PARTIE 5 – LES BOUCLES

**ENONCE DES EXERCICES**

# Exercice 5.1

Ecrire un algorithme qui demande à l’utilisateur un nombre compris entre 1 et 3 jusqu’à ce que la réponse convienne.

Nbr EST\_DU\_TYPE NOMBRE

Nbr PREND\_LA\_VALEUR 0

TANT\_QUE (Nbr < 1 OU Nbr > 3) FAIRE

DEBUT\_TANT\_QUE

AFFICHER "Choisissez un nombre compris entre 1 et 3 :"

LIRE Nbr

FIN\_TANT\_QUE

AFFICHER Nbr

# Exercice 5.2

Ecrire un algorithme qui demande un nombre compris entre 10 et 20, jusqu’à ce que la réponse convienne. En cas de réponse supérieure à 20, on fera apparaître un message : « Plus petit ! », et inversement, « Plus grand ! » si le nombre est inférieur à 10.

Nbr EST\_DU\_TYPE NOMBRE

Nbr PREND\_LA\_VALEUR 0

AFFICHER "Choisissez un nombre compris entre 10 et 20 :"

LIRE Nbr

TANT\_QUE (Nbr < 10 OU Nbr > 20) FAIRE

DEBUT\_TANT\_QUE

SI (Nbr < 10) ALORS

DEBUT\_SI

AFFICHER "nombre trop petit. Saisissez a nouveau:"

LIRE Nbr

FIN\_SI

SINON

DEBUT\_SINON

AFFICHER "nombre trop grand. Saisissez a nouveau:"

LIRE Nbr

FIN\_SINON

FIN\_TANT\_QUE

AFFICHER Nbr

# Exercice 5.3

Ecrire un algorithme qui demande un nombre de départ, et qui ensuite affiche les dix nombres suivants. Par exemple, si l'utilisateur entre le nombre 17, le programme affichera les nombres de 18 à 27.

Nbr EST\_DU\_TYPE NOMBRE

Boucle EST\_DU\_TYPE NOMBRE

Boucle PREND\_LA\_VALEUR 0

AFFICHER "Choisissez un nombre :"

LIRE Nbr

TANT\_QUE (Boucle < 10) FAIRE

DEBUT\_TANT\_QUE

Nbr PREND\_LA\_VALEUR Nbr + 1

Boucle PREND\_LA\_VALEUR Boucle + 1

AFFICHER\* Nbr

FIN\_TANT\_QUE

# Exercice 5.4

Ecrire un algorithme qui demande un nombre de départ, et qui ensuite écrit la table de multiplication de ce nombre, présentée comme suit (cas où l'utilisateur entre le nombre 7) :

Table de 7 :

7 x 1 = 7

7 x 2 = 14

7 x 3 = 21

…

7 x 10 = 70

Nbr EST\_DU\_TYPE NOMBRE

Boucle EST\_DU\_TYPE NOMBRE

Boucle PREND\_LA\_VALEUR 1

AFFICHER "Choisissez un nombre:"

LIRE Nbr

TANT\_QUE (Boucle <= 10) FAIRE

DEBUT\_TANT\_QUE

AFFICHER Nbr

AFFICHER " X "

AFFICHER Boucle

AFFICHER " = "

AFFICHERCALCUL\* Nbr \* Boucle

Boucle PREND\_LA\_VALEUR Boucle + 1

FIN\_TANT\_QUE

# Exercice 5.5

Ecrire un algorithme qui demande un nombre de départ, et qui calcule la somme des entiers jusqu’à ce nombre. Par exemple, si l’on entre 5, le programme doit calculer :

1 + 2 + 3 + 4 + 5 = 15

NB : on souhaite afficher uniquement le résultat, pas la décomposition du calcul.

Nbr EST\_DU\_TYPE NOMBRE

Boucle EST\_DU\_TYPE NOMBRE

Resultat EST\_DU\_TYPE NOMBRE

Boucle PREND\_LA\_VALEUR 1

AFFICHER "Choisissez un nombre :"

LIRE Nbr

TANT\_QUE (Boucle <= Nbr) FAIRE

DEBUT\_TANT\_QUE

Resultat PREND\_LA\_VALEUR Resultat + Boucle

Boucle PREND\_LA\_VALEUR Boucle + 1

FIN\_TANT\_QUE

AFFICHER Resultat

# Exercice 5.6

Ecrire un algorithme qui demande un nombre de départ, et qui calcule sa factorielle.

NB : la factorielle de 8, notée 8 !, vaut

1 x 2 x 3 x 4 x 5 x 6 x 7 x 8

Nbr EST\_DU\_TYPE NOMBRE

Boucle EST\_DU\_TYPE NOMBRE

Resultat EST\_DU\_TYPE NOMBRE

Boucle PREND\_LA\_VALEUR 1

Resultat PREND\_LA\_VALEUR 1

AFFICHER "Choisissez un nombre :"

LIRE Nbr

TANT\_QUE (Boucle <= Nbr) FAIRE

DEBUT\_TANT\_QUE

Resultat PREND\_LA\_VALEUR Resultat \* Boucle

Boucle PREND\_LA\_VALEUR Boucle + 1

FIN\_TANT\_QUE

AFFICHER Resultat

# Exercice 5.7

Ecrire un algorithme qui demande successivement 20 nombres à l’utilisateur, et qui lui dise ensuite quel était le plus grand parmi ces 20 nombres :

Entrez le nombre numéro 1 : 12 Entrez le nombre numéro 2 : 14 etc.

Entrez le nombre numéro 20 : 6

Le plus grand de ces nombres est : 14

Modifiez ensuite l’algorithme pour que le programme affiche de surcroît en quelle position avait été saisie ce nombre :

C’était le nombre numéro 2

Nbr EST\_DU\_TYPE LISTE

Boucle EST\_DU\_TYPE NOMBRE

Position EST\_DU\_TYPE NOMBRE

Resultat EST\_DU\_TYPE NOMBRE

LIRE Nbr[1]

LIRE Nbr[2]

LIRE Nbr[3]

LIRE Nbr[4]

LIRE Nbr[5]

LIRE Nbr[6]

LIRE Nbr[7]

LIRE Nbr[8]

LIRE Nbr[9]

LIRE Nbr[10]

LIRE Nbr[11]

LIRE Nbr[12]

LIRE Nbr[13]

LIRE Nbr[14]

LIRE Nbr[15]

LIRE Nbr[16]

LIRE Nbr[17]

LIRE Nbr[18]

LIRE Nbr[19]

LIRE Nbr[20]

TANT\_QUE (Boucle < 20) FAIRE

DEBUT\_TANT\_QUE

SI (Nbr[Boucle] > Resultat) ALORS

DEBUT\_SI

Resultat PREND\_LA\_VALEUR Nbr[Boucle]

Position PREND\_LA\_VALEUR Boucle

FIN\_SI

Boucle PREND\_LA\_VALEUR Boucle + 1

FIN\_TANT\_QUE

AFFICHER "Le nombre le plus grand de la liste est :"

AFFICHER\* Resultat

AFFICHER "C'était le nombre numéro "

AFFICHER Position

# Exercice 5.8

Réécrire l’algorithme précédent, mais cette fois-ci on ne connaît pas d’avance combien l’utilisateur souhaite saisir de nombres. La saisie des nombres s’arrête lorsque l’utilisateur entre un zéro.

Nbr EST\_DU\_TYPE LISTE

Boucle EST\_DU\_TYPE NOMBRE

Position EST\_DU\_TYPE NOMBRE

Resultat EST\_DU\_TYPE NOMBRE

LIRE Nbr[1]

Boucle PREND\_LA\_VALEUR 1

TANT\_QUE (Nbr[Boucle] != 0) FAIRE

DEBUT\_TANT\_QUE

SI (Nbr[Boucle] > Resultat) ALORS

DEBUT\_SI

Resultat PREND\_LA\_VALEUR Nbr[Boucle]

Position PREND\_LA\_VALEUR Boucle

FIN\_SI

Boucle PREND\_LA\_VALEUR Boucle + 1

lire Nbr[Boucle]

FIN\_TANT\_QUE

AFFICHER "Le nombre le plus grand de la liste est :"

AFFICHER\* Resultat

AFFICHER "C'était le nombre numéro "

AFFICHER Position

# Exercice 5.9

Lire la suite des prix (en euros entiers et terminée par zéro) des achats d’un client. Calculer la somme qu’il doit, lire la somme qu’il paye, et simuler la remise de la monnaie en affichant les textes "10 Euros", "5 Euros" et "1 Euro" autant de fois qu’il y a de coupures de chaque sorte à rendre.

Nbr EST\_DU\_TYPE LISTE

Boucle EST\_DU\_TYPE NOMBRE

PrixTotal EST\_DU\_TYPE NOMBRE

PrixPaye EST\_DU\_TYPE NOMBRE

ARendre EST\_DU\_TYPE NOMBRE

Billet10 EST\_DU\_TYPE NOMBRE

Billet5 EST\_DU\_TYPE NOMBRE

Piece1 EST\_DU\_TYPE NOMBRE

AFFICHER " Entrez la valeur de vos articles. Lorsque vous avez fini, tapez 0."

Boucle PREND\_LA\_VALEUR 0

TANT\_QUE (Nbr[Boucle] != 0) FAIRE

DEBUT\_TANT\_QUE

Boucle PREND\_LA\_VALEUR Boucle + 1

LIRE Nbr[Boucle]

PrixTotal PREND\_LA\_VALEUR PrixTotal + Nbr[Boucle]

FIN\_TANT\_QUE

AFFICHER "Le prix total est de : "

AFFICHER PrixTotal

AFFICHER\* " euros "

LIRE PrixPaye

ARendre PREND\_LA\_VALEUR PrixPaye - PrixTotal

Billet10 PREND\_LA\_VALEUR (ARendre-(ARendre % 10))/10

ARendre PREND\_LA\_VALEUR ARendre - Billet10 \* 10

Billet5 PREND\_LA\_VALEUR (ARendre-(ARendre % 5))/5

Piece1 PREND\_LA\_VALEUR ARendre - Billet5 \* 5

AFFICHER "Billet de dix a rendre : "

AFFICHER\* Billet10

AFFICHER "Billet de cinq a rendre : "

AFFICHER\* Billet5

AFFICHER "Piece de un a rendre : "

AFFICHER\* Piece1

TANT\_QUE (Billet10 > 0) FAIRE

DEBUT\_TANT\_QUE

AFFICHER\* "Billet de dix"

Billet10 PREND\_LA\_VALEUR Billet10 - 1

FIN\_TANT\_QUE

TANT\_QUE (Billet5 > 0) FAIRE

DEBUT\_TANT\_QUE

AFFICHER\* "Billet de cinq"

Billet5 PREND\_LA\_VALEUR Billet5 - 1

FIN\_TANT\_QUE

TANT\_QUE (Piece1 > 0) FAIRE

DEBUT\_TANT\_QUE

AFFICHER\* "Piece de 1"

Piece1 PREND\_LA\_VALEUR Piece1 - 1

FIN\_TANT\_QUE

FIN\_ALGORITHME

# Exercice 5.10

Écrire un algorithme qui permette de connaître ses chances de gagner au tiercé, quarté, quinté et autres impôts volontaires.

On demande à l’utilisateur le nombre de chevaux partants, et le nombre de chevaux joués. Les deux messages affichés devront être :

Dans l’ordre : une chance sur X de gagner

Dans le désordre : une chance sur Y de gagner

X et Y nous sont donnés par la formule suivante, si n est le nombre de chevaux partants et p le nombre de chevaux joués (on rappelle que le signe ! signifie "factorielle", comme dans l'exercice 5.6 ci-dessus) :

1. = n ! / (n - p) !
2. = n ! / (p ! \* (n – p) !)

NB : cet algorithme peut être écrit d’une manière simple, mais relativement peu performante. Ses performances peuvent être singulièrement augmentées par une petite astuce. Vous commencerez par écrire la manière la plus simple, puis vous identifierez le problème, et écrirez une deuxième version permettant de le résoudre.

ChevauxPartants EST\_DU\_TYPE NOMBRE

ChevauxJoues EST\_DU\_TYPE NOMBRE

Ordre EST\_DU\_TYPE NOMBRE

desordre EST\_DU\_TYPE NOMBRE

LIRE ChevauxJoues

LIRE ChevauxPartants

Ordre PREND\_LA\_VALEUR ALGOBOX\_FACTORIELLE(ChevauxPartants)/ALGOBOX\_FACTORIELLE(ChevauxPartants - ChevauxJoues)

desordre PREND\_LA\_VALEUR ALGOBOX\_FACTORIELLE(ChevauxPartants)/(ALGOBOX\_FACTORIELLE(ChevauxJoues)\*ALGOBOX\_FACTORIELLE(ChevauxPartants - ChevauxJoues))

AFFICHER\* Ordre

AFFICHER\* desordre