



## Examen de rattrapage

Aucun document n'est autorisé  
Les solutions doivent être rédigées en C  
Les appareils portables doivent être éteints

### 1 Affichage

10 pts. ☹️50'

Qu'affichent les deux programmes suivants :

```
1 #include <stdio.h>
2
3 int main(){
4     int A[4] = {4, 3, 2, 1};
5     int i, *p = A, *q = A;
6     for (i=0; i<4; i++){
7         printf("%d ", *A);
8         p++;
9     }
10    printf("\n");
11    for (i=0; i<4; i++){
12        printf("%d ", *q);
13        q++;
14    }
15    printf("\n");
16    for (i=0; i<4; i++){
17        p--;
18        printf("%d ", *p);
19    }
20 }
```

```
1 #include <stdio.h>
2 void Toto(char *S, char c, int *P, int *n)
3 {
4     int i=0;
5     *n=0;
6     while(S[i]!='\0')
7     {
8         if(S[i] == c)
9             *(P + (*n)++) = i;
10        i++;
11    }
12 }
13 int main(){
14     char S[100]="Toto&Lolo";
15     char c='o';
16     int i, n, T[10];
17     Toto(S, c, T, &n);
18     for (i = 0; i < n; i++)
19         printf("%d ", *(T + i));
20 }
```

### 2 Intersection de deux tableaux

10 pts. ☹️40'

On dispose de deux tableaux T et S d'entier à une seule dimension. Les tableaux T et S sont supposés déjà triés dans l'ordre croissant ↗ et sans doublons. La taille du tableau T est t et celle du tableau S est s. Les variables t et s doivent être inférieures ou égales à la taille maximale des tableaux T et S (fixée ici à 100).

Écrire une fonction Tab\_Intersection qui prend en entrée deux tableaux T et S ainsi que leurs tailles réelles t et s puis elle construit un tableau I de taille i qui contient l'**intersection** des deux tableaux T et S. Le tableau I doit rester trié par construction.

Exemple :

T: 

3	8	11	17	23	48	56	61	87	93	98
---	---	----	----	----	----	----	----	----	----	----

 t = 11.

S: 

4	6	8	23	53	56	76	87	90
---	---	---	----	----	----	----	----	----

 s = 9.

I: 

8	23	56	87
---	----	----	----

 i = 4.

« Bon courage »