

# Compte rendu TACHE 2

Nader Marie Ange – Perret Florian

module geometrie2d.h:

```
typedef struct Point_  
{  
double abs,ord;  
} Point;  
  
typedef struct Vecteur_  
{  
double abs,ord;  
} Vecteur;  
  
// creer un point a partir de deux coordonnées  
Point creer_point(double abs , double ord);  
  
// creer un vect a coordonnées  
Vecteur creer_vecteur(double abs, double ord);  
  
// creer un vecteur a partir de deux points  
Vecteur creer_vecteur_bip (Point a , Point b);  
  
// somme de deux points  
Point somme_point(Point a, Point b);  
  
// somme de deux vecteurs  
Vecteur somme_vecteur(Vecteur u, Vecteur v);  
  
// multiplication d'un point par un réel  
Point mult_reel_p(Point a, double d);  
  
// multiplication d'un vecteur par un réel  
Vecteur mult_reel_v(Vecteur u, double d);  
  
// Produit scalaire de deux vecteur  
double produit_scalaire(Vecteur u , Vecteur v);  
  
// Norme d'un vecteur  
double norme_vect(Vecteur u);  
  
// distance entre deux points  
double dist_points(Point a, Point b);
```

geometrie2d.c :

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

```
#include<stdlib.h>
#include<string.h>
#include<math.h>
#include "geometrie2d.h"
```

```
Point creer_point(double abs , double ord){
Point p={abs,ord};
return p;
}
```

```
Vecteur creer_vecteur(double abs , double ord){
Vecteur v={abs,ord};
return v;
}
```

```
Vecteur creer_vecteur_bip(Point a , Point b){
Vecteur v={b.abs-a.abs,b.ord-a.ord};
return v;
}
```

```
Point somme_point(Point a, Point b){
return creer_point(a.abs+b.abs , a.ord+b.ord);
}
```

```
Vecteur somme_vecteur(Vecteur u, Vecteur v){
return creer_vecteur(u.abs+v.abs , u.ord+v.ord);
}
```

```
Point mult_reel_p(Point a, double d){
return creer_point(d*a.abs,d*a.ord);
}
```

```
Vecteur mult_reel_v(Vecteur u, double d){
return creer_vecteur(d*u.abs,d*u.ord);
}
```

```
double produit_scalaire(Vecteur u , Vecteur v){
return u.abs*v.abs+u.ord*v.ord;
}
```

```
double norme_vect(Vecteur u){
return sqrt((u.abs)*(u.abs)+(u.ord)*(u.ord));
}
```

```
double dist_points(Point a, Point b){
return sqrt(pow(a.abs-b.abs,2)+pow(a.ord-b.ord,2));
}
```

programme de test.c:

```
#include<stdio.h>
#include<stdlib.h>
#include<string.h>
#include<math.h>
#include "geometrie2d.h"

int main(int argc , char*argv[]){
    Point a=creer_point(4,5);
    Point b= creer_point(2,4);
    Vecteur u= creer_vecteur(6,9);
    Vecteur v= creer_vecteur_bip(a,b);

    Point somme_p=somme_point(a,b);
    Vecteur somme_v=somme_vecteur(u,v);
    Point multipli_p=mult_reel_p(a,4);
    Vecteur multipli_v=mult_reel_v(u,4);

    printf("Point a: %f %f\n",a.abs , a.ord);
    printf("Point b: %f %f\n",b.abs,b.ord);
    printf("Vecteur u: %f %f\n",u.abs,u.ord);
    printf("Vecteur v: %f %f\n",v.abs,v.ord);
    printf("Somme de a et b: %f %f\n",somme_p.abs,somme_p.ord);
    printf("Somme de u et v: %f %f\n" ,somme_v.abs,somme_v.ord);
    printf("Multiplier a par 4: %f %f\n",multipli_p.abs,multipli_p.ord);
    printf("Multiplier u par 4: %f %f\n",multipli_v.abs,multipli_v.ord);
    printf("Produit scalaire de u et v: %f\n",produit_scalaire(u,v));
    printf("Norme de u: %f\n",norme_vect(u));
    printf("Distance entre a et b: %f\n",dist_points(a,b));

}
```

résultat des tests : (les résultats ont été vérifiés et sont correctes)

```
Creation de l'executable test_geometrie
-----
clang test_geometrie.o geometrie2d.o -L. -lm -o test_geometrie
● ange@ange-GL502VY:~/Desktop/Map401$ ./test_geometrie
Point a: 4.000000 5.000000
Point b: 2.000000 4.000000
Vecteur u: 6.000000 9.000000
Vecteur v: -2.000000 -1.000000
Somme de a et b: 6.000000 9.000000
Somme de u et v: 4.000000 8.000000
Multiplier a par 4: 16.000000 20.000000
Multiplier u par 4: 24.000000 36.000000
Produit scalaire de u et v: -21.000000
Norme de u: 10.816654
Distance entre a et b: 2.236068
○ ange@ange-GL502VY:~/Desktop/Map401$
```