

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:			
2	1:			
	0:			
	B:			
3	1:			
	0:			
	B:			
4	1:			
	0:			
	B:			
5	1:			
	0:			
	B:			
6	1:			
	0:			
	B:			
7	1:			
	0:			
	B:			
8	1:			
	0:			
	B:			
9	1:			
	0:			
	B:			
10	1:			
	0:			
	B:			
11	1:			
	0:			
	B:			

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:			
2	1:			
	0:			
	B:			
3	1:			
	0:			
	B:			
4	1:			
	0:			
	B:			
5	1:			
	0:			
	B:			
6	1:			
	0:			
	B:			
7	1:			
	0:			
	B:			
8	1:			
	0:			
	B:			
9	1:			
	0:			
	B:			
10	1:			
	0:			
	B:			
11	1:			
	0:			
	B:			

Ma technique de perforation des programmes.

A titre d'information je me suis amusé à estimer le nombre de trous que j'ai pratiqué pour réaliser les feuilles de programme servant à mettre au point ou dépanner le matériel, et les grilles relatives aux 53 programmes que je vous propose et qui ont été dument validés sur la machine. On arrive à la bagatelle de 2699 trous, **à multiplier par deux car une fois avoir plastifié les feuilles il faut les repercer !** Et encore, à ces 5398 trous ne sont pas prises en compte les grilles mauvaises qui ont fini dans la corbeille à papier. La première approche envisageable consiste à utiliser l'outil bien connu de la Fig.1, encore que cette technique manque certainement un tantinet de convivialité. (Il m'arrive toutefois de l'utiliser quand il faut percer deux ou trois trous pour corriger une grille.) La deuxième technique consiste à réaliser la perforatrice spéciale montrée dans la vidéo de Monsieur Marc Raynaud sur : <https://www.youtube.com/watch?v=L5O04P2ASRc>.

Pour m'éviter d'avoir à me fabriquer un tel dispositif, je me suis contenté d'utiliser l'outil de poinçonnage de la Fig.2 qui est disponible dans mon atelier. Dans la pratique je dispose d'un jeu complet d'emporte-pièces, il suffit de choisir celui qui présente le bon diamètre.

- 1) On protège l'établi avec un "martyr" composé de six ou huit épaisseurs de feuilles de journal **1** sur lesquelles on ajoute un carton "épais" **2**.
- 2) Avec le marteau **5**, on frappe avec mesure sur le poinçon **4** tenu bien vertical et centré sur le cercle à percer.
- 3) Tous les quatre trous on débourre **4** avec l'ustensile **3**.

C'est facile, rapide et précis, en un mot : CONVIVIAL.



Ben Môa môa, je sais que le Nulentout est le seul informaticien au monde à programmer son ordinateur avec un marteau !

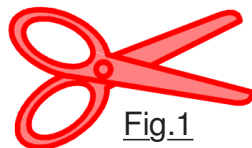


Fig.1

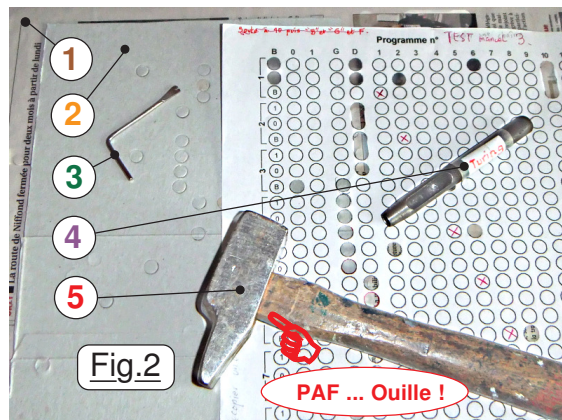


Fig.2

Chère lectrice, cher lecteur,

Initialement je n'avais pas du tout l'intention de fournir des programmes mis à part le petit algorithme proposé dans le chapitre 41 en page 27 du didacticiel, exemple de logiciel avec aiguillage sur deux options possibles. En effet, l'Internet regorge d'exemples tous aussi séduisants les uns que les autres. Puis, au moment de tester sévèrement le prototype, j'ai puisé sans vergogne dans la richesse mise en ligne.

Il a fallu se rendre à l'évidence : Une proportion considérable ne fonctionne pas correctement. Du reste, avant de m'engager dans la perforation d'une grille, je testais l'algorithme sur la machine de Turing "papier" décrite dans le chapitre 26.

Aussi, après avoir validé 66 programmes différents pour considérer que le prototype était fiable et que je pouvais envisager de publier en ligne, j'ai pensé qu'il serait bienvenu pour mes lecteurs d'avoir des exemples "qui fonctionnent". Sur ma machine il n'y a aucun problème, mais je ne suis absolument pas à l'abri, comme ceux qui ont mis en ligne leurs grilles, de m'être trompé ne serait-ce que d'un trou sur les documents que je vous propose. J'ai donc lu et relu trois fois "ma production" pour me convaincre qu'elle est sans erreur, espérant ainsi ne pas avoir laisser persister une erreur.

À mon sens, seules les mini-fiches d'Utilisation au format A6 doivent être imprimées et éventuellement plastifiées. Elles sont conçues Recto/Verso.

Les fiches au format A5 contiennent les tables de transition pour perforer vos feuilles de programme. À mon avis il n'est pas du tout indispensable de les imprimer. (Recto /Verso dans ce cas.) En effet, face à l'écran de l'ordinateur on coche sur la grille les trous à perforer ... et vogue la galère ...

Rien n'interdit avant, de les tester, et je vous le conseille fortement, avec une Machine de Turing version papier pour détecter une erreur de ma part toujours envisageable.

Comme j'ai horreur du vide, ces fiches sont saturées de divers renseignements complémentaires.

Programme utilisateur n°1.

A lgorithme prévu comme démonstrateur d'aiguillage, cette feuille perforée héberge deux logiciels binaires élémentaires. Un premier pion de donnée (*En jaune sur La Fig.1 de la mini-fiche.*) sert à sélectionner l'algorithme traité. Il est suivi d'une ou plusieurs séparations de type "**B**" avant la donnée représentée en rouge. La donnée BINAIRE est quelconque et comporte un nombre libre de BITS.

P lacer le BIT de sélection à l'état "**0**" ou à l'état "**1**". Grand classique de base en traitements binaires :

- Si le bit est à "**0**" tous les BITS de la donnée sont forcés à "**0**".
- Si le bit est à "**1**" tous les BITS de la donnée sont forcés à "**1**".
- Processus continu jusqu'à LECTURE d'un état séparateur "**B**".

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T4
	B:		G	
2	1:	1	G	T3
	0:	1	G	T3
	B:		G	
3	1:	1	G	
	0:	1	G	
	B:		D	T6
4	1:	0	G	T5
	0:	0	G	T5
	B:		G	
5	1:	0	G	
	0:	0	G	
	B:		D	T6
6	1:		D	
	0:		D	
	B:		D	T7
7	1:			F
	0:			F
	B:			

(1) : Sauter les séparateurs. (5) : Traite les autres BITS.
 (2) : Traite le premier BIT. (6) : Dégage la tête à gauche.
 (3) : Traite les autres BITS. (7) : FIN.
 (4) : Traite le premier BIT.

37 trous

Programme utilisateur n°3.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T5
	0:		G	T2
	B:		G	
2	1:	0	G	T3
	0:	1	G	T3
	B:		G	
3	1:	0	G	
	0:	1	G	
	B:		D	T4
4	1:		D	
	0:		D	
	B:		D	T11
5	1:			T6
	0:			T6
	B:		G	
6	1:		G	
	0:		G	T7
	B:		G	T10
7	1:		G	T8
	0:		G	
	B:		G	T10
8	1:		G	
	0:		G	
	B:		G	T9
9	1:			
	0:			
	B:	1	D	T11
10	1:			
	0:			
	B:	0	D	T11
11	1:			
	0:			
	B:			F

(1) : Aiguillage. Laisse sur le séparateur.

(2) : Cherche le Début de la donnée.

(4) : Dégage à Droite.

(5) : Cherche le Début de la donnée.

(9) : OUI

(10) : NON

Aiguillage :

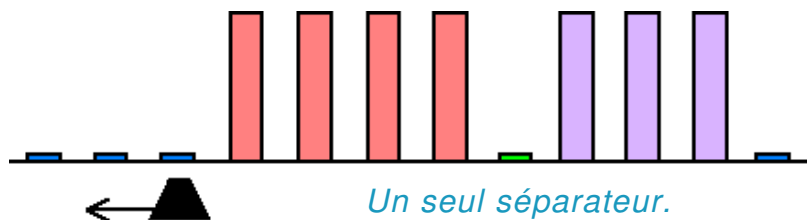
"1" : Trouver la sous-chaîne [01] dans une donnée Binaire.

"0" : Inverser tous les BITS d'une chaîne Binaire.

46 trous

Programme utilisateur n°4.

A lgorithme agencé pour concaténer deux chaînes de caractères **Unaires**. Les deux suites de "1" peuvent avoir une longueur quelconque, mais un seul "B" doit être utilisé pour les séparer.



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		G	T3
3	1:	B	D	T4
	0:			
	B:		D	T6
4	1:			
	0:			
	B:	1	G	T5
5	1:			
	0:			
	B:		G	T3
6	1:			
	0:			
	B:		D	T7
7	1:		D	
	0:			
	B:		D	T8
8	1:			
	0:			
	B:			F

(1) : Déplacer la tête de Lecture/Écriture sur le début de la donnée.
(6) et (7) : Dégager la tête de Lecture/Écriture à gauche de la donnée.

22 trous

Programme utilisateur n°2.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T6
	B:		G	
2	1:			T3
	0:			T3
	B:		G	
3	1:		G	
	0:		G	
	B:		G	4
4	1:			T5
	0:			T5
	B:		G	
5	1:		G	
	0:		G	
	B:		D	T11
6	1:		G	T7
	0:			
	B:		G	
7	1:		G	
	0:			
	B:		D	T8
8	1:	B	G	T9
	0:			
	B:			T11
9	1:			
	0:			
	B:	1	D	T10
10	1:			
	0:			
	B:		D	T8
11	1:			
	0:			
	B:			F

(1) : Aiguillage qui laisse sur "B".

(2) : Début de la première donnée.

(3) : Passe la première donnée.

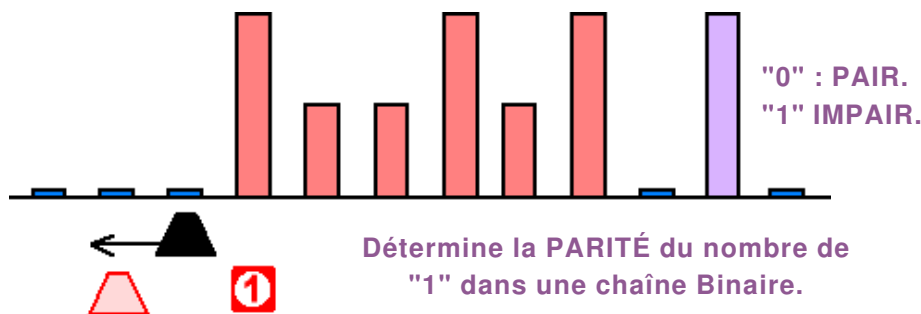
(4) et (5) : Traite la deuxième donnée.

(11) : FIN.

(6) à (10) : réalise le SHITF à Droite d'une suite de "1"

35 trous

Programme utilisateur n°5.



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	T2
2	1:		G	T3
	0:		G	
	B:		G	T4
3	1:		G	T2
	0:		G	
	B:		G	T5
4	1:			
	0:			
	B:	0	D	T6
5	1:			
	0:			
	B:	1	D	T6
6	1:			
	0:			
	B:		D	T7
7	1:		D	
	0:		D	
	B:		D	T8
8	1:			
	0:			
	B:			F

(1) : Chercher le premier BIT de la donnée.

(4) : PAIR.

(5) : IMPAIR.

(6) à (8) : Dégage de deux pions à gauche de la donnée.

26 trous

Programme utilisateur n°8.

Petit programme de Déminage. Un premier BIT d'aiguillage permet de choisir l'option du type de mines. Avec "0" recherche les mines de type "0", et avec "1" détecte les mines de type "1". Si aucune mine n'est trouvée dans le champ, positionne un Drapeau dont le type est celui de l'option de recherche. Si une mine est trouvée, la tête de Lecture/Écriture pointe son emplacement.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T6
	0:		G	T2
	B:		G	
2	1:		G	T3
	0:			T9
	B:		G	
3	1:		G	
	0:			T9
	B:		G	T4
4	1:			
	0:			
	B:	0	D	T5
5	1:			
	0:			
	B:		D	T9
6	1:			T9
	0:		G	T7
	B:		G	
7	1:			T9
	0:		G	
	B:		G	T8
8	1:			
	0:			
	B:	1	D	T5
9	1:			F
	0:			F
	B:			F

(1) : Aiguillage. "0" : Mines de type "0" et "1" : Mines de type "1".

(2) : Cherche le Début du champ de mines.

(3) : Cherche "0".

(4) : traite le Drapeau "0" ou "1".

(5) : Dégage à gauche.

(6) : Cherche le Début du champ de mines.

(7) : Cherche "1".

(8) : Drapeau.

32 trous

Programme utilisateur n°7.

Programme avec aiguillage qui pour "0" translate un code GRAI vers du Binaire pur et qui pour "1" transforme un Binaire pur vers son équivalent en code GRAI. Éviter une donnée de plus de quatre BITS pour simplifier la table de transcodage. Pour les deux données les poids faibles sont à droite et les poids forts à gauche de façon conventionnelle. (La plus grande valeur sera 15.)

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T7
	0:		G	T2
	B:		G	
2	1:			T3
	0:			T3
	B:		G	
3	1:		G	T4
	0:		G	
	B:		D	T5
4	1:	0	G	T3
	0:	1	G	
	B:		D	T5
5	1:		D	
	0:		D	
	B:		D	T6
6	1:		D	
	0:		D	
	B:			F
7	1:			T8
	0:			T8
	B:		G	
8	1:		G	T9
	0:		G	
	B:		D	T5
9	1:	0	G	
	0:	1	G	T8
	B:		D	T5

(1) : Aiguillage. "0" : GRAI vers Binaire. "1" : Binaire vers GRAI.

(2) : Cherche le Début de la donnée.

(5) : Ramène à gauche de la donnée.

(6) : FIN du programme.

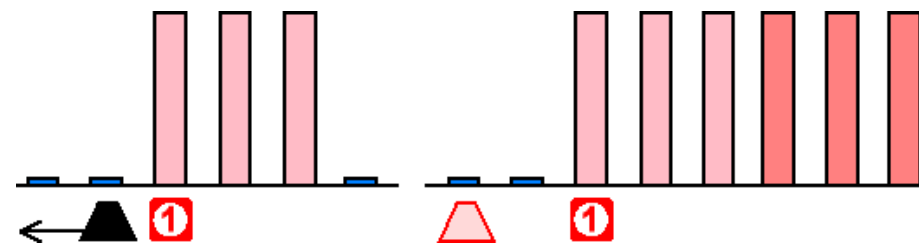
(7) : Cherche le Début de la donnée.

42 trous

(3) et (4) : Code Grai vers Binaire.
(6) et (7) : Code Binaire vers GRAI.

Programme utilisateur n°6.

Algorithme agencé pour doubler une chaîne binaire constituée d'une suite de "1". La tête termine deux séparateurs à gauche de la donnée. Éviter plus de quatre BITS à "1" dans cette multiplication *Unaire par deux* pour éviter un nombre de mouvements prohibitif sur la machine. (Grand maximum : 24 BITS dans la chaîne.)



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		D	T3
3	1:	0	G	T4
	0:		D	
	B:		G	T5
4	1:			
	0:		G	
	B:	0	D	T3
5	1:			
	0:	1	G	
	B:		D	T6
6	1:		D	
	0:			
	B:		D	T7
7	1:			
	0:			
	B:			F

(1) : Aller au début de la donnée.

(2) : Aller à la fin de la chaîne.

(6) et (7) : Dégage de deux pions à gauche de la donnée.

(7) : FIN.

23 trous

Programme utilisateur n°9.

A lgorithme avec sélecteur qui pour "0" effectue une ADDITION Unaire et qui pour "1" fait une SOUSTRACTION Unaire.

ATTENTION : Il faut un seul séparateur entre X et Y et $Y \geq X$.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T6
	0:		G	T2
	B:		G	
2	1:			T3
	0:			
	B:		G	
3	1:	B	G	T4
	0:			
	B:			
4	1:		G	
	0:			
	B:	1		T5
5	1:			F
	0:			
	B:			F
6	1:		G	T7
	0:			
	B:		G	
7	1:		G	
	0:			
	B:		G	T8
8	1:	B	D	T9
	0:			
	B:		G	
9	1:	B	D	T10
	0:			
	B:		D	
10	1:		G	T8
	0:			
	B:			T5

$$R = Y + X$$

$$R = Y - X$$

Aiguillage :
 "0" : ADDITION Unaire.
 "1" : SOUSTRACTION Unaire.

(2) : Aller au Début de la Donnée.
 (3) et (4) : Traite l'ADDITION.
 (6) : Aller sur X.
 (7) : <aller sur Y
 (8) à (10) : Traite la SOUSTRACTION.

32 trous

Programme utilisateur n°12.

A lgorithme de transcodage en BINAIRE pur d'un Entier indiqué en Unaire. (Efface la donnée initiale.) Prévoir de l'espace de type "B" à gauche de la donnée initiale en fonction du résultat attendu.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		D	T3
3	1:	B	D	T4
	0:			
	B:			T8
4	1:		D	
	0:			
	B:		D	T5
5	1:	0	D	
	0:	1	G	T6
	B:	1	G	T6
6	1:		G	
	0:		G	
	B:		G	T7
7	1:		G	
	0:			
	B:		D	T3
8	1:		D	T9
	0:		D	T9
	B:		D	
9	1:		D	
	0:		D	
	B:		D	T10
10	1:			
	0:			
	B:			F

(1) : Aller sur le premier pion.
 (2) : Aller sur le dernier "1".
 (3) à (7) : Transcoder en binaire pur.
 (8) et (9) : Dégager à gauche du résultat.

37 trous

Programme utilisateur n°11.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:	0	D	T2
2	1:			
	0:			
	B:	0	D	T3
3	1:			
	0:			
	B:	0	D	T4
4	1:			
	0:			
	B:		D	T5
5	1:			
	0:			
	B:	1	D	T6
6	1:			
	0:			
	B:	1	D	T7
7	1:			
	0:			
	B:	1	D	T8
8	1:			
	0:			
	B:		D	T9
9	1:			
	0:			
	B:	0	D	T10
10	1:			
	0:			
	B:	0	D	T11
11	1:			
	0:			F
	B:	0		

Codage de **SOS** en Morse. Les "0" représentent des points et les "1" des traits.

(1) à (3) : Lettre S en Morse.
 (4) : Séparation en Morse.
 (5) à (7) : Lettre O en Morse.
 (8) : Séparation en Morse.
 (9) à (11) : Lettre S en Morse.

30 trous

Programme utilisateur n°10.

Programme qui calcule la valeur de 2 élevé à la puissance *N*.
 La valeur de *N* est exprimé en *Unaire*. Éviter de choisir *N* supérieur à 3 car cet algorithme génère beaucoup de mouvements.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		D	T3
3	1:	0	G	T4
	0:	1	D	
	B:		G	T7
4	1:		G	
	0:		G	
	B:		G	T5
5	1:		G	
	0:			
	B:	1	D	T6
6	1:		D	
	0:			
	B:		D	T3
7	1:		G	
	0:			
	B:		G	T8
8	1:		G	
	0:			
	B:	1	G	T9
9	1:		D	
	0:			
	B:		D	T10
10	1:			F
	0:			
	B:			

(1) : Aller au Début de la Donnée.
 (2) : Aller à la Fin de la Donnée.
 (9) : Dégage la tête L/E à Droite du résultat.

35 trous

Programme utilisateur n°13.

Réciproque du programme n°12 cet algorithme effectue le transcodage vers un Entier indiqué en Unaire d'une valeur exprimée en **BINAIRE pur**. Les poids forts sont à gauche et il faut prévoir au moins N+1 espaces de type "**B**" à droite de la donnée.

ATTENTION : pas de "**0**" en tête, c'est à dire à gauche du poids fort.

(Pour la valeur 5 effectue déjà 62 instructions sur la machine.)

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:			
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T3
3	1:	0	D	T4
	0:	1	D	
	B:			T9
4	1:		G	T6
	0:		G	T6
	B:		G	T5
5	1:		G	T6
	0:	B	G	T6
	B:			
6	1:		G	
	0:		G	
	B:		G	T7
7	1:		G	
	0:			
	B:	1	D	T8
8	1:		D	
	0:			
	B:		D	T3
9	1:			
	0:			
	B:			TF

(1) et (2) : Cherche le poids faible.

(3) : Fin du programme.

(3) à (8) : Effectue le transcodage de BINAIRE vers ENTIER Unaire.

(9) : Fin du déroulement et réinitialisation de la machine.

36 trous

Page 9

Programme utilisateur n°16.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	G	T2
	0:		D	
	B:		G	T10
2	1:			
	0:		G	
	B:		G	T3
3	1:		G	
	0:			
	B:		D	T4
4	1:	0	G	T5
	0:		D	
	B:		G	T8
5	1:			
	0:		G	
	B:		G	T6
6	1:		G	
	0:			
	B:	1	D	T7
7	1:		D	
	0:			
	B:		D	T4
8	1:			
	0:	1	G	
	B:		D	T9
9	1:		D	
	0:			
	B:		D	T1
10	1:			
	0:	1	G	
	B:		D	T11
11	1:		D	
	0:			
	B:			F

ATTENTION : Pour la multiplication de 3 x 2 la machine effectue déjà 118 cycles d'HORLOGE.

Multiplication de deux Entiers exprimés en Unaire.
Laisse la tête de L/E sur le début de la première donnée.

41 trous

Programme utilisateur n°15.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:	1	G	T2
2	1:			
	0:			
	B:	0	G	T3
3	1:			
	0:			
	B:	1	G	T4
4	1:			
	0:			
	B:	0	G	T5
5	1:			
	0:			
	B:		G	T6
6	1:			
	0:			
	B:	1	G	T7
7	1:			
	0:			
	B:	1	G	T8
8	1:			
	0:			
	B:	0	G	T9
9	1:			
	0:			
	B:	1	D	T10
10	1:		D	
	0:		D	
	B:		D	T11
11	1:		D	
	0:		D	
	B:			F

Codage de l'appel **CQ** en Morse. Les "0" représentent des points et les "1" des traits.

(1) à (4) : Lettre C en Morse.
 (5) : Séparation en Morse.
 (6) à (9) : Lettre Q en Morse.
 (10) à (11) : Dégagement et Fin.

33 trous

Programme utilisateur n°14.

Programme qui effectue la division euclidienne par 3 d'un **Entier exprimé en Unaire**. Le Quotient est positionné à droite de la donnée. Les BITS de la valeur initiale sont tous remplacés par des "0". Il faut prévoir au moins trois espaces de type "B" à gauche de la **donnée** pour pouvoir "dégager" la valeur du **Reste**.

(Pour la valeur 7 effectuée déjà 33 instructions sur la machine.)

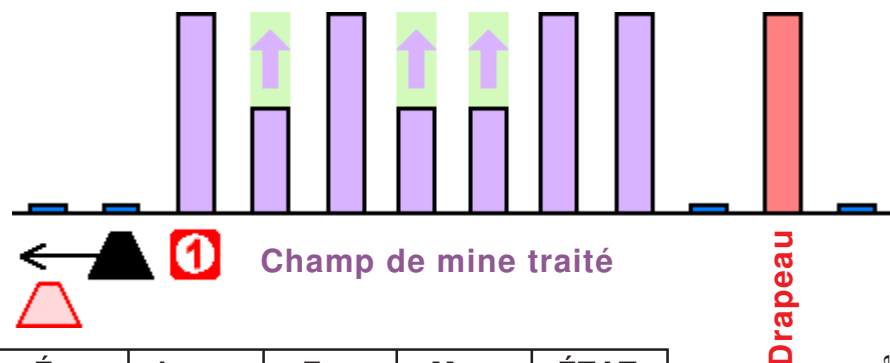
État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	D	T2
	0:		D	
	B:		D	T9
2	1:	0	D	T3
	0:			
	B:		D	T8
3	1:	0	G	T4
	0:			
	B:		D	T7
4	1:			
	0:		G	
	B:		G	T5
5	1:		G	
	0:			
	B:	1	D	T6
6	1:		D	
	0:			
	B:		D	T1
7	1:			
	0:			
	B:	1	D	T8
8	1:			
	0:			
	B:	1		T9
9	1:			F
	0:			
	B:			F

ATTENTION : La tête de lecture/Ecriture doit se trouver à **Droite de la Donnée**.

33 trous

Programme utilisateur n°17.

Petit programme de *Déminage* assez voisin du programme n°8. Toutefois il ne recherche que les mines enterrées de type "0". Toutes les mines du champ contaminé sont traitées et les "trous" remplacés par des "1". Quand la totalité du champ a été sécurisé le signale par un Drapeau positionné à "1" et placé à droite de la zone traitée.



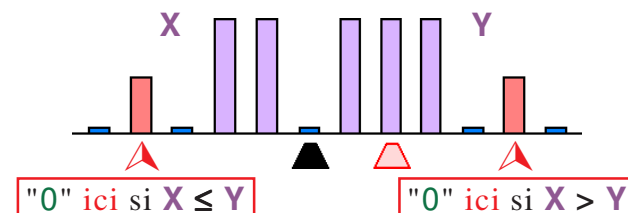
État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	
	0:	1	G	
	B:		G	T3
3	1:			
	0:			
	B:	1	D	T4
4	1:			
	0:			
	B:		D	T5
5	1:		D	
	0:			
	B:		D	T6
6	1:			
	0:			
	B:			F

- (1) : Cherche le début du champ de mines.
 (2) : Démine l'intégralité du champ.
 (3) : Plante le drapeau.
 (4) : Revient au début de la zone traitée.
 (5) : Dégage à gauche du champ traité.
 (6) : Fin du programme et réinitialisation machine.

17 trous

Programme utilisateur n°20.

Réalise la comparaison entre deux Entiers codés sous forme *Unaire*. laisse la tête de Lecture/Ecriture dans une position variable à la fin du traitement.



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	D	T2
	0:		D	
	B:		D	
2	1:		G	T3
	0:			
	B:		D	T5
3	1:	0	G	T4
	0:		G	
	B:		G	
4	1:		D	T1
	0:			
	B:		G	T7
5	1:			
	0:			
	B:	0	G	T6
6	1:			F
	0:	1	G	
	B:		G	
7	1:			
	0:			
	B:	0	D	T8
8	1:			F
	0:	1	D	
	B:		D	

Un seul "B" sépare les données X et Y et la tête de Lecture/Ecriture doit initialement se trouver sous ce séparateur de X et de Y.

32 trous

Programme utilisateur n°19.

A lgorithme qui *inscrit sous forme Unaires sur la machine la suite de FIBONACCI*. Logiquement l'ensemble du barillet doit être initialisé avec des "B".

$$Un+2 = Un + Un+1$$

 **Pas de Fin.** 

1 - 1 ⇒ 2 ⇒ 3 ⇒ 5 ⇒ 8 ⇒ 13 ⇒ 21 ⇒ 34 ...

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	G	T2
	0:			
	B:		G	T8
2	1:		G	
	0:			
	B:		G	T3
3	1:		G	
	0:			
	B:		G	T4
4	1:		G	
	0:			
	B:	1	D	T5
5	1:		D	
	0:			
	B:		D	T6
6	1:		D	
	0:	1	G	T8
	B:		D	T7
7	1:		D	
	0:	1	G	T1
	B:			
8	1:	0	G	T3
	0:			
	B:		D	T9
9	1:		D	
	0:			
	B:		G	T1

Les deux premiers termes sont chacun égaux à un et la tête de L/E doit se trouver sous celui de gauche. Sur la machine on ne dispose au maximum d'une étendue de 56 positions. Il y aura perte par "écrasement" des données et divergence du programme à partir de l'élément 21 inclus.

36 trous

Programme utilisateur n°18.

G rand classique en génie informatique, cet algorithme *réalise la Conjecture de SYRACUSE* découverte par Collatz en 1930, également nommée conjecture de Ulam. (*Le terme de Syracuse fait référence à l'université de Syracuse qui se trouve dans la ville Syracuse est située dans le sud-est de la Sicile, au bord de la mer Ionienne.*)

Prendre un nombre entier **N** :

- Si **N** est PAIR, *le diviser par 2.*
- Si **N** est IMPAIR, prendre $(3 \times N) + 1$.
- Recommencer avec les entiers successifs obtenus. La conjecture de Collatz prédit qu'*à la fin, on obtient toujours 1.*

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		D	T2
	0:	B	D	
	B:			
2	1:		G	T4
	0:		G	T4
	B:			T7
3	1:		D	T4
	0:		D	
	B:		G	T6
4	1:	0	D	T5
	0:	1	D	T3
	B:	1	G	T6
5	1:		D	
	0:		D	T4
	B:	0	D	T4
6	1:		G	
	0:		G	
	B:		D	T1
7	1:			
	0:			
	B:			F

Prévoir beaucoup d'espaces "B" à gauche de la donnée initiale, car le programme va écrire codé en binaire chaque terme de la suite en décalant d'un BIT à gauche sur le barillet.

34 trous

Programme utilisateur n°21.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T3
3	1:	B	G	T4
	0:	B	G	T5
	B:			T6
4	1:			
	0:			
	B:	1	D	T2
5	1:			
	0:			
	B:	0	D	T2
6	1:			T7
	0:			T7
	B:		G	
7	1:		G	
	0:		G	
	B:		D	T8
8	1:	B	D	T9
	0:	B	D	T10
	B:			
9	1:		D	
	0:		D	
	B:	1	D	T11
10	1:		D	
	0:		D	
	B:	0	D	T11
11	1:			
	0:			
	B:			F

Réalise la permutation circulaire
à Droite d'une donnée exprimée
en Binaire pur.

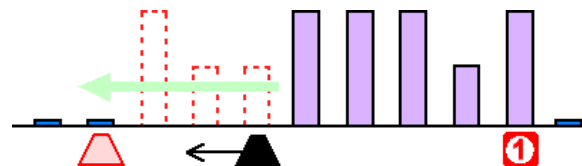
- (1) : Déplace la tête de L/E au Début de la donnée.
 (2) : Déplace la tête de L/E au à la Fin du Binaire.
 (3) à (5) : Décale d'un BIT à Droite la donnée binaire.
 (6) et (7) : La tête de L/E va sur la "Retenue".
 (8) à (10) : Recopie la Retenue à Gauche de la donnée.

44 trous

Page 13

Programme utilisateur n°24.

Calcule cinq fois l'entier N exprimé en Binaire pur. Laisse la tête de L/E à gauche du résultat final. La donnée binaire initiale peut présenter une taille quelconque, mais **il faut prévoir un espace de "B" suffisant situé à gauche de la donnée pour pouvoir écrire les poids forts du résultat.**



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T3
3	1:		D	T4
	0:		D	
	B:			T8
4	1:		D	T6
	0:		D	T5
	B:	0	D	T5
5	1:	0	D	T6
	0:	1	D	T3
	B:	1	D	T8
6	1:	0	D	T7
	0:	1	D	T5
	B:	1	D	T5
7	1:		D	
	0:		D	
	B:	0	D	T4
8	1:			
	0:			
	B:			F

NOTE : Des "0" en tête, à gauche du poids fort sont autorisés mais ils seront conservés.

- (1) : Aller au Début de la donnée.
 (2) : Aller à la Fin de la donnée.

42 trous

Programme utilisateur n°23.

Incrémenter / Décrémenter d'une unité une donnée codée en Binaire en fonction d'un BIT de sélection. Ce BIT d'option est situé **à droite** de la donnée Binaire séparé par un seul "B".

- Sélecteur = "0" : **Décrémente** de 1.
- Sélecteur = "1" : **Incrémente** de 1.

ATTENTION : La tête de lecture/Ecriture doit être à **Droite** du BIT d'**Option**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		D	
2	1:		D	T3
	0:		D	T5
	B:			
3	1:			T4
	0:			T4
	B:		D	
4	1:	0	D	
	0:	1		T7
	B:	1		T7
5	1:			T6
	0:			T6
	B:		D	
6	1:	0		T7
	0:	1	D	
	B:			
7	1:		D	
	0:		D	
	B:			T8
8	1:			
	0:			
	B:			F

- (1) : Aller sur l'aiguillage.
 (2) : Aiguillage qui laisse sur "B".
 (3) : Aller au Début de la donnée.
 (4) Ajouter 1 à la donnée.
 (5) : Aller au Début de la donnée.
 (6) : Ajouter 1 à la donnée.
 (7) : Dégager à gauche.

27 trous

Programme utilisateur n°22.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B	D	T2
	0:	B	D	T3
	B:			
2	1:			
	0:			
	B:	1	G	T4
3	1:			
	0:			
	B:	0	G	T4
4	1:			
	0:			
	B:		G	T5
5	1:			T1
	0:			T1
	B:		D	T6
6	1:			T7
	0:			T7
	B:		D	
7	1:		D	
	0:		D	
	B:		G	T8
8	1:	B	G	T9
	0:	B	G	T10
	B:			
9	1:		G	
	0:		G	
	B:	1	G	T11
10	1:		G	
	0:		G	
	B:	0	G	T11
11	1:			
	0:			
	B:			F

- (1) à (4) : Décale d'un BIT à Gauche la donnée binaire.
 (5) à (7) : La tête de L/E va sur la "Retenue".
 (8) à (10) : Recopie la Retenue à Droite de la donnée.

42 trous

Réalise la permutation circulaire à Gauche d'une donnée exprimée en Binaire pur.

Programme utilisateur n°25.

Calculer la division par cinq de l'Entier **N** exprimé en Binaire pur. Laisse la tête de L/E à gauche du résultat final. *Si le résultat n'est pas entier, positionne un à "0" à gauche de N.*

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T3
3	1:		D	T4
	0:		D	
	B:		G	T8
4	1:		D	T6
	0:		D	T5
	B:		D	T9
5	1:	0	D	T3
	0:	1	D	T6
	B:		D	T9
6	1:	0	D	T5
	0:	1	D	T7
	B:		D	T9
7	1:		D	
	0:		D	T4
	B:		D	T9
8	1:		D	T10
	0:	B	G	
	B:			
9	1:			
	0:			
	B:	0	D	T10
10	1:			F
	0:			
	B:			F

$$R = N / 5$$

Si **R** NON
Entier
positionne
un
drapeau
"0".

- (1) : Déplace la tête de L/E au Début de la donnée Binaire.
(2) : Déplace la tête de L/E à la Fin de la donnée Binaire.

48 trous

Programme utilisateur n°28.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B	D	T6
	0:	B	D	T2
	B:		D	
2	1:	B	D	T3
	0:	B	D	T3
	B:	B	D	T3
3	1:	B	D	T4
	0:	B	D	T4
	B:	B	D	T4
4	1:	0	D	T5
	0:	0	D	T5
	B:	0	D	T5
5	1:	1	D	T4
	0:	1	D	T4
	B:	1	D	T4
6	1:	B	D	T7
	0:	B	D	T7
	B:	B	D	T7
7	1:	B	D	T8
	0:	B	D	T8
	B:	B	D	T8
8	1:	0	D	T9
	0:	0	D	T9
	B:	0	D	T9
9	1:	0	D	T10
	0:	0	D	T10
	B:	0	D	T10
10	1:	1	D	T11
	0:	1	D	T11
	B:	1	D	T11
11	1:	1	D	T8
	0:	1	D	T8
	B:	1	D	T8

Programme avec sélection du motif
qui permet de **construire une frise**
sans FIN sur le barillet.

- (1) : Aiguillage qui laisse un "B".
(2) et (3) : Créé deux "blancs".
(4) et (5) : Trace le motif n°1.
(6) et (7) : Créé deux "blancs".
(8) à (11) : Trace le motif n°2.

97 trous

Programme utilisateur n°27.

Algorithme qui détecte si une donnée **N** écrite en Binaire pur constitue ou non un **PALINDROME**. Laisse la tête de L/E à gauche du BIT traduisant le résultat : "0" : OUI, "1" : NON.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:		D	T2
2	1:	B	D	T6
	0:	B	D	T3
	B:		D	T9
3	1:		D	
	0:		D	
	B:		G	T4
4	1:	B	D	T8
	0:	B	G	T5
	B:		D	T9
5	1:		G	
	0:		G	
	B:		D	T2
6	1:		D	
	0:		D	
	B:		G	T7
7	1:	B	G	T5
	0:		D	T8
	B:		D	T9
8	1:	0	D	
	0:	0	D	
	B:	0	D	T10
9	1:			
	0:			
	B:	1	D	T10
10	1:			
	0:			
	B:			F

ATTENTION : La tête de lecture/Ecriture doit être immédiatement à Droite de **N**.

Efface la donnée initiale **N**.

(8) : NON.
(9) : OUI.

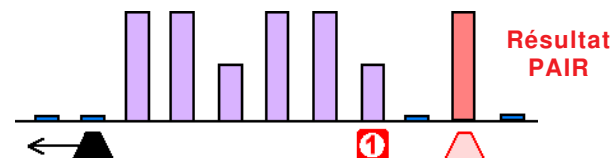
48 trous

Programme utilisateur n°26.

Vérifie si le nombre de "0" est PAIR ET le nombre de "1" est PAIR sur une donnée **N** binaire. Laisse la Tête de L/E à gauche du drapeau (BIT.) traduisant le résultat :

"0" : Le Nbre de "0" ou de "1" n'est pas Pair.

"1" : Le Nbre de "0" ET de "1" est PAIR.



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	T3
	0:		G	T5
	B:		G	T6
3	1:		G	T2
	0:		G	T4
	B:		G	T7
4	1:		G	T5
	0:		G	T3
	B:		G	T7
5	1:		G	T4
	0:		G	T2
	B:		G	T7
6	1:			
	0:			
	B:	1	D	T8
7	1:			
	0:			
	B:	0	D	T8
8	1:			
	0:			
	B:			F

(1) : Chercher le premier BIT de la donnée.

(6) : Positionne le Drapeau PAIR.

(7) : Positionne le Drapeau NON pair.

34 trous


Programme utilisateur n°29.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:	B	G	T7
	0:	B	G	T3
	B:			T11
3	1:		G	
	0:		G	
	B:		G	T4
4	1:		G	
	0:		G	
	B:	0	D	T5
5	1:		D	
	0:		D	
	B:		D	T6
6	1:		D	
	0:		D	
	B:	0	G	T2
7	1:		G	
	0:		G	
	B:		G	T8
8	1:		G	
	0:		G	
	B:	1	D	T9
9	1:		D	
	0:		D	
	B:		D	T10
10	1:		D	
	0:		D	
	B:	1	G	T2
11	1:			
	0:			
	B:			F

- (1) : Aller au début de la chaîne
 (2) : Force le BIT à "B".
 (3) : Va sur la copie.
 (4) : Recopie le "0".
 (6) : Restitue le "0".
- (7) : Va sur la COPIE.
 (8) : Recopie le "1".
 (9) : Retourne sur l'ORIGINAL
 (10) : Restitue le "1".

47 trous

Programme utilisateur TEST 2.

Algorithme vraiment élémentaire, avec aiguillage , qui permet en opérations de maintenance de générer à convenance deux types de vérifications. La première lorsque le BIT de sélection est à "1" consiste à tester l'aptitude d'une ligne avec F en mode AUTOM de traiter une écriture ou un déplacement. La deuxième avec "B" ou "0" d'écrire systématiquement un "0" puis d'effectuer un déplacement à Gauche. Cette option permet de vérifier que LECTURE ne s'arrête plus intempestivement et que l'écriture de "0" se déclenche sur les deux unités et retourne son accusé de réception. (Programme envisagé lors d'apparitions aléatoires de problèmes d'écritures des "0".)


- BIT d'option à "1" : Saut en 2 et **test INFINI** d'écritures de "0".
- BIT d'option à "0" : } Saut en 4, puis en 10. Écriture d'un
- BIT d'option à "B" : } "0", mouvement à Gauche et Fin.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T4
	B:			T4
2	1:			T10
	0:			T10
	B:			T10
3	1:			
	0:			
	B:			
4	1:	0	G	
	0:	0	G	
	B:	0	G	
10	1:	B	G	F
	0:	B	G	F
	B:	B	G	F

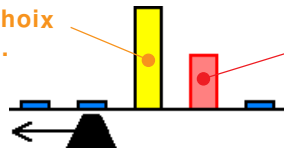
- (1) : BIT d'OPTION.
 (2) : Boucle sans fin sur "0" et "G".
 (3) : "Séparation visuelle" sur la grille.
 (4) : Fait sauter à la Fin quel que soit l'état de lecture.
 (5) : Écriture d'un "B", mouvement à Gauche et Fin.

21 trous

Programme utilisateur **TEST.**

Algorithme vraiment élémentaire, avec aiguillage , qui permet en opérations de maintenance de générer à convenance une erreur de type **ER 1** ou une erreur de type **ER 2**. Deux pions doivent initialement être positionnés sur la machine avant d'exécuter le processus en mode **AUTOM.**

BIT d'option pour le choix entre ER 1 et ER 2.



BIT pour sélectionner la configuration qui engendre l'erreur.

Le BIT d'**option** à "0" provoque **ER 1**.

Le BIT d'**option** à "1" provoque **ER 2**.

Effet du BIT de sélection de la configuration critique :

- * Sur **Option** "0" consigne deux écritures contradictoires.
- * Sur **Option** "1" consigne deux ou trois Transitions simultanées.
- "1" avec **BIT d'option** à "0" : Ecriture d'un "B" et d'un "1".
- "0" avec **BIT d'option** à "0" : Ecriture d'un "B" et d'un "0".
- "B" avec **BIT d'option** à "0" : Ecriture d'un "0" et d'un "1".
- "1" avec **BIT d'option** à "1" : Transition 1 et Transition 11.
- "0" avec **BIT d'option** à "1" : Transition 3 et Transition 9.
- "B" avec **BIT d'option** à "1" : Transition 5, 6 et Transition 7.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:	B	G	T4
	0:	B	G	T3
	B:			
3	1:	B 1		
	0:	B 0		
	B:	0 1		
4	1:			1 11
	0:			3 9
	B:			5 6 7

(1) : Aller sur le BIT d'option.

(2) : Choix type d'erreur désiré.

(3) : Configuration de l'erreur **ER 1**.

(4) : Configuration de l'erreur **ER 2**.

22 trous

Programme utilisateur n°30.

Typique des opérations binaires sur les microcontrôleurs, **ce logiciel simule le fonctionnement d'un Registre FIFO**. Ces opérations matérialisent des files d'attente en mémoire.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:	B	G	T3
	0:	B	G	T7
	B:		G	T9
3	1:		G	
	0:		G	
	B:		G	T4
4	1:	1	D	T5
	0:	1	D	T5
	B:	1	D	T5
5	1:			
	0:			
	B:		D	T6
6	1:		D	
	0:		D	
	B:		G	T2
7	1:		G	
	0:		G	
	B:		G	T8
8	1:	0	D	T5
	0:	0	D	T5
	B:	0	D	T5
9	1:	B	D	T10
	0:	B	D	T10
	B:			
10	1:			
	0:			
	B:			F

Dans cet algorithme **le Registre Binaire est supposé avoir été chargé de la Droite vers la Gauche**, le BIT d'écriture et de lecture étant à Droite.

- (1) : Aller sur le Registre.
 (3) et (4) : Sortir le BIT d'état "1".
 (5) et (6) : Retour sur le Registre.
 (7) et (8) : Sortir le BIT d'état "0".
 (9) : Purge le BIT de sortie.

50 trous

Programme utilisateur n°31.

Exemples de Castors affairés avec deux options. La zone concernée doit contenir au moins cinq "B" à gauche du pion de sélection et au moins quatre "B" à droite. Quand le programme est activé il faut que la tête de Lecture/Écriture soit au dessus du pion de sélection. Si cette position est un "B" il sera considéré comme un "0".

OPTION	États	symboles	Cycles	Score
"0"	2	2	6 + 2	4
"1"	3	2	21 + 2	5

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B		T5
	0:	B		T2
	B:			T2
2	1:		G	T3
	0:			
	B:	1	D	T3
3	1:		D	T8
	0:			
	B:	1	G	T2
4	1:			
	0:			
	B:			
5	1:			T8
	0:			
	B:	1	D	T6
6	1:	B	D	T7
	0:			
	B:	1	G	
7	1:		G	T5
	0:			
	B:	1	G	
8	1:			F
	0:			
	B:			

(1) : Sélection de l'OPTION.

(2) et (3) : Castor affairé six cycles et Score = 4.

(4) : Séparation "visuelle" de trois lignes sur la grille.

(5) à (7) : Castor affairé 21 cycles et Score = 5.

29 trous

Programme utilisateur n°33.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B		T6
	0:	B		T2
	B:			T2
2	1:		G	T3
	0:	1	G	T3
	B:	1	G	T3
3	1:	B	G	T4
	0:	B	G	T4
	B:		G	T4
4	1:	B	G	T5
	0:	B	G	T5
	B:		G	T5
5	1:	B	G	T2
	0:	B	G	T2
	B:		G	T2
6	1:		G	T7
	0:	1	G	T7
	B:	1	G	T7
7	1:	B	G	T8
	0:	B	G	T8
	B:		G	T8
8	1:	B	G	T9
	0:	B	G	T9
	B:		G	T9
9	1:	B	G	T10
	0:	B	G	T10
	B:		G	T10
10	1:	B	G	T11
	0:	B	G	T11
	B:		G	T11
11	1:	B	G	T6
	0:	B	G	T6
	B:		G	T6

(1) : OPTION.

(2) : Plante le piquet.

(3) à (5) : Trois espaces puis retour.

(6) : Plante le piquet.

(7) à (11) : Cinq espaces puis retour.

Simule la réalisation d'une clôture.

OPTION "B" ou "0" : Trois espaces entre les piquets.

OPTION "1" : Cinq espaces entre les piquets.

85 trous

Programme utilisateur n°34.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B	G	T7
	0:	B	D	T2
	B:		D	T2
2	1:		D	T3
	0:			
	B:		D	
3	1:	B	G	T4
	0:	B	G	T4
	B:		G	T4
4	1:		G	T5
	0:			
	B:			
5	1:		G	T6
	0:			
	B:	1	G	
6	1:	B		T11
	0:	B		T11
	B:			T11
7	1:		G	T8
	0:			
	B:		G	
8	1:	B	D	T9
	0:	B	D	T9
	B:		D	T9
9	1:		D	T10
	0:			
	B:			
10	1:		D	T6
	0:			
	B:	1	D	
11	1:			
	0:			
	B:			F

Simule la réalisation d'un mur entre deux limites.

OPTION "B" ou "0" : Réalise deGauche à Droite.

OPTION "1" : Réalise de Droite à Gauche.

(1) : OPTION.

(2) : va sur la limite.

(3) : Efface à Gauche.

(4) : Revient.

(5) : Construit un "pion".

(6) : Efface à Droite ...

48 trous

Programme utilisateur n°32.

Exemples de Castors affairés avec deux options. La zone concernée doit contenir au moins le nombre de "B" précisés dans le tableau donné ci-dessous. Quand le programme est activé il faut que la tête de Lecture/Écriture soit au dessus du pion de sélection.

OPTION	États	symboles	Cycles	Score	"B" à Ga	"B" à Dr
"B" ou "0"	2	2	38 + 2	9	3	7
"1"	4	2	107 + 2	13	4	11

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B		T5
	0:	B		T2
	B:			T2
2	1:			T10
	0:	1	G	T3
	B:	0	D	T3
3	1:	0	G	
	0:	1	D	
	B:	1	G	T2
4	1:		G	T6
	0:			
	B:	1	D	T6
5	1:	B	G	T7
	0:			
	B:	1	G	T5
6	1:		G	T8
	0:			
	B:	1	D	T10
7	1:	B	D	T5
	0:			
	B:	1	D	
8	1:			F
	0:			
	B:			F

(1) : Aiguillage. "B" ou "0" : SCORE = 9 et "1" : SCORE 13.

(2) et (3) : Traite le Castor Affairé de score 9.

(4) : Séparation visuelle dans la grille de programme.

(5) à (8) : Traite le Castor Affairé de score 13.

(9) : Séparation visuelle dans la grille de programme.

42 trous

Programme utilisateur n°35.

Programme de calcul de la valeur $5 * N + 1$ en BINAIRE pur.
Peut calculer des valeurs vraiment très grandes. Maximum 2 élevé à la puissance 52 soit la bagatelle de 9.007.199.254.740.991 !

ATTENTION : Le nombre de cycles d'HORLOGE augmente très rapidement avec la valeur initiale proposée pour **N**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T6
3	1:			T6
	0:			T6
	B:			T9
4	1:		D	T5
	0:		D	
	B:			T9
5	1:		D	T7
	0:		D	T6
	B:	0	D	T6
6	1:	0	D	T7
	0:	1	D	T4
	B:	1	D	T9
7	1:	0	D	T8
	0:	1	D	T6
	B:	1	D	T6
8	1:		D	
	0:		D	T5
	B:	0	D	T5
9	1:			
	0:			
	B:		D	F

Prévoir un nombre suffisant de "B" à
Gauche de la donnée **N**.

- (1) : Trouver la donnée.
(2) : Aller à la fin de la donnée.
(3) à (8) : Calcule $(5 * N) + 1$.
(9) : Décale d'un pion et FIN.

49 trous

Programme utilisateur n°38.

Programme de calcul de la valeur $5 * N + 4$ en BINAIRE pur.
Peut calculer des valeurs vraiment très grandes. Maximum 2 élevé à la puissance 52 soit la bagatelle de 9.007.199.254.740.991 !

ATTENTION : Le nombre de cycles d'HORLOGE augmente très rapidement avec la valeur initiale proposée pour **N**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T8
3	1:			T8
	0:			T8
	B:			T9
4	1:		D	T5
	0:		D	
	B:			T9
5	1:		D	T7
	0:		D	T6
	B:	0	D	T6
6	1:	0	D	T7
	0:	1	D	T4
	B:	1	D	T9
7	1:	0	D	T8
	0:	1	D	T6
	B:	1	D	T6
8	1:		D	
	0:		D	T5
	B:	0	D	T5
9	1:			
	0:			
	B:		D	F

Prévoir un nombre suffisant de "B" à
Gauche de la donnée **N**.

- (1) : Trouver la donnée.
(2) : Aller à la fin de la donnée.
(3) à (8) : Calcule $(5 * N) + 1$.
(9) : Décale d'un pion et FIN.

49 trous

Programme utilisateur n°37.

Programme de calcul de la valeur $5 * N + 3$ en BINAIRE pur.
Peut calculer des valeurs vraiment très grandes. Maximum 2 élevé à la puissance 52 soit la bagatelle de 9.007.199.254.740.991 !

ATTENTION : Le nombre de cycles d'HORLOGE augmente très rapidement avec la valeur initiale proposée pour **N**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T7
3	1:			T7
	0:			T7
	B:			T9
4	1:		D	T5
	0:		D	
	B:			T9
5	1:		D	T7
	0:		D	T6
	B:	0	D	T6
6	1:	0	D	T7
	0:	1	D	T4
	B:	1	D	T9
7	1:	0	D	T8
	0:	1	D	T6
	B:	1	D	T6
8	1:		D	
	0:		D	T5
	B:	0	D	T5
9	1:			
	0:			
	B:		D	F

Prévoir un nombre suffisant de "B" à
Gauche de la donnée **N**.

- (1) : Trouver la donnée.
(2) : Aller à la fin de la donnée.
(3) à (8) : Calcule $(5 * N) + 1$.
(9) : Décale d'un pion et FIN.

49 trous

Programme utilisateur n°36.

Programme de calcul de la valeur $5 * N + 2$ en BINAIRE pur.
Peut calculer des valeurs vraiment très grandes. Maximum 2 élevé à la puissance 52 soit la bagatelle de 9.007.199.254.740.991 !

ATTENTION : Le nombre de cycles d'HORLOGE augmente très rapidement avec la valeur initiale proposée pour **N**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T5
3	1:			T5
	0:			T5
	B:			T9
4	1:		D	T5
	0:		D	
	B:			T9
5	1:		D	T7
	0:		D	T6
	B:	0	D	T6
6	1:	0	D	T7
	0:	1	D	T4
	B:	1	D	T9
7	1:	0	D	T8
	0:	1	D	T6
	B:	1	D	T6
8	1:		D	
	0:		D	T3
	B:	0	D	T5
9	1:			
	0:			
	B:		D	F

Prévoir un nombre suffisant de "B" à
Gauche de la donnée **N**.

- (1) : Trouver la donnée.
(2) : Aller à la fin de la donnée.
(3) à (8) : Calcule $(5 * N) + 2$.
(9) : Décale d'un pion et FIN.

49 trous

Programme utilisateur n°39.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		D	T2
	0:		D	
	B:		D	T5
2	1:		D	T1
	0:		D	T3
	B:		D	T4
3	1:		D	
	0:		D	T2
	B:		D	T4
4	1:			
	0:			
	B:	0		F
5	1:			
	0:			
	B:	1	G	T6
6	1:			
	0:			
	B:		G	T7
7	1:		G	
	0:		G	
	B:		D	T8
8	1:		D	T9
	0:		D	
	B:		G	T11
9	1:	0	D	T8
	0:	1	D	T10
	B:			
10	1:		D	
	0:		D	T9
	B:			
11	1:			F
	0:	B	G	
	B:			

Effectue une division par 3 en BINAIRE avec test de divisibilité. Indicateur = "0" : NON divisible et donnée inchangée. Indicateur = "1" : Divisible et le résultat est calculé. Efface les "0" en tête inutiles.

- (1) à (3) : Vérifie la divisibilité.
 (4) : NON divisible, écrit "0".
 (5) : Divisible, écrit "1".
 (7) : Va à la fin de la donnée.
 (8) à (10) : Réalise la division par 3.
 (11) : Efface les zéros inutiles.

44 trous

Programme utilisateur n°42.

Déplace une donnée BINAIRE à gauche dans la mémoire vive de la machine. Le décalage se fait à deux BITS à gauche du poids fort de la donnée initiale qui est effacée de la mémoire.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T3
3	1:	B	D	T8
	0:	B	D	T4
	B:			
4	1:		D	
	0:		D	
	B:		D	T5
5	1:		D	
	0:		D	
	B:	0	G	T6
6	1:		G	
	0:		G	
	B:		G	T7
7	1:			T2
	0:			T2
	B:			T10
8	1:		D	
	0:		D	
	B:		D	T9
9	1:		D	
	0:		D	
	B:	1	G	T6
10	1:			
	0:			
	B:			F

- (1) : Cherche la chaîne BINAIRE.
 (2) : Va sur le premier BIT.
 (3) : Aiguille pour "0" ou "1".
 (4) : Va à la fin.
 (5) : Recopie le "0".
 (6) et (7) : Va à la fin.
 (8) : Va à la fin.
 (9) : Recopie le "1".
 (10) : Fin du programme.

39 trous

Programme utilisateur n°41.

Réalise un Compteur / Décompteur **BINAIRE** prépositionnable.
Peut être initialisé à **N** > 0 même en OPTION de Comptage. **Il faut au moins un BIT de donnée à "0" ou à "1"**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:	B	G	T7
	0:	B	G	T3
	B:			
3	1:			T4
	0:			T4
	B:		G	
4	1:		G	
	0:		G	
	B:		D	T5
5	1:	0		T4
	0:	1	D	
	B:		G	T6
6	1:	B	G	
	0:			
	B:		D	T10
7	1:			T8
	0:			T8
	B:		G	
8	1:		G	
	0:		G	
	B:		D	T9
9	1:	0	D	
	0:	1		T8
	B:	1		T8
10	1:			
	0:			
	B:			F

(1) : Va sur l'OPTION.

(2) : Efface l'OPTION et dirige le programme.

(3) : Va sur la donnée BINAIRE.

(4) : Va sur le poids faible.

(5) : Décompte.

(6) : Efface tous les "0" et Fin.

(7) : Va sur la donnée BINAIRE.

(8) : Va sur le poids faible.

(9) : Compte.

Option = "0" : Décomptage.

Option = "1" : Comptage.

En Option Comptage le programme est infini.

40 trous

Programme utilisateur n°40.

Construit la suite des entiers naturels codés en **UNAIRE**. La machine recopie à droite l'élément **n** et ajoute un pion à "1".

Attention, ce programme est sans fin.

Il ne faut que des "B" sur le plateau au lancement du programme sauf sous la tête de **L / E** où l'on doit positionner un "1".

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	G	T2
	0:			
	B:			
2	1:		G	
	0:			
	B:		G	T3
3	1:		G	
	0:			
	B:	1	D	T4
4	1:		D	
	0:			
	B:		D	T5
5	1:		D	T9
	0:	1	G	T6
	B:			
6	1:			
	0:			
	B:		G	T7
7	1:		G	
	0:			
	B:	1	D	T8
8	1:		D	
	0:			
	B:		G	T1
9	1:		D	
	0:	1	G	T1
	B:			

(1) : Indique le BIT translaté.

(2) : Va à la prochaine séparation.

(3) : Remonte à **n+1** et recopie le BIT traité.

(4) : Revient sur l'élément **n**.

(5) : Termine sur l'élément **n** si "0".

(6) : Saut le "B".

(7) : Ajoute "1" à **n+1**.

(8) : Revient sur **n**.

(9) : Traite sur **n** le BIT suivant.

31 trous

Programme utilisateur n°43.

Construit la suite des entiers naturels codés en **UNAIRE**. La machine recopie à gauche l'élément **n** et ajoute un pion à "1".

Attention, ce programme est sans fin.

Il ne faut que des "B" sur le plateau au lancement du programme sauf sous la tête de **L / E** où l'on doit positionner un "1".

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	D	T2
	0:			
	B:			
2	1:		D	
	0:			
	B:		D	T3
3	1:		D	
	0:			
	B:	1	G	T4
4	1:		G	
	0:			
	B:		G	T5
5	1:		G	T9
	0:	1	D	T6
	B:			
6	1:			
	0:			
	B:		D	T7
7	1:		D	
	0:			
	B:	1	G	T8
8	1:		G	
	0:			
	B:		D	T1
9	1:		G	
	0:	1	D	T1
	B:			

- (1) : Indique le BIT translaté.
 (2) : Va à la prochaine séparation + 1.
 (3) : recopie le BIT et va sur "B".
 (4) : Revient sur l'élément **n**.
 (5) : Termine sur l'élément **n** si "0".
- (6) : Saute le "B".
 (7) : Ajoute "1" à **n+1**.
 (8) : Revient sur **n**.
 (9) : Traite sur **n** le BIT suivant.

31 trous

Programme utilisateur n°46.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B		T3
	0:	B		T5
	B:	0		T7
2	1:	0		T9
	0:	1		T10
	B:	1		T11
3	1:	0		T11
	0:	0		T11
	B:	B		T10
4	1:	B		T9
	0:	0		T8
	B:	0		T7
5	1:	1		T5
	0:	1		T3
	B:	0		T2
6	1:	0		T1
	0:	B		T1
	B:	B		T1
7	1:	0		T1
	0:	0		T2
	B:	1		T3
8	1:	1		T4
	0:	0		T5
	B:	0		T7
9	1:	B		T9
	0:	B		T10
	B:	0		T11
10	1:	0		F
	0:	1		F
	B:	1		F
11	1:	0		T11
	0:	0		T10
	B:	B		T8

En rouge les seules instructions percées après la plastification de la grille de programme.

Algorithme appartenant à la catégorie des fonctions **NON CALCULABLE**, au contraire d'un Castor Affairé ce programme comporte un **maximum de TRANSITIONS pour aboutir à un minimum de cycles d'HORLOGE** pour l'exécuter. En outre, il totalise dix instructions d'ECRITURE surabondantes et 29 lignes de TRANSITIONS non utilisées.

66 trous

Programme utilisateur n°45.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	G	T2
	0:			
	B:			
2	1:		G	
	0:		D	T3
	B:		D	T3
3	1:	0	D	T4
	0:			
	B:			
4	1:		D	
	0:		G	T5
	B:			
5	1:		G	T6
	0:	1	G	
	B:		D	F
6	1:		D	T1
	0:		D	T7
	B:			
7	1:		D	
	0:		D	
	B:		D	T8
8	1:	B	G	T10
	0:	B	G	T9
	B:		G	T9
9	1:		G	T5
	0:	B	G	
	B:		G	
10	1:		G	T11
	0:	1	G	
	B:		G	
11	1:			F
	0:	B	G	
	B:	D		

(6) : Recommence si NON Fin.

(7) : Va sur l'**OPTION**.

(8) : **OPTION**.

(3) : Force un "0" à droite.

(4) : Va au début.

(5) : Teste pour la FIN.

50 trous

Astuce : Les instructions en roses sont ajoutées sur des lignes "non utilisées" pour restaurer à "1" les BITS mis à "0".

Programme utilisateur n°44.

Divise une chaîne unaire de longueur quelconque en deux "moitiés" **UNAIRES**. Si le nombre de BITS de la donnée est IMPAIR la "moitié la plus grande" sera décalée à gauche et la plus

petite à droite.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:	B	D	T3
	0:			
	B:			
3	1:			
	0:			
	B:	1	G	T4
4	1:			
	0:			
	B:		G	T5
5	1:		G	T6
	0:			
	B:			T10
6	1:		G	
	0:			
	B:	1	D	T7
7	1:	B	D	T8
	0:			
	B:			
8	1:		D	T9
	0:			
	B:			T10
9	1:		D	
	0:			
	B:		G	T2
10	1:			
	0:			
	B:			F

(1) : Cherche la chaîne **UNAIRES**.

(3) : Écrit le BIT de gauche.

(4) : Va sur la donnée.

(5) : Va à la fin ou **FIN IMPAIRE**.

(6) : Va au début et écrit le "1" de droite.

(7) : Efface le BIT.

(8) : Teste pour une **FIN PAIRE**.

(9) : Va à la fin de la donnée.

(10) : Fin du programme.

27 trous

Programme utilisateur n°47.

Addition de deux **BINAIRES**. Calcule la valeur de **X** en faisant la somme **X + Y**. En fin de programme **Y** est forcée à zéro, les données étant toutes codées en BINAIRE pur.

L'une des deux valeurs initiale peut être nulle.

- La tête de L/E doit impérativement se trouver sous la séparation de **X** et **Y**.
- Un seul "**B**" pour séparer **X** et **Y**.
- Prévoir assez de "**B**" à gauche de **X**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:		G	T2
2	1:		G	T3
	0:		G	
	B:			T8
3	1:		G	
	0:		G	
	B:		D	T4
4	1:	0	D	T5
	0:	1	D	
	B:			
5	1:		D	
	0:		D	
	B:		D	T6
6	1:	0	D	
	0:	1	G	T7
	B:	1	G	T7
7	1:		G	
	0:		G	
	B:		G	T2
8	1:			
	0:			
	B:			F

- (1) : Va sur **Y**.
 (2) : Teste pour la FIN.
 (3) : Va au début de **Y**.
 (4) : Enlève 1 à **Y**.
 (5) : Va au début de **X**.
 (6) : Ajoute 1 à **X**.
 (7) : Revient au test de FIN.

38 trous

Programme utilisateur n°50.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:		G	
	B:		D	
2	1:		D	
	0:		G	T3
	B:		G	T5
3	1:		G	
	0:		D	T4
	B:		G	T5
4	1:		G	
	0:		D	T7
	B:		G	T4
5	1:		G	
	0:		D	T6
	B:		D	T7
6	1:		D	
	0:			
	B:	0		T8
7	1:	1		T8
	0:			
	B:		D	
8	1:		D	
	0:		G	T9
	B:		G	T10
9	1:			T2
	0:			
	B:			T2
10	1:		G	
	0:			T11
	B:			
11	1:			
	0:			
	B:			F

- (1) : Cherche la chaîne BINAIRE.
 (2) : Va à gauche de la donnée.
 (3) : Teste la parité des "1".
 (4) : PAIR.
 (5) : IMPAIR.
 (6) : Va à gauche du "1" le plus à droite.
 (7) : Inverse le BIT.
 (8) : Va à gauche de la donnée.
 (9) et (10) : Teste pour la FIN.

39 trous

Réalise un **Compteur**
BINAIRE codé en **GRAY**.

Programme utilisateur n°49.

Réprésentatif des premières tabulatrices électromécaniques utilisées aux USA pour les recensements ou le dépouillement des élections, ce programme simule un **comptage BINAIRE** pour recenser une population **UNAIRE**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	D	T2
	0:			
	B:		G	
2	1:			
	0:			
	B:		D	T3
3	1:			
	0:			
	B:	1	G	T4
4	1:			
	0:			
	B:		G	T5
5	1:	0	D	T6
	0:		G	
	B:		D	T9
6	1:			
	0:		D	
	B:		D	T7
7	1:	0	D	
	0:	1	G	T8
	B:	1	G	T8
8	1:		G	
	0:		G	
	B:		G	T5
9	1:			
	0:	1	D	
	B:			T10
10	1:			
	0:			
	B:			F

- (1) : Va sur la "population".
 (2) et (3) : Compte le premier individu.
 (4) : Revient sur la "population".
 (5) : Cherche le prochain individu.
 (6) : Revient sur le résultat.

36 trous

- (7) : Ajoute 1 en BINAIRE.
 (8) : Revient sur la population.
 (9) : Restitue "les UNAIREs."

Prévoir un nombre de "B" suffisants à gauche pour le compteur BINAIRE.

Programme utilisateur n°48.

Soustraction de deux **BINAIREs**. Calcule la valeur de **X** en faisant la différence **X - Y**. En fin de programme **Y** est forcée à zéro, les données étant toutes codées en BINAIRE pur.

La valeurs initiale de **Y** peut être nulle.

- La tête de L/E doit impérativement se trouver sous la séparation de **X** et **Y**.
- Un seul "**B**" pour séparer **X** et **Y**.
- **X** doit être supérieur ou égal à **Y**.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			
	0:			
	B:		G	T2
2	1:		G	T3
	0:		G	
	B:			T8
3	1:		G	
	0:		G	
	B:		D	T4
4	1:	0	D	T5
	0:	1	D	
	B:			
5	1:		D	
	0:		D	
	B:		D	T6
6	1:	0	G	T7
	0:	1	D	
	B:			
7	1:		G	
	0:		G	
	B:		G	T2
8	1:			
	0:			
	B:			F

- (1) : Va sur **Y**.
 (2) : Teste pour la FIN.
 (3) : Va au début de **Y**.
 (4) : Enlève 1 à **Y**.
 (5) : Va au début de **X**.
 (6) : Enlève 1 à **X**.
 (7) : Revient au test de FIN.

29 trous

Programme utilisateur n°51.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	T3
	0:		G	
	B:			F
3	1:		G	
	0:		G	
	B:		D	T4
4	1:		D	T6
	0:		D	
	B:		G	T5
5	1:		D	T6
	0:		D	
	B:		G	T10
6	1:		D	T5
	0:		D	
	B:		G	T7
7	1:		G	
	0:		G	
	B:		D	T8
8	1:	0	D	T9
	0:	1	D	T9
	B:			
9	1:		D	
	0:		D	
	B:		G	T2
10	1:		G	
	0:		G	
	B:		D	T11
11	1:		D	T8
	0:		D	
	B:			

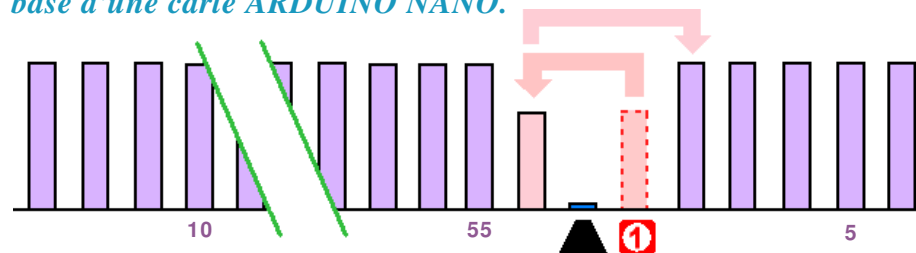
Décompteur BINAIRE
codé en GRAY.

- (1) : Cherche la chaîne BINAIRE. (5) : PAIR.
 (2) : Teste pour la FIN. (6) : IMPAIR.
 Va à gauche de la donnée. (7) et (8) : Teste le BIT de droite.
 (3) : Va à droite de la donnée. (9) : Va à gauche de la donnée.
 (4) : Teste la parité des "1". (10) et (11) : Inverse le BIT à gauche du "1" le plus à droite.

46 trous

Programme utilisateur n°54.

Petit programme élémentaire assez voisin d'un Castor affairé. Le défi consiste à créer un programme avec le minimum d'instructions qui engendre en fonction de la donnée initiale proposée un maximum de cycles d'HORLOGE sur la machine. Transforme une suite composée d'un nombre quelconque de "1" en une suite de taille identique remplacée par des "0". La taille la plus grande sur la machine sera de 55 BITS placés à "1". *Algorithme utilisé comme exemple didactique dans le tutoriel sur la Machine de Turing électronique à base d'une carte ARDUINO NANO.*



1654 cycles d'HORLOGE. Soit une Heure et 20 minutes.

1-55 / 55-2 / 2-54 / 54-3 / 3-53 / 53-4 / 4-52 / 52-6 / 6-51 ...

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:	0	G	T3
	0:			F
	B:			
3	1:		G	
	0:		D	T4
	B:		D	T4
4	1:	0	D	T5
	0:			F
	B:			
5	1:		D	
	0:		G	T2
	B:			

- (1) : Cherche le début de la suite de "1".
 (2) : Force un "0" ou Fin du programme si "0".
 (3) : Va à gauche de la suite si "B" ou "0".
 (4) : Écrit un nouveau "0" ou fin du PGM si "0".
 (5) : Revient vers la gauche de la suite.

18 trous

Programme utilisateur n°53.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	D	T2
	0:		D	
	B:		D	
2	1:		G	T3
	0:			
	B:		G	T5
3	1:	0	G	T4
	0:		G	
	B:		G	
4	1:		D	T1
	0:			
	B:		D	T8
5	1:			
	0:	1	G	
	B:		G	T6
6	1:	B	G	T7
	0:	B	G	
	B:		G	
7	1:		D	T1
	0:			
	B:			T11
8	1:			
	0:	1	D	
	B:		D	T9
9	1:			
	0:			T10
	B:		D	
10	1:			T1
	0:	B	D	
	B:			
11	1:			F
	0:			
	B:		D	

- (1) : Lit "1", écrit "0" et Droite.
 (2) : Vérifie un "1" à Gauche de X.
 (3) : Gauche jusqu'à un "1" sur Y.
 (4) : Lit "1" et revient sur X, ou D et Tr8.
 (5) : Force tous les BITS de X à "1".
 (6) : Efface X ou Y pions sur Y ou X.

42 trous

- (7) : S'il reste un "1" on recommence. Sinon : Tr11.
 (8) : Remet tous les BITS de Y à "1".
 (9) : Revient sur X.
 (10) : Enlève Y à X.
 (11) : Retour à Droite de X et FIN.

PGCD de deux
UNAIRES X et Y.

Programme utilisateur n°52.

Construit la suite des puissances de deux sous forme **UNAIRE**. L'ensemble du carrousel est au préalable forcé à "B" sauf le BIT situé sous la tête de L/E qui est initialisé à "1".

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	0	D	T2
	0:		D	
	B:		G	T6
2	1:		D	
	0:			
	B:		D	T3
3	1:		D	
	0:			
	B:	1	G	T4
4	1:		G	
	0:			
	B:		G	T5
5	1:		G	
	0:		D	T1
	B:		D	T1
6	1:		G	
	0:	1	D	T7
	B:		D	T10
7	1:		D	
	0:			
	B:		D	T8
8	1:		D	
	0:			
	B:	1	G	T9
9	1:		G	
	0:			
	B:		G	T6
10	1:		D	
	0:			
	B:		D	T1

- (1) : Met à "0" le BIT de l'élément n.
 (2) : Va sur l'élément n+1.
 (3) : Ajoute un "1" à gauche de l'élément n+1.
 (4) : Revient sur l'élément n.
 (5) : Va sur le "1" le plus à gauche de n.
 Si que des "0" passe en (1) puis en (6).

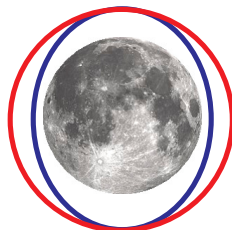
- (6) : Remet à "1" le BIT "le plus à droite" de n.
 (7) : Revient sur l'élément n+1.
 (8) : Force un "1" à gauche de n+1.
 (9) : Revient sur l'élément n.
 (10) : Si tous à "1" va sur n+1 et recommence.

40 trous

Programme utilisateur n°55.

Simule le programme qui sur Apollo gérait la circularisation de l'orbite du train spatial. Pour des raisons de stabilité du séjour du module resté en orbite basse, c'est le petit axe de l'ellipse qui était

augmenté et égalisé.



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		G	T3
3	1:	0	D	
	0:		D	
	B:		D	T4
4	1:	0	G	T5
	0:		D	
	B:			
5	1:			
	0:		G	
	B:		G	T6
6	1:	0	D	T7
	0:		G	
	B:		D	T9
7	1:			
	0:		D	
	B:		D	T8
8	1:	0	G	T5
	0:		D	
	B:	0	G	T5
9	1:			
	0:	1	D	
	B:			F
10	1:			
	0:			
	B:			

(7) : Revenir sur le Petit AXE.

(9) : Replace les BITS à "1".

(1) : Cherche la chaîne UNAIRE.

(2) : Aller sur le Grand AXE.

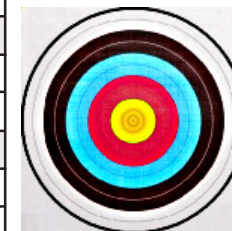
(4) : Va à la fin.

(5) : Revenir sur le Grand AXE.

36 trous

Programme utilisateur n°58.

Simule le tir d'une flèche en tir à l'arc au centre d'une cible. La cible est symbolisée par une suite de "0" en nombre IMPAIR. La flèche plantée au centre prendra l'état "1" et dépassera en hauteur.



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:			
	B:		G	
2	1:			
	0:	B	G	T3
	B:			
3	1:			T4
	0:			
	B:	0		T2
4	1:	0	G	T5
	0:			
	B:			
5	1:		G	
	0:		D	T6
	B:		D	T6
6	1:	0	D	T7
	0:			
	B:			
7	1:		D	T8
	0:			
	B:			
8	1:		D	T9
	0:			T10
	B:			
9	1:		D	
	0:		G	T4
	B:			
10	1:		D	
	0:		D	
	B:			F

(1) : Sauter l'archer et aller sur la flèche

(2) et (3) : Vol jusqu'à la cible.

(4) à (6) : Forcer un "0" aux extrémités.

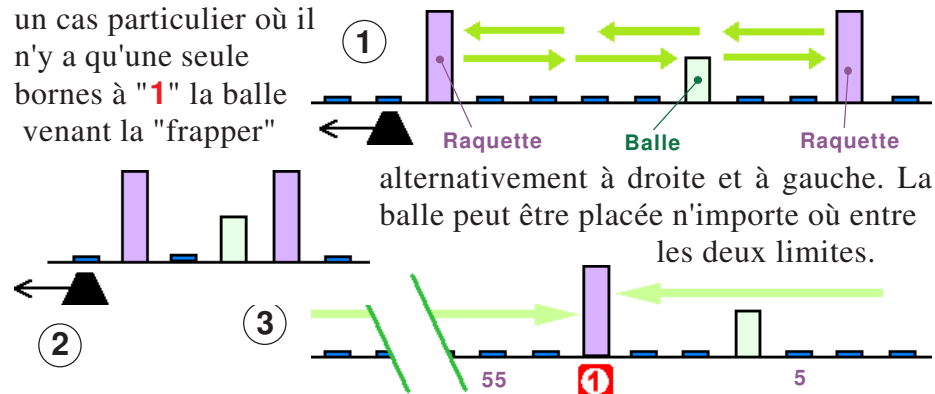
(7) à (9) : Test pour la Fin.

(10) : Replace la tête L/E à gauche de la cible.

39 trous

Programme utilisateur n°57.

Simule symboliquement une partie de Ping-pong entre deux bornes à "1" qui représentent les joueurs. La distance minimale entre deux bornes est de deux pions. (Voir ②) La balle est simulée par un "0". L'espace maximal peut faire 55 BITS, comme en ③ et c'est un cas particulier où il n'y a qu'une seule borne à "1" la balle venant la "frapper"



État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:			
	B:		G	
2	1:			
	0:			T3
	B:		G	
3	1:			
	0:	B	G	T4
	B:			
4	1:		D	T5
	0:			
	B:	0		T3
5	1:		G	T4
	0:	B	D	
	B:	0		T6
6	1:		G	T4
	0:	B	D	
	B:	0		T5

(1) et (2) : Cherche la raquette de gauche.
 (3) et (4) : Déplace la balle vers la Droite.
 (5) et (6) : Déplace la balle vers la Gauche.

24 trous

Programme utilisateur n°56.

Simule une transaction commerciale et calcule la monnaie à rendre sur un trop perçu. La somme versée par l'acheteur doit être égale ou supérieure au prix de vente. Données codées en UNAIRE.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	T3
	0:			T10
	B:			
3	1:		G	
	0:		G	
	B:		D	T4
4	1:	0		T5
	0:		D	
	B:			
5	1:			
	0:		G	
	B:		G	T6
6	1:		G	
	0:			
	B:		D	T7
7	1:	B	D	T8
	0:			
	B:			
8	1:		D	
	0:			
	B:		D	T2
9	1:		D	
	0:		D	
	B:		G	T10
10	1:	0	G	T5
	0:		D	
	B:			F

(1) : Cherche la donnée du PRIX.
 (2) : Teste pour la fin.
 (3) : Cherche le premier BIT à effacer.
 (4) : Met à "0" ce premier BIT.
 (5) : Revient sur la somme versée.
 (6) : Va à la fin de la somme versée.
 (7) : Efface le BIT pour soustraction..
 (8) et (9) : Revient sur le Prix de vente.
 (10) : Force un "0" de plus ou FIN.

33 trous

Programme utilisateur n°59.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			T2
	B:		G	
2	1:		G	T5
	0:		G	T3
	B:		D	T7
3	1:	0	D	T4
	0:			T2
	B:		D	T7
4	1:			
	0:	1	G	T2
	B:			
5	1:			T2
	0:	1	D	T6
	B:		D	T7
6	1:	0	G	T2
	0:			
	B:			
7	1:	B	G	T9
	0:	B	G	T8
	B:			F
8	1:			
	0:			
	B:	0	D	T10
9	1:			
	0:			
	B:	1	D	T10
10	1:			
	0:			
	B:		D	T11
11	1:		D	
	0:		D	
	B:		G	T2

Inverser latéralement une donnée binaire. (La retourner horizontalement.) Par rapport à sa position initiale, la donnée est décalée d'un BIT à droite. La tête de L/E se trouvera à deux "B" à gauche du résultat final.

(1) : Aller sur la donnée.
Chaque caractère est déplacé de la gauche vers la droite par une suite de transpositions.

46 trous

Programme utilisateur n°62.

Construire en UNAIRE la suite $U^{n+1} = U^n + 1$. Globalement, la technique consiste à recopier la valeur de U^n à gauche de cette dernière et à y concaténer deux BITS "1" à sa gauche.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		D	T3
3	1:	B	D	T4
	0:			
	B:		D	T8
4	1:		D	
	0:			
	B:		D	T5
5	1:		D	
	0:			
	B:	1	G	T6
6	1:		G	
	0:			
	B:		G	T7
7	1:		G	
	0:			
	B:	1	D	T3
8	1:		D	
	0:			
	B:	1	D	T9
9	1:			
	0:			
	B:	1	G	T10
10	1:		G	
	0:			
	B:		D	T3

(1) et (2) : Se place à droite de la donnée UNAIRE..
(2) : Va sur le premier BIT.
(3) : Efface le "1".
(5) : Ecrit "1".
(7) : Restore le "1" effacé.
(8) à (10) : Ajoute deux "1" à la valeur recopiée.

34 trous

Programme utilisateur n°61.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B	D	T6
	0:	B	D	T2
	B:		D	T10
2	1:		D	
	0:		D	
	B:		D	T3
3	1:		D	
	0:		D	
	B:	0	G	T4
4	1:		G	
	0:		G	
	B:		G	T5
5	1:		G	
	0:		G	
	B:	0	D	T1
6	1:		D	
	0:		D	
	B:		D	T7
7	1:		D	
	0:		D	
	B:	1	G	T8
8	1:		G	
	0:		G	
	B:		G	T9
9	1:		G	
	0:		G	
	B:	1	D	T1
10	1:	0	D	
	0:	1	G	T11
	B:	1	G	T11
11	1:		G	
	0:		G	
	B:		D	T1

Construire la suite des Entiers codés en BINAIRE pur. Noter que ce programme n'a pas de Fin. La configuration de départ débute avec la première valeur initialisée manuellement à "1". La tête de L/E doit impérativement être sous ce BIT.

56 trous

Page 34

Programme utilisateur n°60.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B	D	T3
	0:	B	D	T2
	B:		D	T11
2	1:			
	0:			
	B:	0	G	T4
3	1:			
	0:			
	B:	1	G	T4
4	1:			
	0:			
	B:		G	T5
5	1:		G	
	0:		G	
	B:		D	T6
6	1:	B	G	T8
	0:	B	G	T7
	B:			T11
7	1:			
	0:			
	B:	0	D	T9
8	1:			
	0:			
	B:	1	D	T19
9	1:			
	0:			
	B:		D	T10
10	1:		D	
	0:		D	
	B:		G	T1
11	1:			
	0:			
	B:		D	F

Algorithme qui sépare une donnée BINAIRE quelconque en deux moitiés de tailles analogues séparées par deux "B". Si le nombre de BITS et PAIR les deux entités seront de même taille. Si la chaîne est de taille IMPAIRE la donnée la plus "longue" sera celle de gauche. La tête de L/E sera à droite du morceau situé à gauche après le traitement.

41 trous

Programme utilisateur n°63.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:			T2
	0:			
	B:		G	
2	1:		G	
	0:			
	B:		G	T3
3	1:	0	G	T4
	0:		G	
	B:		G	
4	1:		D	T5
	0:			
	B:		G	T7
5	1:	0	D	T6
	0:		D	
	B:		D	
6	1:		G	T3
	0:			
	B:		G	T11
7	1:		G	
	0:			
	B:	1	D	T8
8	1:		D	
	0:			
	B:		D	T9
9	1:			
	0:	1	D	
	B:		D	T10
10	1:	B	D	T6
	0:	B	D	
	B:		D	
11	1:			
	0:	1	G	
	B:			F

Division euclidienne de l'entier A par l'entier B.
Les deux entiers A et B sont codés en UNAIRE.

- (1) et (2) : Aller sur la donnée.
 (3) à (6) : Force à "0" un "1" de B puis force à "0" un "1" de A.
 (7) : Ajoute un "1" à B.
 (9) : Remet à "1" les "0" de B.
 10) : Enlève "B" BITS à A.
 (11) : Force à "1" les "0" du Reste.

43 trous

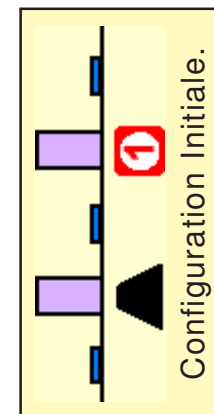
Programme utilisateur n°66.

Établir une Bijection entre les entiers naturels N et les points du plan de coordonnées X, Y entières et positives.

Cet algorithme ne fonctionne que jusqu'à la valeur 10 obtenue

après 57 cycles d'HORLOGE.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	1	D	
	0:	1	G	T2
	B:	1	G	T2
2	1:		G	
	0:		G	
	B:		G	T3
3	1:	0	D	T4
	0:	1	G	
	B:		D	T5
4	1:		D	
	0:		D	
	B:		D	T1
5	1:	0	D	
	0:			
	B:		G	T5
6	1:	0	G	
	0:	1	D	T7
	B:	1	D	T7
7	1:		D	
	0:		D	
	B:		D	T8
8	1:	0	G	T9
	0:	1	D	
	B:		G	T10
9	1:		G	
	0:		G	
	B:		G	T6
10	1:	0	G	
	0:			
	B:		D	T1



- (1) : Réalise X + 1.
 (2) : Aller sur Y.
 (3) : Réalise Y - 1.
 (4) : Aller sur X.
 (6) : Réalise Y + 1.
 (7) : Aller sur X.
 (8) : Réalise X - 1.
 (9) : Aller sur Y.

54 trous

Programme utilisateur n°65.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:	B	D	T7
	0:	B	D	T2
	B:			
2	1:			
	0:			
	B:		D	T3
3	1:			T5
	0:			T5
	B:			
4	1:		D	T15
	0:		D	
	B:		D	F
5	1:	0	D	T6
	0:	1	D	T4
	B:	1	D	F
6	1:		D	
	0:		D	T5
	B:	0	D	T5
7	1:			
	0:			
	B:		D	T8
8	1:			T11
	0:			T11
	B:			
9	1:		D	T10
	0:		D	
	B:		D	F
10	1:	0	D	T11
	0:	1	D	T9
	B:	1	D	F
11	1:		D	
	0:		D	T10
	B:	0	D	T10

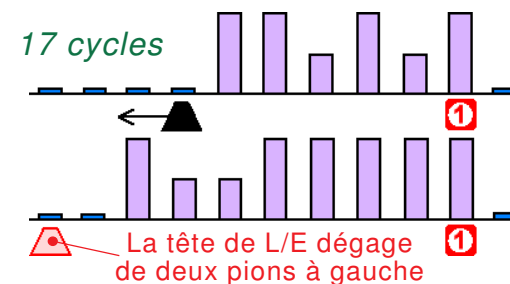
Conction d'un BIT d'option calcule les valeurs de $3 * N$ + 1 ou de $3 * N$ + 2 codées en BINAIRE pur. Peut calculer sur des valeurs BINAIRES vraiment très élevées et traite très rapidement. (52 BITS.)

- (1) : Va sur $(3*N)+1$ ou sur $(3*N)+2$.
 (2) : Va à droite de la donnée..
 (3) à (6) : Calcule $(3 * N) + 1$.
 (7) : Va à droite de la donnée.
 (8) à (11) : Calcule $(3 * N) + 2$.

54 trous

Programme utilisateur n°64.

L' algorithme calcule $3 * N$ en BINAIRE pur. Il peut traiter des valeurs vraiment très grandes. Maximum 2 élevé à la puissance cinquante deux soit la bagatelle de 9.007.199.254.740.991 !



Comme le calcul ne consomme que quatre transitions et que les rotations du plateau sont rapides, on dégage de deux pions la tête de L/E à gauche du résultat.

État	Lectr	Ecrt	Mvt	ÉTAT
1	1:		G	T2
	0:		G	T2
	B:		G	
2	1:		G	
	0:		G	
	B:		D	T3
3	1:		D	T4
	0:		D	
	B:		D	T6
4	1:	0	D	T5
	0:	1	D	T3
	B:	1	D	T6
5	1:		D	
	0:		D	T4
	B:	0	D	T4
6	1:			
	0:			
	B:		D	F

- (1) : Va sur la donnée BINAIRE.
 (2) : Déplace la tête de L/E à droite de la donnée.
 (3) à (5) : Calcule $3 * N$.
 (6) : Dégage d'un deuxième pion à gauche.

31 trous