

Compte Rendu – Tâche 2

geometrie2d.c :

```
#include <stdio.h>
#include "geom2d.h"
#include <math.h>

Point set_point ( double x , double y ){
    Point P = {x,y} ;
    return P ;
}

Vecteur set_vecteur ( double x , double y ){
    Vecteur V = {x,y} ;
    return V ;
}

Point add_point ( Point P1 , Point P2 ){
    return set_point ( P1.x+P2.x , P1.y+P2.y ) ;
}

Vecteur add_vecteur ( Vecteur V1, Vecteur V2){
    return set_vecteur ( V1.x+V2.x , V1.y+V2.y ) ;
}

Point Produit_avec_un_reel_point ( double a, Point P){
    return set_point ( a* P.x, a* P.y);
}

Vecteur Produit_avec_un_reel_vecteur ( double a, Vecteur V){
    return set_vecteur ( a* V.x, a* V.y);
}

double produit_scalaire (Vecteur V1, Vecteur V2){
    return V1.x*V2.x + V1.y*V2.y;
}

Vecteur vect_bipoint ( Point A, Point B){
    Vecteur V = {B.x-A.x , B.y-A.y } ;    return
V;
}

double norme_vecteur (Vecteur V){
    return sqrt((V.x*V.x) + (V.y*V.y));
}

double distance_entre_points(Point P1, Point P2){
    double a = (P1.x - P2.x) * (P1.x - P2.x);    double
b = (P1.y - P2.y) * (P1.y - P2.y);    return sqrt (
a + b );
}
```

```
}
```

geometrie2d.h :

```
#include <stdio.h>

typedef struct Vecteur {
    double x , y ; // coordonnees
} Vecteur ;

typedef struct Point {    double
x , y ; // coordonnees
} Point ;

// cree le point de coordonnées (x,y)
Point set_point ( double x , double y );
// cree le vecteur de coordonnées (x,y)
Vecteur set_vecteur ( double x , double y );

// somme P1+P2
Point add_point ( Point P1 , Point P2 );

// somme V1+V2
Vecteur add_vecteur ( Vecteur V1, Vecteur V2);

// Produit aP
Point Produit_avec_un_reel_point ( double a, Point P);

// Produit aV
Vecteur Produit_avec_un_reel_vecteur ( double a, Vecteur V);

// Produit scalaire entre V1 et V2
double produit_scalaire (Vecteur V1, Vecteur V2);

// Norme de V
double norme_vecteur (Vecteur V);

// vecteur correspondant au bipoint AB */
Vecteur vect_bipoint ( Point A, Point B ) ;
// distance entre P1 et P2
double distance_entre_points(Point P1, Point P2);
```

Programme principal de test :

```
#include "geom2d.h"
```

```

int main(int argc, char **argv){
    Point A, B, C, D, E;
    Vecteur U, V1, V2, VR;
    A = set_point(1.0, -3.0);
    double x=3, norme, d;
    printf(" test set_point : ");
    if (A.x == 1.0 && A.y == -3.0){
        printf("Pass \n");
        printf("A = (%f,%f) \n", A.x, A.y);
    }
    else
        printf("Fail \n");

    printf(" test set_point : ");
    B = set_point(4.0, 1.0);
    if(B.x==4.0 && B.y == 1.0 ){
        printf("Pass \n");
        printf("B = (%f,%f) \n", B.x, B.y);
    }
    else
        printf("Fail \n");

    C = add_point(A,B);    /* -> C = (5.0, -2.0) */
    printf(" test somme de 2 points : ");
    if(C.x==5.0 && C.y == -2.0 ){
        printf("Pass \n");
        printf("C : somme de A et B \t= (%f,%f) \n", C.x, C.y);
    }
    else
        printf("Fail \n");

    V1= set_vecteur(0.0, 2.0);
    V2 = set_vecteur (0.0, 4.0);
    printf(" test set_vecteur : ");
    if (V1.x == 0.0 && V1.y == 2.0)
        printf("Pass \n");
    else
        printf("Fail \n");

    printf(" test add_vecteur : ");
    VR = add_vecteur(V1,V2);
    if (VR.x == 0.0 && VR.y == 6.0)
        printf("Pass \n");
    else
        printf("Fail \n");

    printf(" test produit_avec_un_reel_vecteur : ");

```

```

VR = Produit_avec_un_reel_vecteur(x,V1);
if (VR.x == 0.0 && VR.y == 6.0)
    printf("Pass \n");
else
    printf("Fail \n");

printf(" test vect_bipoint : ");
U = vect_bipoint(A,B); /* -> U = (-3.0, 4.0) */
if(U.x==3.0 && U.y == 4.0 )
    printf("Pass \n");
else
    printf("Fail \n");

printf(" test norme_vecteur : ");
norme = norme_vecteur(V1);
if (norme == 2)
    printf("Pass \n");
else
    printf("Fail \n");

D = set_point(0.0, 1.0);
E = set_point(0.0, 2.0);

printf(" test distance_entre_points: ");

d = distance_entre_points(D, E);
if(d == 1)
    printf("Pass \n");
else
    printf("Fail \n");

printf(" test Produit_avec_un_reel_point: ");
A = Produit_avec_un_reel_point(x, A);
if(A.x == 3.0 && A.y == -9.0)
    printf("Pass \n");
else
    printf("Fail \n");

printf(" test produit_scalaire: ");
norme = produit_scalaire(V1, V2);
if(norme == 8)
    printf("Pass \n");
else
    printf("Fail \n");

return 0;
}

```

Résultat jeu de test :

```
test set_point : Pass A = (1.000000,-3.000000) test set_point : Pass B =  
(4.000000,1.000000) test somme de 2 points : Pass C : somme de A et B =  
(5.000000,-2.000000) test set_vecteur : Pass test add_vecteur : Pass test  
produit_avec_un_reel_vecteur : Pass test vect_bipoint : Pass test  
norme_vecteur : Pass test distance_entre_points: Pass test  
Produit avec un reel_point: Pass test produit_scalaire: Pass
```