

Algorithmique en seconde

Question 1

/ 1

Un algorithme est :

- une suite d'actions à exécuter, de la première à la dernière.
- un programme écrit en langage codé
- un programme écrit en langage machine

Question 2

/ 1

Un algorithme se construit :

- l'ordre de construction n'a pas d'importance
- avant l'écriture d'un programme
- après l'écriture d'un programme

Question 3

/ 1

Parmi les éléments suivants, quel est celui qui permet de stocker une valeur ?

- expression
- variable
- constante

Question 4

/ 1

Parmi les éléments suivants, quel est celui qui permet de calculer ?

- expression
- constante
- variable

Question 5

/ 1

Quelle est l'écriture qui permet d'ajouter 5 à la variable *prix* ?

prix ← 5

prix ← *prix* + 5

prix ← *prix* 5

Question 6

/ 1

Quelle est le bloc qui permet de choisir une action parmi deux possibles ?

- un bloc conditionnel
- un bloc séquentiel
- un bloc répétitif

Question 7

/ 1

Après l'exécution du code suivant, quelle sera la valeur de C si A vaut 7 ?

```
Si A ≠ 7
  Alors C ← A * A
  Sinon C ← -A
Fin Si
```

- 49
- 7
- 7

Algorithmique en seconde**Question 8**

/ 1

Après l'exécution du code suivant, quelle sera la valeur de C si A vaut -5 ?

```
Si A ≠ 7
  Alors C ← A*A
  Sinon C ← -A
Fin Si
```

- 25
 -5
 5

Question 9

/ 1

Sachant qu'un traitement doit être répété 50 fois, quelle est la structure la plus appropriée ?

- Pour
 Tant que
 Si

Question 10

/ 1

Qu'affiche le script suivant après avoir été exécuté ?

```
somme ← 0
Pour i allant de 0 à 3 :
  somme ← somme + 5
FinPour
Afficher(somme)
```

- 3
 20
 5

Question 11

/ 1

Quelle sera la valeur affichée à la fin de l'exécution de l'algorithme suivant ?

```
Si A > 5 :
  Alors Si B ≠ 0 :
    Alors C ← A-B
    Sinon C ← C+A
  Fin Si
Sinon
  C ← C-B
Fin Si
```

- 7
 8
 -1

Algorithmique en seconde

Question 12

/ 1

Après l'exécution du code suivant, quel sera le montant de la réduction sachant que je suis membre et que j'ai 15 ans ?

```
Si membre="oui" Alors
  Si age ≥ 18 Alors
    reduction = 50
  Sinon
    reduction = 40
Fin Si
Sinon
  reduction = 30
Fin Si
```

- 50
 30
 40

Question 13

/ 1

Quel est le couple de valeurs qui permet d'obtenir une réduction de 20 ?

```
Si membre="oui" Alors
  Si age ≥ 18 Alors
    reduction = 50
  Sinon
    reduction = 40
Fin Si
Sinon Si age ≥ 18 Alors
  reduction = 30
Sinon
  reduction = 20
Fin Si
Fin Si
```

- membre vaut oui et age vaut 10, par exemple
 membre vaut non et age vaut 17, par exemple
 membre vaut non et age vaut 18, par exemple

Question 14

/ 1

Quelle sera la valeur affichée à la fin de l'exécution de l'algorithme suivant ?

```
cpt ← 1
Tant que cpt ≤ 4:
  cpt ← cpt+3 :
  [Traitement]
Fin tant que
```

- 4
 1
 7

Algorithmique en seconde

Question 15

/ 1

L'utilisateur a encodé les valeurs 10, 20, -5, 40 et -1. Quelle sera la valeur affichée par cet algorithme ?

```
total ← 0
Saisir nbr
Tant que nbr ≠ -1 :
    total ← total + nbr
    Saisir nbr
Fin tant que
Afficher total
```

- 75
 65
 64

Question 16

/ 1

Dans le code suivant, combien de fois le traitement sera-t-il effectué ?

```
cpt ← 1
Tant que cpt ≠ 8 :
    [Traitement]
    cpt ← cpt+2
Fin tant que
```

- à l'infini
 1
 7

Question 17

/ 1

Quelle sera la valeur affichée à la fin de l'exécution de l'algorithme suivant ?

```
cpt ← 15
Tant que cpt ≥ 10 :
    cpt ← cpt-2
Fin tant que
```

- 4
 1
 9

Question 18

/ 1

Que renvoie l'appel : "fonction1(641)" ?

```
def fonction1(s) :
    Tant que s>0 :
        s ← s-100
    Fintantque
    renvoyer s
```

- 159
 141
 41
 -59

Algorithmique en seconde**Question 19**

/ 1

Que renvoie l'appel : "fonction2(4)" ?

```
def fonction2(x) :  
  Si  $-5x+10>0$  :  
    renvoyer "image positive"  
  Sinon :  
    renvoyer "image négative"  
  FinSi
```

- 10
 image positive
 4
 image négative

Question 20

/ 1

Que renvoie l'appel : "fonction3(x)" ?

```
def fonction3(x) :  
   $x \leftarrow 500$  :  
  Pour i allant de 1 à n :  
     $x \leftarrow 0,8+50$   
  FinPour  
  renvoyer x
```

- 410
 352.4
 315.536
 331.92