21 -1 A 20 -1
UNE CHAINE EQUIVALENTE A
 A 21 -1 A 20 -1
CA= A 21 -1 A 20 -1
*********************************************
    21                          20                          2
 A 21  1
LE PROBLEME EST EQUIVALENT A UN PROBLEME DE REDUCTION  LIBRE
CUI= A 20  1
CUN= A 20 -1
CU = A 21  1
CA= A 20  1 A 20 -1
MOT EQUIVALENT A 1
CA=
MOT EQUIVALENT A 1



TRAITEMENT DU DEUXIEME PROBLEME,S'IL Y EN A,EN CHANGEANT LA RELATION
R=2;

1
2,

2
2,

1
-2,

2
-2,

0
0,

*FIN DE LECTURE DE LA RELATION R***
 A  1  2 A  2  2 A  1 -2 A  2 -2
  PR( 1) = 0  TA( 1) = 2
  PR( 2) = 0  TA( 2) = 2
     2                          1                          2
K= 0  R= 0
M=1;
```

```
LECTURE DU MOT W 1
1
3,

2
3,

1
-3,

2
-3,

0
0,


*** FIN DE LECTURE DU MOT W 1
 A  1  3 A  2  3 A  1 -3 A  2 -3
  PW( 1) =   C  PW( 2) =   0

LE MOT
 A  1  3 A  2  3 A  1  3 A  2 -3
PEUT ETRE EQUIVALENT A 1
RPO= A 22  2



RPO= A 22  2 A 20 -2
RO= A 22  2 A 20 -2
MAX2= 22
MIN2= 20
K2=  3
K= 0  R= 0
PAS DE REMPLACEMENT DANS CE CAS POUR LE MOT W
RPO= A 23  3
RPO= A 23  3 A 20 -3
W= A 23  3 A 20 -3
NG=  2
C= A 23  3
LA TRANCHE   A 23  3  DU MOT   A 23  3 A 20 -3  N,EST PAS EQUIVALENT A 1
**************************************************
W4= A 23  3
C= A 20 -3
LA TRANCHE   A 20 -3  DU MOT   A 23  3 A 20 -3  N,EST PAS EQUIVALENT A 1
**************************************************
W4= A 23  3 A 20 -3
MOT NON EQUIVALENT A 1



TRAITEMENT DU DEUXIEME PROBLEME,S'IL Y EN A,EN CHANGEANT LA RELATION
N=3:

1
2,

2
1,

3
```

```
1,

2
-2,

1
1,

2
1,

0
0,

*FIN DE LECTURE DE LA RELATION R***
 A  1  2 A  2  1 A  3  1 A  2 -2 A  1  1 A  2  1
  PR( 1) = 3  TA( 1) = 2
  PR( 2) = 0  TA( 2) = 3
  PR( 3) = 1  TA( 3) = 1
     3
K= 1  R= 3                    1
 A  3  1                                          3
LE PROBLEME EST EQUIVALENT A UN PROBLEME DE REDUCTION  LIBRE
CUI= A  2 -2 A  1  1 A  2  1 A  1  2 A  2  1
CUN= A  2 -1 A  1 -2 A  2 -1 A  1 -1 A  2  2
M=4;

LECTURE DU MOT W  1
1
1,

2
2,

3
1,

2
2,

0
0,

***FIN DE LECTURE DU MOT W 1
 A  1  1 A  2  2 A  2 -1 A  1 -2 A  2 -1 A  1 -1 A  2  2 A  2  2
MOT N,EST PAS EQUIVALENT A 1 MAIS EQUIVALENT A
UNE CHAINE EQUIVALENTE A
 A  1  1 A  2  1 A  1 -2 A  2 -1 A  1 -1 A  2  4
```

APPENDICE

NOTIONS SUR LA RESOLUTION DU PROBLEME DU MOT POUR UN MONOIDE M [[1] PP. 47]

Soit un monoïde M présenté par un ensemble de générateurs

$E = \{ a_1, a_2, \ldots, a_n \}$ et un ensemble de relations définies

$\Sigma = \{ A_i = B_i$ pour $i = 1, 2, \ldots, m$ où A_i, B_i sont différents du mot vide $\}$.

E est appelé l'alphabet du monoïde. A une telle présentation de monoïde, on associe un groupe G défini par la présentation

$$< E \; ; \; \Sigma >$$

Définition :

1) Soient X et Y deux mots quelconques de M, (X et Y $\in E^*$), et soit $A_i = B_i$ de Σ, alors la transformation du mot $X\,A_i\,Y$ au mot $X\,B_i\,Y$ (ou l'inverse) est appelée transformation élémentaire de M.

$$X\,A_i\,Y \longrightarrow X\,B_i\,Y \text{ ou } X\,B_i\,Y \longrightarrow X\,A_i\,Y.$$

On appelle $X \longrightarrow X$ la transformation identique.

2) Soient X et Y deux mots de E^*, X et Y sont équivalents (et on note $X \equiv Y$) si et seulement si on peut trouver une séquence de transformations élémentaires de M telle que :

$$X = X_0 \longrightarrow X_1 \longrightarrow \ldots \longrightarrow X_i \longrightarrow X_{i+1} \longrightarrow \ldots \longrightarrow X_k = Y.$$

A toute relation $A = B$ de Σ, on associe une paire droite (a,b) (éventuellement réduite à un singleton a = b) où a et b sont les premières lettres respectivement de A et B. De même on associe une paire gauche (a',b') où a' et b' sont les dernières lettres de A et B. Une paire gauche (resp. droite) peut être associée à plus d'une relation de Σ ; pour cela, on va attribuer à chaque paire ainsi associée le nombre de relations définies de Σ qu'elle représente à gauche (resp. à droite).

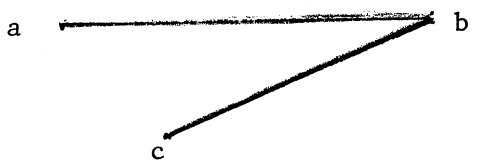
On définit alors deux graphes symétriques, éventuellement avec boucles, gauche et droite. On considère tous les générateurs qui figurent dans les relations définies de Σ comme les sommets de ces deux graphes. Le graphe gauche (resp. droite) a pour branches les paires (a,b) gauches (resp. droites) par analogie avec la théorie des graphes, on dit que Σ n'a pas de cycles gauches (resp. droites) si le graphe gauche (resp. droit) associé n'en a pas ; si Σ n'a ni cycles gauches, ni de cycles droits, on dira que Σ n'a pas de cycles (On considère les boucles comme des cycles).

<u>Exemples</u> :

1) Soit le monoïde présenté par

$$E = \{ a,b,c \} \text{ et } \Sigma = \{ b^2 = c^2a,\ a^2b = b^2c \}.$$

- Les paires gauches sont : (b,c) et (a,b).
  Le graphe gauche associé est le suivant :

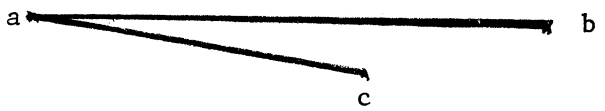


- Les paires droites sont : (b,a) et (b,c).
  Le graphe droit est le suivant :

On remarque qu'ils sont identiques.

2) Soit E = { a,b,c } et soit $\Sigma = \{ab = b,\ ac = c,\ bc = a\}$; on a respectivement comme précédemment :

- (a,b), (a,c) et (b,a) ; d'où le graphe gauche :



- (b,b), (c,c) et (c,a) ; d'où le graphe droit qui a deux boucles.

Adjan [1] a démontré la propriété suivante :

Supposons que le système des relations définies Σ du monoïde M n'a pas de cycles et soient X, Y deux mots par rapport à l'alphabet E de M. Si l'équation $X \equiv Y$ est vraie dans M, alors il est nécessaire et suffisant qu'elle soit vraie dans le groupe

$$G = \langle E \; ; \; \Sigma \rangle.$$

Ensuite, il a démontré que sous ces mêmes conditions, M est plongeable dans le groupe G (c'est-à-dire il existe un isomorphisme entre M et le sous-groupe H de G engendré par l'alphabet positif E seulement).

En particulier, considérons le cas d'un monoïde $[E \; ; \; \Sigma]$ où Σ ne comporte qu'une seule relation.

On dit que la relation définie $A = B$ de Σ est réduite à gauche (resp. à droite) dans M si les premières (resp. les dernières) lettres de A et B sont distinctes. La relation $A = B$ est dite réduite si elle est à la fois réduite à gauche et à droite. Il est évident que $A = B$ (A et B sont différents du vide) sera réduit si et seulement si Σ n'a pas de boucles.

Ainsi, si Σ ne comporte qu'une seule relation définie réduite $A = B$ (A et B sont différents du vide), G sera présenté par

$$\langle E \; ; \; A\overline{B} \rangle .$$

Etant donné que le problème du mot est résolu dans $\langle E \; ; \; A\overline{B} \rangle$ et M est isomorphe à H ; alors, il existe un algorithme pour résoudre le problème du mot d'un monoïde M qui a une seule relation définie réduite $A = B$.

BIBLIOGRAPHIE

[1] - ADJAN S.I.
Defining relations and algorithmic problems for groups and semigroups.
American Mathematical Society, providence, R.I., 1967.

[2] - BENZAKEN C.
Synthèse des réseaux de contact
Université de GRENOBLE, Institut de Mathématiques Appliquées.
Séminaire de Logique 20 Décembre 1965.

[3] - BRITTON J.L.
Solution of the word problem for certain types of Groups I.
Proc. Glasgow Math. Assoc., 3 (1956), 45-54.

[4] - BRITTON J.L.
Solution of the word problem for certain types of Groups II.
Proc. Glasgow Math. Assoc., 3 (1957), 68-90.

[5] - BRITTON J.L.
The word problem for Groups.
Proc. London Math. Soc., 8, No. 32 (3rd series) (1958), 493-506.

[6] - BRITTON J.L.
The word problem
Ann. of Math. (2) 77 (1963), 16-32.

[7] - BOONE WILLIAM W.
The word problem
Ann. of Math. (2) 70 (1959), 207-265.

[8] - BOONE WILLIAM W.
Certain simple unsolvable problems of group theory.
Indig. Math. 16, (1955), 231-237, 492-497 ;
17, (1955), 252-256 ;
19, (1955), 22-27, 227-232.

[9]- BENSON CLARK T. ; MENDELSOHN N.S.
A calculus for a certain class of word problems in Groups.
J. Combinatorial Theory 1 (1966), 202-208.

[10]- BAUMSLAG Gilbert
Groups with one defining relator.
J. Austral. Math. Soc. 4 (1964), 385-392.

[11]- CHOMSKY N. and SCHUTZENBERGER M.P.
The algebraic theory of context-free languages, in "computer programming and Formal Systems".
(F. Braffort et al., eds.)
North-Holland publ., Amsterdam, 1963.

[12]- GREENDLINGER M.
Dehn's Algorithm for the word problem.
Comm. pure and appl. Math., 13 (1960), 67-83.

[13]- GREENDLINGER M.
On Dehn's algorithms for the conjugacy and word problems, with applications
Comm. pure and Appl. Math., 13 (1960), 641-677.

[14] - GRINDLINGER E.I.
The word problem for a class of semigroups with a finite number of defining relations (Russian).
Sibirsk. Mat. Ẑ. 5 (1964), 77-85.

[15] - GRINDLINGER M.
Solutions of the word problem for a class of groups by generalization of Dehn's algorithm. (Russian)
Dokl. Akad. Nauk SSSR 154 (1964), 507-509.

[16] - GROSS M. LENTIN A.
Notions sur les grammaires formelles
Gauthier - Villars Paris.

[17] - LYNDON ROGER C.
On Dehn's algorithm.
Math. Ann. 166 (1966), 208-228.

[18] - LYNDON ROGER C.
Cohomology Theory of groups with a single defining relation.
Annals of Math., 52 (1950), 650-665.

[19] - MAGNUS W. , KARRASS A. , SOLITAR D.
Combinatorial Group Theory (presentations of groups in terms of Generators and Relations).

[20] - MENDELSOHN N.S.
An algorithmic Solution for a word problem in group Theory.
Canad. J. Math. 16 (1964), 509-516.

[21] - MENDELSOHN N.S.
Correction to : An algorithmic solution for a word problem in group Theory.
Canad. J. Math. 17 (1965), 505.

[22] - NOVIKOV P.S.
On the Algorithmic unsolvability of the word problem in group theory.
Trudy Mat. Inst. im Steklov No 44, 143 PP.
Izdat. Akad. Nauk. SSSR, Moscow (Russian), MR 17, (1955), 706.
English text in "Russian translations", series 2, Vol. 9, 1958.
Edited by the American Math. Soc.

[23] - TRUFFAULT Bernard
Sur le problème des mots pour les groupes de Greendlinger.
C.R. Acad. Sci. Paris, Sér. A-B 267 (1968), A1-A3.

[24] - VAUQUOIS B.
Calculabilité des langages.
Université de Grenoble - Service de Mathématiques Appliquées, 1970.