

Exercice 1:

Exécute le programme suivant :

```

#include<stdio.h>
#include<stdlib.h>
int main(){

int *p=malloc(4*sizeof(int));
*(p+0)=1;|
*(p+1)=2;
*(p+2)=3;
*(p+3)=7;
printf("%d ",*(p+2));
free(p);
printf("\n %d ",*(p+2));
}

```

```

1  #include<stdio.h>
2  #include<stdlib.h>|
3  main(){
4  | printf("\n-----malloc-----\n");
5  | int *a = malloc(2*sizeof(int));
6  | if (a == NULL){
7  | | printf("La memoire n'est pas allouée");
8  | | }
9  | else{
10 | | printf("%d\n",*(a+0));
11 | | printf("%d\n",*(a+1));
12 | | *(a+0)=1;
13 | | *(a+1)=2;
14 | | printf("%d\n",*(a+0));
15 | | printf("%d\n",*(a+1));
16 | | free(a);
17 | | }

```

```

18 | printf("\n-----calloc-----\n");
19 | int *b = calloc(2, sizeof(int));
20 | if (b == NULL){
21 | | printf("La memoire n'est pas allouée");
22 | | }
23 | else{
24 | | printf("%d\n",*(b+0));
25 | | printf("%d\n",*(b+1));
26 | | *(b+0)=1;
27 | | *(b+1)=2;
28 | | printf("%d\n",*(b+0));
29 | | printf("%d\n",*(b+1));
30 | | free(b);
31 | | }

```

```
45 | p=realloc(p,5*sizeof(int));
46 |
47 | printf("%d\n",*(p+0));
48 | printf("%d\n",*(p+1));
49 | printf("%d\n",*(p+2));
50 | printf("%d\n",*(p+3));
51 | printf("%d\n",*(p+4));
52 | free(p);
53 | }
54 | }
55 |
```

Exercice2 :

Ecrire un programme qui alloue dynamiquement la mémoire (avec malloc() ou calloc) d'un tableau de n éléments . Ensuite le programme calcule et affiche la somme et la moyenne des éléments du tableau .

Veillez Structurer le programme de la manière suivante :

void remplissage (float *t,int n)

void affichage (float *t,int n)

float * somme (float *t,int n)

float * moyenne (int *n)