



Auteur : Christian MATHIEU

OBJECTIFS :

Le programme TORTUE est un programme permettant l'initiation par le graphique aux techniques de programmation.

Plutôt que l'invention d'un nouveau symbolisme, l'auteur a choisi d'utiliser les termes du langage LOGO, réputé par ses qualités pédagogiques et son aptitude à l'acquisition de notions informatiques structurées trop souvent absentes des langages d'initiation, avec le risque de déformation des élèves qui cette absence implique.

Les ordres de base retenus sont ceux en usage dans l'éducation nationale française.

Bien entendu, ce programme ne comporte qu'un sous-ensemble des possibilités de LOGO, mais il possède néanmoins les éléments de base d'un bon langage : variables globales et locales, procédures avec paramètres, récursivité. Les ordres ont été limités à la partie tortue de LOGO, le traitement de listes (textes) n'étant pas implémenté.

GENERALITES :

Une tortue portant un crayon sous sa carapace se déplace selon les ordres qu'on lui donne.

Quand elle avance (AV), elle avance droit devant elle, c'est à dire dans la direction où se trouve sa tête.

Quand elle tourne, elle tourne sur place, vers la droite (D) ou vers la gauche (G) par une rotation autour de son crayon qui ne trace donc rien.

On peut lever la plume (L). Dans ce cas, le plume ne touche plus le papier, on avance sans rien tracer, et ce jusqu'au prochain ordre pour baisser le crayon (BC).

On peut écrire une seule commande suivie de (ENTER) pour voir immédiatement son action, mais le plus souvent on écrit une liste de commandes (qui correspond à la liste d'instructions d'un programme), suivie de (ENTER). Dans ce dernier cas, la tortue exécutera l'ensemble des commandes données. Tant que (ENTER) n'a pas été actionnée, on peut corriger la liste par appui sur les touches (CTRL) et (--) ou (CTRL) et (<--), comme pour la plupart des programmes QL. L'ordre (ENTER) est pour l'ordinateur le signal indiquant qu'il peut exécuter.

La liste des commandes que l'on peut afficher, dans le cadre de commandes, au bas de l'écran, est strictement limitée à 3 lignes de 72 caractères. Cette limite peut paraître contraignante. Toutefois, outre une simplification des modes d'édition qui rebute certains débutants en LOGO, ce choix présente un énorme avantage pédagogique : IL EST IMPOSSIBLE D'ECRIRE DE LONGUES LISTES D'INSTRUCTIONS DANS LESQUELLES ON SE PERD. Dès que l'on dépasse la capacité d'écriture, on est obligé de découper le problème en sous-programmes auxquels on donne un nom. L'HABITUDE DE STRUCTURER EST IMPOSEE PAR CETTE CONTRAINTE qui reste cependant large dans un univers de débutants.

MISE EN OEUVRE (sans utilisation):

- 1- allumer la machine ou appuyer sur RESET (pas de micro-cassette en place)
- 2- aucune micro-cassette n'étant en place, appuyer sur (F2)
- 3- mettre la micro-cassette TORTUE à droite, une micro-cassette vierge à gauche
- 4- taper LRLIN MOV2 CLONE
- 5- ranger la micro-cassette originale en lieu sûr

UTILISATION HABITUELLE:

- 1- allumer la machine ou appuyer sur RESET (pas de micro-cassette)
- 2- mettre la copie de TORTUE à gauche
- 3- appuyer sur (F2), après une présentation, vous aurez l'écran d'utilisation de LOGO, la tortue au centre.
- 4- comme le demande le PANNEAU, en haut, appuyez sur (CTRL) et (C) en même temps.
- 5- tapez les commandes LOGO à votre convenance.
- 6- la commande TERMINE finit l'utilisation LOGO, (CTRL) (C) vous redonnera le BASIC.

DEMONSTRATION:

Après avoir exécuté les directives ci-dessus jusqu'à 4 inclus, tapez RAVENE, puis à la demande "Nom du fichier où lire les mots...:", répondre: MOV1, DEMO, LOG. Laisser le programme lire le fichier, puis taper DEMO.

CARACTERES SPECIAUX:

- Certaines combinaisons de touches doivent être connues:
- (CTRL) (←) : efface le caractère qui PRECEDE le curseur.
 - (CTRL) (→) : efface le caractère qui est SOUS le curseur.
(on appelle curseur le carré cliquant qui indique où on va écrire)
 - (CTRL) ([) : crochet ouvrant.
 - (CTRL) (]) : crochet fermant.
(il est pratique que le crochet s'obtienne avec la touche portant la parenthèse correspondante)
 - (CTRL) (C) : la première et la dernière des combinaisons à frapper: doit être utilisé en PREMIER pour permettre l'accès au cadre d'instructions LOGO, doit être frappé après TERMINE pour revenir au BASIC.

LA TOUCHE (CTRL) EST A APPUYER EN MEME TEMPS
QUE LA TOUCHE INDIQUEE AVEC ELLE.

(ESC) permet d'interrompre l'exécution d'un programme.

(→) : la flèche droite permet de déplacer le curseur d'écriture vers la droite.

(←) : la flèche gauche permet de déplacer le curseur d'écriture vers la gauche.

I COMMANDES ELEMENTAIRES:

AV (avance) : est suivie d'une indication de distance. Ex.: AV 50
RE (recule) : est suivie d'une indication de distance. Ex.: RE 150
TD (tourne droite) : est suivie d'une indication d'angle en degrés.
Ex.: TD 90
TG (tourne gauche) : est suivie d'une indication d'angle en degrés.
Ex.: TG 180

** Un espace est indispensable entre la commande et la valeur, en effet, TD90 serait pris comme un nom global composé de lettres et de chiffres et non pour la commande tourner à droite. Le programme dirait probablement: je ne sais pas quoi faire pour TD90, à moins que ce mot n'ait été défini par l'utilisateur.

** Le terme valeur désigne une expression, c'est à dire:

- un nombre ; exemple : 10
 - une variable ; exemple : LONGUEUR
 - une suite d'opérations portant sur des nombres ou des variables:
:LONGUEUR + 1 ; :LARGEUR
- (voir ci-après les remarques sur les variables).

II COMMANDES DE DESSIN:

VE (vide-écran) : efface l'écran et ramène la tortue au centre, face au nord.
NETTOIE : efface l'écran sans changer la position de la tortue.
ORIGINE : ramène la tortue au centre de l'écran, face au nord, en traçant si le crayon est baissé.
LC : lève la plume qui ne tracera plus si la tortue avance.
BC : baisse la plume pour retracer après un ordre LC.
FCC : Fixe Couleur Crayon: change la couleur du tracer, selon la valeur qui suit, :NOIR (=0), :ROUGE (=2), :VERT (=4), :BLANC (=6) et qui correspond aux conventions couleur de SINCLAIR en mode Haute résolution.
ATTENTION: Le choix volontaire ou accidentel de la couleur noire amène à une tortue noire sur fond noir que l'on a peu de chances de voir. De plus, le tracer sera de même invisible, sauf sur un dessin dense. Cette possibilité permet de "GOMMER" une erreur: EX.: après AV 50, les commandes FCC 0 RE 50 FCC 7 effacent le trait tracé.

III COMMANDES DE PROGRAMMATION:

REPETE : suivie d'un nombre n, répète n fois la séquence d'instructions entre crochets qui suit.
exemple: REPETE 4 [AV 50 TD 90]
tracera les 4 cotés d'un carré de dimension 50.

POUR : crée un nouveau mot de vocabulaire (sous-programme). Cette commande ne trace jamais rien, elle APPREND seulement à l'ordinateur ce qu'il doit faire quand on lui donne le mot à exécuter.
exemple: POUR CARRE REPETE 4 [AV 50 TD 90]
apprendra à l'ordinateur que faire pour CARRE. Ensuite, si on écrit CARRE, seul ou dans une liste d'instructions, on verra se tracer un CARRE de 50.

Si le mot défini ne fait pas ce que l'on attend, il suffira de refaire POUR CARRE ... pour que la nouvelle définition remplace l'ancienne.
Le nom du mot défini peut être suivi d'une liste de paramètres permettant par exemple d'obtenir des CARRÉS de dimensions différentes (voir ci-après les remarques sur les variables).

DONNE : donne à la "boîte" (=variable) dont le nom suit la valeur du nombre, de la variable, ou de l'expression qui suit.
PAR SOUCI D'HOMOGENEITE AVEC LES LANGAGES LOGO, ON ADMETTRA QUE:
 *VAR désigne le NOM de la variable VAR
 *VAR désigne le CONTENU de la variable VAR.

exemples: **DONNE "COTE 5** crée un symbole de nom COTE et lui associe la valeur 5

DONNE "SOMME :PRIX1 + :PRIX2 -100 crée un symbole SOMME et lui affecte le total de l'addition des contenus PRIX1 et PRIX2 puis retranche 100.

DONNE "TOTAL :PRIX1 + 100 \$:PRIX2 / 5 ajoute le contenu de PRIX1 à 100, multiplie le résultat par le contenu de PRIX2, divise le tout par 5 et range le résultat dans TOTAL.
LE "SOUS-ENSEMBLE TORTUE N'A PAS DE PARENTHESES, IL EFFECTUE LES OPERATIONS DANS L'ORDRE OU IL LES TROUVE, SANS PRIORITE D'OPERATEURS. Cette hypothèse est simple pour des enfants non formés aux techniques de l'algèbre.

REMARQUES SUR LES VARIABLES;

- Les opérations possibles sont: +(addition), -(soustraction), *(multiplier), /(diviser).
- En logo, les signes des opérations doivent être entourés d'espaces, et "+ 5" signifie ajouter 5, alors que "+5" est seulement le chiffre positif 5. Cette nécessité n'est pas exigée par l'interpréteur TORTUE. Il est cependant souhaitable de prendre des habitudes conformes aux LOGOS courants.
- Les calculs décimaux sont faits en interne avec une grande précision. Toutefois, compte-tenu de la population visée, les décimales sont limitées au millième pour les entrées et sorties. La manipulation de longs nombres non significatifs n'est pas souhaitable à ce stade.
- La convention de notation décimale est la convention américaine: 325.45 = 325 unités et 45 centièmes. Cette option a été retenue par souci d'homogénéité avec les matériels en usage dans les écoles.
- Pour le nombre 325.637,45 on écrira toutefois 325637.45 et non 325,637.45 comme l'exigerait le strict respect des normes américaines.
- Dans **DONNE "COMPTE :COMPTE + 1**, l'ordinateur calcule d'abord :COMPTE + 1, et indiquera une erreur s'il n'y a rien dans :COMPTE.
- On peut affecter une valeur à une variable par **DONNE**, mais aussi par affectation en tant que **PARAMETRE ASSOCIE A UN MOT**.
- exemple 1:** **POUR CARRE :COTE REPETE 4 (AV :COTE TD 90)** déclare qu'au mot CARRE sera associé une valeur :COTE, et quand on exécutera **AV :COTE**, on ira chercher ce qu'il y a dans :COTE. On n'utilisera donc plus CARRE, mais CARRE 50 pour tracer un CARRE de 50, CARRE 70 pour tracer un carré de 70 etc...
- On peut également utiliser comme valeur le contenu d'une variable: **DONNE "X 80 CARRE :X** tracera un carré de 80, puisque :X envoyé à CARRE contient 80.

exemple 2:
POUR POLY :LONGUEUR :NOMBRE DONNE "ANGLE 360 / :NOMBRE REPETE :NOMBRE (AV :LONGUEUR TD :ANGLE)
 permettra de tracer tout polygone régulier :
POLY 50 3 trace un triangle (3 côtés) de 50
POLY 20 8 trace un octogone (8 côtés) avec des côtés de 20 .
explications:
POUR POLY :LONGUEUR :NOMBRE déclare le mot POLY et indique qu'avec lui on recevra deux valeurs que l'on appellera LONGUEUR et NOMBRE dans POLY.
DONNE "ANGLE 360 / :NOMBRE crée une variable ANGLE, dans laquelle on calcule de quel angle il faut tourner pour réaliser ce polygone. Cet ANGLE dépend du nombre de côtés reçu avec la demande d'exécution de POLY.
REPETE :NOMBRE (AV :LONGUEUR TD :ANGLE) répète NOMBRE fois le tracer d'un côté du polygone et la rotation de tortue qui correspond. Les paramètres sont un moyen de fabriquer des outils efficaces et leur absence constitue l'un des défauts majeurs des BASIC de première génération. (Le BASIC du QL en a tenu compte).

STOP : provoque une sortie du mot en cours d'exécution. En particulier, il permet la fin d'un processus récursif. Il est dangereux de laisser la récursion se poursuivre indéfiniment, dans la mesure où le débordement de mémoire serait alors géré par le PASCAL, et se solderait par une sortie de TORTUE avec perte du travail en cours.

SI ... [] ou **SI ... [] [] :**
exemple: **SI :COMPTE < 1000 (AJOUTER) (FACTURER)**
 si l'expression est vraie, la première liste d'instructions est exécutée, sinon ce sera la seconde si elle existe.
NOTEZ LA FORME DE LA COMPARAISON, UTILISABLE EN LOGO CLASSIQUE. LA FORME PLP? :COMPTE 1000 CHANTE INCONTESTABLEMENT LA LOUANGE DE LA POLONAISE INVERSE, MAIS SA MAITRISE COMPLIQUE SINGULIEREMENT LA TACHE DES DEBUTANTS VISES PAR CE PROGRAMME.
 Les signes de comparaison sont: < > <= >=
POUR SPIRE DONNE "C :C + 5 AV :C TD 90 SI :C < 100 (SPIRE) (STOP)
 définit le mot SPIRE que l'on utilisera par **DONNE "C 5 SPIRE**
explications: **DONNE "C 5** met une valeur de départ dans C et évite que dans SPIRE on ne soit arrêté par **DONNE "C :C + 1** qui trouve une variable vide. (Une variable est vide si on n'a rien mis dedans, on ne suppose jamais qu'il y a zéro ou un autre valeur par défaut).
dans SPIRE: - on met dans "C :C + 5, ce qui revient à chaque fois à augmenter la valeur de C de 5, donc à tracer des segments qui augmentent de 5 à chaque appel.
 - on avance de :C et on tourne de 90, pour tracer des segments à angle droit de plus en plus grands.
 - Si le contenu de :C est plus petit que 100, on revient dans SPIRE tracer le segment suivant, sinon STOP sort de spire et rend la main à l'opérateur.

III FACILITES D'UTILISATION:

LISTER: permet d'obtenir une liste:

- soit des dernières lignes écrites par l'opérateur (on ne garde que les 12 dernières).

- soit des mots enregistrés.

Les lignes ou mots sont vus un par un; on peut (voir panneau en haut de l'écran) revenir au précédent, passer au suivant ou RECUPERER le texte pour modification. Cette possibilité évite bien des frappes et donc des erreurs de manipulation.

REMARQUES: Dans la liste des mots enregistrés, POUR MACHIN (?) signifie que le mot MACHIN a été utilisé (par erreur ou parce qu'un mot défini comporte MACHIN) mais que MACHIN n'est pas (pas encore?) défini. Si on frappait un ligne contenant MACHIN, on obtiendrait le message "Je ne sais pas quoi faire pour MACHIN".

Il n'a pas été prévu de pouvoir effacer un mot écrit par erreur et non utilisé, mais, dans la mesure où la place prise est très faible, ce point n'est pas trop pénalisant. En cas de besoin, utiliser la commande SAUVE et ne pas sauver les mots non définis; utiliser ensuite TERMINE, puis relancer le programme; RAMENE permettra alors de récupérer les mots sauvés, les mots inutiles étant disparus.

SAUVE: permet de sauver dans un fichier les mots définis. Ces mots peuvent être récupérés par la commande RAMENE. Faire remarquer que les mots définis peuvent être gardés, et récupérés, ce qui est un encouragement (un de plus) à découper un problème en sous-programmes.

Le programme demande un nom de fichier, exemples: MDV1 SAUV LOG, puis pour chaque mot défini si on le garde ou pas. On peut ainsi libérer la machine de mots devenus inutiles. Veiller à ce que le fichier n'existe pas déjà; cette commande ne peut détruire un fichier pour le remplacer.

RAMENE: permet de RAMENER des mots SAUVES sur cassette.
Essayez: RAMENE

.... Nom du fichier: MDV1 DEMO LOG

Quand les mots auront été lus, tapez: DEMO.

Vous pourrez alors voir des programmes donnés comme exemple se dérouler.

Les fichiers LOGO peuvent être copiés sur imprimante avec la commande COPY de BASIC. (Mettre l'imprimante en mode LF RC auto, car les listes de commandes sont souvent plus longues qu'une ligne).

TERMINE: Termine l'exécution de TORTUE et retourne au moniteur de la machine. Tous les mots non SAUVES sont perdus.

IV CONTROLE DE DEROULEMENT:

Un programme qui se déroule mal ou trop longtemps peut être interrompu par appui sur la touche (ESC).

Parfois, on veut savoir la position exacte de la tortue. Les commandes POS (position) et CAP permettent respectivement l'affichage des coordonnées ou du cas dans le cadre PANNEAU. Ces commandes ne sont pas comme en LOGO de véritables FONCTIONS rendant une valeur (la notion de fonction n'a pas été développée), elles permettent cependant de connaître avec plus de précision l'état de la tortue et facilitent les mises au point.

V FICHIERS FOURNIS:

T EXE	: interrupteur de commandes de type LOGO.
BOOT	: permet le démarrage du programme depuis le micro-drive gauche.
CLONE	: recopie dans le drive gauche de l'original placé à droite (commande sous BASIC LRUN MDV2,CLONE)
GRAPHES.DAT	: définition des règles de syntaxe admises.
COMMANDES.DAT	: liste des commandes utilisées.
DEMO.LOG	: programmes LOGO donnés en exemple.