

## LOGO, LISTES ET AFFECTATIONS

Danielle ORTOLLAND

But : donner une représentation de la notion d'affectation dans un langage qui utilise le traitement de listes et faire travailler sur les structures arborescentes.

Préalables supposés connus : mots et listes en LOGO.

### MISE EN SITUATION SANS ORDINATEUR.

#### 1. Décodage

##### *Situation matérielle*

De grandes enveloppes A, B, C...portant des noms de régions.  
(exemple : A "Nord-Pas-de-Calais B "Rhône-Alpes C "Languedoc-Roussillon)

Des enveloppes moyennes  $A_i$ ,  $B_j$ ,  $C_k$  portant des noms de départements de ces régions.

(exemple :  $A_1$  "Nord       $A_2$  "Pas-de-Calais  
 $B_1$  "Savoie       $B_2$  "Isère       $B_3$  "Rhône       $B_4$  "Ain  
 $C_1$  "Aude       $C_2$  "Hérault )

De petites enveloppes  $A_{ij}$ ... portant des noms de villes de ces départements.

(exemple :

$A_{11}$ "Lille	$A_{12}$ "Douai	$A_{13}$ "Dunkerque
$A_{21}$ "Calais	$B_{21}$ "Grenoble	$B_{31}$ "Lyon
$B_{41}$ "Nantua	$B_{42}$ "Bourg	$A_{22}$ "Arras
$B_{11}$ "Chambéry	$C_{11}$ "Carcassonne	$C_{21}$ "Narbonne)

On recherche Monsieur X, dont le nom est écrit sur un carton blanc, et qui réside dans l'une des villes (le carton est dans l'une des petites enveloppes). Chaque enveloppe représentant une ville est placée dans une enveloppe-département. Chaque enveloppe de région contient

les enveloppes-département de cette région. Sont donc seules visibles les enveloppes A, B, C.

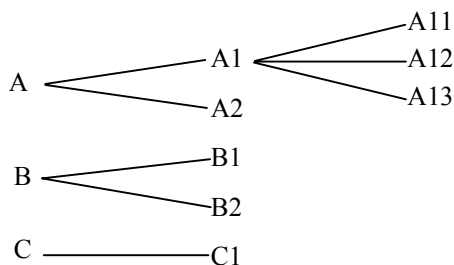
### ***La situation***

On se répartit en deux équipes + un meneur de jeu. Chaque équipe à tour de rôle interroge le meneur de jeu en lui donnant le nom d'une enveloppe visible (Exemple : "Nord-Pas-de-Calais). Le meneur de jeu ouvre l'enveloppe et lit les étiquettes des enveloppes apparentes ("Nord "Pas-de-Calais). La même équipe continue jusqu'à l'ouverture d'une enveloppe vide ; puis c'est à l'autre équipe d'interroger le meneur de jeu. L'équipe gagnante est la première qui découvre Monsieur X.

### ***Intérêt***

Mettre en évidence la nécessité de représenter la situation pour ne pas poser plusieurs fois les mêmes questions, pour explorer de façon systématique, ni omettre de possibilité.

Une représentation arborescente est intéressante :



## **2. Codage**

Une équipe  $E_1$  dispose d'enveloppes de trois ou quatre tailles différentes, de crayons de bois, de gommes. Elle doit de la même façon inventer un univers dans lequel elle place un objet. Elle va représenter cet univers grâce aux enveloppes. Une autre équipe  $E_2$  devra découvrir où a été placé l'objet comme précédemment. On note le nombre de questions posées par  $E_2$  pour découvrir l'objet ; puis on inverse les rôles de  $E_1$  et  $E_2$ .

L'équipe gagnante est celle qui totalise le moins de questions avec le même nombre d'enveloppes à coder au départ. Exemple d'univers possibles : Course au Trésor, Tintin enquête...

Le but de ces deux situations, outre la représentation arborescente et la recherche d'une stratégie pour gagner, est de proposer une représentation matérielle de l'activité qui suit, et qui concerne l'affectation en LOGO.

### ***Le codage en logo***

Les mêmes situations seront reprises mais sur ordinateur, grâce à l'instruction "DONNE

#### 1. Décodage

On a enregistré le programme suivant ;

```
DONNE "FRANCE [NORD-PAS-DE-CALAIS RHONE-ALPES LANGUEDOC-ROUSSILLON]
DONNE "NORD-PAS-DE-CALAIS [NORD PAS-DE-CALAIS]
DONNE "NORD [LILLE DOUAI DUNKERQUE]
DONNE "DUNKERQUE [MONSIEUR X N'Y EST PAS]
DONNE "LILLE [OUI, TU AS TROUVÉ MONSIEUR X]
etc.
```

Chacun interrogera l'ordinateur grâce à une première information : le nom de la variable qui permet d'entrer dans l'arborescence (ici : "FRANCE) et grâce à l'instruction ECRIS CHOSE ou ECRIS

```
Exemple :   ECRIS CHOSE "FRANCE
             ECRIS CHOSE "NORD-PAS-DE-CALAIS
```

### ***2. Codage***

Chacun pourra de la même façon inventer un univers et y placer un objet grâce à l'instruction "DONNE

### ***3. Bilan***

On voit que l'instruction DONNE revient ici à affecter une liste, celle qui figure entre crochets, à une variable dont le nom est le mot qui figure juste après DONNE. Une représentation matérielle peut être une enveloppe ou une boîte sur laquelle on inscrit le nom de la variable, la liste étant placée à l'intérieur de l'enveloppe ou de la boîte.

L'instruction CHOSE "X rend l'objet placé dans l'enveloppe ou la boîte, ou la variable "X.

## RÉFLEXIONS POUR ALLER PLUS LOIN

1. Ajoutons les affectations supplémentaires suivantes :

DONNE "ALSACE [BAS-RHIN]

DONNE "BAS-RHIN [STRASBOURG]

DONNE "STRASBOURG [NON, PERSONNE EN VUE]

Que va répondre l'ordinateur aux ordres suivants :

? ÉCRIS CHOSE "ALSACE

? ÉCRIS CHOSE "BAS-RHIN

ÉCRIS CHOSE "ALSACE ?

ÉCRIS CHOSE :ALSACE

2. Autre problème : il faut modifier le contenu de "FRANCE et y ajouter "ALSACE. Comment le faire sans retaper la liste déjà enregistrée ? Il y a au moins trois possibilités :

x on édite "FRANCE ED ["FRANCE

on tape la correction dans l'éditeur on sort par CNT Q

x on utilise une autre variable "Z

DONNE "Z :FRANCE

DONNE "FRANCE PH :Z "ALSACE

On fabrique ici une nouvelle liste dont les éléments sont ceux de la liste "FRANCE initiale à laquelle on adjoint "ALSACE.

x Mais en fait cette variable Z est inutile ; on peut taper directement (en revenant à l'initial précédent) :

DONNE "FRANCE PH :FRANCE "ALSACE

:FRANCE sera ici interprété comme l'ancien contenu de la variable "FRANCE et "FRANCE comme le nom de la variable dont on va changer le contenu.

3. Donnons une autre représentation de la situation initiale avec les enveloppes et comparons-la à la précédente.

Appelons FRANCEBIS cette nouvelle représentation obtenue en tapant :

DONNE "FRANCEBIS [[ [LILLE DOUAI DUNKERQUE] [CALAIS ARRAS]]  
[[ [CHAMBERY] [GRENOBLE] [LYON] [NANTUA BOURG]] [[CARCASSONNE  
NARBONNE] [ ] ] ] ] ]

← *liste vide représentant l'Hérault*

Rappelons que la première représentation était obtenue par :

DONNE "FRANCE NORD-PAS-DE-CALAIS RHONE-ALPES LANGUEDOC-ROUSSILLON]

DONNE "NORD-PAS-DE-CALAIS [NORD PAS-DE-CALAIS]

etc. Que va répondre l'ordinateur aux ordres suivants :

EC PREM "FRANCEBIS	COMPTE "FRANCEBIS
EC PREM "FRANCE	COMPTE "FRANCE
EC PREM PREM "FRANCEBIS	EC PREM PREM "FRANCE

Pouvez-vous l'expliquer ?

EC DER SP "FRANCEBIS	EC DER SP "FRANCE
----------------------	-------------------

## UN JEU UTILISABLE AVEC DES ENFANTS PLUS JEUNES

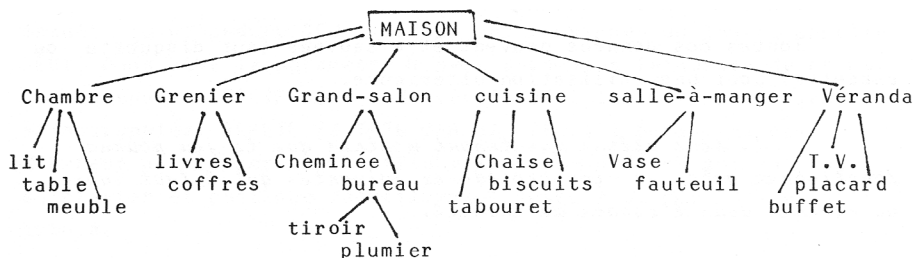
Nous nous proposons de construire un jeu où c'est l'ordinateur qui placera au hasard un objet de notre choix à l'une des terminaisons d'un arbre. Le but du jeu est toujours de retrouver cet objet, mais nous voulons éviter aux élèves de taper

EC CHOSE " ou EC :

De plus l'ordinateur comptera le nombre de coups mis pour trouver l'objet. Pour cela :

### 1. prévoir l'arbre avec tes mots désirés

Ce peut être, par exemple, l'arbre suivant :



Choisir l'objet qui sera caché ; par exemple, une souris.

### 2. Enregistrer toutes ces données

Pour l'instant, nous ne nous préoccupons pas de l'objet caché, ni des messages à placer aux extrémités de l'arbre. Tout ceci sera fait lors de l'introduction des procédures permettant le jeu au paragraphe 3.

Ainsi :

DONNE "MAISON [CHAMBRE GRENIER GRAND-SALON CUISINE SALLE-A-MANGER VERANDA]  
 DONNE "VERANDA [PLACARD BUFFET TELEVISION]  
 DONNE "SALLE-A-MANGER [FAUTEUIL VASE]  
 DONNE "CUISINE [TABOURET BISCUITS CHAISE]  
 DONNE "GRENIER [LIVRES COFFRE]

DONNE "CHAMBRE [LIT TABLE MEUBLE]

DONNE "BUREAU [TIROIR PLUMIER]

Continuer les affectations de la façon suivante enregistrer le mot de passe qui permet de débiter par exemple :

DONNE "MOT-DE-PASSE "MAISON

Puis le message de départ M1, par exemple :

DONNE "M1 [OU SE TROUVE LA SOURIS]

Enregistrer toutes les terminaisons ou feuilles de l'arbre, par exemple :

DONNE "L [LIT TABLE MEUBLE LIVRES COFFRE CHEMINEE TIROIR PLUMIER  
TABOURET BISCUITS CHAISE FAUTEUIL VASE PLACARD BUFFET TV]

Ensuite tous les mots figurant dans l'arbre, pour que l'ordinateur sanctionne les fautes d'orthographe, par exemple

DONNE "M [LIT TABLE MEUBLE LIVRES COFFRE CHEMINEE BUREAU TIROIR  
PLUMIER TABOURET BISCUITS CHAISE FAUTEUIL VASE PLACARD BUFFET  
TV CHAMBRE GRENIER GRAND-SALON CUISINE VERANDA SALLE-A-MANGER  
MAISON]

Toutes ces données peuvent être sauvées (sur disquette ou cassette) pour une utilisation ultérieure.

3. Nous allons maintenant montrer que le jeu pourra fonctionner grâce au cinq procédures suivantes qu'il faut taper ou ramener dans l'espace de travail.

POUR JEU

NY :L

NYESTPAS :L

ME 25 FOURS [2 5] EC :M1

FOURS [1 10] TAPE [A TOI DE LE TROUVER EN TAPANT :]

TAPE CAR 32 EC : MOT-DE-PASSE FOURS [3 20] FCT 1

JEU 1

FIN

POUR NY :L

DONNE "CACHETTE ITEM 1 + HASARD COMPTE :L :L FIN

POUR NYESTPAS :L

SI VIDE? :L [STOP]

SI EGAL? PREM :L :CACHETTE [DONNE PREM :L [BRAVO ! TU AS TROUVE]]

[DONNE PREM :L [NON, TU N'AS PAS TROUVE]]

NYESTPAS SP :L

FIN

```

POUR JEU1 :N
AFFECTE ME 25 FOURS [3 10] FCT 3 EC CHOSE :MOT
SI EGAL ? :MOT :CACHETTE [FOURS [3 15] TAPE [TU AS GAGNE EN]
TAPE CAR 32 TAPE :N TAPE CAR 32 EC "COUPS STOP]
FOURS [3 20] FCT 1
JEU1 :N + 1
FIN

POUR AFFECTE
DONNE "MOT PREM LL
SI NON MEMBRE ? :MOT :M [FOURS [3 22] "RECOMMENCE FOURS[3 20]
REPETE 36 [TAPE CAR 32] FOURS [3 20] AFFECTE]
FIN

```

La procédure d'appel est la procédure JEU. Elle appelle dans son déroulement les procédures NY, NYESTPAS, et JEU1.

NY tire au hasard un mot de la liste L. Ici, L étant la liste des terminaisons de l'arbre, NY triera au hasard l'une de ces terminaisons qui sera la cachette de la souris. La procédure NYESTPAS affecte à la cachette la liste [BRAVO! TU AS TROUVE] et à tous les autres éléments de la liste : [NON, TU N'AS PAS TROUVE].

Ensuite la procédure JEU affiche les consignes du jeu et appelle JEU1. Dans JEU1 le paramètre N sert à compter le nombre de coups. La procédure AFFECTE permet de vérifier si les mots sont bien orthographiés. REPETE 36 TAPE CAR 32 efface l'écran, en cas d'erreur au clavier. JEU1 affiche le contenu de ce qui est tapé au clavier et continue récursivement jusqu'à ce que l'objet soit trouvé.

Danielle ORTOLLAND  
E.N. Lille

## BIBLIOGRAPHIE :

*Réseau Nord* n° 9, Cahier spécial LOGO "Tintin enquête", E. Ortolland  
*Bull. APMEP* n° 351 (déc. 85), J.N. Gers : Pluridisciplinarité avec LOGO.