

DS: Programmation C

1 Décembre 2022 - 08h30/10h00

Nom: Prénom:

Documents autorisés : une feuille A4 recto/verso manuscrite.

Question 1 Considérez le programme suivant :

```
#include <stdio.h>
int main()
{
    printf("%d", printf("%d", printf("%d", printf("%s", "Welcome to geeks"))));
    return (0);
}
```

Parmi les propositions suivantes laquelle correspond à ce que le programme va afficher ?

- Welcome to geeks2131
- Welcome to geeks1621
- Welcome to geeks2124
- Welcome to geeks1622

Question 2 Considérez le programme suivant :

```
int main()
{
    unsigned int x = 3;
    while (x-- >= 0) {
        printf("%d ", x);
    }
    return 0;
}
```

Quel est le résultat associé à ce programme ?

- 3 2 1 0
- 2 1 0 -1
- Boucle infinie
- 65535

Question 3 Considérez le programme suivant :

```
#include <stdio.h>
void main()
{
    printf("value is = %d", (10 ++));
    return 0;
}
```

Quelle est le résultat de ce programme ?

- 10
- 11
- erreur à la compilation
- erreur à l'exécution

Question 4

```
int main()
{
    int y = 10;
    if (y++ > 9 && y++ != 11 && y++ > 11)
        printf("%d", y);
    else
        printf("%d", y);
    return 0;
}
```

- 11
- 12
- 13
- 14

Question 5 Considérez la fonction suivante :

```
#include <stdio.h>
int main()
{
    char a[] = { 'A', 'B', 'C', 'D' };
    char* ppp = &a[0];
    *ppp++; // Line 1
    printf("%c %c ", *++ppp, --*ppp); // Line 2
}
```

Parmi les propositions suivantes laquelle est correcte ?

- C B
- B A
- B C
- C A

Question 6 Considérez la fonction suivante :

```
#include <stdio.h>
void mystery(int arr[], int len) {
    int *x = arr-1;
    int *y = arr + len;
    while (x++ < y--) {
        int t = *x;
        *x = *y;
        *y = t;
    }
}
```

Expliquez ce que fait cette fonction en précisant la valeur des arguments.

Question 7 Considérez le programme suivant :

```
#include <stdio.h>

int main()
{
    char *p;
    p= "Hello";

    printf("%c\n", *&*p);
}
```

- Hello
- H
- Une adresse va être imprimée.
- Aucun des cas précédents.

Question 8 Considérez les lignes de code suivantes :

```
struct car
{ int speed;
  car type[10];} vehicle;
struct car *ptr;
ptr = &vehicle;
```

Quelle instruction permet d'affecter la valeur 200 au vehicle?

- (*ptr).speed = 200
- (*ptr) ->speed = 200
- *ptr .speed = 200
- &ptr .speed = 200

Question 9 Considérez le programme suivant :

```
#include <stdio.h>
void main(){
static int var = 5;
printf("%d", var--);
if (var)
    main();
}
```

Que produit ce programme ?

- Une erreur de compilation
- 55555
- 54321
- Une boucle infinie.

Question 10 Considérez le programme suivant :

```
#include <stdio.h>

void main(){
    int a[]={1,2,3,4,5},*p;
    p=a;
    ++*p;
    printf("%d ",*p);
    p += 2;
    printf("%d ",*p)
}
```

que produit l'exécution de ce programme ?

- 2 4
- 3 4
- 2 2
- 2 3
- 3 3

Question 11 Que produit le programme suivant ?

```
#include <stdio.h>
void solve() {
    int first = 10, second = 20;
    int third = first + second;
    {
        int third = second - first;
        printf("%d ", third);
    }
    printf("%d", third);
}
void main() {
    solve();
}
```

- 30 10
- 10 20
- 10 30
- 20 10

Question 12 Quelles expressions parmi les suivantes calculent la taille d'un tableau ?

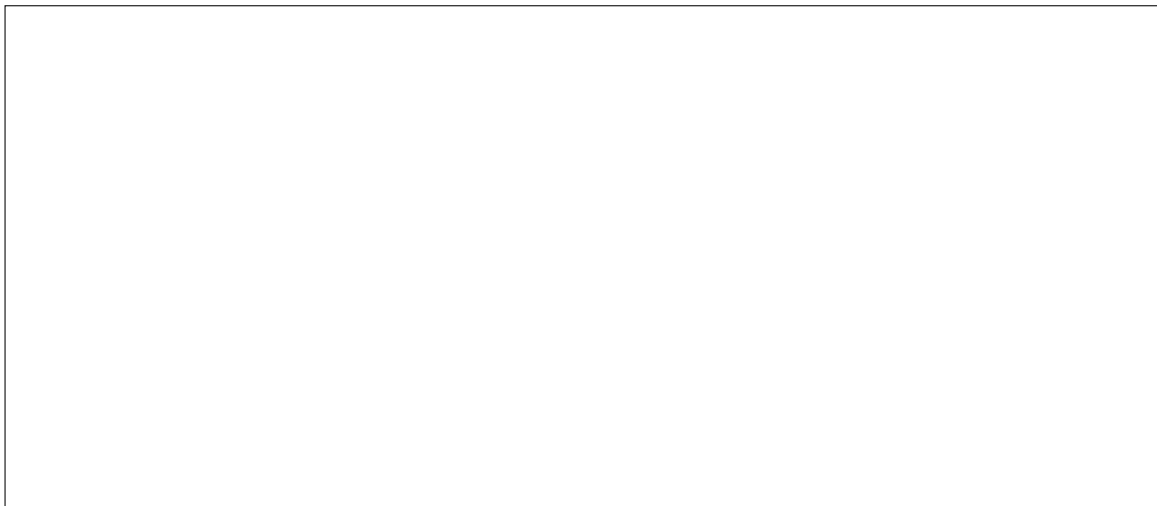
- `sizeof(a[0])`
- `sizeof(a)/sizeof(a[0])`
- `sizeof(a)*sizeof(a[0])`
- `sizeof(a)`

Question 13 Que va imprimer le programme suivant ?

```
#include <strings.h>
#include <stdio.h>
int main()
{
    char str1 []="JAMES, ";
    char str2 [15]="BOND ";
    strcat(str2, str1);
    printf("%s", str2);
    printf("%s", str1);
}
```

- JAMES, BOND JAMES,
- JAMES, JAMES,
- BOND BOND JAMES,
- BOND JAMES, JAMES,

Question 14 Ecrire une fonction clone qui étant donnés deux arguments, un tableau d'entier et sa taille, agrandit cette taille en la multipliant par deux, et recopie l'ensemble des données originales du tableau dans la deuxième partie du tableau. Par exemple si le tableau t contient 1,2,3,4,5 après l'appel de la fonction, le tableau t contiendra 1,2,3,4,5,1,2,3,4,5.



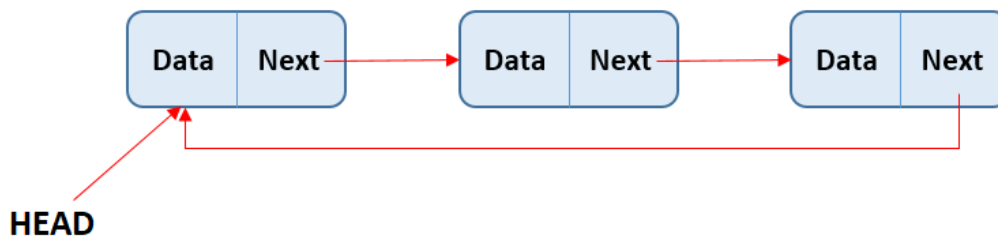
1 Problème : listes chaînées circulaires.

On considère une structure de liste chaînée circulaire. Pour représenter de telles listes des cellules contenant un entier et un pointeur vers la cellule suivante. La liste est assimilée à un pointeur sur la cellule de tête (qui est aussi la cellule de fin). Un pointeur vers NULL indique que la liste est vide.

L'implantation du type `list` permettant d'implanter une telle structure est la suivante :

```
struct circ_list_{
    int data;
    circ_list_ *next;};
typedef struct circ_list_ circ_list;
```

Une liste circulaire peut être représentée graphiquement de la manière suivante :



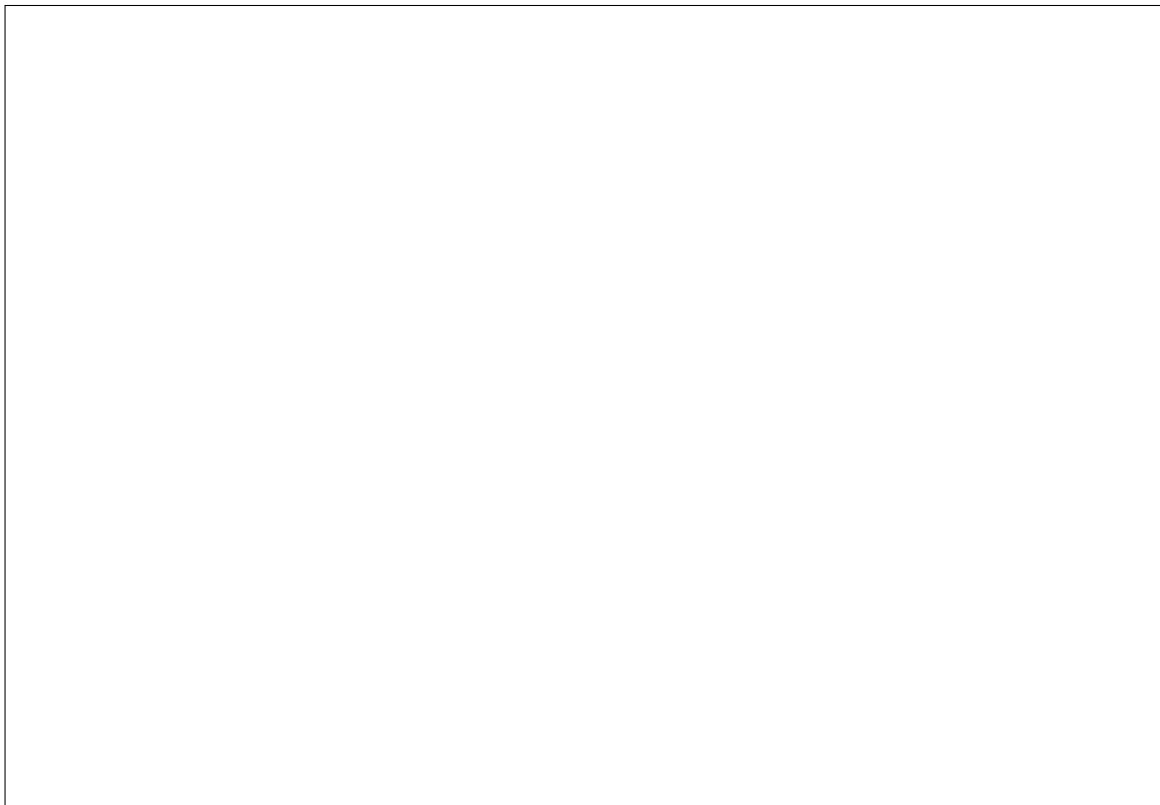
Question 15 Etant donnée une variable ptr de type `circ_list *`, donnez la suite d'instructions pour implanter une liste circulaire dont l'impression donne 132,45,-43.

Question 16 Donnez le code d'une fonction affiche qui affiche tous les éléments d'une liste circulaire dans l'ordre en partant du premier élément pointé.

Question 17 Ecrire une fonction concatenation qui étant donnée deux listes circulaires pointées par lc1 ,lc2 construit une liste contenant d'abord tous les éléments de lc1 suivis de ceux de lc2.



Question 18 Ecrire une fonction inverse qui étant donnée une liste pointée par lc1 inverse la liste. C'est à dire que l'impression de l'inversion d'une liste contenant 1,2,3,4 affichera 4,3,2,1.



Question 19 Utilisez le cadre suivant pour insérer toutes les indications utiles que vous n'avez pas eu la place de mettre avant. Notez bien le numéro de la question suivi du commentaire que vous jugez bon d'ajouter.

