**Compte rendu TP3 - Rayane Ben Chaabane**

***Exercice 1 :***

|  |  |
| --- | --- |
| **&a** | Adresse de la variable a |
| **\*pa** | Valeur de l’adresse qu’elle pointe = a = 1 |
| **&pa** | Adresse de la variable pa |
| **\*ppa** | Valeur de l’adresse qu’elle pointe = pa = &a |
| **\*\*ppa** | Valeur de l’adresse qu’elle pointe = a = 1 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **a** | **pa** | **ppa** |
| **&a** |  | *=* |  |
| **\*pa** | *=* |  |  |
| **&Pa** |  |  | *=* |
| **\*ppa** |  | *=* |  |
| **\*\*ppa** | *=* |  |  |

*\*= : égal*

***Exercice 2 :***

1. La fonction ***echange1*** réalise un échange des valeurs de variable tandis que la fonction *echange2* fait un échange des valeurs par les adresses en utilisant les pointeurs.
2. La fonction ***echange1*** ne remplit pas son objectif

|  |  |  |
| --- | --- | --- |
|  | **u** | **v** |
| **Echange1** | 3 | 4 |
| **Echange2** | 4 | 3 |

1. Pour la **fonction *echange1***, on utilise la méthode de passage des paramètres par valeur, les variables qui sont en paramètre sont copiées dans des variables locales à la fonction et ne modifierons pas les variables utilisées en paramètres.

Pour la **fonction *echange2***, on utilise la méthode par adresse cette fois ci. On utilise plus de copie de variable en locale cette fois-ci on manipule directement les variables.

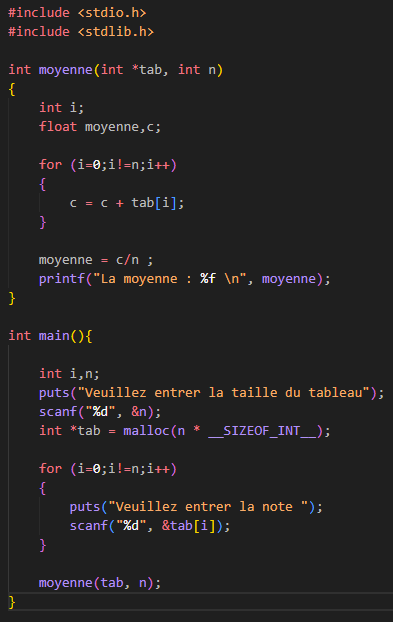
Concernant la méthode par adresse, celle-ci est plus économe et rapide en mémoire que l’autre méthode car nous ne créons pas de variable en plus. Cependant, la méthode par valeurs peut être un avantage quand on ne veut pas perdre la valeur de la variable sans faire exprès.

***Exercice 3 :***

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **A** | **B** | **C** | **P1** | **P2** |
| **Init.** | 1 | 2 | 3 | / | / |
| **P1=&A** | 1 | 2 | 3 | &A | / |
| **P2=&C** | 1 | 2 | 3 | &A | &C |
| **\*P1=(\*P2)++** | 3 | 2 | 4 | &C | &C |
| **P1=P2** | 3 | 2 | 4 | &C | &B |
| **P2=&B** | 3 | 2 | 4 | &C | &B |
| **\*P1-=\*P2** | 3 | 2 | 2 | &C | &B |
| **++\*P2** | 3 | 3 | 2 | &C | &B |
| **\*P1\*=\*P2** | 3 | 3 | 6 | &C | &B |
| **A=++\*P2\*\*P1** | 24 | 4 | 6 | &C | &B |
| **P1=&A** | 24 | 4 | 6 | &A | &B |
| **\*P2=\*P1/=\*P2** | 6 | 6 | 6 | &A | &B |

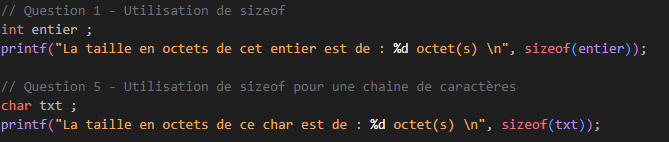
**Exercice 4 :**

On utilise deux paramètres qu’on fait rentrer en paramètre le tableau en pointeur, il nous faut sa taille afin de pouvoir l’exploiter correctement dans la boucle et effectuer le calcul.



**Exercice 5 :**

*Quelques questions ont été répondues sur le code*

**

**

*2. pi est égal à 547354876 alors que i = 547354872*

6.



Il affiche ***Bleu*** cela donc affiche la couleur qui correspond au code hexa contenu dans la variable tab

7.



Il affiche ***Bleuet*** cela donc affiche la couleur qui correspond au code hexa contenu dans la variable pc