



Examen final

Aucun document n'est autorisé
Les solutions doivent être rédigées en C
Note CC= Max(Exo1, Exo2) multiplié par deux
Note EF = Exo1 + Exo2

1 Affichage

10 pts. ⌚40'

1. Montrer l'historique d'exécution ainsi que l'affichage des deux programmes suivants : (8 pts)

```
1 int main(){
2   int T[8]={1,2,3,4,5,6,7};
3   int i, *p = T, *q = &T[7];
4   int A[4][3]={{1,2},{3,4},{1}};
5   printf("%d, %d\n", T[0], T[7]);
6   while(p < q){
7     i = *q;
8     *q = *p;
9     *p = i;
10    p++;
11    q--;
12  }
13  for (i=0; i<8 ;i++)
14    printf("%d ", T[i]);
15  printf("\n p-q = %d\n", p-q );
16  printf("%d, %d\n", A[2][1], A[3][0]);
17  printf("%d, %d\n", **A, *(A+1));
18  printf("%d, %d\n", *((A+1)+1),*(A+1));
19 }
```

```
1 void Toto(char str[], char c){
2   int i, j;
3   int len = strlen(str);
4   for(i=0; i<len; i++)
5     if(str[i] == c){
6       for(j=i; j<len; j++)
7         str[j] = str[j+1];
8       len--;
9       i--;
10    }
11 }
12 int main(){
13  char str[100]="Toto&Loulou", *str2;
14  str2 = strchr(str, 'o');
15  Toto(str, 'o');
16  printf("str = %s\n", str);
17  Toto(str, 'u');
18  printf("str = %s", str2);
19 }
```

2. Expliquer brièvement le rôle de la boucle while et la fonction Toto (2 pts).

2 Rotation circulaire à droite

10 pts. ⌚50'

On dispose d'un tableau T de n entiers, une rotation circulaire à droite consiste à décaler la place des éléments du tableau T en considérant que les deux extrémités du tableau (T[0] et T[n-1]) sont attachées. Un décalage est effectué avec un 'pas' correspondant au nombre de cases vers la droite dont chaque élément est décalé. Ainsi, tout élément dépassant la fin du tableau est réintroduit par le début du tableau. Le décalage se déroule de la gauche vers la droite.

Exemple : soit le tableau T suivant :

T :

7	2	8	1	4	9	3	5
---	---	---	---	---	---	---	---

si on applique une rotation circulaire de pas = 3, on obtient le tableau R suivant :

R :

9	3	5	7	2	8	1	4
---	---	---	---	---	---	---	---

1. Écrire une fonction Rotation_Droite_1 permettant de remplir un tableau R par une rotation circulaire à droite d'un tableau T passé en paramètre. (Pensez à ajouter les paramètres nécessaires) (3 pts).
2. Écrire une fonction Rotation_Droite_2 permettant d'effectuer une rotation circulaire à droite d'un tableau T passé comme paramètre. Le résultat de cette rotation sera dans le même tableau T. (Ne pas utiliser un tableau intermédiaire, et pensez à ajouter les paramètres nécessaires) (4 pts).
3. Écrire la fonction main qui appelle les deux fonctions ci-dessus pour effectuer la rotation circulaire d'un tableau déclaré et initialisé dans la fonction main. Le pas (nombre de case à décaler) est à demander à l'utilisateur (3pts).

« Bon courage »



Correction de l'examen final

Aucun document n'est autorisé
Les solutions doivent être rédigées en C
Note CC= Max(Exo1, Exo2) multiplié par deux
Note EF = Exo1 + Exo2

1 Affichage

10 pts. ⌚40'

1. Montrer l'historique d'exécution ainsi que l'affichage des deux programmes suivants

```
1 int main(){
2   int T[8]={1,2,3,4,5,6,7};
3   int i, *p = T , *q = &T[7];
4   int A[4][3]={{1,2},{3,4}, {1}};
5   printf("%d, %d\n", T[0], T[7]);
6   while(p < q){
7     i = *q;
8     *q = *p;
9     *p = i;
10    p++;
11    q--;
12  }
13  for (i=0; i<8 ;i++)
14    printf("%d ", T[i]);
15  printf("\n p-q = %d\n", p-q );
16  printf("%d, %d\n", A[2][1], A[3][0]);
17  printf("%d, %d\n", **A, *(A+1));
18  printf("%d, %d\n", *(*A+1)+1,*(A+1));
19 }
```

```
1 void Toto(char str[], char c){
2   int i, j;
3   int len = strlen(str);
4   for(i=0; i<len; i++)
5     if(str[i] == c){
6       for(j=i; j<len; j++)
7         str[j] = str[j+1];
8       len--;
9       i--;
10    }
11 }
12 int main(){
13   char str[100]="Toto&Loulou", *str2;
14   str2 = strchr(str, 'o');
15   Toto(str, 'o');
16   printf("str = %s\n", str);
17   Toto(str, 'u');
18   printf("str = %s", str2);
19 }
```

2. Expliquer brièvement le rôle de la boucle while et la fonction Toto.

Solution

Affichage

```
1,0
0 7 6 5 4 3 2 1
p-q = 1
0,0
1,3
4,2
```

Affichage

```
str = Tt&Lulu
str = t&Ll
```

- La boucle while permet d'inverser les éléments du tableau T.
- La fonction Toto supprime toutes les occurrences du caractère c de la chaîne de caractères str.

2 Rotation circulaire à droite

10 pts. ⌚50'

On dispose d'un tableau T de n entiers, une rotation circulaire à droite consiste à décaler la place des éléments du tableau T en considérant que les deux extrémités du tableau (T[0] et T[n-1]) sont attachées. Un décalage est effectué avec un 'pas' correspondant au nombre de cases vers la droite dont chaque élément est décalé. Ainsi, tout élément dépassant la fin du tableau est réintroduit par le début du tableau. Le décalage se déroule de la gauche vers la droite.

Exemple : soit le tableau T suivant :

T : 7 2 8 1 4 9 3 5

si on applique une rotation circulaire de pas = 3, on obtient le tableau R suivant :

R :

9	3	5	7	2	8	1	4
---	---	---	---	---	---	---	---

- Écrire une fonction `Rotation_Droite_1` permettant de remplir un tableau R par une rotation circulaire à droite d'un tableau T passé comme paramètre. (Pensez à ajouter les paramètres nécessaires) (3 pts).
- Écrire une fonction `Rotation_Droite_2` permettant d'effectuer une rotation circulaire à droite d'un tableau T passé comme paramètre. Le résultat de cette rotation sera dans le même tableau T. (Ne pas utiliser un tableau intermédiaire, et pensez à ajouter les paramètres nécessaires) (4 pts).
- Écrire la fonction `main` qui appelle les deux fonctions ci-dessus pour effectuer la rotation circulaire d'un tableau déclaré et initialisé dans la fonction `main`. Le pas (nombre de case à décaler) est à demander à l'utilisateur (3pts).

Solution

```
1 #include <stdio.h>
2 void rotation_droite_1(int T[], int t, int R[], int pas){
3     int i;
4     for(i=0; i<t; i++)
5         R[(i+pas)%t] = T[i];
6 }
7
8 void rotation_droite_2(int T[], int t, int pas){
9     int i;
10    for(i=0; i<pas; i++)
11        Rotation(T, t);
12 }
13
14 void Rotation(int T[], int t){
15    int i, dernier;
16    dernier = T[t-1];
17    for(i=t-1; i>0; i--)
18        T[i] = T[i - 1];
19    T[0] = dernier;
20 }
21
22 int main(){
23    int i, pas;
24    int T[10] = {0,1,2,3,4,5,6,7,8,9};
25    int R[10];
26    printf("Donner le pas de rotation : ");
27    scanf("%d", &pas);
28    pas = pas % 10;
29    printf("\n");
30    printTab(T, 10);
31    rotation_droite_1(T, 10, R, pas);
32    printf("\n");
33    printTab(R, 10);
34    rotation_droite_2(T, 10, pas);
35    printf("\n");
36    printTab(T, 10);
37 }
38
39 void printTab(int T[], int t){
40    int i;
41    for(i=0; i<t; i++)
42        printf("%d ", T[i]);
43 }
```