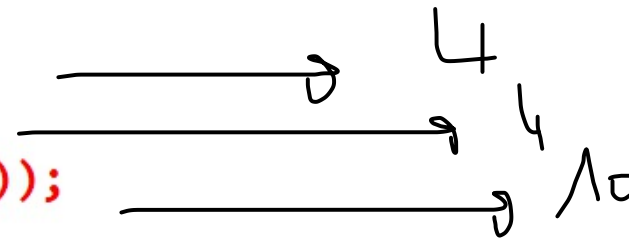


Chapitre 3. Gestion dynamique de la mémoire

```
#include<stdio.h>
main()
{
    int a;
    float x;
    char nom[10]="ali";
    printf("la taille de a est %d\n",sizeof(a));
    printf("la taille de x est %d\n",sizeof(x));
    printf("la taille de nom est %d\n",sizeof(nom));
}
```

```
la taille de a est 4
la taille de x est 4
la taille de nom est 10
```



```

1 #include<stdio.h>
2 //Réservation statique
3 main()
4 {
5     int T[1000],N,i;
6     printf("Donnez le nombre des elements:");
7     scanf("%d",&N);
8     //Remplir le tableau
9     for(i=0;i<N;i++)
10    {
11        printf("T[%d]=",i);
12        scanf("%d",&T[i]);
13    }
14    //Affichage
15    for(i=0;i<N;i++)
16        printf("%d\t",T[i]);
17 }

```

Donnez le nombre des elements:5

```

T[0]=10
T[1]=20
T[2]=30
T[3]=5
T[4]=2
10      20      30      5      2
-----

```

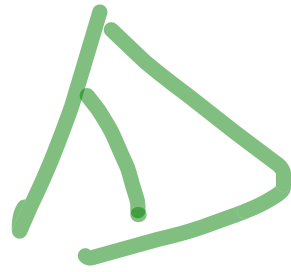
$N=5$

on a réservé $4 \times 1000 = 4000$ octets

on a utilisé $N \times 4 = 20$ octets

Reste d'espace: $4000 - 20$
 $= 3980$ octets

L'inconvénient majeur de la réserve statique est la perte de l'espace mémoire



Solution : on utilise la réserve dynamique

Réservation statique

1. Perte de l'espace
2. Se fait au moment de l'usage
3. Zone statique
4. Pas de pointeur

Réservation dynamique

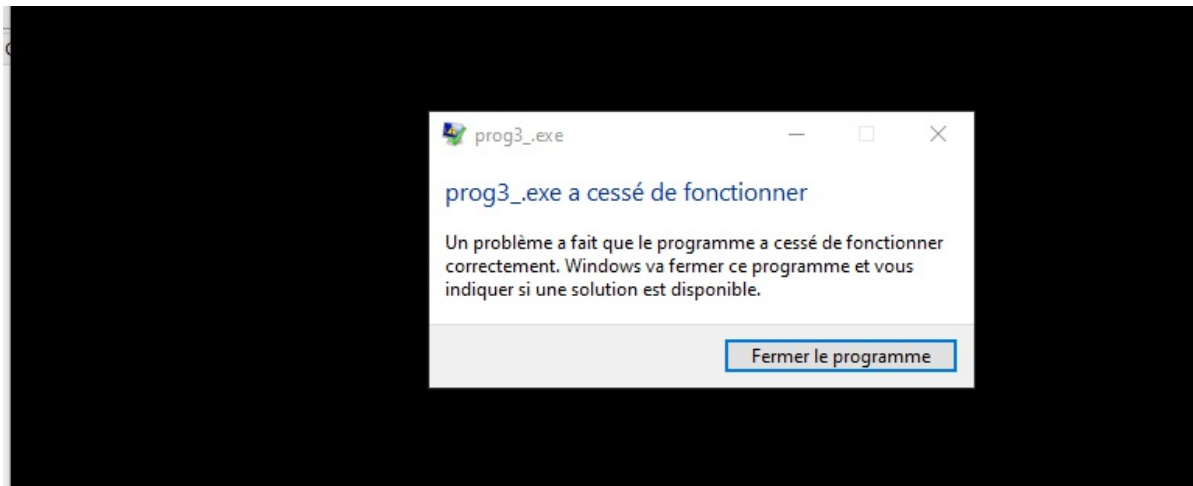
1. aucune Perte
2. Se fait au moment d'exécution
3. Tas
4. Pointeur

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 //Réservation dynamique
4 main()
5 {
6     int *p;
7     p=(int*)malloc(4);
8     printf("la valeur de p est %p",p);
9 }
```

```
la valeur de p est 0000000000C21410
```

```
-----
```

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 //Réservation dynamique
4 main()
5 {
6     int *p;
7     *p=10 ;
8     printf("le contenu pointe par p est %d",*p);
9 }
```



```
//Réservation statique  
int T[10];
```

```
//Réservation dynamique
```

```
int *T;  
T=(int*)malloc(10*sizeof(int))
```

```

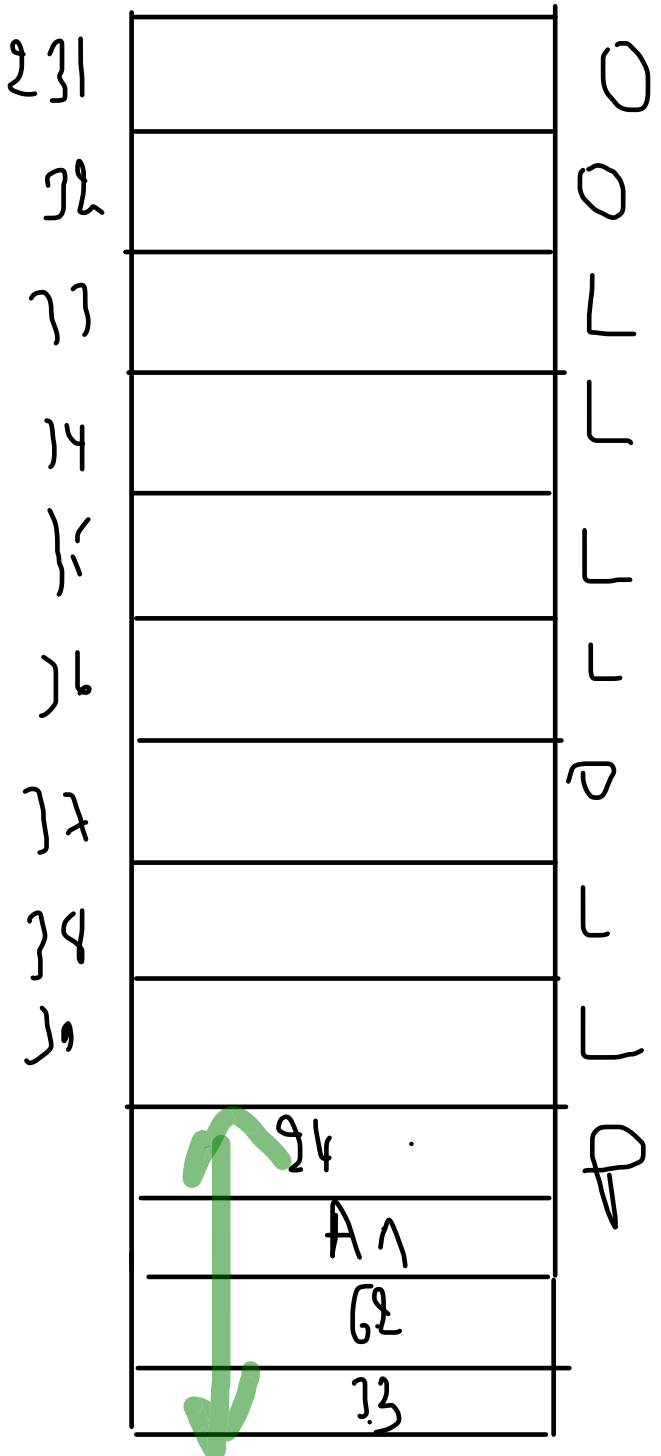
1 #include<stdio.h>
2 #include<stdlib.h>
3 //Réservation dynamique
4 main()
5 {
6 char *p;
7 p=(char*)malloc(4);
8 printf("la valeur de p est %p",p);
9 }

```

donc

$p = 24A16233$

24A16233




```

#include<stdio.h>
#include<stdlib.h>
//Réservation dynamique
main()
{
    int *T,N,i;
    printf("Donnez le nombre des elements:");
    scanf("%d",&N);
    ///La réservation
    T=(int*)malloc(N*4);
    //Remplir le tableau
    for(i=0;i<N;i++)
    {
        printf("T[%d]=",i);
        scanf("%d",&T[i]);
    }
    //Affichage
    for(i=0;i<N;i++)
        printf("%d\t",T[i]);
}

```

```

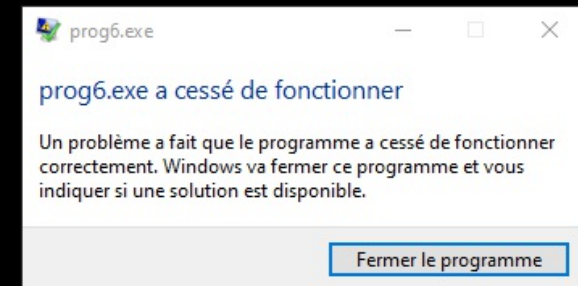
Donnez le nombre des elements:4
T[0]=10
T[1]=20
T[2]=30
T[3]=5
10      20      30      5
-----

```

```
#include<stdio.h>
#include<stdlib.h>
//Réserveation dynamique
main()
{
int *T,N,i;
printf("Donnez le nombre des elements:");
scanf("%d",&N);
//La réservation
T=(int*)malloc(N*4);
//Affichage
for(i=0;i<N;i++)
    printf("%d\t",T[i]);
}
```

```
Donnez le nombre des elements:5
10511952      0      10486096      0      1867275361
```

```
#include<stdio.h>
#include<stdlib.h>
//Réserveation dynamique
main()
{
int *p1,*p2;
*p1=10;
*p2=*p1+5;
printf("le contenu pointé par p1 est %d\n",*p1);
printf("le contenu pointé par p2 est %d\n",*p2);
}
```



```
1 #include<stdio.h>
2 #include<stdlib.h>
3 //Réserveation dynamique
4 main()
5 {
6     int *p1,*p2;
7     //On doit réserver l'espace
8     p1=(int*)malloc(4);
9     p2=(int*)malloc(4);
10    *p1=10;
11    *p2=*p1+5;
12    printf("le contenu pointé par p1 est %d\n",*p1);
13    printf("le contenu pointé par p2 est %d\n",*p2);
14 }
```

```
le contenu pointé par p1 est 10
le contenu pointé par p2 est 15
```

```
#include<stdio.h>
#include<stdlib.h>
//Problème de Réserveation: on doit réserver;
main()
{
    int *p1,*p2;
    p1=(int*)malloc(sizeof(int));
    p2=(int*)malloc(sizeof(int));
    *p1=10;
    *p2=*p1+5;
    printf("le contenu pointé par p1 est %d\n",*p1);
    printf("le contenu pointé par p2 est %d\n",*p2);
}
```



```

#include<stdio.h>
#include<stdlib.h>
typedef struct complexe{
    float re;
    float im;}cplx;

////main()
main()
{
    cplx z1,*z2;
    z1.re=10;
    z1.im=4;
    (*z2).re=5;
    (*z2).im=3;
    printf("Z1=%f+%fi\n",z1.re,z1.im);
    printf("Z2=%f+%fi\n",z2->re,z2->im);
}

```

enum

on doit noter
l'es pte //

```

1  #include<stdio.h>
2  #include<stdlib.h>
3  typedef struct complexe{
4      float re;
5      float im;}cplx;
6
7  ///main()
8  main()
9  {
10     cplx z1,*z2;
11     z1.re=10;
12     z1.im=4;
13     z2=(cplx*)malloc(sizeof(cplx));
14     (*z2).re=5;
15     (*z2).im=3;
16     printf("Z1=%f+%fi\n",z1.re,z1.im);
17     printf("Z2=%f+%fi\n",z2->re,z2->im);
18 }

```

C:\Users\user\Desktop\S4_2021\gestion_Memoire\prog8_.exe

Z1=10.000000+4.000000i

Z2=5.000000+3.000000i

Process exited after 0.04465 seconds with return value

Appuyez sur une touche pour continuer...

$z2 \rightarrow re = 5$

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  typedef struct complexe{
4      float re;
5      float im;}cplx;
6
7  ///main()
8  main()
9  {
10     cplx z1,*z2;
11     z1.re=10;
12     z1.im=4;
13     z2=(cplx*)malloc(sizeof(cplx));
14     z2->re=5; //(*z2).re=5;
15     z2->im=3; //(*z2).im=3;
16     printf("Z1=%f+%fi\n",z1.re,z1.im);
17     printf("Z2=%f+%fi\n",z2->re,z2->im);
18 }
```



```
1 #include<stdio.h>
2 #include<stdlib.h>
3 //Réservation dynamique
4 main()
5 {
6     int *T,N,i;
7     printf("Donnez le nombre des elements:");
8     scanf("%d",&N);
9     ///La réservation
10    T=(int*)calloc(N,sizeof(int));
11    //Affichage
12    for(i=0;i<N;i++)
13        printf("%d\t",T[i]);
14 }
```

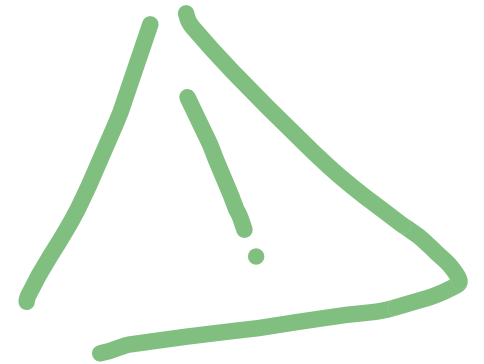
```
Donnez le nombre des elements:5
0      0      0      0      0
-----
```

```

#include<stdio.h>
#include<stdlib.h>
//Réserveation dynamique
main()
{
int N,i;
char *T;
printf("Donnez le nombre des elements:");
scanf("%d",&N);
///La réservation
T=(char*)malloc(N*sizeof(char));
//Affichage
for(i=0;i<N;i++)
    printf("%c\t",T[i]+65);
}

```

caractères aléatoires



```

Donnez le nombre des elements:6
æ      o      Â      A      A      A
-----

```

```

1  #include<stdio.h>
2  #include<stdlib.h>
3  //Réservation dynamique
4  main()
5  {
6  int N,i;
7  char *T;
8  printf("Donnez le nombre des elements:");
9  scanf("%d",&N);
10 //La réservation
11 T=(char*)calloc(N,sizeof(char));
12 //Affichage
13 for(i=0;i<N;i++)
14     printf("%d\t",T[i]);
15 }

```

```

Donnez le nombre des elements:5
0      0      0      0      0
-----

```

Remarque:
0 est le code ascii de
caractère '\0'

```

1  #include<stdio.h>
2  #include<stdlib.h>
3  //Réservation dynamique
4  main()
5  {
6  int N,i;
7  float *T;
8  printf("Donnez le nombre des elements:");
9  scanf("%d",&N);
10 //La réservation
11 T=(float*)calloc(N,sizeof(float));
12 //Affichage
13 for(i=0;i<N;i++)
14     printf("%f\t",T[i]);
15 }

```

```

Donnez le nombre des elements:5
0.000000      0.000000      0.000000      0.000000      0.000000
-----

```

Pour libérer l'espace réservée on utilise la
fonction `free(p)`

