

Examen Aout 2015

Prénom : Nom :

Formation : ☐ Électronique ☐ Télécom

Vous avez exactement **2h30** pour répondre à toutes les questions de cet examen. Vous n'avez le droit à rien si ce n'est de quoi écrire. N'oubliez pas d'écrire vos nom et prénom de manière lisible sur la première page.

Bonne chance !

1 Extraits de code (3 points)

Pour chacun des extraits de code suivants, vous devez indiquer ce que son exécution affiche à l'écran.

```

int f = 1;
while (f * -2 > -10)
{
    f += 1;
    printf("%d", f);
}
  
```

Q1(a) : _____

```

int s;
for (s = 16; s / 2 >= 1; s /= 2)
{
    printf("%d;", s);
}
  
```

Q1(b) : _____

```

int tab[] = {1, 3, 5, 7, 11, 13};
printf ("Valeur : %d", tab[(tab[3] - tab[1]) / 2]);
  
```

Q1(c) : _____

2 Somme spéciale (5 points)

Écrivez une fonction qui commence par calculer la somme de tous les éléments d'un tableau "tab" et stocker cette somme dans une variable "sum". Ensuite, calculez et renvoyez la somme des carrés de la différence entre chaque valeur et "sum".

Par exemple, pour le tableau $[1, -5, 2, -1, 0]$, le programme calcule d'abord la somme de ses valeurs $(1 - 5 + 2 - 1 + 0 = -3)$. Ensuite, on calcule la somme des carrés des différences : $(1 - (-3))^2 + (-5 - (-3))^2 + (2 - (-3))^2 + (-1 - (-3))^2 + (0 - (-3))^2 = 58$. La fonction doit donc renvoyer 58 dans ce cas.

```
float sum (int *tab, int N)
{
```

```
}
```

3 Représentation des données et pointeurs (6 points)

1. Donnez la représentation en bit de signe sur 7 bits des nombres décimaux -15 et 131 .

2. Calculez, en passant par la représentation en complément à deux sur 8 bits, $5 - 10$ (donnez les détails du calcul).

3. Qu'affiche le code suivant après exécution ?

```

int data[] = {1, 2, 3, 4, 5};

int t = 0;
int *e = data;
while (e <= data + 5)
{
    t += *e;
    e += 2;
}

printf ("Somme : %d", t);

```

Q3(c) : _____

4 Mémoire et pointeurs (5 points)

Soit la variable `char **data` et la situation suivante en mémoire (pour rappel, un `char` occupe 1 octet en mémoire et un pointeur occupe 8 octets en mémoire).

| | |
|-------------|------|
| | ... |
| data : 2000 | 4000 |
| | ... |
| 4000 | 9000 |
| | 7000 |
| | ... |
| 7000 | 'D' |
| | '%' |
| | '3' |
| | ... |
| 9000 | '+' |
| | 'A' |
| | ... |

Quelles sont les valeurs des expressions suivantes :

1. `data`
2. `*data`
3. `&data`
4. `data[2]`
5. `data + 1`
6. `*(data + 1)`
7. `data[1][1]`
8. `&(data[0][2])`
9. `*(data + 1) + 2`
10. `*(*(data + 1) + 2)`

5 Chambres d'hôtel (11 points)

Soit la structure suivante représentant une chambre d'hôtel dont on donne le numéro et le nombre de lits :

```

struct room {
    int number;      // Numéro de la chambre
    int beds;        // Nombre de lits
};
  
```

Écrivez une procédure qui reçoit une chambre en paramètre et qui permet de l'afficher. Par exemple, la procédure doit afficher :

Chambre #10 (3 lits)

```

void printRoom (struct room r)
{

}
  
```

Écrivez une fonction qui construit une nouvelle chambre à partir des deux paramètres `n` et `b` représentant le numéro de la chambre et le nombre de lits.

```

struct room* createRoom (int n, int b)
{

}
  
```

Enfin, complétez la fonction `main` suivante, de manière à ce qu'elle affiche la chambre qui est créée et stockée dans la variable `c`.

```
int main()
{
    struct room *c = createRoom (10, 13);

    return 0;
}
```