

Exercice 3:

```
#include <stdio.h>
void main() {
    int t [3];
    int i, j;
    int *adt;
    for (i=0, j=0 ; i<3 ; i++) t[i] = j++ + i ;/*1*/
    for (i=0 ; i<3 ; i++) printf("%d\n", t[i]);/*2*/
    for (i=0 ; i<3 ; i++) printf("%d\n",*(t+i));/*3*/
    for (adt = t ; adt < t+3 ; adt++) printf("%d\n",*adt);/*4*/
    for (adt = t+2 ; adt>=t ; adt--) printf("%d\n",*adt);/*5*/
}
```

1)- $t[i] = j++ + i \implies t[i] = j+i; j++;$

Donc $i=0$: $t[0]=0; j=1;$

- $i=1$: $t[1]=2; j=2;$
- $i=2$: $t[2]=4; j=3;$

2)- l'affichage de $t[i]$:

- 0
- 2
- 4

3)- Rappel: $t=&t[0]$, $t+i=&t[i]$, $*(t+i)=t[i]$

Donc le programme va afficher:

- 0
- 2
- 4

4)- Dans cas on $adt=&t[0]$ et $t+3=&t[3]$ et $*(adt+i)=t[i]$

Donc le programme va afficher:

```
pour adt=t:0 /* adt=&t[0] ==> * adt=t[0];adt++ ==> adt=&t[1]*/
pour adt=t+1:2 /*adt=&t[1]==> *adt=t[1];adt++ ==> adt=&t[2]*/
pour adt=t+2:4 /*adt=&t[2]==> *adt=t[2];adt++ ==> adt=&t[3]*/
```

5)- Ici on commence par la dernière valeur:

```
pour adt=t+2:4/*adt=&t[2]==> *adt=t[2];adt--==> adt=&t[1]*/
pour adt=t+1:2/* adt=&t[1]==> *adt=t[1];adt--==> adt=&t[0]*/
pour adt=t:0/* adt=&t[0]==> *adt=t[0]*/
```

Exercice 4:

```
#include <stdio.h>
#define M100
void main() {
    int t[M],n,i,min,max,s=0;
    printf("Donnez la dimension <100\n");
    scanf("%d", &n);//saisie
    printf("Tapper les elements de T: \n");
    for (i=0; i<n; i++){
        printf("T[%d] = ",i);
        scanf("%d",&t[i]);
    }
    //calcul
    max = min = t[0];
    for (i=0 ; i<n; i++){
        s=s+t[i];
        if(t[i]>max) max = t[i]; //max = t[i]>max?t[i]:max
        if(t[i]<min) min= t[i]; //min= t[i]<min?t[i]:min
    }
    printf("La somme: %d, la moyenne:%.2f\n",s,(float)s/n);
    printf("La valeur max: %d\n",max);
    printf("La valeur min: %d\n",min);
}
```

Exercice 5:

```
#include <stdio.h>
#define M100
void maxmin(int t[], int n, int *admax, int *admin){
    int i,max,min;
    max = t[0];
    min = t[0];
    for (i=1;i<n; i++){
        if(t[i]>=max)max = t[i]; //max= t[i]>max ? t[i]:max
        if(t[i]<=min)min = t[i]; //min= t[i]<min ? t[i]:min
    }
    *admax=max;
    *admin=min*;
}
int main() {
    int t[M],n,i,min,max;
    printf("Donnez la dimension <100\n");
```

```

scanf("%d", &n);//saisie
printf("Tapper les elements de T: \n");
for (i=0; i<n; i++){
    printf("T[%d] = ",i);
    scanf("%d",&t[i]);
}
maxmin(t,n,&max,&min);
printf("La valeur max: %d\n",max);
printf("La valeur min: %d\n",min);
}

```

Exercice 6:

```

#define N 10
int tab[N] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
int *p = &tab[0], *q = &tab[N-1], temp;
while (p < q){
    temp = *p;
    *(p++) = *q;
    *(q--) = temp;
}

```

On a le tableau:

1	2	3	4	5	6	7	8	9	10
---	---	---	---	---	---	---	---	---	----

```

// C pas un code valide, c juste une explication
temp=*p=&tab[0] = 1;
*(p++)= *q ==>*p= *q;
p++= &tab[0]++;==>*p= *q= &tab[9] = 10;
p=&tab[1]*(q--)= temp==>*q= temp;
q--= &tab[9]--;==>*q= temp= 1;
q=&tab[8]
// Et ainsi de suite...

```

En fin, on obtient le tableau:

10	9	8	7	6	5	4	3	2	1
----	---	---	---	---	---	---	---	---	---

Exercice 7:

a)- En utilisant 2 tableaux:

```

#include <stdio.h>
#define M 100
void saisie(float t[], int n){
    int i;
    for(i=0;i<n;i++){
        printf("t[%d] = ",i);
        scanf("%f",&t[i]); //t+i
    }
}
void invrser(float t1[], float t2[], int n){
    int i;
    for(i=0; i<n; i++)
        t2[i]=t1[n-1-i];
}
void affiche(float t[], int n){
    int i;
    for(i=0;i<n;i++){
        printf("t[%d] = ",i);
        printf("%f\n",t[i]);
    }
}
int main() {
    int n;
    float t1[M],t2[M];
    printf("Donnez la dimension <100\n");
    scanf("%d", &n);
    saisie(t1,n);
    invrser(t1,t2,n);
    affiche(t2,n);
    return 0;
}

```

b)- En utilisant un tableau:(voir exercice 6)

Exercice 8:

```

#include <stdio.h>
int restQ(int a, int b, int q, intr){
    if(a>b){
        r=a%b;q=a/b;
    }else{
        r=b%a;q=b/a;
    }
}

```

```
}  
int main() {  
    int x,y,r,q;  
    printf("Donnez les deux nombres: ");  
    scanf("%d%d", &x,&y);  
    printf("Le quotient: %d et le reste: %d",restQ(x,y,q,r));  
    return 0;  
}
```