



lesson 49 C Practice 2 (Tracing)

لكي تكون مبرمج متميز عليك أن تتعلم أن تقوم بعمل **trace** لأي كود برمجي، أي أنك تستطيع تعقب أي برنامج في ذهنك أو على الورق بدون الحاجة الى عمل **run** للكود، وتكون قادر على فهمه وماذا سيفعل إذا قمنا بعمل **run** وبهذا تستطيع أن تكتشف إذا كان هناك خطأ أثناء كتابتك للكود

لاحظ بعض الأخطاء لا يظهر مباشرة في الكود و هي الأخطاء المنطقية (Logic Errors) ، هذه الأنواع من الأخطاء يحتاج منك أن تكون قوى الملاحظة و تستطيع أن تكتشف الخطأ بمجرد النظر أو بالتحليل المنطقي للكود الذي أمامك

في هذا الدرس سنتعرف على بعض الأمثلة للتدريب على **trace** :

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

```
//تضمين المكتبة التي تحتوي على printf
```

```
int main( ) {
```

```
    int i, x;
```

```
    //هنا قمنا بحجز متغيرين لا يوجد بهما أي قيم
```

```
    x = 0;
```

```
    //هنا قمنا بجعل المتغير بصفر
```

```
    for (i = 1; i < 5; i++) {
```

```
        //أمر for سيقوم بالدوران 4 لفات
```

```
        x += i;
```

```
        //هنا قم بإضافة كل قيمة ل i في x
```

```
    }
```



```
printf("%d\n", x);
// هنا قم طباعة قيمة x

}
```

فى امر **for**

يجعل قيمة **i=1** يفحص الشرط هل قيمة **5 < i**
 إذا كانت نعم فينتقل إلى الأقواس { **x += i** ; } وتنفيذ الأوامر التى بداخلها
 ثم يعود إلى أمر **for** ويقوم بعمل **i++**

الدورة الثانية :

يجعل قيمة **i=2** يفحص الشرط هل قيمة **5 < i**
 إذا كانت نعم فينتقل إلى الأقواس { **x += i** ; } وتنفيذ الأوامر التى بداخلها
 ثم يعود إلى أمر **for** ويقوم بعمل **i++**

الدورة الثالثة :

يجعل قيمة **i=3** يفحص الشرط هل قيمة **5 < i**
 إذا كانت نعم فينتقل إلى الأقواس { **x += i** ; } وتنفيذ الأوامر التى بداخلها
 ثم يعود إلى أمر **for** ويقوم بعمل **i++**

الدورة الرابعة:

يجعل قيمة **i=4** يفحص الشرط هل قيمة **5 < i**
 إذا كانت نعم فينتقل إلى الأقواس { **x += i** ; } وتنفيذ الأوامر التى بداخلها
 ثم يعود إلى أمر **for** ويقوم بعمل **i++**



الدورة الخامسة :

يجعل قيمة $i = 5$ يفحص الشرط هل قيمة $i < 5$

هنا يكون لا، يكون الشرط **false** وبالتالي يتوقف أمر **for**

ويقوم بتنفيذ الأوامر الموجودة بعده

وهو أمر الطباعة

عندما كنا نقوم بعمل loops وكان أمر $x += i$ يتكرر في كل دورة

كانت قيم i و x

كالتالى :

ترتيب الدوران	قيمة i	قيمة x	قيمة x النهائية
الدوران الأول	1	0	1
الدوران الثاني	2	1	3
الدوران الثالث	3	3	6
الدوران الرابع	4	6	10

وبالتالى هنا قمنا بعمل **trace** ، وهو معرفة ما سيفعله البرنامج قبل أن يقوم بالطباعة