Série N°2
TD Algorithmique & Programmation

Exercice 1

Corriger les instructions erronées et donner le schéma d'exécution dans la mémoire :

Instructions	Schéma
float x = 6, y = 4 ; **float *p1, *p2 ;**	x 6 y 4 p1 ☐ p2 ☐
p1 = x ; **p2 = (float)malloc(int) ;** ***p2 = 20 ;**	x 6 y 4 p1 ☐ p2 ☐
p1 = p2 ;	x 6 y 4 ☐ p1 ☐ p2 ☐
***p1 = 300 ;**	x 6 y 4 ☐ p1 ☐ p2 ☐
***p2 = 200 ;**	x 6 y 4 ☐ p1 ☐ p2 ☐
p1=y **free(*p2) ;** **p2 = &y ;**	x 6 y 4 ☐ p1 ☐ p2 ☐

Exercice 2

Compléter le tableau suivant :

	A	**B**	**C**	**P1**	**P2**
Initialisation	1	2	3	/	/
P1=&A	1	2	3	&A	/
P2=&C					
***P1=(*P2)++**					
P1=P2					
P2=&B					
***P1-=*P2**					
++*P2					

Exercice 3

1. Ecrire une fonction ***fct*** qui permet de calculer la formule suivante :

$$\frac{x^2+1}{2} \text{ avec } x \epsilon \mathbb{R}$$

2. Tester la fonction ***fct*** :

…….. fct (…….)
{
…….
}
	int main()
	{ ……..
	……..
	……..
	……..
	}

Exercice 4

1. Ecrire une procédure ***message*** qui permet d'afficher le terme « Bonjour MIP »***n*** fois.
2. Tester la procédure ***message*** :

…….. message (……..)
{
……..
……..
……..
}
	int main()
	{
	……..
	……..
	……..
	……..
	……..
	}

Exercice 5

Donner le résultat d'exécution du programme suivant :

```
#include <stdio.h>
int x=4;
int avancer1( ){
   x+=1;
   return x ;
}
void avancer2 (int y){
   y+=2;
}
void reculer2 ( ){
   x-=2;
}
int reculer3 (int  x){
    x-=3;
   return x ;
}
```

```
main ( ){
      int  y;
      y=avancer1 ()+2;
      printf("\n x=%d",x);
      printf("\n y=%d",y);
      avancer2(y);
      printf("\n x=%d",x);
      printf("\n y=%d",y);
      reculer2( );
      printf("\n x=%d",x);
      printf("\n y=%d",y);
      x=reculer3(y)-3;
      printf("\n x=%d",x);
      printf("\n y=%d",y);
}
```