



البرمجة الإجرائية Procedural Programming

IPG202

الفصل الثالث تمارين عملية على بنى المعطيات المركبة Practice On Compound Data Types

الكلمات المفتاحية

بنية معطيات مركبة، مصفوفة، مصفوفة أحادية البعد، مصفوفة ثنائية البعد، فهرس، وصول مباشر، foreach، سلسلة محرفية، string.

ملخص الفصل

يتضمن هذا الفصل مجموعة من التمارين التطبيقية على بنى المعطيات المركبة وذلك بهدف تعزيز المفاهيم النظرية التي تم عرضها خلال الفصلين السابقين واكتساب المهارات العملية في التعامل أنماط المسائل التي تتضمن تعاملًا مع مجاميع قيم كبيرة وكتابة الشيفرة البرمجية بلغة C# التي تعبر عن حلول هذه المسائل.

أهداف الفصل

بنهاية هذا الفصل سيكون الطالب قادراً على:

- تعزيز مهاراته في تعريف المصفوفات والسلاسل المحرفية.
- تصميم حلول لمسائل التعامل مع مجموعات قيم كبيرة.
- كتابة الشيفرة البرمجية بلغة C# للمسائل التي تتضمن مصفوفات وسلاسل.
- تنفيذ البرامج واكتشاف الأخطاء وتصحيحها.

محتويات الفصل

1. التمرين الأول : مسألة تقييمات مصفوفة أحادية.
2. التمرين الثاني : مسألة قانون أوم.
3. التمرين الثالث : جدول النسب المثلثية.
4. التمرين الرابع : مسألة حساب مجموع مصفوتين.
5. التمرين الخامس : مسألة المصفوفات المربعة.
6. التمرين السادس : مسألة ترتيب قائمة أسماء الطلاب أجدياً.

1- التمرين الأول: مسألة تقييمات مصفوفة أحادية

ترغب مديرية الأرصاد الجوية بتسجيل معدلات هطولات الأمطار اليومية على مدار عام كامل وإجراء بعض التقييمات عليها.

والمطلوب كتابة برنامج يسمح بمايلي:

- تسجيل الهطولات المطرية اليومية.
- حساب مجموع الأمطار المتساقطة على مدار العام.
- حساب نسبة الأيام الماطرة.

```
//Declaring Storage structure
int[] rain_record = new int[365];

//Declaring other Variables
int sum =0,count_rainy=0;
float rainyDaysAvg;

//Entering and Processing Values
Console.WriteLine("Enter Rain Records Day by Day");
for (int i = 0; i < rain_record.Length; i++)
{
    rain_record[i] = Int32.Parse(Console.ReadLine());
    sum += rain_record[i];
    if (rain_record[i] > 0)
        count_rainy++;
}

//Printing Results
Console.WriteLine("Sum of Year Rain Quantities: " + sum);
rainyDaysAvg = (float)((count_rainy * 100) / rain_record.Length);
Console.WriteLine("Percent of Rainy Days: " + rainyDaysAvg);
```

2- التمرين الثاني: مسألة قانون أوم

أكتب برنامجاً يستخدم ثلاث مصفوفات أحادية البعد CURRENT ، RESISTANCE ، VLOT لتخزين قيم التيارات، المقاومات، الجهود على التوالي.

قم بإدخال قيم للمصفوفتين CURRENT ، RESISTANCE ومن ثم حساب قيم المصفوفة VOLT حيث أن:
 $VOLT[i] = CURRENT[i] * RESISTANCE[i]$

قم بطباعة قيم المصفوفات الثلاث.

```
//Declaring Storage structure
float[] current = new float[10];
float[] resistance = new float[10];
float[] volt = new float[10];

//Entering and Processing Values
for (int i = 0; i < 10; i++)
{
    Console.Write("Current Value =");
    current[i] = float.Parse(Console.ReadLine());
    Console.Write("Resistance Value =");
    resistance[i] = float.Parse(Console.ReadLine());

    //Calculating and storing volt value
    volt[i] = current[i] * resistance[i];
}

//Printing Results
Console.Write("Resistances : ");
foreach (float r in resistance)
    Console.Write(r + "\t");
Console.WriteLine();

Console.Write("Currents : ");
foreach (float c in current)
    Console.Write(c + "\t");
Console.WriteLine();

Console.Write("Volts : ");
foreach (float v in volt)
    Console.Write(v + "\t");
Console.WriteLine();
```

3- التمرين الثالث: جدول النسب المثلثية

نرغب بإنشاء جدول لتخزين النسب المثلثية للزوايا المحصورة بين 0 و 90 درجة، ومن ثم طباعته على الشاشة. الجدول يجب أن يكون من الشكل:

| x | Sin(x) | Cos(x) | Tan(x) |
|-------|-------------------|-------------------|-------------------|
| 0 | 0 | 1 | 0 |
| 1 | 0.841470984807897 | 0.54030230586814 | 1.5574077246549 |
| | | | |
| | | | |
| 90 | 0.893996663600558 | -0.44807361612917 | -1.99520041220824 |

ملاحظة: استقد من توابع الصنف مسبق التعريف Math.

```
//Decalring Storage structure
double[,] ratios = new double[91,4];

//Cakculating and storing ratios
for (int i = 0; i < 91; i++)
{
    ratios[i,0] = i;
    ratios[i,1] = Math.Sin(i);
    ratios[i,2] = Math.Cos(i);
    ratios[i,3] = Math.Tan(i);
}

//Displaying Stored Values
Console.WriteLine("x \t\t Sin(x) \t\t Cos(x) \t\t Tan(x)");
for (int i = 0; i < 91; i++)
    Console.WriteLine("{0} \t\t {1} \t\t {2} \t\t {3}", ratios[i, 0], ratios[i, 1], ratios[i, 2], ratios[i, 3]);
```

4- التمرين الرابع : مسألة حساب مجموع مصفوفتين

يمكن أن يتم جمع مصفوفتين من خلال إنتاج مصفوفة يكون فيها قيمة كل عنصر هي ناتج جمع القيمتين المتقابلتين كما يبين المثال التالي:

$$\begin{array}{ccc}
 3 & 5 & 5 \\
 4 & 2 & 1 \\
 5 & 0 & 2 \\
 1 & 2 & 4
 \end{array}
 +
 \begin{array}{ccc}
 3 & 0 & 7 \\
 2 & 4 & 2 \\
 7 & 1 & 1 \\
 1 & 2 & 0
 \end{array}
 =
 \begin{array}{ccc}
 6 & 5 & 12 \\
 6 & 6 & 3 \\
 14 & 1 & 3 \\
 2 & 4 & 4
 \end{array}$$

حيث أن المصفوفات الثلاث يجب أن تكون لها نفس الأبعاد.
أكتب برنامجاً يقوم بإدخال قيمتي مصفوفتين وإيجاد المصفوفة المعبرة عن ناتج جمعهما، وطباعة النتائج.

```
//Decalring Storage structure
int[,] mat1 = new int[4,3];
int[,] mat2 = new int[4, 3];
int[,] mat3 = new int[4, 3];

//Entering the 2 arrays and calculating sum
for (int i = 0; i < 4; i++)
    for (int j = 0; j < 3; j++)
    {
        Console.Write("mat1[ " + i + " , " + j + " ]= ");
        mat1[i, j] = Int32.Parse(Console.ReadLine());
        Console.Write("mat2[ " + i + " , " + j + " ]= ");
```

```

        mat2[i, j] = Int32.Parse(Console.ReadLine());
        mat3[i, j] = mat1[i, j] + mat2[i, j];
    }

    //printing arrays
    Console.WriteLine("Mat1 =");
    foreach (int row in mat1)
        Console.WriteLine(row + " ");
    Console.WriteLine();

    Console.WriteLine("Mat2 =");
    foreach (int row in mat2)
        Console.WriteLine(row + " ");
    Console.WriteLine();

    Console.WriteLine("Mat3 =");
    foreach (int row in mat3)
        Console.WriteLine(row + " ");
    Console.WriteLine();

```

5- التمرين الخامس : مسألة المصفوفات المربعة

تفترض هذه المسألة وجود مصفوفة مربعة من القيم الصحيحة ذات حجم معين حيث يطلب تنفيذ العمليات التالية:

- 1- إدخال القيم لعناصر المصفوفة.
- 2- طباعة قيم عناصر المصفوفة سطراً سطراً.
- 3- حساب مجموع قيم عناصر القطر الرئيسي.
- 4- إيجاد القيمة العظمى في القطر الثانوي.
- 5- اختبار فيما إذا كانت المصفوفة متناظرة.

ملاحظات الحل:

- المصفوفة المربعة هي مصفوفة ثنائية البعد عدد أسطرها وأعمدتها متساويين.
- في حال كان لدينا مصفوفة مربعة بعدها $n \times n$ فإن:
- ✓ عناصر القطر الرئيسي هي عناصر يتساوى فيها فهرس السطر مع فهرس العمود.
 - ✓ عناصر القطر الثانوي هي عناصر يكون فيها مجموع قيمتي فهرس السطر والعمود يساوي $n-1$.
 - ✓ تكون المصفوفة المربعة متناظرة إذا كان قيمة كل عنصرين متقابلين بالنسبة للقطر الرئيسي متساويين (أي أن $x[i,j] = x[j,i]$).

```

//DEclaring Variables
int[,] MAT = new int[4, 4];
int i,j;

//Entering Array Elements
for (i=0;i<4;i++)
    for (j=0;j<4;j++)
    {
        Console.Write("MAT( "+i+" , "+j+" )=");
        MAT[i,j] =Int32.Parse(Console.ReadLine());
    }

//Printing Array Elements
for (i = 0; i < 4; i++)
{
    for (j = 0; j < 4; j++)
        Console.Write(MAT[i, j] + " ");
    Console.WriteLine();
}

//Finding Sum
int sum_diag=0;
for (i=0;i<4;i++)
    for (j=0;j<4;j++)
        if (i==j)
            sum_diag+=MAT[i,j];
Console.WriteLine("MAIN DIAG SUM IS : " + sum_diag);

//Finding Max
int max=0;
for (i=0;i<4;i++)
    for (j=0;j<4;j++)
        if (i+j==3)
            if ( MAT[i,j]>max)
                max=MAT[i,j];
Console.WriteLine("SECOND DIAG MAX IS : " + max);

//Testing Symetric
bool symetric=true;
for (i=0;i<4;i++)
    for (j=0;j<4;j++)
        if (MAT[i,j]!=MAT[j,i])
            symetric=false;
if (symetric==true)
    Console.WriteLine("ARRAY IS SYMETRIC");
else
    Console.WriteLine("ARRAY IS ASSYMETRIC");

```

6- التمرين السادس : مسألة ترتيب قائمة أسماء الطلاب أبجدياً

نرغب بإدخال أسماء الطلاب في الصف ومن ثم طباعة قائمة باسمائهم مرتبة أبجدياً.

```
//Declaring Storage strutcure
string[] students = new string[10];

//Entering Students Names
for (int i=0;i<students.Length;i++)
    students[i] =Console.ReadLine();

//Sorting Students Names
string temp;
for (int pass=1;pass<students.Length;pass++)
    for(int i=0;i<students.Length-1;i++)
        if (students[i].CompareTo(students[i+1])>1)
        {
            temp = students[i];
            students[i] = students[i + 1];
            students[i + 1] = temp;
        }

//Printing Students Names
for (int i = 0; i < students.Length; i++)
    Console.WriteLine(students[i]);
```