

TP1 SR01: programmation C

A remettre le : 01 novembre 2018

- Ce TP peut se faire en binôme. Les deux étudiants formant un binôme doivent appartenir au même groupe de TD.
- Les différents binômes devront remettre un seul fichier ***.zip** contenant les codes sources ainsi qu'un rapport électronique (pdf) de quelques pages.
- Le dernier délai pour la remise des TP est fixé pour le **1 novembre 2018**.

1 Exercice 1

L'objectif de cet exercice est de collecter un ensemble de données relatives aux notes des étudiants dans les différentes UV et de les manipuler par la suite. Les notes de chaque UV sont stockées dans un fichier (SR01.txt par exemple) selon le format suivant :

La première ligne de chaque fichier contient le nombre des étudiants inscrits dans l'UV en question. Les lignes qui suivent contiennent des informations sur l'étudiant (identifiant unique, nom, prénom) ainsi que sa note finale.

Exemple de fichiers:

Nb_etudiant=30			
1	Jack	jean	10
2	Marc	douglas	5
5	Emma	watson	15
.	.	.	.
.	.	.	.

Figure 1: SR01.txt

Nb_etudiant=30			
2	Marc	douglas	8
5	Emma	watson	15
1	Jack	jean	5
.	.	.	.
.	.	.	.

Figure 2: LO21.txt

Dans ce programme vous êtes sensés :

- Déclarer une structure Etudiant conenant les champs suivants: identifiant unique, nom, prénom et moyenne.
- A partir des données stockées dans le fichier, récupérer l'identifiant, le nom et le prénom de chaque étudiant puis calculer sa moyenne.
- Produire un fichier en sortie (Fichier_final.txt) qui contient les moyennes des étudiants ordonnées par ordre décroissant.

Remarque : on suppose que tous les étudiants sont inscrits dans les mêmes UV. Par contre, l'ordre des étudiants dans les fichiers correspondant à chaque UV n'est pas forcément le même. vous pouvez utiliser les fichiers (SR01.txt, LO21.txt, ...) disponibles sur moodle pour réaliser l'exercice.

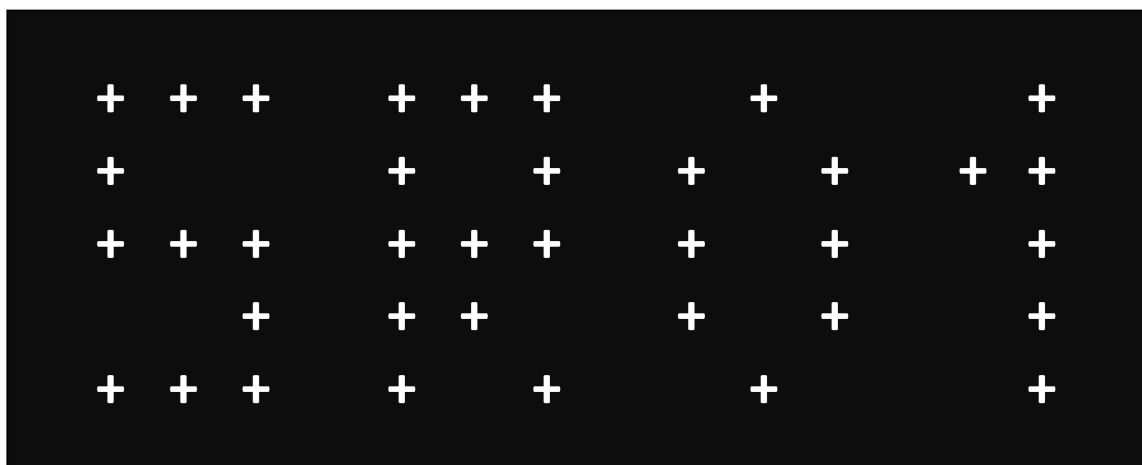
Nb_etudiant=30			
1	Emma	watson	15
2	Marc	douglas	14
.	.	.	.
.	.	.	.
.	.	.	.

Figure 3: Fichier_final.txt

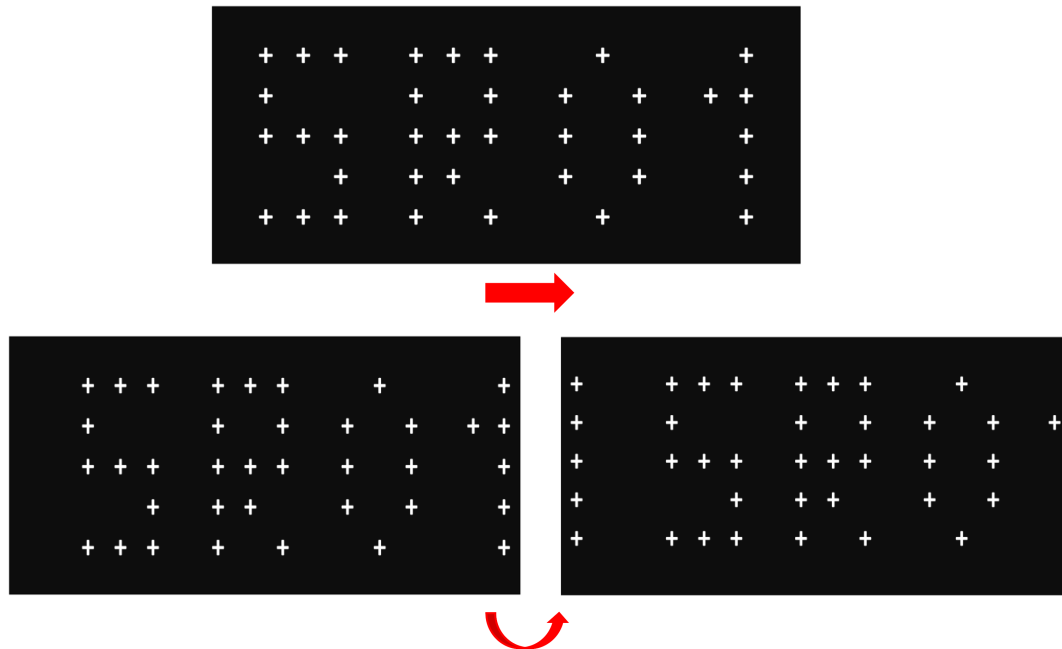
2 Exercice 2

L'objectif de cet exercice est de réaliser l'affichage du mot "SR01" puis de l'animer en le glissant d'une manière circulaire vers la droite comme suit:

1. Affichage:



2. Animation:



Cet exercice peut être réalisé en utilisant une matrice de caractères, où chaque case est remplie par un espace (' ') ou un '+' (voir l'exemple *test* disponible sur moodle). Penser à attribuer le droit d'exécution pour pouvoir lancer le programme *test* (*chmod +x test*).

3 Exercice 3

Pour chacun de ces programmes, exécuter le programme et donner une explication du résultat obtenu:

```

1 #include<stdio.h>
2 int main(){
3     int A=20,    B=5;
4     int C=!--A/++!B;
5     printf("A=%d B=%d c=%d \n", A,B,C);
6 }

```

Listing 1: programme 1

```

1 #include<stdio.h>
2 int main(){
3     int A=20,    B=5,    C=-10,    D=2;
4     printf("%d \n", A&&B||!0&&C++&&!D++);
5     printf("c=%d d=%d \n", C, D);
6 }

```

Listing 2: programme 2

```
1 #include<stdio.h>
2 int main(){
3     int p[4]={1,-2,3,4};
4     int *q=p;
5     printf("c=%d\n", *++q**q++);
6     printf("c=%d \n",*q);
7 }
```

Listing 3: programme 3

```
1 #include<stdio.h>
2 int main(){
3     int p[4]={1,-2,3,4};
4     int *q=p;
5     int d=*q&*q++|*q++;
6     printf("d=%d\n", d);
7     printf("q=%d \n",*q);
8 }
```

Listing 4: programme 4

```
1 #include<stdio.h>
2 int main(){
3     int a=-8,b=3;
4     int c= ++a&&--b ? b--:a++;
5     printf("a=%d b=%d c=%d\n",a,b,c);
6 }
```

Listing 5: programme 5

```
1 #include<stdio.h>
2 int main(){
3     int a=-8,b=3;
4     a>>=2^b;
5     printf("a=%d\n",a);
6 }
```

Listing 6: programme 6