

Examen de Langage C (*V. Granet*)

Durée : 1h30

Aucun document autorisé

Mobiles interdits

- 1. Aux États-Unis (entre autres), les étudiants sont notés avec des lettres qui définissent un grade : A = excellent, B = Bien, C = Passable, D = Insuffisant et F = Très Faible). Écrivez un programme C qui lit un grade sur l'entrée standard, et qui écrit sur la sortie standard l'avis correspondant. Vous utiliserez l'énoncé **switch**. Vous penserez aussi à traiter le cas d'erreur.

---

```
#include <stdio.h>
#include <stdlib.h>

int main(void) {
    char grade;
    scanf("%c", &grade);
    switch(grade) {
        case 'A' : printf("Excellent\n");
                    break;
        case 'B' : printf("Bien\n");
                    break;
        case 'C' : printf("Passable\n");
                    break;
        case 'D' : printf("Insuffisant\n");
                    break;
        case 'F' : printf("Très faible\n");
                    break;
        default  : fprintf(stderr, "Grade inconnu\n");
                    break;
    }
    return EXIT_SUCCESS;
}
```

---

On représente les coordonnées d'un point  $p$  du plan cartésien par deux réels (double)  $x$  et  $y$  tels que  $p = (x, y)$ .

- 2. Écrivez en C la fonction `distance` qui renvoie la distance entre 2 points  $p_1 = (x_1, y_1)$  et  $p_2 = (x_2, y_2)$  telle que  $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ . L'en-tête de cette fonction est le suivant :

```
/*
 * Antécédent : (x1,y1) et (x2,y2) 2 points du plan
 * Rôle : ...
 */
double distance(double x1, double y1, double x2, double y2)
```

---

```
/*
 * Antécédent : (x1,y1) et (x2,y2) 2 points du plan
 * Rôle : renvoie la distance les points (x1,y1) et (x2,y2)
 */
double distance(double x1, double y1, double x2, double y2) {
    return sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1));
}
```

---

- 3. On appelle  $\bar{p}$  la distance d'un point  $p = (x, y)$  par rapport à l'origine  $(0, 0)$ . Écrivez en C la fonction `compare` qui compare 2 points (4 réels double)  $p_1 = (x_1, y_1)$  et  $p_2 = (x_2, y_2)$  et renvoie :

-1 si  $\bar{p}_1 < \bar{p}_2$

0 si  $\bar{p}_1 = \bar{p}_2$

1 si  $\bar{p}_1 > \bar{p}_2$

---

```
/*
 * Antécédent : (x1,y1) et (x2,y2) 2 points du plan
 * Rôle : renvoie :
 *          -1 si (x1,y1) < (x2,y2)
 *           0 si (x1,y1) = (x2,y2)
 *           1 si (x1,y1) > (x2,y2)
 */
int compare(double x1, double y1, double x2, double y2) {
    double d1 = distance(0, 0, x1, y1);
    double d2 = distance(0, 0, x2, y2);
    if (d1 < d2) return -1;
    if (d1 > d2) return 1;
    // d1==d2
    return 0;
}
```

---

- 4. Écrivez un programme C qui lit sur l'entrée standard 4 réels (**double**) qui représentent les coordonnées de 2 points du plan cartésien  $(x_1, y_1)$  et  $(x_2, y_2)$ . Votre programme affiche d'abord la comparaison des 2 points, puis calcule la distance  $d$  entre les 2 points et affiche sur la sortie standard :
- *petite distance*, si  $d < 1$
  - *moyenne distance*, si  $1 \leq d < 10$
  - *grande distance*, si  $d \geq 10$

---

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include <assert.h>

int main(void) {
    double x1, y1, x2, y2, dist;
    // lire les coordonnées de 2 points du plan
    printf("x1 = "); scanf("%lf", &x1);
    printf("y1 = "); scanf("%lf", &y1);
    printf("x2 = "); scanf("%lf", &x2);
    printf("y2 = "); scanf("%lf", &y2);
    // afficher la comparaison des 2 points
    printf("%d\n", compare(x1,y1,x2,y2));
    // calculer la distance entre les 2 points
    dist = distance(x1, y1, x2, y2);
    // tester la distance
    assert(dist>=0);
    if (dist<1)
        printf("petite distance\n");
    else
        if (dist<10)
            printf("moyenne distance\n");
        else
            // dist>=10
            printf("grande distance\n");
    //
    return EXIT_SUCCESS;
}
```

.....