



مقدمة في البرمجة Introduction To Programming

IPG101

الفصل الثامن تمارين عملية على بنى التحكم Practice On Control Structures

الكلمات المفتاحية

تدفق، تحكم، مجال رؤية، تعبير بولياني، شرط، أمر شرطي، if، else، switch، تكرار، حلقة، حلقة لا نهائية، while، do while، for، break، continue.

ملخص الفصل

يتضمن هذا الفصل مجموعة من التمارين التطبيقية على بنى التحكم وذلك بهدف تعزيز المفاهيم النظرية التي تلقاها خلال الفصلين السابقين واكتساب المهارات العملية في التعامل أنماط المسائل التي تتضمن قرار أو تكرار وتصميم الحلول لها وكتابة الشيفرة البرمجية بلغة C# التي تعبر عن حلول هذه المسائل.

أهداف الفصل

بنهاية هذا الفصل سيكون الطالب قادراً على:

- تعزيز مهاراته في صياغة الشروط والتعبير البوليانية.
- تصميم حلول لمسائل القرار والتكرار.
- كتابة الشيفرة البرمجية بلغة C# للمسائل التي تتضمن تحكماً.
- تنفيذ البرامج واكتشاف الأخطاء وتصحيحها.

محتويات الفصل

1. التمرين الأول : مسألة قابلية القسمة.
2. التمرين الثاني : مسألة حساب قيمة تابع.
3. التمرين الثالث: مسألة حساب تقدير طالب.
4. التمرين الرابع : مسألة طباعة أيام الأسبوع.
5. التمرين الخامس : مسألة طباعة قواسم عدد صحيح موجب.
6. التمرين السادس : مسألة إيجاد معكوس رقم صحيح.
7. التمرين السابع : مسألة حساب أعداد الناجحين.

1- التمرين الأول: مسألة قابلية القسمة

أكتب برنامجاً يطلب إلى المستخدم إدخال قيمة صحيحة من لوحة المفاتيح، ثم يقوم بطباعتها إذا كانت تقبل القسمة على 6 و 7 في آن معاً.

```
Console.WriteLine("ENTER A NUMBERS: ");
int num;
num = Int32.Parse(Console.ReadLine());
if (num % 6 == 0 && num % 7 == 0)
    Console.WriteLine("IS MULTIPLE TO 6 and 7 ");
else
    Console.WriteLine("IS NOT MULTIPLE TO 6 and 7 ");
```

مابعد الحل:

قم بتعديل الحل السابق بحيث يكون ملائماً لحل المسائل التالية:

- ✓ طباعة العدد المدخل إذا كان يقبل القسمة على 6 أو 7.
- ✓ إختبار فيما إذا كان أحد عددين مدخلين من مضاعفات الآخر.
- ✓ طباعة الأعداد التي تقبل القسمة على 6 و 7 معاً والمحصورة بين عددين مدخلين.

2- التمرين الثاني: حساب قيمة تابع

ليكن لدينا التابع المعرف على النحو التالي:

| | | | |
|----|-----|-----|-----|
| Y= | X+1 | حيث | X>0 |
| | 0 | حيث | X=0 |
| | X*X | حيث | X<0 |

المطلوب: أكتب برنامجاً يقوم بحساب قيمة Y من أجل قيمة مدخلة لـ X وطباعته على الشاشة.

```
int x,y;
Console.Write("x= ");
x = Int32.Parse(Console.ReadLine());
if (x > 0)
    y = x + 1;
else if (x == 0)
    y = 0;
else
    y = x*x;
Console.WriteLine("y = "+y);
```

3- التمرين الثالث: حساب تقدير طالب

تقوم الجامعة الافتراضية السورية باحتساب علامات الطلاب في مادة مقدمة في البرمجة على أساس أن الطالب ينفذ وظيفة فصلية ويتقدم لامتحان نهائي، حيث يتم احتساب نسبة من العلامة الكلية 25% للوظيفة و 75% للامتحان وتكون علامه النهائية هي ناتج جمع العلامتين بعد احتساب النسب. المطلوب كتابة برنامج يقوم بإدخال علامات الطالب في الوظيفة والامتحان فيقوم بحساب العلامة النهائية ويمنحه تقديراً وفق الجدول التالي:

| finalMark | grade |
|---------------------|-------|
| finalMark ≥ 90 | A |
| 80 ≤ finalMark < 90 | B |
| 70 ≤ finalMark < 80 | C |
| 60 ≤ finalMark < 70 | D |
| finalMark < 60 | F |

```
const double HomeworkRate = 0.25;
const double ExamRate = 0.75;
double HomeworkMark, ExamMark, FinalMark;
char grade;
HomeworkMark = Double.Parse(Console.ReadLine());
ExamMark = Double.Parse(Console.ReadLine());
if (HomeworkMark < 0 || HomeworkMark > 100 || ExamMark < 0 || ExamMark > 100)
    Console.WriteLine("Invalid Homework Or Exam Marks");
else
{
    FinalMark = HomeworkRate * HomeworkMark + ExamRate * ExamMark;
    if (FinalMark > 90)
        grade = 'A';
    else if (FinalMark >= 80 && FinalMark < 90)
        grade = 'B';
    else if (FinalMark >= 70 && FinalMark < 80)
        grade = 'C';
    else if (FinalMark >= 60 && FinalMark < 70)
        grade = 'D';
    else
        grade = 'F';
    Console.WriteLine("Mark = {0} \t grade = {1}", FinalMark, grade);
}
```

مابعد الحل:

قم بإجراء التعديل اللازم على البرنامج بحيث يقوم بإدخال جميع علامات الصف المؤلف من 50 طالب وحساب تقديراتهم.

4- التمرين الرابع : طباعة أيام الأسبوع

أكتب برنامجاً يعطى رقم الشهر في السنة فيقوم بطباعة اسمه.

```
int month_num;
Console.WriteLine("ENTER A MONTH NUMBER :");
month_num = Int32.Parse(Console.ReadLine());
switch (month_num)
{
    case 1:
        Console.WriteLine("JANUARY"); break;
    case 2:
        Console.WriteLine("FEBRUARY"); break;
    case 3:
        Console.WriteLine("MARCH"); break;
    case 4:
        Console.WriteLine("APRIL"); break;
    case 5:
        Console.WriteLine("MAY"); break;
    case 6:
        Console.WriteLine("JUNE"); break;
    case 7:
        Console.WriteLine("JULY"); break;
    case 8:
        Console.WriteLine("AUGUST"); break;
    case 9:
        Console.WriteLine("SEPTEMBER"); break;
    case 10:
        Console.WriteLine("OCTOBER"); break;
    case 11:
        Console.WriteLine("NOVEMBER"); break;
    case 12:
        Console.WriteLine("DECEMBER"); break;
    default:
        Console.WriteLine("NOT A VALID MONTH NUMBER"); break;
}
```

مابعد الحل:

إذا علمت أن أشهر الربيع هي الأشهر 3-4-5 وأشهر الصيف هي الأشهر 5-6-7-8 وأشهر الخريف هي الأشهر 9-10-11 وأشهر الشتاء هي الأشهر 12-1-2.
قم بإجراء التعديل اللازم على البرنامج بحيث يقوم بطباعة الفصل عند إدخال رقم الشهر.

5- التمرين الخامس : طباعة قواسم عدد صحيح موجب

أكتب برنامجاً يقوم بطباعة قواسم عدد صحيح موجب.

```
uint number;  
Console.WriteLine("ENTER A NUMBER :");  
number = UInt32.Parse(Console.ReadLine());  
for (uint divisor = 1; divisor <= number; divisor++)  
    if (number % divisor == 0)  
        Console.WriteLine(divisor);
```

مابعد الحل:

إن مسألة إيجاد قواسم عدد ما هي مسألة متعددة التطبيقات وتتنوع المسائل التي تستند إلى هذه الفكرة.

قم بإجراء التعديلات اللازمة على البرنامج بحيث يقوم بحل المسائل التالية:

- ✓ حساب مجموع قواسم العدد المدخل.
- ✓ يقال عن العدد أنه تام perfect إذا كان مجموع قواسمه (ماعدا العدد نفسه) يساوي العدد نفسه.
- ✓ طباعة جميع الأعداد التامة المحصورة بين 1 و 100000000.
- ✓ اختبار فيما إذا كان العدد أولياً أم لا.

6- التمرين السادس : إيجاد معكوس رقم

أكتب برنامج لعكس ترتيب الأرقام في عدد صحيح موجب. مثلاً: إذا كان العدد المدخل هو 543 فإنه يطبع 345.

```
Console.WriteLine("A PROGRAM TO REVERSE A NUMBER: ");  
int number;  
Console.WriteLine("ENTER NUMBER TO REVERSE: ");  
number = Int32.Parse(Console.ReadLine());  
int d, n = 0;  
Console.Write(number);  
while (number > 0)  
{  
    d = number % 10;  
    number = number / 10;  
    n = 10 * n + d;  
}  
Console.WriteLine(" AFTER REVERSING IS: {0}", n);
```

مابعد الحل:

يقال عن عدد صحيح ما أنه متناظر palindrome إذا كان يمكن قراءته من اليمين ومن اليسار بنفس الطريقة.
كمثال على الأعداد المتناظرة:

23254845 , 976679 , etc

طور البرنامج في التمرين التاسع عشر بحيث يقوم باختبار عدد مدخل فيما إذا كان متناظراً أم لا.
أعد كتابة البرنامج مجدداً بحيث يقوم بطباعة جميع الأعداد المتناظرة المحصورة بين عددين مدخلين.

7- التمرين السابع : حساب أعداد الناجحين في الامتحان

بفرض أن لديك مجموعة من 10 طلاب، اكتب برنامجاً بلغة C# يستعرض أرقام الطلاب من 1 إلى 10 رقماً رقماً، بحيث يقوم المستخدم من أجل كل طالب بإدخال رقم 1 في حال كان الطالب ناجحاً، وإدخال رقم 2 في حال كان الطالب راسباً، وبحيث يعطي البرنامج في النهاية عدد الناجحين وعدد الراسبين والنسبة المئوية للنجاح.

```
int passes = 0,           // number of passes
    failures = 0,         // number of failures
    student = 1,          // student counter
    result;               // one exam result

// process 10 students; counter-controlled loop
while (student <= 10)
{
    Console.WriteLine("student "+student+" result (1=pass, 2=fail)= ");
    result = Int32.Parse(Console.ReadLine());
    if (result == 1)
        passes = passes + 1;
    else
        failures = failures + 1;
    student = student + 1;
}
// termination phase
Console.WriteLine();
Console.WriteLine("Passed: " + passes);
Console.WriteLine("Failed: " + failures);
Console.WriteLine("Raise Percentage\n" + passes * 10);
```

مابعد الحل:

طور البرنامج السابق بحيث يقوم بإدخال علامات كل طالب في 4 مواد، ثم يقوم بحساب معدل الطالب ويحدد أعداد الناجحين والراسبين بحسب كون المعدل أكبر من 60 أم لا.