

Algorithmes paramétrés (1) [ss]

Exercices de cours

Karine Zampieri, Stéphane Rivière

Unisciel  algoprogram  Version 16 mai 2018

Table des matières

1	Appréhender le cours	2
1.1	Autour de la fonction evalBool / pgevalbool	2
1.2	Bulbes du logo d'Ascq / pgparterre	3
2	Appliquer le cours	5
2.1	Autour de la fonction max2i / pgminmax2i	5
2.2	Autour de la fonction divisible / pgdivisible	7
3	Approfondir le cours	8
3.1	Autour de la fonction signe / pgsigne	8

C++ - Exercices de cours (TP)



Mots-Clés Algorithmes paramétrés ■

Difficulté ●○○ (1 h 30) ■

1 Appréhender le cours

1.1 Autour de la fonction evalBool / pgevalbool



Écrivez le **profil** d'une fonction `evalBool(b)` qui renvoie 1 si `b` est `Vrai`, 0 sinon.



Écrivez le corps de la fonction de sorte que :

```
evalBool(true) ==> 1  
evalBool(false) ==> 0
```



Écrivez un programme qui teste votre fonction.



Testez. Résultat d'exécution :

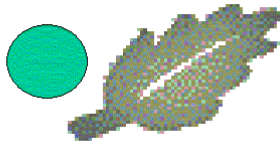
```
==> evalbool(false) est 0  
==> evalbool(true) est 1
```

1.2 Bulbes du logo d'Ascq / pgparterre



Objectif

C'est le moment de planter les bulbes. La mairie de Villeneuve d'Ascq désire en mettre dans tous les parterres de la ville où fleurit habituellement son logo constitué essentiellement d'une feuille verte et d'un cercle bleu. Seuls cette feuille et ce cercle doivent être fleuris :



Vous êtes embauché pour aider le service des espaces verts à calculer le nombre de bulbes nécessaires pour chaque parterre :

- On sait que le diamètre d du cercle doit être égal à la moitié de la largeur ℓ du parterre.
- On admet que la feuille a la même surface qu'un losange dont la grande diagonale $d1$ est égale à la moitié de la longueur L du parterre, et la petite diagonale $d2$ à la moitié de la largeur ℓ du parterre.
- Le nombre de bulbes au m^2 est de 100.

Cet exercice calcule et affiche le nombre de bulbes nécessaires pour un parterre.



Donnez les expressions de d , $d1$ et $d2$ en termes de ℓ et L .



Écrivez une fonction `surfaceC(r)` qui calcule et renvoie la surface d'un cercle de rayon r . Utilisez la formule $\pi \cdot r^2$.

Outil C++

Selon les compilateurs :

GNU C++ La constante `M_PI` de π est définie dans la bibliothèque `<cmath>`.
(inutile de la définir si vous incluez la bibliothèque)

ISO C++ La constante `M_PI` n'est pas standard.
Il faut la définir :

```
const double PI=3.141592653589793116;
```



Écrivez une fonction `surfaceL(d1,d2)` qui calcule et renvoie la surface d'un losange de diagonales $d1$ et $d2$. Utilisez la formule $\frac{1}{2} \cdot d1 \cdot d2$.



Écrivez une fonction `surface(L,lgr)` qui calcule et renvoie la surface à fleurir d'un parterre de longueur L et largeur lgr .



Écrivez une fonction `bulbes(L, lgr)` qui calcule et renvoie le nombre de bulbes nécessaires d'un parterre de longueur `L` et largeur `lgr`.



Écrivez un programme qui saisit la longueur et la largeur d'un parterre puis calcule et affiche la surface à fleurir et le nombre de bulbes nécessaires.

2 Appliquer le cours

2.1 Autour de la fonction `max2i` / `pgminmax2i`

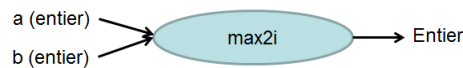


Objectif

Cet exercice détermine le plus grand de deux, trois ou quatre entiers.



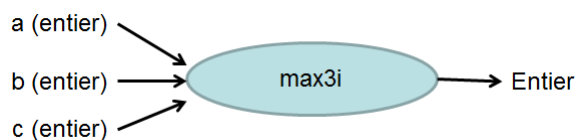
Écrivez le **profil** d'une fonction `max2i(a,b)` qui renvoie le plus grand des entiers `a` et `b`.



Écrivez le corps de la fonction.



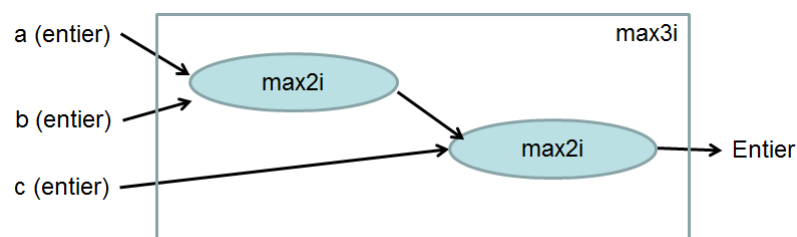
Écrivez le **profil** d'une fonction `max3i(a,b,c)` qui renvoie le plus grand des trois entiers `a`, `b` et `c`.



Écrivez le corps de la fonction en utilisant la fonction `max2i`

Aide méthodologique

Composez deux à deux.



Écrivez une fonction `max4i(a,b,c,e)` qui calcule et renvoie le plus grand des quatre entiers `a`, `b`, `c` et `e` en utilisant les fonctions `max2i` et/ou `max3i`.



Écrivez un programme qui demande quatre entiers puis calcule et affiche le plus grand des deux premiers, des deux derniers et des quatre entiers.



Testez. Exemple d'exécution :

```
Quatre entiers? 3 -2 8 4  
Max des deux premiers 3  
Max des deux derniers 8  
Max des quatre entiers 8
```



Écrivez une fonction `min2i(a,b)` qui calcule et renvoie le minimum (= le plus petit) des entiers `a` et `b`.

2.2 Autour de la fonction divisible / pgdivisible



Écrivez le **profil** d'une fonction `divisible(n,d)` qui renvoie `Vrai` si un entier `n` est divisible par un entier `d`, `Faux` sinon.



Propriété

Un entier n est **divisible** par un entier d si (et seulement si) le reste de la division entière de n par d (c.-à-d. le modulo) est nul.



Écrivez le corps de la fonction.



Propriété

Deux entiers sont **multiples** l'un l'autre si l'un est divisible par l'autre.



Écrivez une fonction `multiple(a,b)` qui renvoie `Vrai` si deux entiers `a` et `b` sont multiples l'un l'autre, `Faux` sinon.



Écrivez un programme qui saisit deux entiers.
Affichez l'invite :

```
Deux entiers?
```



Affichez les divisibilités et la multiplicité des deux entiers.



Testez. Exemple d'exécution :

```
Deux entiers? 132 33
==> divisible(132,33) est 1
==> divisible(33,132) est 0
==> multiple(132,33) est 1
```

3 Approfondir le cours

3.1 Autour de la fonction signe / pgsigne



Écrivez une fonction `signe(n)` qui renvoie -1 , 0 ou 1 selon qu'un entier `n` est strictement négatif, nul ou strictement positif. Exemples :

```
signe(5) ==> 1
signe(-8) ==> -1
signe(0) ==> 0
```



Écrivez une fonction `vabs(n)` qui renvoie la valeur absolue d'un entier `n` par appel à la fonction `signe`.



Écrivez une fonction `signeProduit(a,b)` qui calcule et renvoie le signe du produit des entiers `a` et `b` sans calculer le produit.

Aide méthodologique

Pour le produit, on a :

$$\text{signe}(a \times b) = \text{signe}(a) \times \text{signe}(b)$$

En effet, lorsque les deux nombres sont non nuls mais de même signe, la fonction `signe` renvoie le même entier -1 ou 1 pour les deux, dont le produit vaut $+1$, indiquant ainsi un produit positif. Lorsque l'un des deux nombres est nul, la fonction `signe` renvoie 0 et le produit des signes est nul indiquant que le produit des nombres est nul. Enfin, lorsque les deux nombres sont de signes contraires, on obtient un résultat égal à -1 .



De même, écrivez une fonction `signeSomme(a,b)` qui calcule et renvoie le signe de la somme des entiers `a` et `b` sans calculer la somme.

Outil C++

La fonction `abs(x)` de la valeur absolue est définie dans la bibliothèque `<cstdlib>`.

Aide méthodologique

Le signe d'une somme de deux nombres est le signe du nombre qui a la plus grande valeur absolue. Lorsque les valeurs absolues sont égales, la somme est nulle si les nombres ont des signes différents, comme -15 et 15 par exemple. Sinon, c'est le signe commun.



Écrivez un programme qui saisit deux entiers puis teste vos fonctions.



Testez. Exemple d'exécution :

```
Deux entiers? -2 6
==> signe(-2) est -1
==> signe(6) est 1
==> signe(0) est 0
==> signeProduit est -1
==> signeSomme est 1
==> vabs(-2) est 2
==> vabs(6) est 6
```