

ALGOBOX DAY 3

BATONNETS

VARIABLES

```
playerOne EST_DU_TYPE CHAINE
playerTwo EST_DU_TYPE CHAINE
winner EST_DU_TYPE CHAINE
currentPlayer EST_DU_TYPE NOMBRE
stick EST_DU_TYPE CHAINE
totalSticks EST_DU_TYPE NOMBRE
selectedSticks EST_DU_TYPE NOMBRE
DEBUT_ALGORITHME
// BASE
stick PREND_LA_VALEUR " !"
playerOne PREND_LA_VALEUR 1
playerTwo PREND_LA_VALEUR 2
winner PREND_LA_VALEUR " "
totalSticks PREND_LA_VALEUR 16
currentPlayer PREND_LA_VALEUR 1
AFFICHER* "Bienvenue dans le jeu des bâtonnets"
AFFICHER* "Vous jouez chacun votre tour et vous devez sélectionner entre 1 et 3 bâtonnets"
AFFICHER* "Celui qui récupère le dernier bâtonnet à perdu !"
AFFICHER* "Veuillez rentrer votre pseudo avant de commencer"
LIRE playerOne
AFFICHER "le joueur 1 est : "
AFFICHER* playerOne
LIRE playerTwo
AFFICHER "le joueur 2 est : "
AFFICHER* playerTwo
AFFICHER* " "
AFFICHER* "-----"
AFFICHER* " "
// Début de la gestion du jeu
TANT_QUE (totalSticks>0) FAIRE
    DEBUT_TANT_QUE

        // Changement de joueur
        SI (currentPlayer==1) ALORS
            DEBUT_SI
                currentPlayer PREND_LA_VALEUR 2
            FIN_SI
        SINON
            DEBUT_SINON
                currentPlayer PREND_LA_VALEUR 1
            FIN_SINON

        //Affichage consigne/tour + choix du nombre de bâtonnets à enlever pour chaque tour
        SI (currentPlayer==1 ET totalSticks>0) ALORS
            DEBUT_SI
                AFFICHER "Il reste : "
                AFFICHER totalSticks
                AFFICHER* " bâtonnets"
                AFFICHER playerOne
                AFFICHER " Combien de bâtonnets enlèves-tu ?"
```

```

    FIN_SI
    SINON
        DEBUT_SINON
        AFFICHER "Il reste : "
        AFFICHER totalSticks
        AFFICHER* " bâtonnets"
        AFFICHER playerTwo
        AFFICHER* " Combien de bâtonnets enlèves-tu ?"
        FIN_SINON

        LIRE selectedSticks
//Condition d'affichage du message d'erreur si la saisie n'est pas cohérente
        TANT_QUE (selectedSticks<1 OU selectedSticks>3 OU selectedSticks>totalSticks)
FAIRE
        DEBUT_TANT_QUE
        AFFICHER* "Impossible ! Tu dois sélectionner entre 1 et 3 bâtonnets ... Et tu
ne peux pas prendre plus de bâtonnets qu'il n'en reste !"
        AFFICHER "En tout, il y a : "
        AFFICHER totalSticks
        AFFICHER " bâtonnets"
        LIRE selectedSticks
        FIN_TANT_QUE

        //Gestion des bâtonnets
        totalSticks PREND_LA_VALEUR totalSticks-selectedSticks

        //Fin de partie
        SI (totalSticks==0) ALORS
            DEBUT_SI
            SI (currentPlayer==1) ALORS
                DEBUT_SI
                winner PREND_LA_VALEUR playerTwo
                FIN_SI
            SINON
                DEBUT_SINON
                winner PREND_LA_VALEUR playerOne
                FIN_SINON
                AFFICHER "C'est terminé ! "
                AFFICHER "le gagnant est : "
                AFFICHER winner
                FIN_SI
        FIN_TANT_QUE

```

Exo dessin 4-A

FONCTIONS_UTILISEES

VARIABLES

x EST_DU_TYPE NOMBRE

z EST_DU_TYPE NOMBRE

y EST_DU_TYPE NOMBRE

p EST_DU_TYPE NOMBRE

n EST_DU_TYPE CHAINE

b EST_DU_TYPE CHAINE

```

DEBUT_ALGORITHME
LIRE p
    n PREND_LA_VALEUR "■"
    b PREND_LA_VALEUR "□"
    z PREND_LA_VALEUR 1

    POUR y ALLANT_DE z A p
        DEBUT_POUR
        POUR x ALLANT_DE z A p
            DEBUT_POUR
            SI (x==z OU x==p OU y==z OU y==p) ALORS
                DEBUT_SI
                AFFICHER n
                FIN_SI
            SINON
                DEBUT_SINON
                SI (x == z+1 OU x == p-1 OU y == z+1 OU y == p-1) ALORS
                    DEBUT_SI
                    AFFICHER b
                    FIN_SI
                SINON
                    DEBUT_SINON
                    AFFICHER n
                    FIN_SINON
            FIN_SINON
        FIN_POUR
        AFFICHER* " "
    FIN_POUR

```

FIN_ALGORITHME

Exo 4-b - La croix

FONCTIONS_UTILISEES

VARIABLES

x EST_DU_TYPE NOMBRE

z EST_DU_TYPE NOMBRE

y EST_DU_TYPE NOMBRE

p EST_DU_TYPE NOMBRE

n EST_DU_TYPE CHAINE

b EST_DU_TYPE CHAINE

DEBUT_ALGORITHME

LIRE p

```

    n PREND_LA_VALEUR "■"
    b PREND_LA_VALEUR "□"
    z PREND_LA_VALEUR 1

```

```

    POUR y ALLANT_DE z A p
        DEBUT_POUR
        POUR x ALLANT_DE z A p
            DEBUT_POUR
            SI (x==z OU x==p OU y==z OU y==p) ALORS
                DEBUT_SI
                AFFICHER n
            FIN_SI
        FIN_POUR
    FIN_POUR

```

```

        FIN_SI
    SINON
        DEBUT_SINON
        // x==y permet d'avancer sur la diagonale principale x1=y1 puis x2=y2 etc
        // p-y permet d'avancer sur la diagonale secondaire en inversant la
progression on commence à y=1 donc p-1 = avant dernier etc
        // rajout du z pour s'assurer qu'on commence bien au début sinon ça décale.
        SI (x==y OU x==(p-y+z)) ALORS
            DEBUT_SI
            AFFICHER n
            FIN_SI
        SINON
            DEBUT_SINON
            AFFICHER b
            FIN_SINON
    FIN_SINON
FIN_POUR
AFFICHER* " "
FIN_POUR

```

FIN_ALGORITHME

Exo 5 - Alternance par rapport au bord

FONCTIONS_UTILISEES

VARIABLES

x EST_DU_TYPE NOMBRE

z EST_DU_TYPE NOMBRE

y EST_DU_TYPE NOMBRE

p EST_DU_TYPE NOMBRE

n EST_DU_TYPE CHAINE

b EST_DU_TYPE CHAINE

distance1 EST_DU_TYPE NOMBRE

distance2 EST_DU_TYPE NOMBRE

distance3 EST_DU_TYPE NOMBRE

distance4 EST_DU_TYPE NOMBRE

visuel EST_DU_TYPE NOMBRE

DEBUT_ALGORITHME

LIRE p

n PREND_LA_VALEUR "■"

b PREND_LA_VALEUR "□"

z PREND_LA_VALEUR 1

POUR y ALLANT_DE z A p

DEBUT_POUR

POUR x ALLANT_DE z A p

DEBUT_POUR

//distance donne la position du point par rapport au bord

//x-z = distance ligne à gauche (minValue)

distance1 PREND_LA_VALEUR x-z

//p-x ligne à droite (maxValue)

distance2 PREND_LA_VALEUR p-x

```

//y-z distance colonne vers le haut (minValue)
distance3 PREND_LA_VALEUR y-z
//y-p distance colonne vers le bas (maxValue)
distance4 PREND_LA_VALEUR p-y

//Determine la plus petite distance pour former une grille colorée en
fonction de la distance avec le bord
// La plus petite distance permet d'indiquer à quel "visuel" le point
appartient: la plus petite valeur est toujours la distance au bord.
visuel PREND_LA_VALEUR distance1
SI (distance2>visuel) ALORS
    DEBUT_SI
    visuel PREND_LA_VALEUR distance2
    FIN_SI
    SI (distance3>visuel) ALORS
        DEBUT_SI
        visuel PREND_LA_VALEUR distance3
        FIN_SI
        SI (distance4>visuel) ALORS
            DEBUT_SI
            visuel PREND_LA_VALEUR distance4
            FIN_SI

//Determine couleur du visuel
SI (visuel%2==0) ALORS
    DEBUT_SI
    AFFICHER b
    FIN_SI
SINON
    DEBUT_SINON
    AFFICHER n
    FIN_SINON
    FIN_POUR
    AFFICHER* " "
FIN_POUR

FIN_ALGORITHME

```

Exo 6