



lesson 81 Practice 22 (Statistics)

فى هذا الدرس نريد عمل برنامج يقوم المستخدم بإدخال **n numbers** أى أن **n** عدد الأرقام التي سيدخلها المستخدم ويسأل المستخدم سؤال : ماهو الرقم الذي تبحث عنه من داخل هذه الأرقام؟
يجيب المستخدم بالرقم الذي يريده ثم يقوم البرنامج بطباعة بعض البيانات الخاصة بهذا الرقم.

- كم عدد الأرقام التي تقبل القسمة على هذا الرقم وليكن **x**
- كم عدد الأرقام التي أكبر من **x**
- كم عدد الأرقام التي أصغر من **x**
- كم عدد الأرقام التي تساوي **x**

دعنا نحلل هذا السؤال :

سيقوم المستخدم بإدخال الأرقام أولاً ثم الرقم الذي يريد عمل الإحصائيات عليه وبالتالي هنا هل سنحتاج إلى **Array** أم لا؟

هنا يقوم المستخدم بإدخال الأرقام أولاً
فى هذه الحالة اذا لم نسجل الأرقام لن نستطيع البحث عن **x** داخل مجموعة الأرقام لأننا لم نسجلها

وبالتالى لن نستطيع عمل الإحصائيات بدون استخدام **Array**

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

```
int main( ) {
```

```
int i, n, z[100], x, gx = 0, sx = 0, ex = 0, dx = 0;
```



```
// المكان الذي نسجل فيه الأرقام z[100]
// الرقم الذي ستجرى عليه الاحصائيات x
// gx ومعناها grater than x أكبر من x
// sx ومعناها smaller than x أصغر من x
// ex ومعناها equal x يساوي x
// dx ومعناها divide x تقبل القسمة على x
```

```
printf("Enter number of numbers: ");
// إحياناً يقوم البعض بكتابة #
// بدلاً من number
// printf("Enter # of numbers: ");
scanf("%d", &n);
```

```
for (i = 0; i < n; i++) {
    printf("Enter # %d: ", i + 1);
    // نطبع ترتيب الأرقام
    scanf("%d", &z[i]);
    // قام المستخدم بإدخال كل الأرقام
}
```

```
printf("Enter x: ");
scanf("%d", &x);
// يقوم المستخدم بإدخال الرقم المراد العمل عليه
```

```
/*
```

هنا خطأ لأن إذا كان شرط في if صحيح الأمر الذي يليه else if لن ينتقل إليه *



```
for (i = 0; i < n; i++) {  
    if (z[i] > x)  
        gx++;  
    // نقوم بعد الأرقام التي أكبر من x  
    else if (z[i] < x)  
        sx++;  
    // نقوم بعد الأرقام التي أصغر من x  
    else if (z[i] % x == 0)  
        dx;  
    // إذا كان باقى القسمة صفر معناه أنه يقبل القسمة  
    else if (z[i] == x)  
        ex++;  
    // نقوم بعد الأرقام التي تساوي x  
  
    وبالتالى وضعنا else if أن أمر واحد فقط سينفذ  
}  
*/
```

```
for (i = 0; i < n; i++) {  
    if (z[i] > x)  
        gx++;  
    // نقوم بعد الأرقام التي أكبر من x  
    // هنا سواء الشرط صحيح أو لا ينتقل إلى الأمر الذى يليه  
    if (z[i] < x)  
        sx++;  
    // نقوم بعد الأرقام التي أصغر من x
```



// هنا سواء الشرط صحيح أو لا ينتقل إلى الأمر الذي يليه //

```
if (z[i] % x == 0)
```

```
dx++;
```

// إذا كان باقى القسمة صفر معناه أنه يقبل القسمة //

// هنا سواء الشرط صحيح أو لا ينتقل إلى الأمر الذي يليه //

```
if (z[i] == x)
```

```
ex++;
```

// نقوم بعد الأرقام التي تساوي x //

// هنا سواء الشرط صحيح أو لا ينتقل إلى الأمر الذي يليه //

//else if لذلك لا نستخدم أمر //

```
}
```

```
printf(
```

```
    "%d Greater than %d\n"
```

```
    "%d Smaller than %d\n"
```

```
    "%d Divisible by %d\n"
```

```
    "%d Equal to %d\n",gx,x,sx,x,dx,x,ex,x);
```

```
}
```

input:

Enter number of numbers: 10

Enter # 1: 9

Enter # 2: 6

Enter # 3: 3

Enter # 4: 1



Enter # 5: 2

Enter # 6: 1

Enter # 7: 2

Enter # 8: 1

Enter # 9: 2

Enter # 10: 1

Enter x: 3

output:

2 Greater than 3

7 Smaller than 3

3 Divisible by 3

1 Equal to 3

(قم بتجربة الكود بنفسك واضغط هنا)

يمكننا اختصار الكود أكثر من ذلك باستخدام Ternary Operator
ويمكننا أيضاً حجز array بالعدد الذي يريده المستخدم لكن هناك بعض ال compiler قد
يظهر خطأ لأنه لا يقبل تعريف المتغيرات إلا في السطر الأول من البرنامج بعد main
مباشرة

وسيكون بهذه الطريقة :

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

```
int main( ) {
```

```
    int i, n, x, gx = 0, sx = 0, ex = 0, dx = 0;
```

```
    // الرقم الذي ستجرب عليه الاحصائيات x
```

```
    // gx ومعناها grater than x أكبر من x
```



// x ومعناها smaller than x أصغر من x

// x ومعناها equal x يساوي x

// x ومعناها divide x تقبل القسمة على x

printf("Enter number of numbers: ");

// أحياناً يقوم البعض بكتابة #

//number بدلاً من

// printf("Enter # of numbers: ");

scanf("%d", &n);

int z[n];

// بهذه الطريقة قمنا بحجز array لكن بالعدد الذي يطلبه المستخدم

// منعاً لإهدار المساحة

// قد يعمل البرنامج هنا بدون أخطاء

// لكن في بعض compiler التي تستخدم إصدار اللغة القديم لاتعمل

for (i = 0; i < n; i++) {

printf("Enter # %d: ", i + 1);

// نطبع ترتيب الأرقام

scanf("%d", &z[i]);

// قام المستخدم بإدخال كل الأرقام

}

printf("Enter x: ");

scanf("%d", &x);

// يقوم المستخدم بإدخال الرقم المراد العمل عليه

for (i = 0; i < n; i++) {

gx += z[i] > x;



```
// x بعد الأرقام التي أكبر من x
sx += z[i] < x;
// x بعد الأرقام التي أصغر من x
dx += z[i] % x == 0;
// إذا كان باقى القسمة صفر معناه أنه يقبل القسمة
ex += z[i] == x;
// x بعد الأرقام التي تساوي x

}
printf(
    "%d Greater than %d\n",
    "%d Smaller than %d\n",
    "%d Divisible by %d\n",
    "%d Equal to %d\n", gx, x, sx, x, dx, x, ex, x);
}
```

input:

Enter number of numbers: 10

Enter # 1: 9

Enter # 2: 6



Enter # 3: 3

Enter # 4: 1

Enter # 5: 2

Enter # 6: 1

Enter # 7: 2

Enter # 8: 1

Enter # 9: 2

Enter # 10: 1

Enter x: 3

output:

2 Greater than 3

7 Smaller than 3

3 Divisible by 3

1 Equal to 3

(قم بتجربة الكود بنفسك واضغط هنا)