

## Corrigé de l'Examen 2<sup>ème</sup> Semestre 2019-2020

Matière	Langage C avancé
Durée	1 h30
Niveau	PII
Enseignant	Youssef ALJ

### Documents autorisés

#### Consignes :

1. Les exercices sont indépendants et peuvent être traités dans l'ordre souhaité.
2. Mentionner le numéro de l'exercice ainsi que celui de la question traitée.

#### Exercice 1 (4 points):

Trouver les 8 erreurs dans le programme suivant :

```
#include <stdio.h> //1 #include à la place de #includure
int main() {
    int a;
    char b; c; //2 , qui sépare b et c au lieu de ;
    printf("bonjour,\n"); //3 " après \n
    printf("veuillez saisir deux caractères ");
    scanf("%c", &b);
    scanf("%c", c); //4 &c au lieu de c
    //5 pas de & pour afficher b
    printf("le caractère saisi est %c\n", &b);
    printf("saisir une valeur entière: ");
    scanf("%d", &a);
    printf("vous avez saisi \n", a); //6 oubli de %d
    printf("au revoir.\n"); //7 guillemets après \n
    return 0;
} //8 } au lieu de )
```

#### Exercice 2 (9 points) :

On demande à une classe d'écrire un programme qui calcule la somme des nombres pairs de 0 jusqu'à un nombre  $n$  défini par l'utilisateur. Deux étudiants Katy et Ahmed proposent leur programme (voir programmes ci-dessous), mais ils n'arrivent pas à remplir certaines parties laissées en pointillés ....

On se propose dans les questions qui suivent d'aider les deux étudiants.

#### **Partie A : aidons Ahmed (voir programme d'Ahmed ci-dessous):**

Ahmed propose de répondre au problème posé en parcourant tous les entiers de 0 jusqu'au nombre  $n$  défini par l'utilisateur et tester à chaque fois si cet entier est pair.

1. Pouvez-vous aider Ahmed à remplir les lignes 11 et 12 de son programme? **Cf programme d'Ahmed**
2. Dans la ligne 13 de son programme, Ahmed souhaite calculer la somme demandée en mettant à jour la variable `somme` déclarée dans la ligne 5. Pouvez-vous l'aider à remplir la ligne 13 de son programme? **Cf programme d'Ahmed**

## Partie B : aidons Katy (voir programme de Katy ci-dessous):

Katy, quant à elle, propose d'organiser son programme sous forme de fonctions. Elle déclare une fonction `somme_pairs` qui prend comme paramètre un entier `n` qui représente le nombre défini par l'utilisateur.

De plus, elle prétend qu'elle n'a pas besoin d'un test `if` à l'intérieur de la boucle `for` comme suggéré par Ahmed.

3. Pouvez-vous aider Katy en remplissant la définition de la fonction `somme_pairs` ci-dessous? Cf programme de Katy
4. Comment appeler la fonction `somme_pairs` dans la ligne 12 pour que le programme de Katy fonctionne correctement? Cf programme de Katy

Programme d'Ahmed	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18	<pre>#include &lt;stdio.h&gt;  int main() {     int i, n;     int somme=0;     printf("veuillez saisir un entier\n");     scanf("%d", &amp;n);     // on parcourt les nombres de 0 jusqu'a n     // si on tombe sur un nombre pair     // on le rajoute a la somme     for (i=0; i&lt;n; i++){         if (i % 2== 0){             somme = somme + i;         }     }     printf("somme des pairs jusqu'à %d est %d", n, somme);     return 0; }</pre>

Programme de Katy	
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24	<pre>#include &lt;stdio.h&gt;  // delcaration de la fonction int somme_pairs(int n);  int main() {     int n;     printf("veuillez saisir un entier\n");     scanf("%d", &amp;n);     int somme;     // appel de la fonction     somme = somme_pairs(n);     printf("somme des pairs jusqu'à %d est %d", n, somme);     return 0; }  //definition de la fonction int somme_pairs(int n){     int i, somme = 0;     for (i=0; i&lt;n;i=i+2){         somme = somme + i;     }     return somme; }</pre>

### Partie C : analyse des deux programmes :

5. Concernant le choix de Katy d'utiliser une fonction, pensez-vous que ce choix est meilleur que celui d'Ahmed? Pourquoi? **Le choix de Katy est meilleur car les fonctions permettent la réutilisabilité des programmes.**
6. Concernant le choix de Katy de ne pas utiliser un test `if` dans la boucle `for`, pensez-vous que ce choix est meilleur que celui d'Ahmed? Pourquoi? **Le choix de Katy est meilleur car permet d'éviter le test `if` qui peut ralentir le programme.**

### Exercice 3 (7 points + 2 points bonus):

Ecrire un programme en C qui fait les opérations suivantes :

1. déclare trois variables `a`, `b` et `c` de type entier. (cf programme 3 ci-dessous)
2. demande à l'utilisateur de saisir les valeurs de `a` et de `b`. (cf programme 3 ci-dessous)
3. lit ce que l'utilisateur a saisi au clavier pour les valeurs `a` et `b`. (cf programme 3 ci-dessous)
4. crée deux pointeurs `ptr_a` et `ptr_b` qui pointent sur `a` et `b` respectivement (cf programme 3 ci-dessous).

#### Programme 3

```
#include <stdio.h>

int main() {
    int a, b, c;
    printf("veuillez saisir a et b\n");
    scanf("%d", &a);
    scanf("%d", &b);
    int *ptr_a = &a;
    int *ptr_b = &b;
    c = *ptr_a;
    *ptr_a = *ptr_b;
    *ptr_b = c;
    printf("les valeurs de a et b sont %d %d", *ptr_a, *ptr_b);
}
```

On rajoute après les opérations précédentes les lignes suivantes :

```
c = *ptr_a;
*ptr_a = *ptr_b;
*ptr_b = c;
printf("les valeurs de a et b sont %d %d", *ptr_a, *ptr_b);
```

On suppose que le programme compile correctement, et à l'exécution l'utilisateur a saisi la valeur de 1 pour `a`, et la valeur de 2 pour `b`.

5. Quel résultat voit-on à l'écran à la fin de l'exécution de ce programme?  
**Les valeurs de a et b sont 2 1**
6. En une phrase, décrire ce que fait ce programme.  
**Ce programme échange les valeurs de a et b.**

**Question bonus (à traiter une fois toutes les questions de l'examen ont été traitées) :**  
Ré-organiser le programme précédent sous forme de fonctions tout en respectant le comportement voulu à la question 6.

**Question bonus**

```
#include <stdio.h>
void echange(int *p_a, int *p_b){
    int c;
    c = *p_a;
    *p_a = *p_b;
    *p_b = c;
}
int main(){
    int a, b;
    printf("veuillez saisir a et b\n");
    scanf("%d", &a);
    scanf("%d", &b);
    int *ptr_a = &a;
    int *ptr_b = &b;
    echange(ptr_a, ptr_b);
    printf("les valeurs de a et b sont %d %d", *ptr_a, *ptr_b);
}
```