UNIVERSITE LOUIS PASTEUR
I.R.E.M. de Strasbourg
10, rue du Général Zimmer
67084 STRASBOURG CEDEX

ANNEE 1981/82

Les Machines Programmables

Groupe I.R.E.M.
Animé par Gérard NAFFZGER

TABLE DES MATIERES

INTRODUCTION

Ce fascicule présente quelques programmes mis au point par le groupe I.R.E.M. "Utilisation de machines programmables dans le 2e cycle" durant l'année scolaire 1980/1981.

Il s'adresse à des lecteurs déjà initiés à la programmation, mais peut éventuellement être utilisé par le débutant : il suffit alors de se limiter au mode d'emploi et de jouer avec la machine.

En prévision d'utilisation d'autres types de machines que les calculatrices TI 57 ou 58, nous accompagons chaque programme d'un organigramme.

CONVENTIONS DE NOTATION UTILISEES

1° L'affectation

Nous désignons généralement le contenu d'un registre de données déterminé par une lettre. Une telle lettre est appelée variable.

Supposons que nous prenions la lettre u pour désigner le contenu du registre n° 3, et que ce contenu, à un moment donné, soit 7,2. Nous dirons alors que la valeur de u est 7,2. La lettre X désignera toujours le contenu du registre d'affichage.

Une notation telle que

$$u :\leftarrow u + 5$$

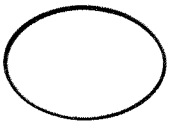
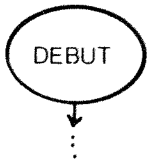
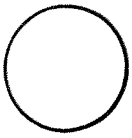
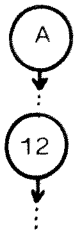
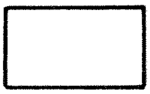
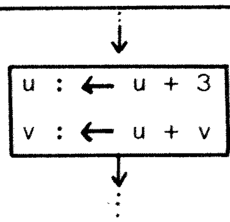
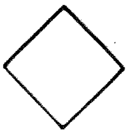
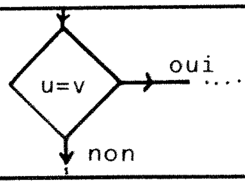
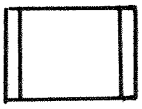
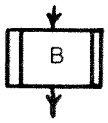
représente une instruction. Cette instruction consiste à donner à u la valeur calculée à droite de la flèche à partir de l'ancienne valeur de u. Dans notre exemple, u prendrait donc la valeur 7,2 + 5, soit 12,2 après exécution de cette instruction.

Une telle instruction est appelée affectation.

Exceptionnellement, nous désignerons également le contenu du registre 3 par la notation R_3.

2° Symboles utilisés dans les organigrammes

Nous dressons ci-aprés un tableau des principaux symboles utilisés dans ce fascicule.

SYMBOLE	SIGNIFICATION	EXEMPLE
	Ovale servant à marquer le début ou la fin d'un organigramme	
	Le cercle sera utilisé pour désigner une étiquette (label) ou une adresse numérique	 label A adresse numérique : pas n° 12
	Le rectangle contient une instruction ou un bloc d'instructions considérées <u>comme étant exécutées simultanément</u>	
	Le losange est utilisé pour les tests	
	Ce symbole désigne un sous-programme	 Exécution du sous-programme labellé B.

3° <u>Convention de notation pour les arrêts et les fins de sous-programme</u>.

Le mot Stop utilisé dans un organigramme symbolise un arrêt obtenu par la touche R/S.

Le mot Fin représente les touches INVSBR.

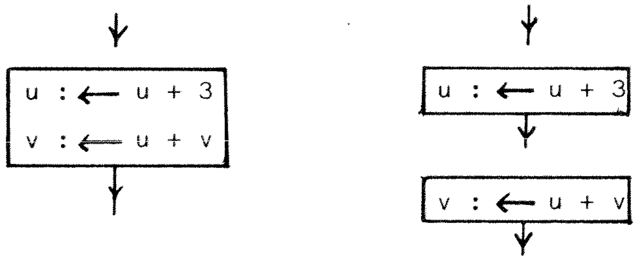
4° Notations relatives aux adressages indirects.

Si u est une variable, la notation R(u) désigne le contenu du registre de numéro u.

Exemple : Supposons que u désigne le contenu du registre n°3, et que ce contenu soit 5. Alors, R(u) désigne le contenu du registre 5.
On peut étendre cette notation à d'autres types d'instructions, par exemple Fix u, SBRu signifieraient dans ce cas Fix 5, SBR5.

5° Questions

Quelle différence y-a-t-il entre les deux portions d'organigramme suivantes ?



Traduisez chacune de ces portions par des séquences de touches sur votre machine, en supposant que u représente le contenu du registre 1, et v celui du registre 2.

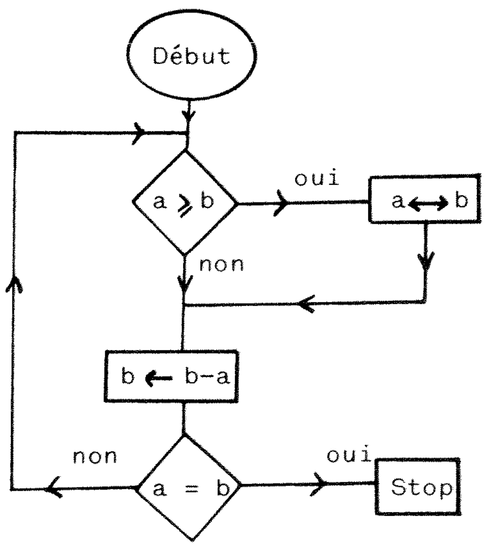
PREMIERE PARTIE

Présentation de trois programmes utilisables en classe

P.G.C.D. en 6 pas sur T.I. 57

Objet du programme : on se donne deux entiers naturels distincts a et b, et l'on cherche le P.G.C.D. de a et b.

Principe : on utilise la méthode des différences, basée sur le théorème suivant : le P.G.C.D. des naturels a et b est égal au P.G.C.D. des naturels a et $|b-a|$
Ce théorème suggère l'algorithme indiqué par l'organigramme ci-dessous



Mode d'emploi du programme.

Introduire a. Appuyer sur x⟷t

Introduire b. Appuyer sur RST R/S

La machine affiche le P.G.C.D. de a et b.

Pour un nouveau calcul, il n'est pas nécessaire d'appuer sur RST.

Listing du programme

Instructions	N° du pas
$x \geqslant t$	00
$x \longleftrightarrow t$	01
INV Sum 7	02
$x = t$	03
R/S	04
RST	05

Commentaires : pour des entiers dont la différence n'est pas très grande, on constatera l'extraordinaire rapidité d'exécution.

Si l'écart des nombres a et b est important, il est préférable d'utiliser l'algorithme d'Euclide.

Exemple : Pour calculer le P.G.C.D. de 480 et 6006, la machine met environ 10 secondes, ce qui est très honorable avec cette méthode.

Question

Que se passerait-il avec ce programme si l'on introduisait par mégarde a et b avec a = b ?

CALCULS DANS L'ENSEMBLE DES RATIONNELS SUR T.I. 57

1° Principe

Voici maintenant un programme permettant d'additionner et de multiplier deux rationnels $\frac{a}{b}$ et $\frac{c}{d}$, où a et c sont des entiers relatifs et où b et d sont des naturels. A la fin du calcul, on rend la fraction irréductible en divisant les deux termes du résultat par le P.G.C.D. de leurs valeurs absolues.

Les formules utilisées sont :

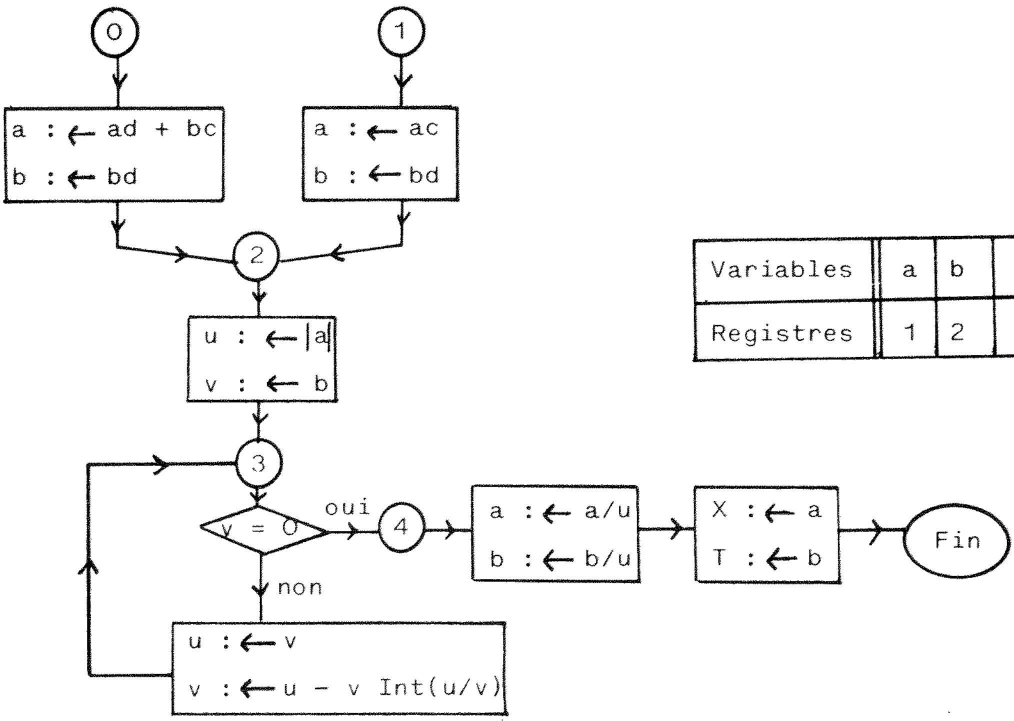
$$\frac{a}{b} + \frac{c}{d} = \frac{ad + bc}{bd} \qquad \text{et} \qquad \frac{a}{b} \times \frac{c}{d} = \frac{ac}{bd}$$

Les termes a et b sont stockés au départ dans les registres 1 et 2 et ils sont remplacés par les termes du résultat à la fin du calcul : cela permet de faire des calculs en chaîne.

Les termes d et c sont conservés dans les registres X et T.

Le P.G.C.D. est calculé par l'algorithme d'Euclide : les diviseurs et restes successifs seront notés u et v et stockés dans les registres 3 et 4.

2° Organigramme



Variables	a	b	u	v	d	c
Registres	1	2	3	4	X	T

3° Mode d'emploi

a) Au début du calcul, stocker a en R 1 et b en R 2

b) Introduire c, appuyer sur x ⟷ t, puis introduire d

c) Pour additionner, appuyer sur SBR 0

d) Pour multiplier, appuyer sur SBR 1

A la fin du calcul, le numérateur du résultat est dans le registre d'affichage X, ainsi que dans R 1 ; le dénominateur se trouve dans le registre T ainsi que dans R 2.

4° Listing du programme sur T.I. 57

Instructions	N° du pas	Commentaires
Label 1	00	Calcul de $\frac{a}{b} \times \frac{c}{d}$
Prd 2	01	
x ⟷ t	02	
Prd 1	03	
Label 2	04	Initialisation pour le calcul du P.G.C.D de $\|a\|$ et b.
Rcl 1	05	
$\|x\|$	06	
Sto 3	07	
Rcl 2	08	
Sto 4	09	
ct	10	
Label 3	11	Boucle de calcul du P.G.C.D. des nombres a et b par l'algorithme d'Euclide.
x=t	12	
GTO 4	13	
Rcl 3	14	
-	15	
(	16	
CE	17	
:	18	
Rcl 4	19	
Sto 3	20	

Instructions	N° du pas	Commentaires
)	21	
Int	22	
X	23	
Rcl 4	24	
=	25	
Sto 4	26	
GTO 3	27	
Label 4	28	Simplification des deux termes a, b par le P.G.C.D de $\|a\|$ et b.
Rcl 3	29	
INV Prd 1	30	
INV Prd 2	31	
Rcl 2	32	Affichage du résultat
$x \leftrightarrow t$	33	
Rcl 1	34	
R/S	35	
Label O	36	
Prd 1	37	
$x \leftrightarrow t$	38	Calcul de $\frac{a}{b} + \frac{c}{d}$
X	39	
Rcl 2	40	
=	41	
Sum 1	42	
$x \leftrightarrow t$	43	
Prd 2	44	
GTO 2	45	

5° Questions

Pourquoi ce programme permet-il d'effectuer les 4 opérations usuelles dans Q ?

Donnez le mode d'emploi pour effectuer des calculs en chaîne.

Exemple : $\frac{2}{3} - \frac{5}{6} + \frac{7}{12}$

METHODE DE RESOLUTION D'UNE EQUATION A UNE INCONNUE REELLE PAR DICHOTOMIE, EN PLUSIEURS VERSIONS.

1° Principe

Considérons l'équation $f(x) = 0$. Si nous connaissons deux réels a et b tels que f(a) et f(b) soient de signes contraires et si nous savons que la fonction f est continue dans le segment $[a,b]$, nous pouvons affirmer qu'il existe au moins une racine de l'équation dans le segment $[a,b]$.

Le nombre $c = \frac{a+b}{2}$ est une valeur approchée de la racine avec une erreur inférieure, en valeur absolue, à $\frac{|a-b|}{2}$.

Ces remarques suggèrent déjà un premier plan de calcul :

1- Déterminer les nombres a et b satisfaisant aux conditions indiquées, en procédant à des essais.

2- Se donner une limite supérieure de l'incertitude, en tenant néanmoins compte des possibilités de la machine : ainsi, sur T.I. 57, il serait illusoire de prendre une incertitude inférieure à 10^{-8}.
Désignons par ε cette limite supérieure.

3- Calculer $c = \frac{a+b}{2}$

4- Tester si $|b-a|$ est inférieur à ε
Si oui, afficher c : Fin du calcul
Si non, continuer

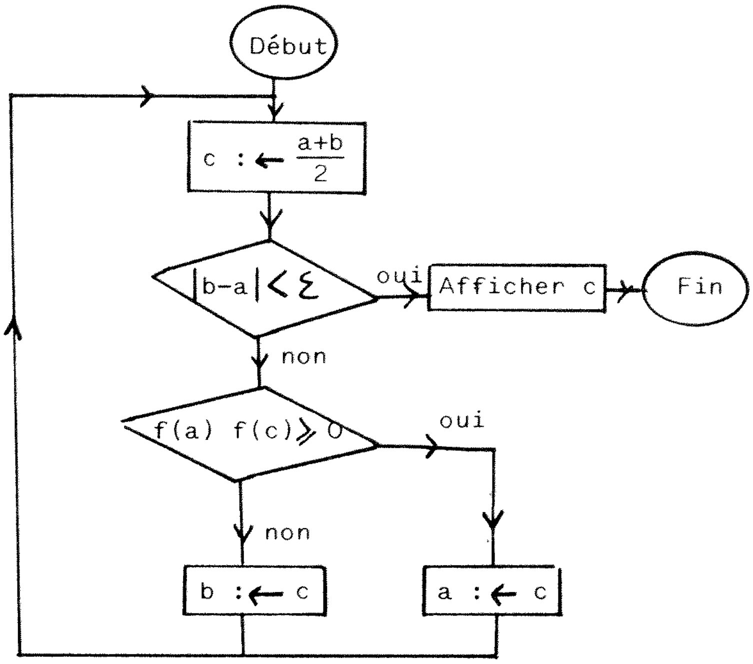
5- Tester si $f(a)\ f(c) \geqslant 0$
Si oui, remplacer a par c
Si non, remplacer b par c

6- Revenir à l'étape 3.

2° Première version

Les valeurs de la fonction f seront ici calculées par sous-programme.

L'organigramme pourrait être le suivant :



Malheureusement, cet organigramme exige au moins 32 pas sur T.I. 57.
D'autre part, son excécution est assez lente, car dans chaque boucle, on calcule plusieurs valeurs de la fonction f.

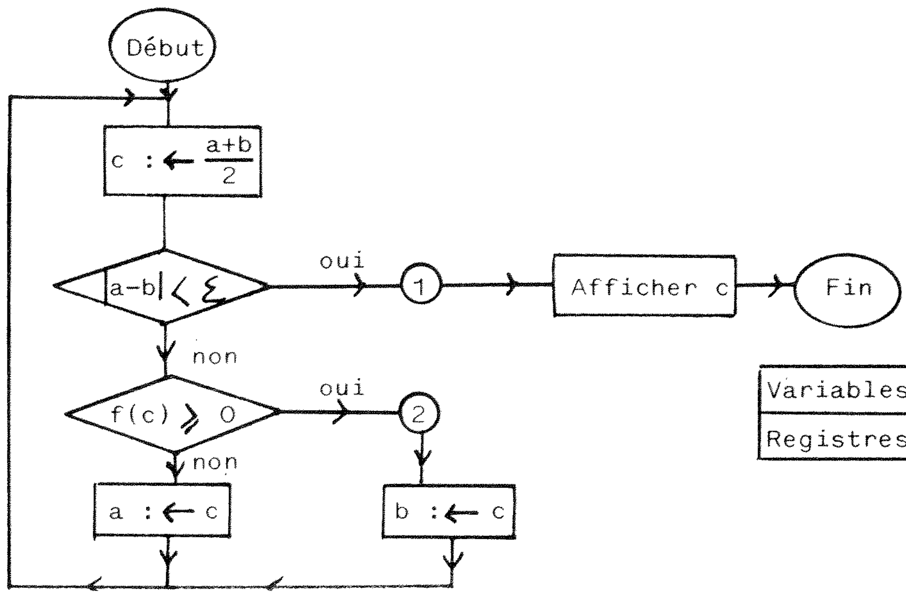
3° Deuxième version

On arrive à un programme de 28 pas seulement, si l'on ne compte pas le sous-programme de calcul de f(x), grâce au mode d'emploi modifié suivant :

1- Déterminer a et b tels que $f(a) < 0$ et $f(b) > 0$.

2- Stocker manuellement a dans le registre 1, b dans le registre 2 et ε dans le registre 3.

Voici un organigramme convenable :



Variables	a	b	ε	c
Registres	1	2	3	4

Démarrer le programme par RST R/S

Avant utilisation, il faut être certain que le registre T (mémoire R7) est effacé.

Programme sur T.I. 57.

Instructions	N° du pas	Commentaires
RCL 1	00	
STO 4	01	
-	02	Calcul simultané de $c = \frac{a+b}{2}$
RCL 2	03	et de $\lvert a-b \rvert - \varepsilon$
SUM 4	04	
=	05	
$\lvert x \rvert$	06	
-	07	
2	08	
INV Prod 4	09	
RCL 3	10	
=	11	
INV $x \geqslant t$	12	Si $\lvert a-b \rvert - \varepsilon < 0$, on affiche c.
GTO 1	13	
RCL 4	14	
SBR 0	15	Si $f(c) \geqslant 0$, branchement au label 2
$x \geqslant t$	16	
GTO 2	17	
RCL 4	18	a prend la valeur c, puis retour
STO 1	19	au début
RST	20	
Lbl 2	21	b prend la valeur c, puis retour
RCL 4	22	au début.
STO 2	23	
RST	24	
Lbl 1	25	Affichage de c et arrêt.
RCL 4	26	
R/S	27	

Commentaires : le calcul de f(X) doit être programmé en sous-programme labellé O à partir du pas 28. Ce sous-programme doit remplacer X par f(X). On pourra utiliser le registre O pour éventuellement conserver la valeur initiale de X.

Exemple : Pour résoudre l'équation $x^3 + 2x^2 - x - 1 = 0$, on peut ajouter les instructions suivantes à partir du pas 28 :

Lbl O (STO O x^2 x (RCL O + 2) - RCL O - 1) INV SBR

Pour calculer des valeurs de f(x), introduire x et appuyer sur SBR O. On trouve $f(0) < 0$ et $f(1) > 0$. Avec $\varepsilon = 10^{-6}$, on trouve la racine 0,8019376.

L'arithmétique en mémoire permet de calculer simultanément c et $|a-b|$, d'où une économie de pas.

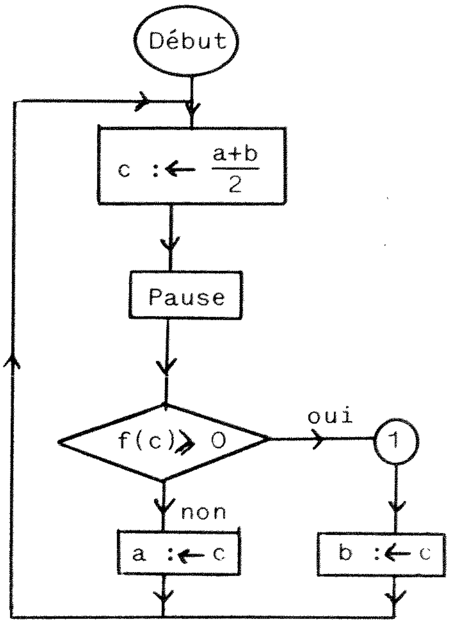
4° Troisième version

Si l'on n'exige pas le test sur ε, il est possible d'écrire un programme de 19 pas seulement. Le voici :

Instructions	N° du pas	Commentaires
RCL 1	00	
+	01	
RCL 2	02	Calcul de $c = \frac{a+b}{2}$
=	03	et affichage en pause
:	04	
2	05	
=	06	
STO 3	07	
Pause	08	
SBR O	09	Si $f(c) \geqslant 0$, branchement
$x \geqslant t$	10	au label 1
GTO 1	11	
RCL 3	12	a prend la valeur c, puis
STO 1	13	retour au début
RST	14	
Lbl 1	15	b prend la valeur c, puis
RCL 3	16	retour au début
STO 2	17	
RST	18	

Mode d'emploi : programmer f(x) en sous-programme labellé 0 à partir du pas 19. Déterminer a et b tels que f(a) < 0 et f(b) > 0. Stocker a en R_1 et b en R_2. Appuyer sur RST R/S. Les valeurs successives de c sont affichées en pause. Dès que les décimales ne bougent plus, arrêter le programme par R/S.

Organigramme :



Variables	a	b	c
Registres	1	2	3

5° Question

Résolvez une équation du 3e degré à coefficients entiers à l'aide des deux dernières versions.

Pourquoi la 3e version est-elle particulièrement commode ?

DEUXIEME PARTIE

Quelques Jeux

BLACK-JACK NUMERIQUE SUR T.I. 57

Ce jeu consiste à essayer d'atteindre un total aussi proche de 13 que possible, mais sans dépasser 13, en tirant des nombres entiers au hasard parmi les entiers de 1 à 9.

Le jeu se joue contre la machine selon la règle suivante.

Première version : le joueur commence.

Dans ce cas, il fait un certain nombre de tirages jusqu'à obtention d'un total satisfaisant. Puis il passe la main à la machine : celle-ci effectue alors des tirages jusqu'à ce que son total dépasse celui du joueur.

Deuxième version : la machine commence.

La machine fait alors les tirages selon une certaine stratégie qui lui donne de bonnes chances de gagner. Puis le joueur effectue ses propres tirages en essayant d'améliorer le score de la machine.

1° Variables du programme.

Les tirages sont effectués à l'aide d'un générateur de nombres aléatoires de $[0,1[$ grâce à l'algorithme $s \leftarrow$ FRAC(997s). s est donc un nombre de $[0, 1[$

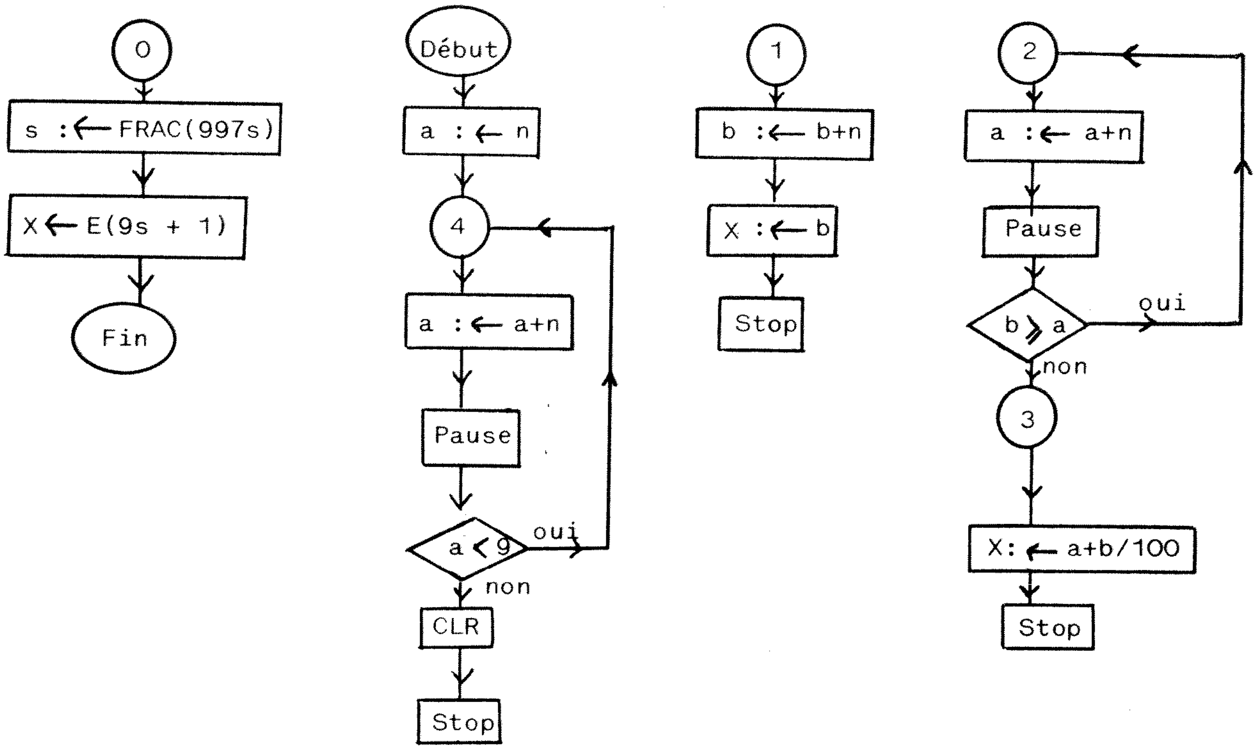
Dans l'organigramme qui suit, n est l'entier aléatoire E(9s + 1).

a est le total de la machine, et b est le total du joueur.

La valeur λ = 997 est stockée au départ dans le registre 3.

Variables	a	b	λ	s	
Registres	1	2	3	4	

2° Organigramme



3° Mode d'emploi

a) Initialisation avant la première partie

Effacer les mémoires par INV ct

Stocker 100 en R_0 et 997 en R_3

Stocker un nombre source compris entre 0 et 1 en R_4

Mettre la machine à 2 décimales par Fix 2

b) Jeu

Si le joueur commence, faire les tirages par SBR1, SBR1, ...
La machine affiche chaque fois le total obtenu.
Dès l'obtention d'un total satisfaisant, passer la main à la machine par SBR2. Celle-ci peut alors afficher un ou plusieurs résultats de tirages en pause. Dès l'arrêt, le score final est affiché sous la forme a.b, où a est le total atteint par la machine, et b le total du joueur.

Pour faire commencer la machine, faire RST R/S, puis après arrêt faire SBR1, SBR1,... pour les tirages du joueur. Enfin, faire SBR3 pour afficher le score.

c) Initialisation pour une nouvelle partie

Après avoir joué une partie, on peut en recommencer une nouvelle en faisant CLR Sto1 Sto2 avant le début du nouveau jeu.

4° Listing du programme sur T.I. 57

Instructions	N° du pas	Commentaires
SBRO Sto 1	00 01	Le total de la machine est initialisé à l'aide d'un premier tirage
label 4 SBRO Sum 1 Pause 9 $x \leftrightarrow t$ Rcl 1 INV $x \geqslant t$ GTO 4	02 03 04 05 06 07 08 09 10	Boucle servant à faire effectuer à la machine des tirages successifs affichés en pause.
CLR R/S	11 12	A la sortie de la boucle précédente, la machine s'arrête en affichant 0
label 1 SBR 0 Sum 2 Rcl 2 R/S	13 14 15 16 17	Tirage effectué par le joueur et affichage du total obtenu
label 2 SBR 0 Sum 1 Pause Rcl 1 $x \leftrightarrow t$ Rcl 2 $x \geqslant t$ GTO 2	18 19 20 21 22 23 24 25 26	Autre boucle servant à faire effectuer à la machine des tirages successifs affichés en pause

Instructions	N° du pas	Commentaires
label 3	27	Affichage du score sous la forme a.b
Rcl 1	28	
+	29	
Rcl 2	30	
:	31	
Rcl 0	32	
=	33	
R/S	34	
label 0	35	Sous-programme permettant de tirer au hasard un entier compris, au sens large, entre 1 et 9.
Rcl 3	36	
x	37	
Rcl 4	38	
=	39	
INV Int	40	
Sto 4	41	
x	42	
9	43	
+	44	
1	45	
=	46	
Int	47	
INV SBR	48	

5° Questions

Comparez les deux boucles servant à faire effectuer des tirages par la machine.

Quelle est la stratégie employée par celle-ci ?

Cette stratégie vous semble-t-elle judicieuse ?

JEU DE DORNIM SUR T.I. 57

1° Règle du jeu

Ce jeu se joue à deux sur papier quadrillé, les sommets du quadrillage représentant les points de coordonnées entières dans un repère orthonormé. Le terrain de jeu se limite au quart de plan $x \geqslant 0$, $y \geqslant 0$, dans lequel les joueurs déplacent à tour de rôle un pion sur les points à coordonnées entières selon la règle suivante.

A partir d'une position donnée, on peut déplacer le pion soit horizontalement vers la gauche (abscisses décroissantes), soit verticalement vers le bas (ordonnées décroissantes), soit en diagonale à gauche et vers le bas, et ceci d'un nombre quelconque de cases tout en restant dans le quart de plan précité.

Le but du jeu est d'atteindre l'origine. Au départ, le pion est placé en un point choisi à l'avance de façon que le gain ne paraisse pas évident pour l'un ou l'autre joueur.

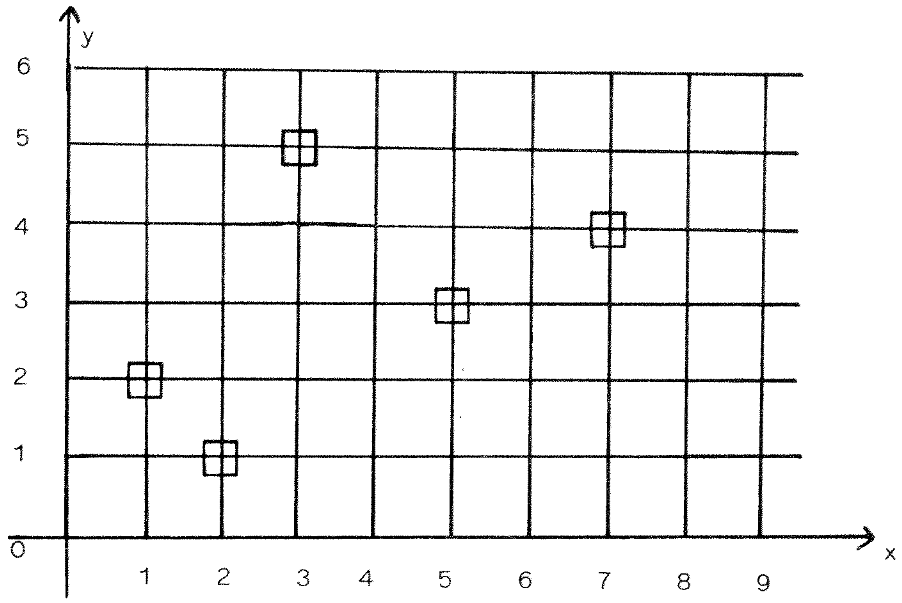
2° Exemple

Désignons les deux joueurs par A et B et supposons que le joueur A commence en plaçant le pion au point de coordonnées (10,15).
Le joueur B peut alors le placer au point de coordonnées (8,13) en descendant de deux cases en diagonale.
Si le joueur A déplace le pion en (5,13), le joueur B peut alors, par un déplacement vers le bas, le poser en (5,3). Si maintenant A joue en (5,2), le joueur B est assuré de gagner en plaçant le pion en (1,2).

3° Positions gagnantes et positions perdantes

On démontre que si l'un des joueurs (par exemple B) joue en plaçant le pion sur certaines positions dites "positions gagnantes", il peut gagner contre toute défense. Nous appellerons "position perdante" toute position non gagnante, l'origine étant considérée comme la position gagnante finale.

Ci-dessous, nous avons marqué par un petit carré quelques positions gagnantes.



4° Forme générale des positions gagnantes

On démontre les résultats suivants.

a) Si l'on joue à partir d'une position gagnante, on obtient toujours une position perdante.

b) A partir d'une position perdante, il est toujours possible de rejoindre une position gagnante en respectant les règles de déplacement du jeu.

c) Désignons par α le nombre d'or $\frac{1+\sqrt{5}}{2}$ et posons $\beta = \alpha + 1 = \alpha^2$.
Les positions gagnantes sont les points dont les coordonnées sont de la forme $(E(n\beta), E(n\alpha))$, où n est un naturel quelconque, ainsi que leurs symétriques par rapport à la première bissectrice.

5° Remarques importantes

a) Considérons une position gagnante du huitième de plan $x \geqslant 0$, $y \geqslant 0$, $x \geqslant y$.
On a $E(n\beta) - E(n\alpha) = E(n(\alpha + 1)) - E(n\alpha) = E(n\alpha + n) - E(n\alpha) = n$

Un tel point vérifie donc la relation $x - y = n$.
Or, lorsqu'on se déplace en diagonale, la différence $x - y$ est constante. Il en résulte que si l'on peut rejoindre une position gagnante en diagonale, l'entier n est égal à $x - y$ et y est supérieur à $E((x - y)\alpha)$.

b) Si, à partir d'une position perdante du huitième de plan précédent on ne peut rejoindre de position gagnante en diagonale, on montre alors qu'on se trouve nécessairement à droite d'une position gagnante : l'ordonnée de cette position est donc y, ordonnée du point (x,y) d'où l'on part. Pour connaître son abscisse, il suffit de calculer l'entier n.

Deux possibilités se présentent :

- si la position gagnante se trouve à droite de la première bissectrice, alors $y = E(n\alpha)$, d'où l'on déduit que $n = E(\frac{y+1}{\alpha})$ et l'abscisse de cette position est $y + n$.
- sinon on trouve $n = E(\frac{y+1}{\beta})$ et l'abscisse de la position gagnante est $y - n$.

6° Première ébauche de programme

Nous nous proposons de programmer ce jeu de façon que l'utilisateur du programme puisse jouer contre la machine, celle-ci adoptant la meilleure tactique possible.

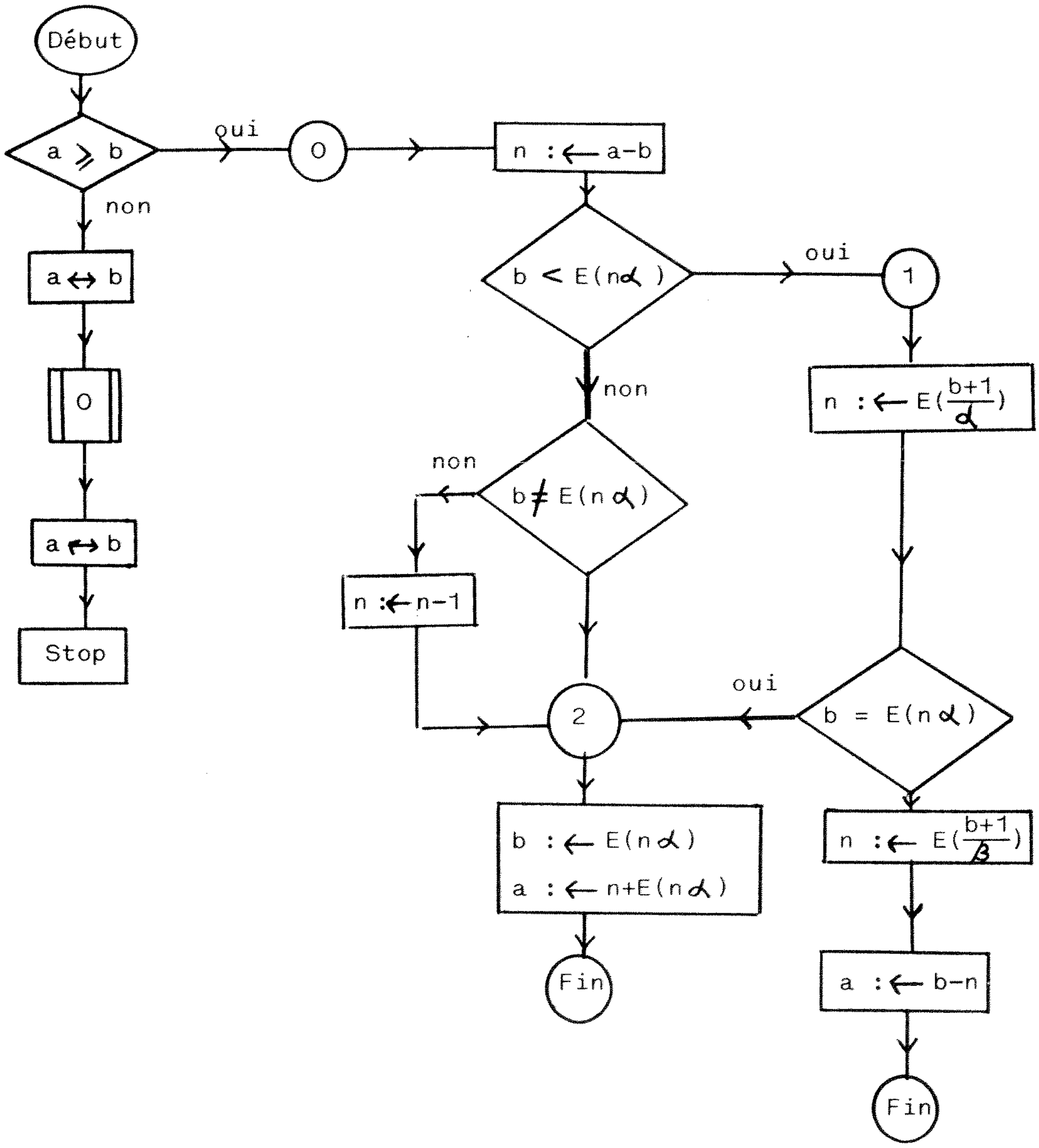
Voici déjà un premier plan de calcul.

a) Se placer dans le huitième de plan considéré ci-dessus, en échangeant au besoin les coordonnées avant et après le calcul.

b) A partir d'un point (x,y) de ce huitième de plan dont les coordonnées sont données à la machine, celle-ci doit d'abord tester si $y \geqslant E((x - y)\alpha)$.
Si oui, tester s'il y a égalité. Dans la négative, la machine indique la position gagnante obtenue en se déplaçant en diagonale, c'est-à-dire le point de coordonnés $(E((x - y)\alpha) + x - y,\ E((x - y)\alpha))$.
Dans l'affirmative, la machine doit jouer à regret sur une position perdante, puisqu'on se trouvait déjà sur une position gagnante ! Dans ce cas, on diminuera donc l'abscisse d'une unité.

c) Si le test $y \geqslant E((x - y)\alpha)$ est négatif, le point (x, y) se trouve nécessairement à droite d'une position gagnante. On fait alors le test $y = E(n\alpha)$ où $n = E(\frac{y+1}{\alpha})$.
Si la réponse est oui, l'abscisse cherchée est $y + n$. Sinon l'entier n est égal à $E(\frac{y+1}{\beta})$ et l'abscisse cherchée est $y - n$.

7° Organigramme



Variables	α	b, puis $\frac{b+1}{\alpha}$	n	b
Registres	0	1	2	7

8° Programme sur T.I. 57

Instructions	N° du pas	commentaires
$x \geqslant t$	00	
GTO 0	01	
$x \leftrightarrow t$	02	Echange des coordonnées si $a < b$
SBR 0	03	
$x \leftrightarrow t$	04	Nouvel échange et arrêt
R/S	05	
Label 0	06	
−	07	
Rcl 7	08	$n :\leftarrow a - b$
STO 1	09	
=	10	
SBR 3	11	Calcul de $E(n\alpha)$ par le sous-
INV $x \geqslant t$	12	programme 3
GTO 1	13	Tests $b < E(n\alpha)$ et $b \neq E(n\alpha)$
INV $x = t$	14	
GTO 2	15	
1	16	
INV SUM 2	17	
Label 2	18	
Rcl 7	19	
SUM 2	20	
Rcl 2	21	
INV SBR	22	
Label 1	23	
$x \leftrightarrow t$	24	
1	25	
SUM 1	26	
Rcl 0	27	
INV Prd 1	28	$\frac{b+1}{\alpha}$ est mis dans R 1
Rcl 1	29	
Int	30	
SBR 3	31	n prend la valeur $E(\frac{b+1}{\alpha})$
$x = t$	32	Test $b = E(n\alpha)$

Instructions	N° du pas	Commentaires
GTO 2	33	
–	34	
$x \leftrightarrow t$	35	
Rcl 0	36	Calcul simultané de $n = E(\frac{b+1}{\beta})$
INV Prd 1	37	et de b-n
Rcl 1	38	
Int	39	
=	40	
INV SBR	41	
Label 3	42	
STO 2	43	Sous-programme permettant déconomiser
X	44	quelques pas !
Rcl 0	45	
=	46	
Int	47	
$x \leftrightarrow t$	48	
INV SBR	49	

9° Mode d'emploi

a) Initialisation

Stocker le nombre d'or $\frac{1+\sqrt{5}}{2}$ dans le registre 0.

b) Jeu

Introduire l'ordonnée b dans le registre T par $x \leftrightarrow t$.

Introduire l'abscisse a.

Appuyer sur RST R/S.

Résultat du jeu de la machine : l'ordonnée se trouve dans T et l'abscisse dans le registre d'affichage X.

10° Remarque

Le sous-programme 3 n'apparaît pas dans l'organigramme. Il permet de terminer le calcul de $E(n\alpha)$.

LA BATAILLE NAVALE SUR T.I. 58

1° Règle du jeu

Il s'agit d'une bataille navale simplifiée. Le programme proposé ici permet à l'utilisateur de la T.I. 58 de jouer contre la machine. Le joueur et la machine sont supposés disposer d'une "mer" de 25 cases sur laquelle sont disposés cinq bateaux occupant chacun une seule case.
Le but du jeu est de couler la flotte ennemie. Pour cela, on vise à tour de rôle une case de l'adversaire : si cette case est occupée par un bateau, celui-ci est coulé.

Remarquons que dans ce jeu simplifié, deux bateaux peuvent être contigus.

2° Codes utilisés

Au cours du jeu, les cases de la machine ainsi que celles du joueur sont représentées par les registres 1 à 25 contenant chacun un code permettant de connaître à chaque instant la situation du jeu.

En règle générale, une case vide est codée 2 et une case occupée est codée 1.

De façon plus précise, soit x le contenu du registre représentant une certaine case au cours du jeu. Les valeurs possibles pour x sont
-2,5 ; -2 ; -1,5 ; -1 ; 1 ; 1,5 ; 2 ; 2,5.

Posons $n = |\mathrm{Int}(x)|$. Si $n = 2$, la case est vide sur la mer de la machine, et si $n = 1$, elle est occupée.

Si $x > 0$, la case n'a pas encore été visée par la machine, et si $x < 0$, elle a déjà été visée.

Si $\mathrm{FRAC}(x) = 0$, la case n'a pas encore été visée par le joueur, elle a été visée dans le cas contraire.

3° Mode d'emploi

Préparer sur papier quadrillé deux grilles de 25 cases, l'une pour la flotte du joueur, l'autre pour celle de la machine. Disposer dans celle du joueur 5 bateaux : chaque case étant repérée par son numéro (entier de 1 à 25), cela revient donc à choisir 5 entiers parmi les entiers de 1 à 25.

Avant la première partie, initialiser en introduisant un nombre source compris entre 0 et 1 et en appuyant sur E.

a) Jeu du joueur

Pour tirer, le joueur doit entrer un numéro de case et appuyer sur A. Si la case visée est vide, la machine répond 0. Si elle est occupée, elle affiche 1.

Au cas où elle aurait déjà été jouée, la machine affiche 505 (S O S) invitant ainsi le joueur à viser une autre case.

b) Jeu de la machine

Pour faire jouer la machine, appuyer sur B. Elle affiche alors le numéro de la case du joueur visée.

4° Variables utilisées

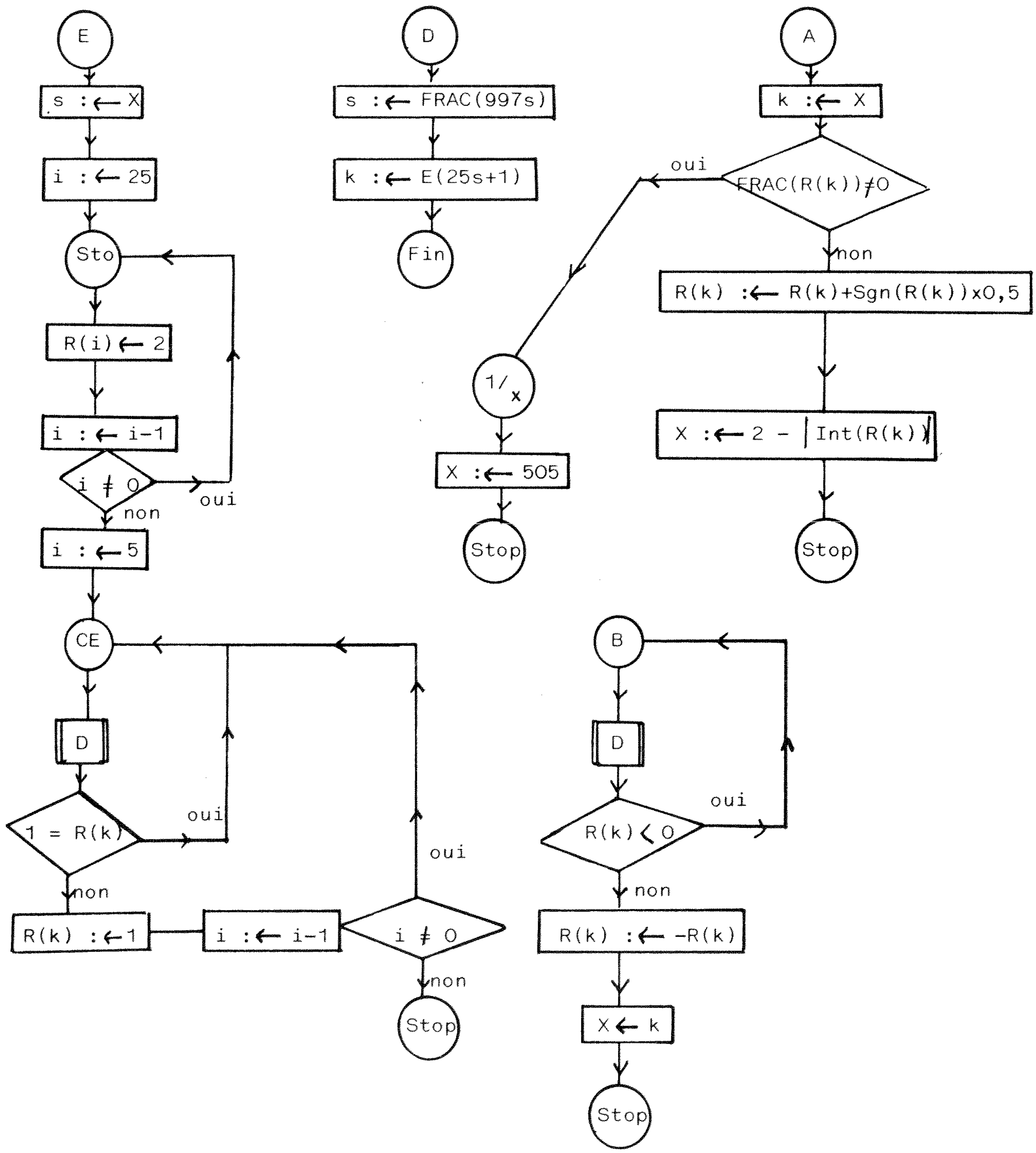
s est le nombre source du générateur de nombres aléatoires.

i est un compteur

k est un pointeur pour les adressages indirects.

Variables	i	s	k
Registres	0	26	27

5° Organigramme


E
s :← X
i :← 25
Sto
R(i) ← 2
i :← i-1
i ≠ 0
oui
non
i :← 5
CE
D
1 = R(k)
oui
non
R(k) :← 1
i :← i-1
i ≠ 0
oui
non
Stop
D
s :← FRAC(997s)
k :← E(25s+1)
Fin
1/x
X :← 505
Stop
A
k :← X
FRAC(R(k))≠0
oui
non
R(k) :← R(k)+Sgn(R(k))x0,5
X :← 2 - |Int(R(k))|
Stop
B
D
R(k) < 0
oui
non
R(k) :← -R(k)
X ← k
Stop

6° Programme sur T.I. 58

Instructions	N° du pas	Instructions	N° du pas
Label D	000-001	CLR	058
(	002	R/S	059
(	003	Label A	060-061
9	004	STO 27	062-063
9	005	CP	064
7	006	Rcl Ind 27	065-066
X	007	INV Int	067-068
Rcl 26	008-009	INV x=t	069-070
)	010	1/x	071
INV Int	011-012	Rcl Ind 27	072-073
STO 26	013-014	Op 10	074-075
X	015	X	076
25	016-017	.	077
+	018	5	078
1	019	=	079
)	020	SUM Ind 27	080-081
Int	021	(	082
STO 27	022-023	2	083
INV SBR	024	-	084
Label E	025-026	RCL Ind 27	085-086
STO 26	027-028	Int	087
2	029	$\lvert x \rvert$	088
5	030	)	089
STO 00	031-032	R/S	090
2	033	Label 1/x	091-092
Label STO	034-035	5	093
STO Ind 0	036-037	0	094
Dsz 0 STO	038-040	5	095
5	041	$\lvert x \rvert$	096
STO 00	042-043	R/S	097
Label CE	044-045	Label B	098-099
D	046	D	100
Rcl Ind 27	047-048	CP	101
$x \leftrightarrow t$	049	Rcl Ind 27	102-103
1	050	INV $x \geq t$	104-105
x = t	051	B	106
CE	052	+/-	107
STO Ind 27	053-054	STO Ind 27	108-109
Dsz 0 CE	055-057	Rcl 27	110-111
		R/S	112

7° <u>Questions</u>

Décrivez les avantages du codage utilisé.

Expliquez les calculs décrits dans l'organigramme.

"LE COMPTE EST BON" SUR T.I. 58

1° Principes du jeu.

Il s'agit d'une variante du jeu "Le compte est bon" de l'émission télévisée "Chiffres et Lettres".

Dans le programme proposé ici, la machine choisit cinq nombres parmi 2, 3, 4, 5, 6, 7, 8, 9, 10, 25, 50, 75, 100, l'un des nombres pouvant être pris plusieurs fois.

Les nombres choisis sont les opérandes utilisés dans le calcul d'un résultat obtenu en utilisant uniquement les opérations +, - et x, c'est-à-dire l'addition, la soustraction et la multiplication. La machine choisit 4 opérations parmi +, - et x : l'une au moins de celles-ci est donc au moins prise deux fois.

Le résultat n du calcul étant affiché, le joueur peut connaître les opérandes utilisés, puis il doit deviner les opérations utilisées pour parvenir au résultat. (Remarquons que ce résultat est un entier de signe quelconque).

Ce jeu est intéressant pour deux raisons au moins.
Il permet de s'entraîner au calcul mental tout en jouant avec la machine, ce qui peut motiver de jeunes élèves.
D'autre part, du point de vue de la programmation, il donne un exemple de choix aléatoire d'opérations, justifiant ainsi l'utilité de l'instruction SBR Ind.

2° Mode d'emploi du programme

a) Initialisation

Avant la première partie, introduire un nombre entier compris entre 0 et 100.000 et appuyer sur E.

b) Génération du résultat

Appuyer sur A. La machine choisit alors 5 opérandes et 4 opérations, puis affiche le résultat n du calcul.

Exemple : n = 164

c) Affichage des opérandes utilisés par la machine

Appuyer sur B. Le premier opérande utilisé est alors affiché. Appuyer sur R/S pour obtenir les suivants.

Exemple : 100, 6, 5, 10, 75.

Un appui de trop sur R/S provoquerait le clignotement sur T.I. 58, l'affichage de O sur T.I. 59.

d) Obtention d'une solution

Appuyer sur C. La machine indique les opérations utilisées à l'aide d'un nombre de 4 chiffres, grâce à la convention suivante :

le chiffre 1 représente +

le chiffre 2 représente -

le chiffre 3 représente x

Dans notre exemple, la machine pourrait donc afficher 2121, ce qui signifie qu'elle a utilisé les opérations -, +, -, +.

Vérification : 100-6+5-10+75 = 164

Remarque : Ces opérations sont supposées être effectuées de gauche à droite.

Autre Exemple :

Action	Affichage	Signification
Introduire 136	136	Nombre source
Appuyer sur E	0	Initialisation
Appuyer sur A	6500	Nombre n calculé
Appuyer sur B	5	Premier opérande
Appuyer sur R/S	10	Deuxième opérande
Appuyer sur R/S	5	Troisième opérande
Appuyer sur R/S	10	Quatrième opérande
Appuyer sur R/S	100	Cinquième opérande
Appuyer sur C	3113	Solution

Vérification :

5 x 10 = 50

50 + 5 = 55

55 + 10 = 65

65 x 100 = 6500

3° Variables utilisées dans le programme

Les principales variables du programme sont s, i, j, k, n.

i est un compteur

j et k sont des pointeurs

s est le nombre source

n est le nombre calculé.

La liste des opérandes susceptibles d'être choisis est stockée dans les registres 12 à 24.

Les opérandes effectivement choisis sont stockés dans les registres 25 à 29.

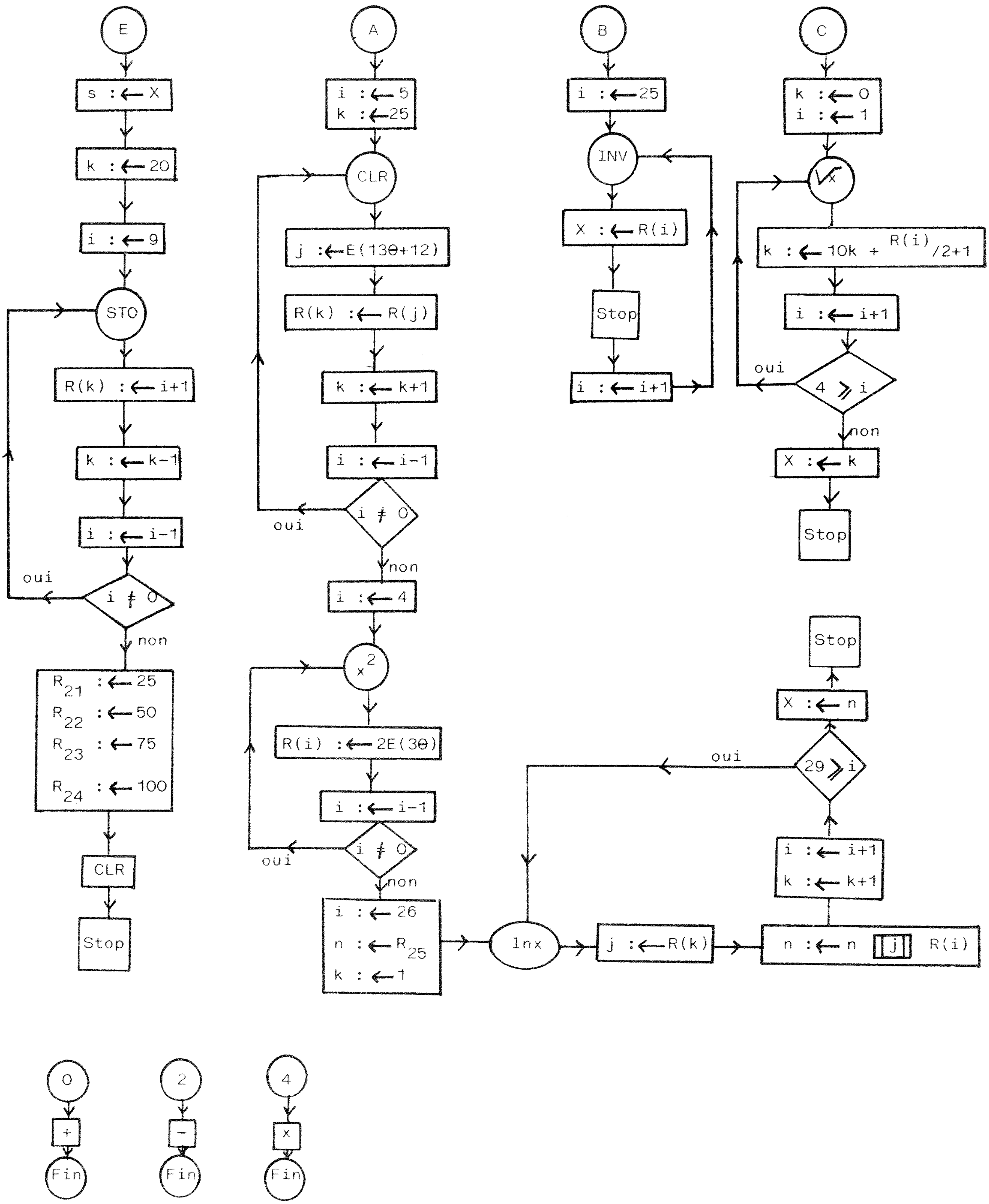
Dans le programme, les opérations +, -, x sont respectivement codées 0, 2, 4.

Les codes des opérations choisies sont stockées dans les registres 1 à 4.

Variables	i	k	j	s	n
Registres	0	5	6	9	11

Dans l'organigramme sui suit, nous avons utilisé le symbole θ pour désigner un nombre aléatoire de $[0, 1[$. Pour générer un tel nombre, le programme utilise le programme machine N° 15 du module de base. Le nombre θ est obtenu par la séquence Pgm 15 SBR DMS.

4° Organigramme


E
s :← X
k :← 20
i :← 9
STO
R(k) :← i+1
k :← k-1
i :← i-1
i ≠ 0
oui
non
R21 :← 25
R22 :← 50
R23 :← 75
R24 :← 100
CLR
Stop
A
i :← 5
k :← 25
CLR
j :← E(13θ+12)
R(k) :← R(j)
k :← k+1
i :← i-1
i ≠ 0
oui
non
i :← 4
x2
R(i) :← 2E(3θ)
i :← i-1
i ≠ 0
oui
non
i :← 26
n :← R25
k :← 1
lnx
j :← R(k)
n :← n [j] R(i)
i :← i+1
k :← k+1
29 ≥ i
oui
X :← n
Stop
B
i :← 25
INV
X :← R(i)
Stop
i :← i+1
C
k :← 0
i :← 1
√x
k :← 10k + R(i)/2+1
i :← i+1
4 ≥ i
oui
non
X :← k
Stop
0
+
Fin
2
−
Fin
4
x
Fin

5° Listing du programme

Instructions	N° du pas	Commentaires
+	000	Sous-programme pour l'addition
INV SBR	001	
-	002	Sous-programme pour la soustraction
INV SBR	003	
X	004	Sous-programme pour la multiplication
INV SBR	005	
Label E	006-007	Initialisations
Sto 09	008-009	
20	010-011	
Sto 05	012-013	
9	014	
Sto 0	015-016	
Label Sto	017-018	Boucle servant à ranger les entiers de 2 à 10 dans les registres 12 à 20
(	019	
Rcl 0	020-021	
+	022	
1	023	
)	024	
Sto Ind 5	025-026	
Op 35	027-028	
Dsz 0 Sto	029-031	
25	032-033	Stockage des nombres 25, 50, 75, 100 dans les registres 21 à 24.
Sto 21	034-035	
50	036-037	
Sto 22	038-039	
75	040-041	
Sto 23	042-043	
100	044-046	
Sto 24	047-048	
CLR	049	Fin de la séquence E
R/S	050	

Instructions	N° du Pas	Commentaires
Label A	051-052	Initialisations pour la boucle labellée CLR
5	053	
Sto 00	054-055	
25	056-057	
Sto 5	058-059	
Label CLR	060-061	
(	062	Boucle permettant de stocker 5 opérandes choisis au hasard.
13	063-064	
X	065	Ces opérandes sont choisis parmi les contenus des registres 12 à 24 et ils sont stockés dans les registres 25 à 29.
Pgm 15	066-067	
SBR DMS	068-069	
+	070	
12	071-072	L'expression $E(13\theta + 12)$ désigne un entier aléatoire compris, au sens large, entre 12 et 24.
)	073	
Int	074	
Sto 6	075-076	
Rcl Ind 6	077-078	
Sto Ind 5	079-080	
Op 25	081-082	
Dsz 0 CLR	083-085	
4	086	Initialisation pour la boucle labellée x^2.
Sto 0	087-088	
Label x^2	089-090	Boucle permettant de stocker dans les registres 1 à 4 des entiers aléatoires choisis dans l'ensemble 0 ; 2 ; 4
(	091	
(	092	
3	093	
X	094	
Pgm 15	095	Un tel entier aléatoire est obtenu par l'expression $2E(3\theta)$
SBR DMS	097-098	
)	099	
Int	100	
X	101	
2	102	
)	103	

Instructions	N° du pas	Commentaires
Sto Ind 0	104-105	
Dsz 0 x^2	106-108	
26	109-110	Initialisations pour la boucle labellée lnx
Sto 0	111-112	
Rcl 25	113-114	
Sto 11	115-116	
1	117	
Sto 5	118-119	
Label lnx	120-121	Boucle de calcul de n. Remarquez l'importante instructions SBR Ind 6.
Rcl Ind 5	122-123	
Sto 6	124-125	
(	126	
Rcl 11	127-128	
SBR Ind 6	129-131	
Rcl Ind 0	132-133	
)	134	
Sto 11	135-136	
Op 25	137-138	
Op 20	139-140	
Rcl 0	141-142	
$x \leftrightarrow t$	143	
29	144-145	
$x \geqslant t$	146	
lnx	147	
Rcl 11	148-149	Affichage de n et fin de la séquence A.
R/S	150	
Label C	151-152	Initialisations pour la boucle labellée $\sqrt{x}$
CLR	153	
Sto 05	154-155	
1	156	
Sto 0	157-158	

Instructions	N° du pas	Commentaires
Label $\sqrt{x}$	159-160	Boucle servant à former le nombre repré-
10	161-162	sentant les opérations choisies par la
Prd 5	163-164	machine.
(	165	
Rcl Ind O	166-167	
:	168	La fonction $x \mapsto \frac{x}{2} + 1$ permet de passer
2	169	de la suite (0, 2, 4) à la suite (1, 2, 3).
+	170	
1	171	
)	172	
Sum 5	173-174	
Op 20	175-176	
Rcl O	177-178	
$x \leftrightarrow t$	179	
4	180	
$x \geqslant t$	181	
$\sqrt{x}$	182	
Rcl 5	183-184	Fin de la séquence C.
R/S	185	
Label B	186-187	Initialisation pour la boucle labellée INV
25	188-189	
Sto O	190-191	
Label INV	192-193	Boucle servant à l'affichage des opérandes
Rcl Ind O	194-195	choisis par la machine.
R/S	196	
Op 20	197-198	
GTO INV		

6° <u>Questions</u>

Est-il nécessaire de stocker la liste des opérandes utilisables ?

Inventez, si possible, une procédure permettant d'éviter ce stockage.

La méthode adoptée ici vous semble-t-elle intéressante ? Si oui, à quel titre ?