

Algorithmique et Programmation – Examen

Session spéciale du 14 septembre 2021

NOTES : Aucun document autorisé. Sont interdits les calculatrices, les téléphones, ainsi que tout autre ustensile de calcul ou de communication.

REMARQUE 1 : Une indication sur le niveau atteint est précisée après chaque exercice. Le niveau de difficulté allant croissant, il est préférable de traiter les exercices dans l'ordre. La réussite de l'exercice 1 est requise pour valider l'examen ; les exercices peuvent ensuite être traités indépendamment de leur ordre.

REMARQUE 2 : Dans la suite, les indications concernant les nombres de lignes sont données en comptant toute ligne de code non-vide (`#include`, prototypes, accolades...).

REMARQUE 3 : Les petits oublis d'un point-virgule, d'une parenthèse, d'une virgule, d'une apostrophe double ou simple, ne seront pas pénalisants tant qu'ils restent **ponctuels**. En revanche, soignez le placement des accolades.

Exercice 1 : Manipulation de rectangles

Nous souhaitons écrire des programmes réalisant des calculs sur des rectangles.

a) L'objectif de cet exercice est d'écrire trois **FONCTIONS** en **LANGAGE C** qui prennent en paramètre la largeur et la hauteur d'un rectangle, et retournent chacune le résultat d'un calcul. Les deux premières fonctions calculent le périmètre et l'aire ; la troisième fonction calcule si le rectangle est un carré (retourne une valeur vraie) ou non (retourne la valeur fausse).

Les prototypes des trois fonctions seront les suivants :

```
int perimetre (int largeur, int hauteur) ;
```

```
int aire (int largeur, int hauteur) ;
```

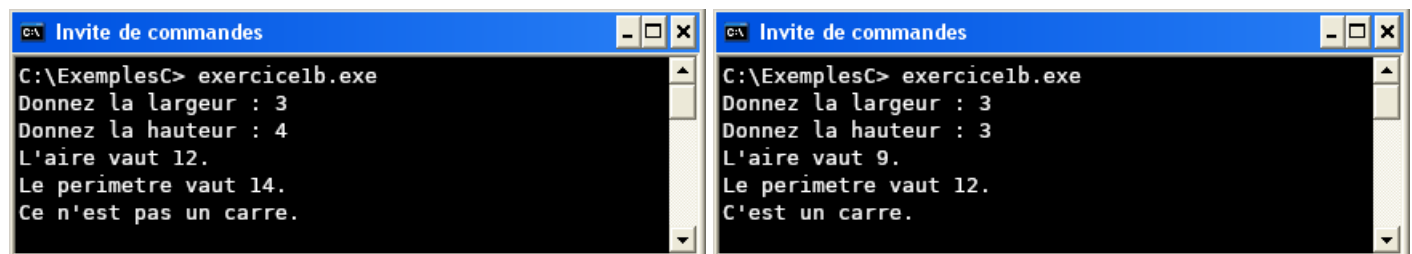
```
char carre (int largeur, int hauteur) ;
```

RAPPELS :

- Le périmètre d'un rectangle est égal à deux fois sa largeur plus deux fois sa hauteur ;
- L'aire d'un rectangle est égale à sa largeur fois sa hauteur ;
- Un rectangle est un carré lorsque sa largeur et sa hauteur sont égales.

b) Pour tester les fonctions du (a), écrire un **PROGRAMME** en **LANGAGE C** dans lequel l'utilisateur commence par indiquer la largeur et la hauteur d'un rectangle. Ensuite, le programme affiche le périmètre et l'aire du rectangle, puis indique s'il s'agit d'un carré ou non.

Ci-après deux exemples d'exécution de ce programme :



```
C:\ExemplesC> exercicelb.exe
Donnez la largeur : 3
Donnez la hauteur : 4
L'aire vaut 12.
Le perimetre vaut 14.
Ce n'est pas un carre.

C:\ExemplesC> exercicelb.exe
Donnez la largeur : 3
Donnez la hauteur : 3
L'aire vaut 9.
Le perimetre vaut 12.
C'est un carre.
```

c) Écrire un nouveau **PROGRAMME** en **LANGAGE C** dans lequel sont déclarés les deux tableaux suivants, définissant les largeurs et hauteurs de 24 rectangles :

```
int L[24] = { 9, 7, 4, 4, 2, 5, 7, 9, 9, 5, 2, 4, 8, 9, 9, 2, 6, 7, 9, 7, 3, 7, 7, 2 } ;
int H[24] = { 1, 7, 3, 1, 1, 9, 4, 2, 3, 9, 3, 1, 6, 9, 2, 7, 2, 6, 2, 1, 6, 7, 4, 9 } ;
```

Par exemple, le premier rectangle est de largeur 9 et de hauteur 1.

Le programme commencera par demander une valeur de seuil à l'utilisateur. Le programme calculera ensuite, à l'aide d'une boucle, l'aire totale des rectangles dont le périmètre dépasse le seuil, ainsi que le nombre de rectangles qui sont des carrés, puis affichera le résultat.

Ci-après quatre exemples d'exécution de ce programme :

Seuil	Aire totale des rectangles dépassant le seuil	Nombre de carrés
10	586	3
20	514	3
30	81	3
40	0	3

N.B. : Un seul parcours de tableau suffit.

Remarques :

- environ 3 lignes par fonction pour le (a)
- plus ou moins 13 lignes (environ) pour le (b)
- plus ou moins 14 lignes (environ) pour le (c)
- les programmes du (b) et du (c) doivent utiliser les fonctions du (a)

E Être capable d'écrire des fonctions et des programmes simples