

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

Code de l'algorithme

```
1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5      AFFICHER "Entrez un nombre compris entre 1 et 3"
6      LIRE nb
7      TANT_QUE (nb<1 OU nb>3) FAIRE
8          DEBUT_TANT_QUE
9              AFFICHER "Entrez une valeur de 1 à 3"
10             LIRE nb
11         FIN_TANT_QUE
12     AFFICHER "Okay"
13 FIN_ALGORITHME
```

Résultats

```
***Algorithme lancé***
Entrez un nombre compris entre 1 et 3
Entrer nb : 9
Entrez une valeur de 1 à 3
Entrer nb : 7
Entrez une valeur de 1 à 3
Entrer nb : 4
Entrez une valeur de 1 à 3
Entrer nb : 2
Okay
***Algorithme terminé***
```

Exercice 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.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5      AFFICHER "Entrez un nombre compris entre 10 et 20"
6      LIRE nb
7      TANT_QUE (nb<10 OU nb>20) FAIRE
8          DEBUT_TANT_QUE
9              SI (nb<10) ALORS
10                 DEBUT_SI
11                 AFFICHER "Plus grand !"
12                 FIN_SI
13                 SINON
14                     DEBUT_SINON
15                     SI (nb>20) ALORS
16                         DEBUT_SI
17                         AFFICHER "Plus petit !"
18                         FIN_SI
19                     FIN_SINON
20             AFFICHER "Entrez une valeur de 10 à 20"
21             LIRE nb
22         FIN_TANT_QUE
23     AFFICHER "Okay"
24 FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez un nombre compris entre 10 et 20
Entrer nb : 2
Plus grand !
Entrez une valeur de 10 à 20
Entrer nb : 25
Plus petit !
Entrez une valeur de 10 à 20
Entrer nb : 15
Okay
***Algorithme terminé***

```

Exercice 3 : 1. 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.

Code de l'algorithme

```
1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      end EST_DU_TYPE NOMBRE
5  DEBUT_ALGORITHME
6      AFFICHER "Entrez un nombre"
7      LIRE nb
8      end PREND_LA_VALEUR nb+10
9      TANT_QUE (nb<end) FAIRE
10         DEBUT_TANT_QUE
11             nb PREND_LA_VALEUR nb+1
12             AFFICHER nb
13         FIN_TANT_QUE
14  FIN_ALGORITHME
```

Résultats

```
***Algorithme lancé***
Entrez un nombre
Entrer nb : 17
18
19
20
21
22
23
24
25
26
27
***Algorithme terminé***
```

Exercice 3 : 1. Ecrire un algorithme qui demande un nombre de départ, et qui ensuite affiche les dix nombres suivants. 2. Réécrire l'algorithme précédent, en utilisant cette fois l'instruction Pour

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      i EST_DU_TYPE NOMBRE
5  DEBUT_ALGORITHME
6      AFFICHER "Entrez un nombre"
7      LIRE nb
8      POUR i ALLANT_DE 1 A 10
9          DEBUT_POUR
10         nb PREND_LA_VALEUR nb+1
11         AFFICHER nb
12         FIN_POUR
13  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez un nombre
Entrer nb : 17
18
19
20
21
22
23
24
25
26
27
***Algorithme terminé***

```

Exercice 4 : Ecrire un algorithme qui demande un nombre de départ, et qui ensuite écrit la table de multiplication de ce nombre

Code de l'algorithme

```
1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      i EST_DU_TYPE NOMBRE
5      val EST_DU_TYPE NOMBRE
6  DEBUT_ALGORITHME
7      AFFICHER "Entrez un nombre"
8      LIRE nb
9      POUR i ALLANT_DE 1 A 10
10         DEBUT_POUR
11             val PREND_LA_VALEUR nb*i
12             AFFICHER nb
13             AFFICHER " x "
14             AFFICHER i
15             AFFICHER " = "
16             AFFICHER val
17         FIN_POUR
18  FIN_ALGORITHME
```

Résultats

```
***Algorithme lancé***
Entrez un nombre
Entrer nb : 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
***Algorithme terminé***
```

Exercice 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.

Code de l'algorithme

```
1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      i EST_DU_TYPE NOMBRE
5      val EST_DU_TYPE NOMBRE
6  DEBUT_ALGORITHME
7      AFFICHER "Entrez un nombre"
8      LIRE nb
9      POUR i ALLANT_DE 1 A nb
10         DEBUT_POUR
11             val PREND_LA_VALEUR val+i
12         FIN_POUR
13     AFFICHER val
14  FIN_ALGORITHME
```

Résultats

```
***Algorithme lancé***
Entrez un nombre
Entrer nb : 5
15
***Algorithme terminé***
```

Exercice 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

Code de l'algorithme

```
1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      i EST_DU_TYPE NOMBRE
5      val EST_DU_TYPE NOMBRE
6  DEBUT_ALGORITHME
7      AFFICHER "Entrez un nombre"
8      LIRE nb
9      val PREND_LA_VALEUR 1
10     POUR i ALLANT_DE 1 A nb
11         DEBUT_POUR
12             val PREND_LA_VALEUR val*i
13         FIN_POUR
14     AFFICHER val
15  FIN_ALGORITHME
```

Résultats

```
***Algorithme lancé***
Entrez un nombre
Entrer nb : 8
40320
***Algorithme terminé***
```

Exercice 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

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      i EST_DU_TYPE NOMBRE
5      max_val EST_DU_TYPE NOMBRE
6  DEBUT_ALGORITHME
7      POUR i ALLANT_DE 1 A 20
8          DEBUT_POUR
9              AFFICHER "Entrez un nombre"
10             LIRE nb
11             SI (nb>max_val) ALORS
12                 DEBUT_SI
13                     max_val PREND_LA_VALEUR nb
14                 FIN_SI
15             FIN_POUR
16             AFFICHER "Le plus grand : "
17             AFFICHER max_val
18  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez un nombre
Entrer nb : 32
Entrez un nombre
Entrer nb : 12
Entrez un nombre
Entrer nb : 65
Entrez un nombre
Entrer nb : 2
Entrez un nombre
Entrer nb : 17
Entrez un nombre
Entrer nb : 499
Entrez un nombre

```


Exercice 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.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      i EST_DU_TYPE NOMBRE
5      max_val EST_DU_TYPE NOMBRE
6  DEBUT_ALGORITHME
7      nb PREND_LA_VALEUR 1
8      TANT_QUE (nb!=0) FAIRE
9          DEBUT_TANT_QUE
10             AFFICHER "Entrez un nombre"
11             LIRE nb
12             SI (nb>max_val) ALORS
13                 DEBUT_SI
14                     max_val PREND_LA_VALEUR nb
15                 FIN_SI
16             FIN_TANT_QUE
17             AFFICHER "Le plus grand : "
18             AFFICHER max_val
19  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez un nombre
Entrer nb : 1
Entrez un nombre
Entrer nb : 4
Entrez un nombre
Entrer nb : 76
Entrez un nombre
Entrer nb : -4
Entrez un nombre
Entrer nb : 0
Le plus grand : 76
***Algorithme terminé***

```

Exercice 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.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb EST_DU_TYPE NOMBRE
4      liste_prix EST_DU_TYPE LISTE
5      i EST_DU_TYPE NOMBRE
6      somme EST_DU_TYPE NOMBRE
7      somme_payee EST_DU_TYPE NOMBRE
8      difference EST_DU_TYPE NOMBRE
9      nb_10 EST_DU_TYPE NOMBRE
10     nb_5 EST_DU_TYPE NOMBRE
11 DEBUT_ALGORITHME
12     nb PREND_LA_VALEUR 1
13     TANT_QUE (nb!=0) FAIRE
14         DEBUT_TANT_QUE
15             AFFICHER "Entrez le prix (0 pour terminer)"
16             LIRE nb
17             liste_prix[i] PREND_LA_VALEUR nb
18             i PREND_LA_VALEUR i+1
19             somme PREND_LA_VALEUR somme+nb
20         FIN_TANT_QUE
21     AFFICHER "La somme : "
22     AFFICHER somme
23     AFFICHER " Euros"
24     AFFICHER "Somme payee : "
25     LIRE somme_payee
26     difference PREND_LA_VALEUR somme_payee-somme
27     TANT_QUE (difference>10) FAIRE
28         DEBUT_TANT_QUE
29             nb_10 PREND_LA_VALEUR nb_10+1
30             difference PREND_LA_VALEUR difference-10
31         FIN_TANT_QUE
32     TANT_QUE (difference>5) FAIRE
33         DEBUT_TANT_QUE
34             nb_5 PREND_LA_VALEUR nb_5+1
35             difference PREND_LA_VALEUR difference-5
36         FIN_TANT_QUE
37     AFFICHER "Monnaie : "
38     SI (nb_10>0) ALORS
39         DEBUT_SI
40             AFFICHER nb_10
41             AFFICHER "x 10 Euros "
42         FIN_SI
43     SI (nb_5>0) ALORS
44         DEBUT_SI
45             AFFICHER nb_5
46             AFFICHER "x 5 Euros "
47         FIN_SI
48     AFFICHER difference
49     AFFICHER "x 1 Euro"
50
51 FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez le prix (0 pour terminer)
Entrer nb : 12
Entrez le prix (0 pour terminer)

```

```
Entrer nb : 35
Entrez le prix (0 pour terminer)
Entrer nb : 0
La somme : 47 Euros
Somme payee :
Entrer somme_payee : 100
Monnaie : 5x 10 Euros 3x 1 Euro
***Algorithme terminé***
```

Généré par AlgoBox

Exercice 10 : Écrire un algorithme qui permette de connaître ses chances de gagner au tiercé, quarté, quinté et autres impôts volontaires.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      n EST_DU_TYPE NOMBRE
4      p EST_DU_TYPE NOMBRE
5      i EST_DU_TYPE NOMBRE
6      x EST_DU_TYPE NOMBRE
7      y EST_DU_TYPE NOMBRE
8      p_facto EST_DU_TYPE NOMBRE
9      n_facto EST_DU_TYPE NOMBRE
10     n_p_facto EST_DU_TYPE NOMBRE
11  DEBUT_ALGORITHME
12     AFFICHER "Entrez le nombre de chevaux partants"
13     LIRE n
14     AFFICHER "Entrez le nombre de chevaux joués"
15     LIRE p
16     p_facto PREND_LA_VALEUR 1
17     n_facto PREND_LA_VALEUR 1
18     n_p_facto PREND_LA_VALEUR 1
19     POUR i ALLANT_DE 1 A n
20         DEBUT_POUR
21             n_facto PREND_LA_VALEUR n_facto*i
22         FIN_POUR
23     POUR i ALLANT_DE 1 A n-p
24         DEBUT_POUR
25             n_p_facto PREND_LA_VALEUR n_p_facto*i
26         FIN_POUR
27     x PREND_LA_VALEUR n_facto/n_p_facto
28     POUR i ALLANT_DE 1 A p
29         DEBUT_POUR
30             p_facto PREND_LA_VALEUR p_facto*i
31         FIN_POUR
32     y PREND_LA_VALEUR x/p_facto
33     AFFICHER "Dans l'ordre : 1 chance sur "
34     AFFICHER x
35     AFFICHER " de gagner."
36     AFFICHER "Dans le désordre : 1 chance sur "
37     AFFICHER y
38     AFFICHER " de gagner."
39  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez le nombre de chevaux partants
Entrer n : 10
Entrez le nombre de chevaux joués
Entrer p : 4
Dans l'ordre : 1 chance sur 5040 de gagner.
Dans le désordre : 1 chance sur 210 de gagner.
***Algorithme terminé***

```

Exercice 10 : Écrire un algorithme qui permette de connaître ses chances de gagner au tiercé, quarté, quinté et autres impôts volontaires. CAS 2

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      n EST_DU_TYPE NOMBRE
4      p EST_DU_TYPE NOMBRE
5      i EST_DU_TYPE NOMBRE
6      x EST_DU_TYPE NOMBRE
7      y EST_DU_TYPE NOMBRE
8      p_facto EST_DU_TYPE NOMBRE
9  DEBUT_ALGORITHME
10     AFFICHER "Entrez le nombre de chevaux partants"
11     LIRE n
12     AFFICHER "Entrez le nombre de chevaux joués"
13     LIRE p
14     x PREND_LA_VALEUR 1
15     p_facto PREND_LA_VALEUR 1
16     POUR i ALLANT_DE n-p+1 A n
17         DEBUT_POUR
18             x PREND_LA_VALEUR x*i
19         FIN_POUR
20     POUR i ALLANT_DE 1 A p
21         DEBUT_POUR
22             p_facto PREND_LA_VALEUR p_facto*i
23         FIN_POUR
24     y PREND_LA_VALEUR x/p_facto
25     AFFICHER "Dans l'ordre : 1 chance sur "
26     AFFICHER x
27     AFFICHER " de gagner."
28     AFFICHER "Dans le désordre : 1 chance sur "
29     AFFICHER y
30     AFFICHER " de gagner."
31  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez le nombre de chevaux partants
Entrer n : 10
Entrez le nombre de chevaux joués
Entrer p : 4
Dans l'ordre : 1 chance sur 5040 de gagner.
Dans le désordre : 1 chance sur 210 de gagner.
***Algorithme terminé***

```

Exercice 11 : Ecrire un algorithme qui demande l'âge d'un enfant à l'utilisateur. Ensuite, il l'informe de sa catégorie

Peut-on concevoir plusieurs algorithmes équivalents menant à ce résultat : OUI

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      age EST_DU_TYPE NOMBRE
4  DEBUT_ALGORITHME
5      AFFICHER "Entrez l'âge d'un enfant"
6      LIRE age
7      SI (age>=6 ET age<=7) ALORS
8          DEBUT_SI
9              AFFICHER "Poussin"
10             FIN_SI
11         SINON
12             DEBUT_SINON
13                 SI (age>=8 ET age<=9) ALORS
14                     DEBUT_SI
15                         AFFICHER "Pupille"
16                         FIN_SI
17                     SINON
18                         DEBUT_SINON
19                             SI (age>=10 ET age<=11) ALORS
20                                 DEBUT_SI
21                                     AFFICHER "Minime"
22                                     FIN_SI
23                                 SINON
24                                     DEBUT_SINON
25                                         SI (age>=12) ALORS
26                                             DEBUT_SI
27                                                 AFFICHER "Cadet"
28                                                 FIN_SI
29                                             SINON
30                                                 DEBUT_SINON
31                                                     AFFICHER "Non defini"
32                                                     FIN_SINON
33                                                 FIN_SINON
34                                     FIN_SINON
35                             FIN_SINON
36             FIN_SINON
37         FIN_SINON
38     FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez l'âge d'un enfant
Entrer age : 8
Pupille
***Algorithme terminé***

```

Exercice 12 : Cet algorithme est destiné à prédire l'avenir, et il doit être infaillible ! Il lira au clavier l'heure et les minutes, et il affichera l'heure qu'il sera une minute plus tard.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      heure EST_DU_TYPE NOMBRE
4      minute EST_DU_TYPE NOMBRE
5      next_min EST_DU_TYPE NOMBRE
6  DEBUT_ALGORITHME
7      AFFICHER "Entrez l'heure"
8      LIRE heure
9      AFFICHER "Entrez les minutes"
10     LIRE minute
11     next_min PREND_LA_VALEUR minute+1
12     SI (next_min>=60) ALORS
13         DEBUT_SI
14             next_min PREND_LA_VALEUR next_min-60
15             heure PREND_LA_VALEUR heure+1
16         FIN_SI
17     AFFICHER "Dans une minute, il sera "
18     AFFICHER heure
19     AFFICHER " heure(s) "
20     AFFICHER next_min
21  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez l'heure
Entrer heure : 18
Entrez les minutes
Entrer minute : 59
Dans une minute, il sera 19 heure(s) 0
***Algorithme terminé***

```

Exercice 13 : Un magasin de reprographie facture 10 F les dix premières photocopies, 9 F les vingt suivantes et 8 F au-delà. Ecrivez un algorithme qui demande à l'utilisateur le nombre de photocopies effectuées et qui affiche la facture correspondante.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      nb_copy EST_DU_TYPE NOMBRE
4      montant EST_DU_TYPE NOMBRE
5      prix_10 EST_DU_TYPE NOMBRE
6      prix_20 EST_DU_TYPE NOMBRE
7      prix_plus EST_DU_TYPE NOMBRE
8  DEBUT_ALGORITHME
9      prix_10 PREND_LA_VALEUR 10
10     prix_20 PREND_LA_VALEUR 9
11     prix_plus PREND_LA_VALEUR 8
12     AFFICHER "Entrez le nombre de photocopies"
13     LIRE nb_copy
14     SI (nb_copy<=10) ALORS
15         DEBUT_SI
16             montant PREND_LA_VALEUR nb_copy*prix_10
17         FIN_SI
18     SINON
19         DEBUT_SINON
20             SI (nb_copy>10 ET nb_copy<=30) ALORS
21                 DEBUT_SI
22                     montant PREND_LA_VALEUR 10*prix_10+(nb_copy-10)*prix_20
23                 FIN_SI
24             SINON
25                 DEBUT_SINON
26                     SI (nb_copy>30) ALORS
27                         DEBUT_SI
28                             montant PREND_LA_VALEUR 10*prix_10+20*prix_20+(nb_copy-30)*prix_plus
29                         FIN_SI
30                     FIN_SINON
31             FIN_SINON
32     AFFICHER "Montant total : "
33     AFFICHER montant
34     AFFICHER " F"
35  FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entrez le nombre de photocopies
Entrer nb_copy : 35
Montant total : 320 F
***Algorithme terminé***

```


Exercice 14 : Le programme demandera donc l'âge et le sexe du Zorglubien, et se prononcera donc ensuite sur le fait que l'habitant est impossible.

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      age EST_DU_TYPE NOMBRE
4      sexe EST_DU_TYPE CHAINE
5  DEBUT_ALGORITHME
6      AFFICHER "Entre votre age"
7      LIRE age
8      AFFICHER "Entrez votre sexe"
9      LIRE sexe
10     SI (sexe=='H' ET age>20) ALORS
11         DEBUT_SI
12             AFFICHER "imposable"
13         FIN_SI
14     SINON
15         DEBUT_SINON
16             SI (sexe=='F' ET age>=18 ET age<=35) ALORS
17                 DEBUT_SI
18                     AFFICHER "imposable"
19                 FIN_SI
20             SINON
21                 DEBUT_SINON
22                     AFFICHER "Non imposable"
23                 FIN_SINON
24             FIN_SINON
25     FIN_ALGORITHME

```

Résultats

```

***Algorithme lancé***
Entre votre age
Entrer age : 25
Entrez votre sexe
Entrer sexe : F
imposable
***Algorithme terminé***

```

Exercice 15 : Vous devez écrire un algorithme qui permette la saisie des scores de quatre candidats au premier tour. Cet algorithme traitera ensuite le candidat numéro 1 (et uniquement lui)

Code de l'algorithme

```

1 FONCTIONS_UTILISEES
2 VARIABLES
3     scores EST_DU_TYPE LISTE
4     max EST_DU_TYPE NOMBRE
5     i EST_DU_TYPE NOMBRE
6 DEBUT_ALGORITHME
7     POUR i ALLANT_DE 1 A 4
8         DEBUT_POUR
9             AFFICHER "Entrez un score"
10            LIRE scores[i]
11            SI (scores[i]>max) ALORS
12                DEBUT_SI
13                    max PREND_LA_VALEUR scores[i]
14                FIN_SI
15            FIN_POUR
16        SI (scores[1]>50) ALORS
17            DEBUT_SI
18                AFFICHER "Elu"
19            FIN_SI
20            SINON
21                DEBUT_SINON
22                    SI (scores[1]>=12.5) ALORS
23                        DEBUT_SI
24                            SI (scores[1]==max) ALORS
25                                DEBUT_SI
26                                    AFFICHER "Favorable"
27                                FIN_SI
28                                SINON
29                                    DEBUT_SINON
30                                        AFFICHER "Défavorable"
31                                    FIN_SINON
32                                FIN_SI
33                            SINON
34                                DEBUT_SINON
35                                    AFFICHER "Battu"
36                                FIN_SINON
37                            FIN_SINON
38            FIN_SINON
39        FIN_POUR
40    FIN_ALGORITHME

```

Résultats

```
***Algorithmme lancé***
Entrez un score
Entrer le terme de rang i de la liste scores : 20
Entrez un score
Entrer le terme de rang i de la liste scores : 40
Entrez un score
Entrer le terme de rang i de la liste scores : 30
Entrez un score
Entrer le terme de rang i de la liste scores : 10
Défavorable
***Algorithmme terminé***
```

Exercice 16 : Ecrire l'algorithme permettant de saisir les données nécessaires (sans contrôle de saisie) et de traiter ce problème.

Code de l'algorithme

```

1 FONCTIONS_UTILISEES
2 VARIABLES
3     age EST_DU_TYPE NOMBRE
4     duree_permis EST_DU_TYPE NOMBRE
5     nb_accidents EST_DU_TYPE NOMBRE
6     fidelite EST_DU_TYPE NOMBRE
7     point EST_DU_TYPE NOMBRE
8 DEBUT_ALGORITHME
9     AFFICHER "Entrez l'age"
10    LIRE age
11    AFFICHER "Entrez la duree du permis"
12    LIRE duree_permis
13    AFFICHER "Entrez le nombre d'accidents"
14    LIRE nb_accidents
15    AFFICHER "Entrez la fidelite"
16    LIRE fidelite
17    SI (fidelite>5) ALORS
18        DEBUT_SI
19            point PREND_LA_VALEUR 1
20        FIN_SI
21    SI ((age<25 ET duree_permis<2 ET nb_accidents==0) OU (age<25 ET duree_permis>2 ET
nb_accidents==1) OU (age>25 ET duree_permis<2 ET nb_accidents==1) OU (age>25 ET
duree_permis>2 ET nb_accidents==2)) ALORS
22        DEBUT_SI
23            point PREND_LA_VALEUR point+1
24        FIN_SI
25        SINON
26            DEBUT_SINON
27                SI ((age<25 ET duree_permis>2 ET nb_accidents==0) OU (age>25 ET duree_permis<2 ET
nb_accidents==0) OU (age>25 ET duree_permis>2 ET nb_accidents==1)) ALORS
28                    DEBUT_SI
29                        point PREND_LA_VALEUR point+2
30                    FIN_SI
31                    SINON
32                        DEBUT_SINON
33                            SI (age>25 ET duree_permis>2 ET nb_accidents==0) ALORS
34                                DEBUT_SI
35                                    point PREND_LA_VALEUR point+3
36                                FIN_SI
37                                SINON
38                                    DEBUT_SINON
39                                        AFFICHER "Refus"
40                                        FIN_SINON
41                            FIN_SINON
42                        FIN_SINON
43    AFFICHER "Le tarif est : "
44    SI (point==1) ALORS
45        DEBUT_SI
46            AFFICHER " Rouge"
47        FIN_SI
48        SINON
49            DEBUT_SINON
50                SI (point==2) ALORS
51                    DEBUT_SI
52                        AFFICHER " Orange"
53                    FIN_SI
54                    SINON
55                        DEBUT_SINON
56                            SI (point==3) ALORS
57                                DEBUT_SI
58                                    AFFICHER " Vert"

```

```
59         FIN_SI
60     SINON
61         DEBUT_SINON
62         SI (point==4) ALORS
63             DEBUT_SI
64             AFFICHER " Bleu"
65             FIN_SI
66         FIN_SINON
67     FIN_SINON
68     FIN_SINON
69     FIN_ALGORITHME
```

Résultats

```
***Algorithme lancé***
Entrez l'age
Entrer age : 31
Entrez la duree du permis
Entrer duree_permis : 7
Entrez le nombre d'accidents
Entrer nb_accidents : 1
Entrez la fidelite
Entrer fidelite : 6
Le tarif est : Vert
***Algorithme terminé***
```

Généré par AlgoBox

Exercice 17 : Toujours à partir de deux tableaux précédemment saisis, écrivez un algorithme qui calcule le schtroumpf des deux tableaux

Code de l'algorithme

```

1  FONCTIONS_UTILISEES
2  VARIABLES
3      i EST_DU_TYPE NOMBRE
4      j EST_DU_TYPE NOMBRE
5      tab1 EST_DU_TYPE LISTE
6      tab2 EST_DU_TYPE LISTE
7      taille_tab1 EST_DU_TYPE NOMBRE
8      taille_tab2 EST_DU_TYPE NOMBRE
9      val EST_DU_TYPE NOMBRE
10     result EST_DU_TYPE NOMBRE
11 DEBUT_ALGORITHME
12     val PREND_LA_VALEUR 1
13     taille_tab1 PREND_LA_VALEUR 1
14     taille_tab2 PREND_LA_VALEUR 1
15     TANT_QUE (val!=0) FAIRE
16         DEBUT_TANT_QUE
17             AFFICHER "Entrez les nombres du tableau 1 (0 pour terminer)"
18             LIRE val
19             tab1[taille_tab1] PREND_LA_VALEUR val
20             taille_tab1 PREND_LA_VALEUR taille_tab1+1
21         FIN_TANT_QUE
22     val PREND_LA_VALEUR 1
23     TANT_QUE (val!=0) FAIRE
24         DEBUT_TANT_QUE
25             AFFICHER "Entrez les nombres du tableau 2 (0 pour terminer)"
26             LIRE val
27             tab2[taille_tab2] PREND_LA_VALEUR val
28             taille_tab2 PREND_LA_VALEUR taille_tab2+1
29         FIN_TANT_QUE
30     POUR i ALLANT_DE 1 A taille_tab1-1
31         DEBUT_POUR
32             POUR j ALLANT_DE 1 A taille_tab2-1
33                 DEBUT_POUR
34                     result PREND_LA_VALEUR result + tab1[i]*tab2[j]
35                 FIN_POUR
36             FIN_POUR
37         AFFICHER "Le schtroumpf est : "
38         AFFICHER result
39 FIN_ALGORITHME

```

Résultats

```

Entrer val : 7
Entrez les nombres du tableau 1 (0 pour terminer)
Entrer val : 12
Entrez les nombres du tableau 1 (0 pour terminer)
Entrer val : 0
Entrez les nombres du tableau 2 (0 pour terminer)
Entrer val : 3
Entrez les nombres du tableau 2 (0 pour terminer)
Entrer val : 6
Entrez les nombres du tableau 2 (0 pour terminer)
Entrer val : 0
Le schtroumpf est : 279
***Algorithme terminé***

```