

# جزوه درس برنامه سازی یک

## C#

تهیه کننده: محمدی



```
Program.cs* -p X
ConsoleApplication3.Program
1 using System;
2
3 namespace ConsoleApplication3
4 {
5     class Program
6     {
7         static void Main(string[] args)
8         {
9             Console.WriteLine("Hello C#!");
10            Console.ReadKey();
11        }
12    }
13 }
```

بهار ۹۴

**فصل اول ( مفاهیم پایه ای پردازش داده ها**

**داده ها (Data):** مجموعه ای از مقادیر در مورد یک موضوع یا شیء است که به صورت کمی با یک مقدار عددی و یا به صورت کیفی نشان داده می شود پس داده ها، مقادیر خام و اولیه در مورد یک موضوع هستند. مانند نمرات یک کلاس

**پردازش (Process):** برای اینکه از داده ها بتوانیم بهتر استفاده کنیم لازم است بر روی آنها محاسبات و یا به طور کلی عملیاتی را انجام دهیم. پس مجموعه محاسبات و عملیاتی که بر روی داده ها صورت می گیرد را پردازش می نامند. ( کامپیوتر پردازشگر داده ها است) مانند جمع و ضرب

**اطلاعات (Information):** نتایج حاصل از عملیات و محاسبات بر روی داده ها را اطلاعات می نامیم که می تواند مورد تفسیر و بررسی قرار گیرد.

و نتیجه بررسی آنها به دانش ختم گردد که دانش می تواند مبنای تصمیم گیری برای انجام کاری شود.



مثال : دانشجویانی که معدل آنها کمتر از ۱۲ باشد مشروط می باشند و در ترم بعد تعداد محدودی واحد درسی می توانند اخذ نمایند. نمرات دانشجویان را وارد سیستم می کنند(داده ها) و پس از انجام محاسبات (پردازش)، معدل دانشجو محاسبه می شود(اطلاعات). با استفاده از معدل می توان وضعیت دانشجو (دانش) را مشاهده نمود.

در فرایند رسیدن از داده ها به اطلاعات، نکات زیر باید رعایت گردد:

- ۱- صحت داده ها: یعنی داده های ورودی به درستی گردآوری شوند. مثلاً نمره دانش آموز نمی تواند عدد منفی باشد.
  - ۲- درستی انجام محاسبات: یعنی محاسبات و یا به طور کلی عملیاتی که بر روی داده ها صورت می گیرد با دقت و بدون اشتباه انجام شوند. در صورت اشتباه در محاسبات، اطلاعات اشتباه بدست می آید.
  - ۳- روش انجام پردازش: با توجه به هدفی که از گردآوری داده ها در نظر داریم باید پردازش مناسب نیز بر روی آنها انجام دهیم تا به اطلاعات مفید برسیم. استفاده از روش های بهینه و الگوریتم های مناسب در عمل پردازش توصیه می شود.
- برنامه :** به مجموعه دستوراتی که به کامپیوتر می فهماند که چه نوع پردازشی را بر روی داده ها انجام دهد و همچنین اطلاعات به دست آمده را چگونه نمایش دهد برنامه گویند.

**زبان ماشین (Machine Language):** زبان قابل فهم کامپیوتر دنباله ای از کدهای صفر و یک است.

**مترجم (Compiler):** جهت ترجمه و تبدیل برنامه ای غیر از زبان ماشین (سطح بالا) به زبان ماشین استفاده می شود. مترجم خود یک برنامه کامپیوتری است که وظیفه آن ترجمه ی برنامه نوشته شده به یک زبان، به کدهای زبان ماشین است.

**برنامه نویسی (Programming):** نوشتن دستورات لازم برای کنترل نحوه کار کامپیوتر، به طوری که کامپیوتر بتواند یک کار مشخص را انجام دهد.

**برنامه نویس (Programmer):** شخصی است که آشنا به دستورات یک زبان برنامه نویسی باشد و با به کارگیری صحیح و مناسب دستورات، برنامه نویسی کند.

**انواع زبان های برنامه نویسی :**

زبان های برنامه نویسی را از نظر این که تا چه اندازه به زبان محاوره ای انسان نزدیک باشند، به صورت زیر دسته بندی می کنند:

**زبان های سطح پایین (Low Level Language):** زبان هایی که به زبان پردازشگر کامپیوتر (CPU) نزدیک باشد و از زبان محاوره ای ما دور هستند مانند زبان ماشین و زبان اسمبلی  
 نکته) زبان ماشین برای اجرا توسط کامپیوتر نیاز به مترجم ندارد.

**زبان های سطح بالا (High Level Language):** زبان برنامه نویسی که به زبان محاوره ای ما نزدیک باشد، مانند VB، Java و C#  
**زبان های سطح میانی (Medium Level Language):** در این زبانها، دستوراتی برای دسترسی راحت تر به سخت افزار پیش بینی

شده باشد و همچنین به زبان عامیانه نزدیک باشند، مانند زبان C.  
 برنامه نویسان از این زبان ها برای کار با سخت افزار کامپیوتر و برنامه ریزی وسایلی که در آنها پردازشگر وجود دارد استفاده می کنند.  
 سوال) با وجود نرم افزارهای آماده مانند مجموعه ی Office، دلیل یادگیری یک زبان برنامه نویسی چیست؟

**تمرین:** عبارات ذکر شده در جدول ستون سمت راست را با گزینه های عبارات سمت چپ تکمیل کنید؟

۱	نتایج حاصل از عملیات و محاسبات بر روی داده ها	<input type="checkbox"/>	الف	زبان ماشین
۲	مقادیر خام و اولیه در مورد یک موضوع	<input type="checkbox"/>	ب	اطلاعات
۳	به کارگیری صحیح و مناسب دستورات	<input type="checkbox"/>	ج	داده ها
۴	زبان قابل فهم سخت افزار کامپیوتر،	<input type="checkbox"/>	د	برنامه نویسی

**تست**

- ۱) زبان برنامه نویسی C# در طبقه بندی سطح زبان ..... قرار گرفته است.  
 الف) پایین (ب) میانی (ج) بالا (د) هیچکدام  
 ۲) در یک برنامه، منظور از ورودی همان ..... و منظور از خروجی همان ..... است.  
 الف) اطلاعات - دانش (ب) اطلاعات - داده (ج) دانش - داده (د) داده - اطلاعات  
 ۳) زبانهایی که به زبان پردازشگر کامپیوتر (cpu) نزدیک باشند زبانهای سطح ..... گفته می شوند.  
 الف) پایین (ب) میانی (ج) بالا (د) الف و ب  
 ۴) در فرایند رسیدن داده به اطلاعات کدام گزینه رعایت نمی شود؟  
 الف) صحت دادهها (ب) درستی انجام محاسبات (ج) خروجی اطلاعات (د) روش انجام پردازش  
 ۵) تصمیم گیری برای انجام یک کار بر مبنای ..... است.  
 الف) داده (ب) اطلاعات (ج) دانش (د) پردازش

## فصل ۲) آشنایی با زبان C#

زبان سی شارپ یک زبان سطح بالا، شی گرا (Object Oriented Language - OOP) و همه منظوره است که به وسیله شرکت مایکروسافت هم زمان با پیدایش لایه نرم افزاری جدید آن به نام NET. ابداع و توسعه پیدا کرده است.  
 این زبان شباهت زیادی به زبان های C و Java دارد و برخلاف زبان C، در هنگام ترجمه و همچنین اجرای برنامه دقت زیادی بر روی تطبیق و بکارگیری داده ها صورت می گیرد. زبان سی شارپ در انحصار شرکت مایکروسافت می باشد.  
 سیستم عامل UNIX با زبان C نوشته شده است. زبان C، یک زبان حرفه ای است که دسترسی به سخت افزار را برای برنامه نویس امکان پذیر می کند و دارای انعطاف بسیار زیادی است به همین دلیل کمتر اشتباهات منطقی برنامه نویس را کنترل می کند(برخلاف زبان C#)

سی شارپ توسط تیمی به مدیریت اندرز هایلزبرگ که قبلاً تجربه ارائه زبان های برنامه نویسی موفقى همچون توروبو پاسکال و دلفى رو داشت، ایجاد شد.

این زبان برنامه نویسی نسخه کامل شده C و ++C می باشد، درمجموعه برنامه های NET می باشد و محیطى بصرى دارد، ساده تر از ++C می باشد، اما قدرتمند.

از ویژگی های این زبان، قابلیت دسترسی مستقیم به حافظه و امنیت می باشد.

می توان برای طراحی برنامه های کاربردى، یعنی مبتنى پرویندوز و برنامه های مبتنى بر وب و سرویس های وب از آن استفاده نمود.

**زبان Java** : متن باز (Open Source) می باشد و بر پایه زبان C می باشد. در ابتدا نام این زبان Oak بود. شعار این زبان " یکبار بنویس و هر جا اجرا کن " می باشد.

به برنامه هایی که کدشان توسط سازندگان در اختیار همه قرار می گیرد تا به دلخواه خود آن را تغییر دهند، متن باز گفته می شود.

### لایه نرم افزارى NET Framework . شرکت مایکروسافت:

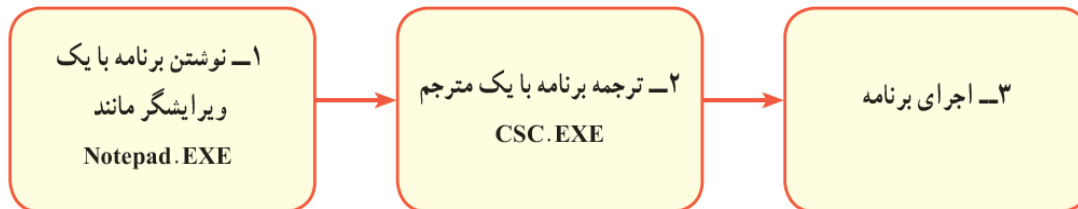
حاوی حجم عظیمی از کدهای نوشته شده به زبان ویژوال می باشد. مثلاً برای نوشتن پنجره ساده باید هزاران کد نوشت که آنها داخل این کتابخانه می باشد. به همین دلیل حجم برنامه های اجرایی NET کم می باشد و برای اجرای فایل های exe حتماً باید این کتابخانه روی سیستم نصب باشد.

برنامه های تولید شده برای دات نت فریم ورک توسط Common Language Runtime اجرا می شوند.

دات نت دارای مترجمی به نام CSC است که کدهای C# را به کدهای زبان ماشین ترجمه می کند.

نکته : زبان C# مانند زبان های C و Java نسبت به حروف کوچک و بزرگ حساس است.

### مراحل نوشتن و ترجمه یک برنامه با استفاده از مترجم دات نت:



مراحل برنامه نویسی، ترجمه و اجرا

```

using System;
class Program
{
    static void Main()
    {
        Console.WriteLine("HELLO!!!");
    }
}
    
```

### یک نمونه برنامه:

این برنامه را در نرم افزار Notepad با دقت تایپ کنید.

سپس این برنامه را با یک نام دلخواه و با پسوند CS ذخیره کنید (مثلاً

test.cs) سپس با مترجم CSC در محیط cmd آن را ترجمه کنید

تا فایل اجرایی test.exe ساخته شود و آن را اجرا کنید.

نکته) در صورت نصب NET . فایل csc.exe را در می توان در مسیر

C:\Windows\Microsoft.NET\Framework\v4.0.30319 مشاهده کرد.

سوال) چگونه مسیر فایل csc.exe را به path ویندوز اضافه کنیم؟

نکته) در صورتی که برنامه نوشته شده دارای خطا باشد توسط مترجم اعلام می شود. به عنوان مثال برنامه زیر ترجمه شده است و خطای آن را مشاهده می کنید.

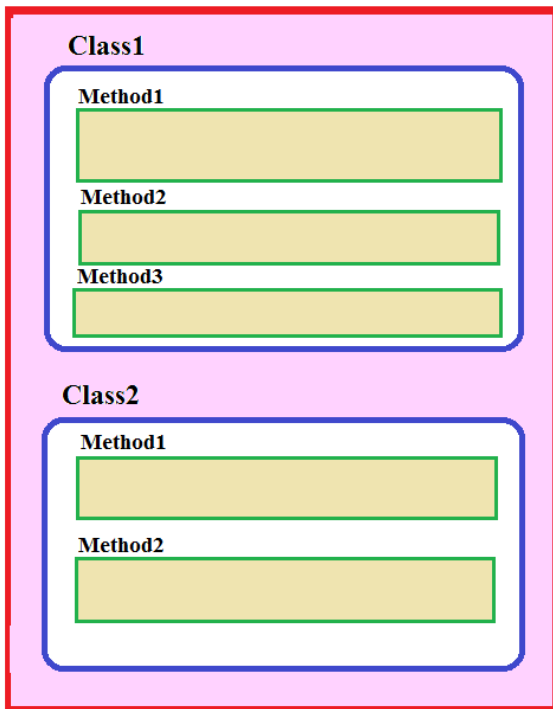
```
using System;
class Program
{
    static void Main()
    {
        Console.WriteLine("HELLO!!!")
        Console.WriteLine("Programing");
    }
}
```

```
F:\>csc test.cs
Microsoft (R) Visual C# Compiler version 4.0.30319.33
for Microsoft (R) .NET Framework 4.5
Copyright (C) Microsoft Corporation. All rights reserved.
test.cs(6,33): error CS1002: ; expected
```

الگوی یک برنامه ساده به زبان C#

یک برنامه کاربردی نوشته شده به زبان C#، شامل مجموعه ای از کلاس ها است که هر یک آنها نیز شامل تعدادی متد هستند.

Namespace



```
class { یک نام دلخواه
    static void Main()
    {
        دستورات مربوط به انجام یک کار
    }
}
```

نکته) بعضی از کلمات برای زبان C# شناخته شده هستند که به آنها کلمات کلیدی یا رزرو شده می گویند. مانند `class` , `static` , `void` و غیره

نکته) نام هایی که برنامه نویس باید انتخاب کند( مانند نام یک کلاس) شناسه گفته می شوند و انتخاب این نام ها دارای ضوابطی است.

**کلاس** : (Class) در واقع یک ایده (Concept) یا ذهنیت می باشد تعریفی از ویژگی ها و خصوصیات و کارایی یک شیء است. کلاس به عنوان یک قالب یا الگویی می باشد که در آن داده هایی تعریف می شود. این داده ها مربوط به یک موضوع است و عملیاتی که می توان بر روی آنها انجام داد. هر کلاس از تعدادی داده و تعدادی متد می تواند تشکیل شود مانند متد `Main` برای تعریف کلاس از کلمه کلیدی `class` استفاده می شود.

```
class نام کلاس {
```

تعریف داده ها و عملیات بر روی آنها

```
}
```

توجه : در نام کلاس نباید از فاصله استفاده نمود( نام کلاس یک شناسه است) بهتر است جهت خوانایی بیشتر نام گذاری با معنی باشد و از روش پاسکال برای نامگذاری استفاده شود( اولین حرف هر کلمه با حرف بزرگ باشد)

**نکته** در زبان C# تعداد زیادی کلاس از قبل آماده وجود دارد که در هر کدام نیز تعداد زیادی متد آماده شده است. مثلا کلاس Console برای عملیات ورودی و خروجی ( بر روی صفحه نمایش و یا صفحه کلید) با تعداد زیادی متد تعریف شده است.

**فضای نام (NameSpace):** برای سازماندهی کتابخانه کلاس ها و قراردادن آنها در یک آرایش مخصوص جهت دسترسی آسان استفاده می شود، همچنین برای جلوگیری از تداخل نام ها در کلاس بهتر است نام کلاس و NameSpace یکسان باشد. از دستور **using** برای شناساندن فضای نام به برنامه استفاده می شود.

**شیء (Object):** نمونه ای از کلاس که ساخته (متولد) می شود.

**متد (Method):**

متد مجموعه ای از دستورات است که برای انجام یک کار لازم است. هر متد مطابق با عملکردش نام گذاری می شود و همچنین دارای یک جفت پرانتز باز و بسته است که در آن ممکن است ورودی هایی ذکر شود که برای انجام کار لازم است.

در برنامه های زبان C#، ممکن است متدهای زیادی تعریف و یا مورد استفاده قرار گیرند، اما حتما باید متدی به نام Main() تعریف شده باشد که نقطه آغاز اجرای برنامه است و اجرای یک برنامه از اولین دستور داخل آن شروع می شود.

در سی شارپ متد ها داخل کلاس تعریف می شود و کلاس هم داخل فضای نام ( namespace ) تعریف می شود. برای دسترسی به متد خاص باید به صورت زیر عمل کنیم :

نام متد . نام کلاس . نام فضای نام

بعنوان مثال System.Console.WriteLine متد WriteLine را که در کلاس Console واقع در فضای نام System می باشد فراخوانی می کند.

این متد برای نمایش اطلاعات استفاده می شود. فرض کنید می خواهیم کلمه Computer در خروجی نمایش داده شود. برای این کار می نویسیم:

```
System.Console.WriteLine("Computer");
```

متد WriteLine حتما نیاز به پرانتز دارد و عبارت Computer بدلیل اینکه یک ثابت رشته ای می باشد حتما باید بین علامت "" قرار بگیرد.

**نکته :** در صورتی که بخواهیم در برنامه توضیحی را بنویسیم می توانیم از علامت // یا /\* \*/ استفاده کنیم. در صورتی که توضیح فقط یک سطر باشد کافی است ابتدای خط علامت // و در صورتی که چند سطر باشد باید بین دو علامت /\* \*/ قرار بگیرد. لازم به ذکر است توضیحات فقط مخصوص برنامه نویس می باشد و ترجمه و اجرا نمی شوند.

**ترجمه و اجرای برنامه :**

برای اجرای برنامه می توان هم از خط فرمان برنامه را کمپایل و اجرا نمود و هم از طریق محیط ویژوال استودیو این کار را انجام داد.

برای این کار در خط فرمان (Command Prompt) از فایل csc.exe برای کمپایل و اجرای برنامه استفاده می کنیم

نام فایل مورد نظر CSC

توجه داشته باشید که فایل های کد سی شارپ دارای پسوند CS می باشند.

**چند متد از کلاس Console:**

**WriteLine:** برای نمایش اطلاعات بکار می رود. این متد داده ها را نمایش می دهد و سپس چشمک زن (Cursor) را به خط بعد منتقل می کند.

```
System.Console.WriteLine("Hi");
```

```
System.Console.WriteLine(125*2);
```

**Write:** این متد مانند WriteLine است ولی مکان نما به خط بعد منتقل نمی شود.

```
System.Console.Write (15+4);
```

**Clear**: برای پاک کردن صفحه خروجی کنسول بکار می رود.

```
System.Console.Clear();
```

**ReadKey**: برای توقف برنامه تا زمانی که کاربر یک کلید را فشار دهد.

```
System.Console.ReadKey();
```

**BackgroundColor**: برای تغییر رنگ زمینه کنسول بکار می رود. رنگ ها در کلاس `ConsoleColor` تعریف شده اند.

```
System.Console.BackgroundColor=System.ConsoleColor.Red;
```

**ForegroundColor**: برای تغییر رنگ متن(قلم) کنسول بکار می رود. رنگ ها در کلاس `ConsoleColor` تعریف شده اند.

```
System.Console.ForegroundColor= System.ConsoleColor.Blue;
```

نکته) ۱۶ رنگ در کلاس `ConsoleColor` وجود دارد.

**دستور using**: برای جلوگیری از نوشتن نام فضای نام قبل از کلاس می توان فضای نام را با این دستور در بالای برنامه تعریف نمود تا دیگر نیازی

به نوشتن فضای نام ها نباشد. بعنوان مثال اگر در بالای برنامه دستور `using System;` را بنویسیم می توانیم برای نمایش اطلاعات کلمه

`System` را دیگر ننویسیم.

```
Console.WriteLine(20/3);
```

پرسش) دستور `Console.WriteLine(20.0/3);` را اجرا کنید. حاصل چیست؟

نکته) یک رشته بین دو علامت نقل قول "" قرار می گیرد که از تعدادی کاراکتر تشکیل می شود. "ALI129"

پرسش) دستور `Console.WriteLine("15+4");` را اجرا کنید. حاصل چیست؟

تمرین) برنامه زیر را در محیط Notepad نوشته و سپس ترجمه و اجرا کنید. چه نتیجه ای می گیرید؟

```
using System;
class Program
{
    static void Main()
    {
        Console.BackgroundColor=ConsoleColor.Blue;
        Console.Write("Press a Key....!!!");
        Console.ReadKey();
        Console.Clear();
        Console.ForegroundColor=ConsoleColor.Black;
        Console.WriteLine("{0} {2} {1}",17,20,25);
    }
}
```

```
Console.WriteLine("{0}*{1}={2}",5,4,5*4);
```

```
Console.WriteLine("{0}*{0}={1}", 4, 4 * 4);
```

تمرین) کلمه ی IRAN را به سه رنگ سبز و سفید و قرمز بر روی صفحه نمایش چاپ کنید.

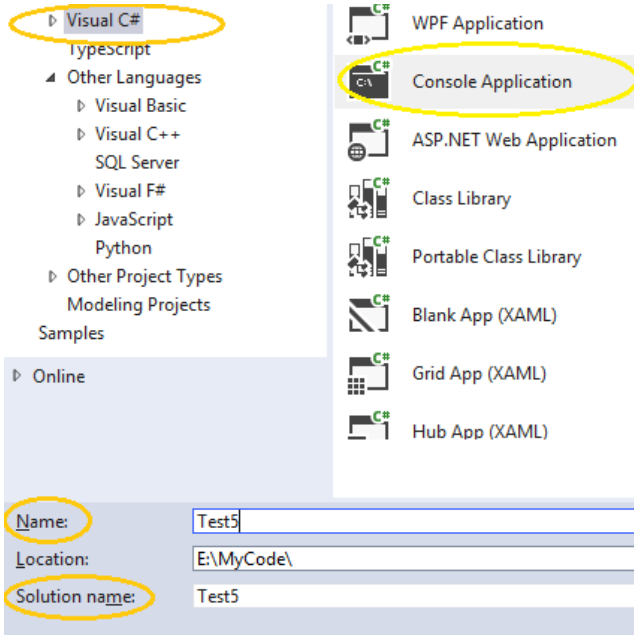
تست) کدامیک از دستورات زیر صحیح است و محاسبه ریاضی انجام می دهد؟

الف) `Console.WriteLine("2+2");` ب) `.WriteLine(2+2);`

ج) `Console.WriteLine("2" + 2);` د) `Console.WriteLine(2+2);`

## فصل ۳) آشنایی با ویژوال استودیو

برای نوشتن برنامه، نیاز به محیطی داریم که همه ابزارها و لوازم مورد نیاز برنامه نویسی در آن گردآوری و متمرکز شده باشد. به چنین محیط برنامه نویسی که در آن می توان تمام مراحل برنامه نویسی، ترجمه، اشکال یابی و متمرکز می باشد. IDE گفته می شود که به معنای محیط تولید برنامه متمرکز است. شرکت مایکروسافت یک IDE بسیار پیشرفته برای برنامه نویسی فراهم کرده است که با کمک آن می توان راحت تر برنامه بنویسیم و ترجمه و اجرا کنیم. VS از چند زبان برنامه نویسی VB، C++، C# پشتیبانی می کند.



برای ایجاد یک برنامه جدید به زبان C#، گزینه New Project را انتخاب می کنیم. و سپس نام راه حل را انتخاب می کنیم.

توجه) در این درس برای ایجاد پروژه از گزینه ی Console Application استفاده می کنیم.

**Solution** (راه حل): در اصطلاح برنامه نویسی شامل چندین پروژه از انواع مختلف می باشد.

راه حل بالاترین سطح دسته بندی در محیط ویژوال استودیو دات نت است. اصولاً وظیفه Solution نگهداری اطلاعات روابط بین پروژه هاست که یک فایل با پسوند sln می باشد و دارای تاثیر در خروجی پروژه خواهد بود.

به هنگام ایجاد یک برنامه باید یک Solution را ایجاد شود. معمولاً نام راه حل با نام پروژه یکسان می باشد.

## معرفی بخش های اصلی ویژوال استودیو :

الف) پنجره ویرایشگر برنامه

ب) پنجره لیست خطاها (Error List): اشکال های تایپی و ساختاری را نمایش می دهد.

ج) پنجره Solution Explorer (مرورگر پروژه): ساختار پروژه و تمام فایل های برنامه را نشان می دهد. چند نکته):

۱- کلمات کلیدی در IDE با رنگ آبی نشان داده می شوند.

۲- فایل متن برنامه به نام program.cs است و می توان نام آن را تغییر داد.

۲- اگر دستوری از لحاظ املایی یا گرامری اشکال داشت خطوط قرمز رنگی زیر آن کشیده می شود.

۳- برای ترجمه برنامه از کلید F6 و برای ترجمه و اجرا کلید F5 یا CTRL+F5 را فشار دهید.

۴- متد ReadKey در کلاس Console جهت گرفتن کلید از صفحه کلید بکار می رود.

۵- متد Beep جهت ایجاد یک صدا با فرکانس خاص بکار می رود و می توان به دو شکل از آن استفاده نمود. (Overload)

Console.Beep();

صدا با فرکانس ۱۴۰۰ هرتز و در مدت ۲ ثانیه پخش می کند Console.Beep(1400 , 2000 );

Console.Beep (مدت زمان برحسب میلی ثانیه ، فرکانس برحسب هرتز )

سؤال) برنامه ای بنویسید که نام و نام خانوادگی خودتان را به زبان فارسی در صفحه نمایش نشان دهد.

Console.WriteLine("100+200/2");

تست) خروجی دستور روبرو چیست؟

د) 100+200/2

ج) 200

ب) 150

الف) 100



## فصل چهارم) آشنایی با انواع داده ها و متغیرها

متغیر (Variable): مکانی از حافظه RAM کامپیوتر است که برای نگهداری موقتی داده ها یا اطلاعات استفاده می شود.

## نام متغیر نوع داده

قبل از استفاده از متغیر باید آن را تعریف نماییم. شکل تعریف متغیر بدین صورت است:

نوع داده یا متغیر بطور کلی سه ویژگی را مشخص می کند:

اندازه متغیر (بر حسب بایت): هر متغیر با توجه به نوع داده اش مقداری از حافظه ی RAM را مصرف می کند.

- نوع اطلاعاتی که در متغیر می توان ذخیره کرد: مثلا صحیح یا اعشاری یا رشته ای یا .....
- نوع عملیاتی را می توان بر روی آن انجام داد. مثلا بر روی اعداد صحیح می توان عمل ضرب را انجام داد و بر روی رشته ها ضرب تعریف نشده است.

در جدول زیر انواع داده ها و ویژگی های آنها ذکر شده است:

نوع داده	کاربرد نوع داده	مقدار حافظه (بایت)	کمترین مقدار	بیشترین مقدار
sbyte	اعداد صحیح	1	-128	127
byte	اعداد صحیح مثبت	1	0	255
short	اعداد صحیح	2	-32768	32767
ushort	اعداد صحیح مثبت	2	0	65535
int	اعداد صحیح	4	-2147483648	2147483647
uint	اعداد صحیح مثبت	4	0	4294967295
long	اعداد صحیح	8	-9223372036854778508	9223372036854778507
ulong	اعداد صحیح مثبت	8	0	18446744073709551615
float	اعداد اعشاری	4	$-3.402823 \times 10^{38}$	$3.402823 \times 10^{38}$
double	اعداد اعشاری با دقت زیاد	8	$-1.79769313486232 \times 10^{308}$	$1.79769313486232 \times 10^{308}$
decimal	اعداد صحیح بزرگ اعداد اعشاری با دقت بسیار زیاد	16	$-79228162514264337593543950335$ $-7.9 \times 10^{28}$	$79228162514264337593543950335$ $+7.9 \times 10^{28}$
bool	مقدار منطقی	1	false	true
char	یک حرف یا علامت (کراکتر)	2	0 کد کراکتر مطابق با سیستم Unicode	65535 کد کراکتر مطابق با سیستم Unicode
string	رشته		-	-
object	آدرس یک داده		-	-

عددی

غیر عددی

مثال:

```
int a, b;          int c,d=11;          int a, b = 7,c=b;          short a=b=c; // syntax error
int a, b = 7,c=a; // syntax error
```

```
byte x = -10; // syntax error
int a=-10, b=7,c=a; byte x = a; // syntax error
byte x=300; // syntax error
uint B,b;
```

نکته) برنامه نویس با توجه به مساله ایی که حل می کند، برای متغیرها نوع داده ی مناسب را انتخاب می کند. مثلا برای نگهداری نمره یک درس چه نوع داده ایی مناسب تر است؟ یا مثلا برای تعیین وضعیت خاموش یا روشن بودن یک لامپ کدام نوع داده مناسب تر است؟

نکته) هر متغیر فقط یک مقدار می تواند در خود نگه دارد و با دستور انتساب این کار انجام می شود. `a=15;`

نکته) نوع داده `float` برای داده های اعشاری با دقت حداکثر ۷ رقم اعشار استفاده می شود و در صورتی که بیشتر از ۷ رقم باشد گرد می شود و نوع داده `double` برای داده های بسیار بزرگ یا بسیار کوچک بکار می رود و با دقت ۱۵ رقم اعشار می باشد.

سوال) خروجی برنامه روبرو چیست؟

```
float a;
a = 16.274138214f;
Console.WriteLine(a);
```

```
float sum;
byte p = 0xE4; // 0XE4
sum = 14.25f;
Console.WriteLine("p=" + p);
```

نکته : در زبان سی شارپ اگر ابتدای عدد صحیح 0X قرار گیرد عدد در مبنای ۱۶ در نظر گرفته می شود. به مثال روبرو توجه کنید:

نوع عدد	نشانه	مثال
عدد صحیح مثبت	u یا U	125U
عدد صحیح بزرگ	l یا L	1700L
عدد صحیح بزرگ مثبت	UL	250000UL
عدد اعشاری با دقت معمولی	f یا F	2.5f
عدد اعشاری با دقت زیاد	d یا D	12.75d
عدد بسیار بزرگ	m یا M	12345678M

جدول زیر نشانه های انواع داده ها را نشان می دهد:

نکته) اگر در برنامه یک عدد اعشاری بدون نشانه بنویسید بطور پیش فرض عدد اعشاری با دقت زیاد (double) در نظر گرفته می شود. به مثال ها دقت کنید:

`float a=12.51;`  
این دستور خطای گرامری دارد زیرا داده های ثابت اعشاری به طور پیش فرض، `double` می باشد

پس می توان یکی از دو دستور زیر را بنویسید:

```
float a=12.51f;                    یا                    double d=12.51;
```

نکته) بطور کلی در زبان سی شارپ امکان انتساب یک نوع داده بزرگتر به نوع داده ی کوچکتر خطا است. در برنامه زیر متغیر `s` از نوع `integer` است و متغیر `h` از نوع `byte` است. خط چهارم خطا است. چرا؟

```
byte h=10;
s = h;
h = s;
```

آیا راه حلی برای برطرف کردن این خطا وجود دارد؟

نکته) می توان اعداد را بصورت نماد علمی (نقطه شناور) نیز نمایش داد. در روش نماد علمی عدد از دو بخش تشکیل می شود. بخش اول یک عدد تک رقمی صحیح بین ۰ تا ۹ است که به آن ماننسیس می گویند. بخش دوم به صورت توانی از ۱۰ است که به آن نما می گویند.

مثال

```
double x=1.234E2; //123.4
float w=4.53E-3f; //0.00453
```

سؤال) برنامه زیر را نوشته و عملگر + را بررسی کنید؟

```
static void Main()
{
    Console.WriteLine("5" + 4 + 7);
    Console.WriteLine(5 + 4 + 7);
    Console.WriteLine(5 +"4" + 7);
    Console.WriteLine("5" + (4 + 7));
    Console.ReadKey();
}
```



```
string st = 13 + "" + 12;
string st = '2'+"a";
string st = 13 + 12; // syntax error
string st = '2'; // syntax error
string st = 2; // syntax error
```

تفاوت 'a' و "a" چیست؟

نکته) نوع داده char فقط یک کاراکتر را نگهداری می کند. اما نوع داده string برای نگهداری یک یا بیش از یک کاراکتر استفاده می شود. (تمرین) با استفاده از متد WriteLine رشته ها یا کاراکترهای زیر را نمایش دهید و حاصل را بنویسید؟

معادل	کد حرف یا رشته	معادل	کد حرف یا رشته
	"0\u00200"		'\u0007'
	"C\u0023"		"b\u0061ck"
	'\u0040'		'\u000a'
	'\u0030'		"12" + "8"

**متد ReadLine :**

این متد جهت دریافت رشته از ورودی استفاده می شود. خروجی این متد از نوع رشته ای می باشد و تا کلید Enter زده نشود کارش به اتمام نمی رسد. البته متد ReadKey نیز از متدهای ورودی می باشد که فقط منتظر دریافت یک کلید می شود و خروجی آن از نوع ConsoleKeyInfo می باشد.

مثال :

```
string st;
Console.WriteLine("Please Enter Number: ");
st = Console.ReadLine();
```

در این مثال پیغام Please Enter Number: به کاربر نشان داده می شود و منتظر دریافت اطلاعات از کاربر می شود. ورودی داخل متغیر st می شود. با توجه به اینکه این متغیر از نوع رشته ای می باشد جهت انجام محاسبات باید به عدد تبدیل گردد.

تمرین) برنامه ای بنویسید که نام و فامیلی را از صفحه کلید دریافت کند و سپس کلمه ی hello را به همراه نام و فامیلی به صورت رنگی نمایش دهد.

**تبدیل رشته به عدد:**

برای تبدیل رشته به عدد از متد Parse نوع int یا float استفاده می شود.

```
int x=int.Parse("721"); float y=float.Parse("2.35");
int x=int.Parse("721a"); //Runtime Error float y=float.Parse("2.35f"); //Runtime Error
```

نکته)نوع های داده مثل int خود یک کلاس هستند که دارای متدهایی نیز هستند. مثل متد Parse Parse به معنای تجزیه کردن است (بررسی کاراکتر به کاراکتر یک رشته).

تمرین) برنامه ای بنویسید که دو عدد صحیح از کاربر دریافت کند و سپس حاصل جمع و ضرب و تفریق و تقسیم آنها را بر روی صفحه نمایش، نشان دهد.

```
int n = int.Parse(Console.ReadLine());
```

تمرین) برنامه ای بنویسید که سه عدد دریافت کند و سپس میانگین آنها را چاپ کند؟





19 >> 1 → 9      25 << 1 → 50      21 << 3 → ?

**عملگرهای انتساب :** =

برای ذخیره یک مقدار یا عبارت در متغیر استفاده می شود. سمت چپ این عملگر باید یک متغیر باشد(ثابت یا عبارت نباشد) حاصل عملگرهای انتساب، مقدار داده ی سمت راست آنها است.

a=5 \*(b=8);

بر خلاف عملگرهای ریاضی که از سمت چپ به راست ارزیابی می شدند در عملگر انتساب از سمت راست به چپ ارزیابی صورت می گیرد(شرکت پذیری راست)

int z, y, x;      z = x = y = 10;

**عملگرهای انتسابی دیگر :** += , -= , \*= , /= , %= , &= , |= , ^= , <<= , >>=

x += 4; → x=x+4;      a\*=2; → a=a\*2;      s <<= 1; → s= s << 1;

x += a; → x=x+a;      n/=10; → n=n/10;      y\*=a+b; → y= y \*(a+b);

```
int a = 14, b = 9;
a = a + b;
b = a - b;
a = a - b;
```

سوال) معادل دستورات x/=(a+b); و x/=y; را بنویسید.

سوال) بررسی کنید برنامه روبرو چه عملی را انجام می دهد؟

تمرین) برنامه ای بنویسید که یک عدد سه رقمی را دریافت کند و حاصل جمع ارقام آن را نمایش دهد؟

تمرین) برنامه ای بنویسید که طول و عرض یک اتاق را دریافت کند و سپس اندازه یک ضلع یک سرامیک مربعی شکل را نیز دریافت کند. سپس تعداد سرامیک های مورد نیاز برای پوشش کف این اتاق را مشخص کند ؟ ( عدد گرد شود)

$$a * \frac{b-c}{s+w} + f$$

سوال) عبارت ریاضی روبرو را به صورت یک عبارت در C# بنویسید؟

```
float b = 461.3762921f;
Console.WriteLine(b);
```

سوال) خروجی دستورات روبرو چیست؟

**تست**

۱- زبان قابل فهم کامپیوتر که دنباله ای از صفر و یک ها است کدام گزینه است؟

الف) زبان اسمبلی      ب) زبان سطح بالا      ج) زبان ماشین      د) زبان سطح میانی

۲- کدامیک از زبان های زیر متن باز است؟

الف) JAVA      ب) C#      ج) C++      د) .NET

۳- به مجموعه ای از دستورات که برای انجام یک کار نوشته می شود ..... گفته می شود.

الف) Object      ب) Method      ج) class      د) namespace

۴- برای نوشتن توضیحات چند خطی در C# از کدام گزینه استفاده می شود؟

الف) /\* \*/      ب) /\* \*/      ج) //      د) \\\

۵- برای ترجمه برنامه در C# از کدام کلید استفاده می شود؟

الف) F5      ب) F6      ج) F2      د) F10

۶- متغیرهای روبرو مجموعا چه مقدار فضا اشغال می کنند؟

```
int a; ushort b;
float c;
bool z;
```

الف) ۹ بایت      ب) ۱۱ بایت      ج) ۱۲ بایت      د) ۸ بایت

۷- خروجی دستور روبرو کدام گزینه است؟

الف) 43b41      ب) CbA      ج) CA      د) این دستور خطا است

**فصل ۶) دستورهای شرطی :**

**عبارت منطقی (بولین):**

عبارتی که حاصلش true یا false است یک عبارت منطقی است. در عبارات منطقی از عملگرهای مقایسه ای (رابطه ای) استفاده می شود. مثلا  $12 > 5$  یک عبارت منطقی است که حاصلش true است. عملگرهای مقایسه ایی در جدول زیر تشریح شده اند:

مثال	نشانه ی عملگر	نام عملگر
$12 == 8 \rightarrow false$	==	برابری
$12 != 8 \rightarrow true$	!=	نا مساوی
$12 < 8 \rightarrow false$	<	کوچکتر
$12 <= 8 \rightarrow false$	<=	کوچکتر یا مساوی
$12 > 8 \rightarrow true$	>	بزرگتر
$12 >= 8 \rightarrow true$	>=	بزرگتر یا مساوی

(مثال ها)

```
Console.WriteLine(10 == 12); //false
Console.WriteLine(10 >= 12); //false
Console.WriteLine(10 <= 12); //true
Console.WriteLine('A' == 'B'-1); //true
Console.WriteLine(65 == 'B' - 1); //true
Console.WriteLine('A' == 65); //true
```

**عملگرهای منطقی: ^ || && !**

این عملگرها برای ترکیب شرط ها استفاده می شوند. و به صورت زیر هستند:

اولویت	نام عملگر	علامت	عملکرد
۱	نقیض	!	ارزش عملوند را معکوس می کند.
۲	یا انحصاری	^	اگر ارزش عملوندها مخالف یکدیگر باشد، نتیجه این عملگر true خواهد بود.
۳	و	&&	تنها در صورتی که هر دو عملوند true باشند نتیجه این عملگر نیز true خواهد بود. در غیر این صورت false است.
۴	یا		اگر حداقل یکی از عملوندها true باشند نتیجه این عملگر نیز true خواهد بود.

در جدول زیر عملکرد عملگرهای && و || و ^ تشریح شده است:

عملوند اول	عملوند دوم	&&		^
true	true	true	true	false
true	false	false	true	true
false	true	false	true	true
false	false	false	false	false

تمرین) حاصل عبارت روبرو را بدست آورید؟  $18 \% 7 > 2 \ || \ 25 != 17 \ \&\& \ 32 <= 11 \ \wedge \ 14 < 5 + 4$

$15 > 11 \ \wedge \ !(22 == 8) \ \&\& \ 16 > 9 \ || \ 20 < 13$



**دستور شرطی if:**

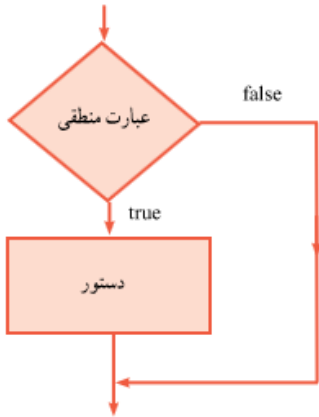
برای کنترل اجرای دستورات استفاده می شود. شکل کلی استفاده از این دستور بدین صورت است :

if( عبارت منطقی )

; یک دستور

این دستور از سه بخش تشکیل شده است. در صورتی که شرط true باشد دستور داخل if اجرا می شود. در صورتی که بخواهیم بیش از یک دستور در بدنه ی if اجرا شود، آن ها را بین علامت های { } قرار می دهیم.

شرط حتما باید بین دو پرانتز قرار بگیرد در غیر اینصورت خطای گرامری داده می شود.



```
int a = int.Parse(Console.ReadLine());
if (a %2 == 0 )
    Console.WriteLine(a+ " is Even");
if (a %2 != 0 )
    Console.WriteLine(a+ " is Odd");
```

مثال (۱)

مثال (۲)

```
int a = int.Parse(Console.ReadLine());
if (a < 10)
{
    Console.WriteLine("Your Mark is "+a);
    Console.WriteLine("Failed ");
}
if (a >= 10)
{
    Console.WriteLine("Your Mark is " + a);
    Console.WriteLine("Passed ");
}
```

تمرین) برنامه ای بنویسید که یک عدد دریافت کند و در صورتی که به ۵ و ۳ پخش پذیر بود پیغام Yes چاپ کند.

تمرین) برنامه ای بنویسید که نمره یک دانش آموز را دریافت کند. اگر نمره کمتر از ۱۰ بود کاراکتر E را نمایش دهد. اگر نمره بین ۱۰ تا ۱۴ بود کاراکتر D را نمایش دهد اگر نمره بین ۱۴ تا ۱۶ بود کاراکتر C را نمایش دهد و اگر نمره بین ۱۶ تا ۱۸ بود کاراکتر B را نمایش دهد و اگر نمره بین ۱۸ تا ۲۰ بود کاراکتر A را نمایش دهد . در نهایت اگر نمره از ۲۰ بیشتر بود پیغام Error صادر شود.

**بلاک:**

```
{
    دستور ;
    دستور ;
    دستور ;
}
```

به تعدادی دستور که داخل علامت های آکولا باز و بسته قرار داشته باشند، بلاک گفته می شود. بلاک می تواند خالی باشد یا حتی فقط شامل یک دستور باشد. برای خوانایی بیشتر، دستورات داخل بلاک را با تورفتگی می نویسند. یک بلاک را می توان مانند یک دستور تلقی کرد.

مثال) خروجی دستورات روبرو چیست؟

```
int grade = 5;
if (grade > 10) ;
    Console.WriteLine("Good");
Console.WriteLine("Bad");
```

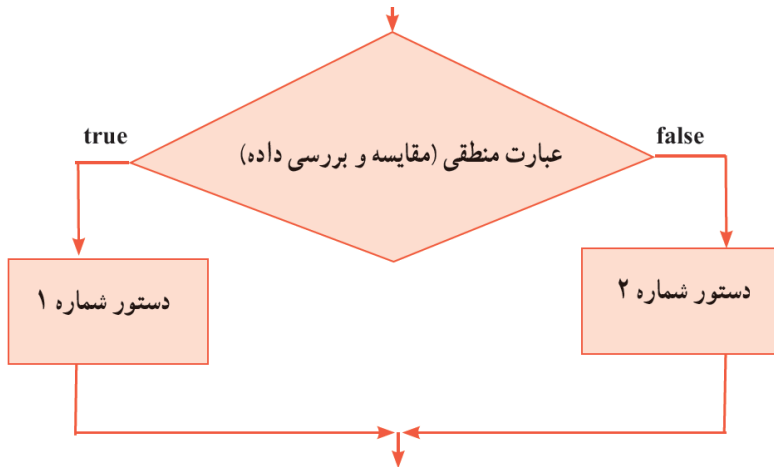
**دستور شرطی if-else:** در صورتی که شرط if برقرار نباشد قسمت else اجرا می شود.

( عبارت منطقی ) if

دستور ۱

else

دستور ۲



در این دستور اگر شرط درست باشد دستور ۱ اجرا می شود و در صورتی که شرط درست نباشد دستور ۲ اجرا می شود. در برنامه زیر عددی از کاربر گرفته می شود و در متغیر a قرار می گیرد. اگر این عدد زوج باشد پیغام is Even و در غیر این صورت پیغام is Odd صادر می شود.

```
int a = int.Parse(Console.ReadLine());
if (a %2==0 )
    Console.WriteLine(a+ " is Even");
else
    Console.WriteLine(a+ " is Odd");
```

نکته) همیشه بین if و else باید یک دستور یا یک بلاک دستور قرار بگیرد در غیر این صورت خطا صادر می شود.

```
if (a %2==0 )
    Console.WriteLine(a+ " is Even");
    Console.WriteLine();
else
    Console.WriteLine(a+ " is Odd");
```

تمرین) برنامه ای بنویسید که سه عدد از کاربر دریافت کند و بزرگترین و کوچکترین آنها را چاپ کند؟

تمرین) برنامه ای بنویسید که درآمد یک کارمند را دریافت کند و سپس ۷ درصد آن به عنوان حق بیمه محاسبه کند. در صورتی که درآمد کارمند از یک میلیون بیشتر است ۵ درصد مالیات مازاد یک میلیون را محاسبه کنید. حقوق دریافتی هر کارمند از فرمول زیر محاسبه می شود.  
مالیات - بیمه - درآمد = حقوق خالص

سوال) برنامه روبرو چه ایرادی دارد؟

```
int k = 12;
if (k>5);
{
    Console.WriteLine("+");
}
else
{
    Console.WriteLine("$");
}
```

تمرین ۱) برنامه ای بنویسید که یک عدد تک رقمی از کاربر دریافت کند و سپس معادل انگلیسی آن را نمایش دهد؟ Five → 5

تمرین ۲) برنامه ای بنویسید که یک عدد از کاربر دریافت کند و سپس قدر مطلق آن را نمایش دهد؟

تمرین ۳) برنامه ای بنویسید که سه عدد از کاربر دریافت کند و بزرگترین و کوچکترین آنها را نمایش دهد؟

**دستورات شرطی تو در تو:**

بدنه ی یک if یا یک else می تواند یک if دیگر باشد. به صورت روبرو:

(مثال)

```

if (عبارت منطقی ۱)
{
    دستورات 1
}
else if (عبارت منطقی ۲)
{
    دستورات 2
}
else if (عبارت منطقی ۳)
{
    دستورات ۳
}
else
{
    دستورات ۴
}
    
```

```

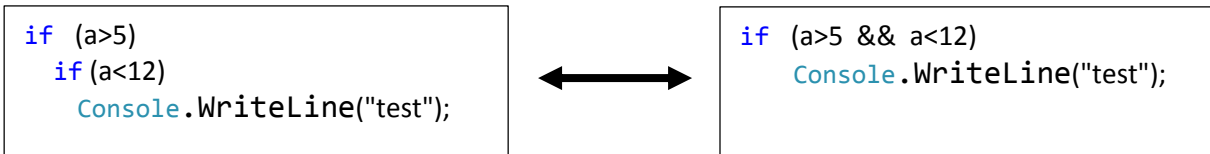
int n = int.Parse(Console.ReadLine());
if (n >= 0 && n < 10)
    Console.Write(n);
else if (n == 10)
    Console.Write('A');
else if (n == 11)
    Console.Write('B');
else if (n == 12)
    Console.Write('C');
else if (n == 13)
    Console.Write('D');
else if (n == 14)
    Console.Write('E');
else if (n == 15)
    Console.Write('F');
else
    Console.Write("Error");
    
```

(مثال دیگر)

```

float b;
b = float.Parse(Console.ReadLine());
if (b < 0 || b > 20)
    Console.WriteLine("Error");
else if (b >= 0 && b < 10)
    Console.WriteLine("BAD");
else if (b >= 10 && b < 15)
    Console.WriteLine("Good!");
else if (b >= 15 && b <= 20)
    Console.WriteLine("very Good!");
Console.ReadKey();
    
```

دستورات زیر معادل هستند:



تمرین) برنامه ای بنویسید که یک کاراکتر حرف انگلیسی از کاربر دریافت کند و در صورتی که حرف کوچک بود آن را به صورت حرف بزرگ نمایش دهید؟  $b \rightarrow B$

تمرین) برنامه ای بنویسید که سه عدد به عنوان ضرایب یک معادله درجه دو دریافت کند و سپس ریشه های معادله را نمایش دهد؟

$$ax^2 + bx + c = 0$$

تست) با توجه به تعریف متغیرهای روبرو، کدام گزینه خطا است؟  
 الف)  $h = a$       ب)  $z = m$       ج)  $a = z$       د)  $h = z$

**دستور switch** : در مواردی که بخواهیم حالت های مختلف یک عبارت را بررسی و بر اساس آن دستورهایی را اجرا کنیم از این دستور استفاده می کنیم.

ابتدا عبارت داخل پرانتز ارزیابی می شود. این عبارت نمی توان اعشاری باشد. سپس بررسی می شود حاصل این عبارت برابر کدام مقدار جلوی **case** می باشد و دستور متناسب با آن تا رسیدن به دستور **break** اجرا می شود. اگر این مقدار برابر هیچ کدام از مقادیر جلوی **case** نبود دستورات بعد از **default** اجرا می شود. مثال

```
switch(عبارت)
{
    case مقدار ۱ :
        دستور۱
        break;
    case مقدار ۲ :
        دستور۲
        break;
    .
    .
    default :
        دستورهای دیگر
        break;
}
```

```
int a = int.Parse(Console.ReadLine());
switch(a){
    case 1:
        Console.WriteLine("One");
        break;
    case 2:
        Console.WriteLine("Two");
        break;
    case 3:
        Console.WriteLine("Three");
        break;
    case 4:
        Console.WriteLine("Four");
        break;
    default :
        Console.WriteLine(a+ " Is > 4 or < 1 ");
        break;
}
```

نکته) عبارت **default** در این دستور اختیاری می باشد.

نکته) از دستور **break** برای خاتمه داده به یک **case** استفاده می شود. اگر دستور **break** نوشته نشود خطا صادر می شود البته اگر بصورت زیر نوشته شود خطا صادر نمی شود:

```
switch(a){
    case 1:
    case 2:
        Console.WriteLine("Two");
        break;}
}
```

در این حالت اگر مقدار **a** برابر یک یا دو باشد عبارت **Two** نمایش داده می شود.

نکته) دستور بعد از **case** می تواند هر دستوری حتی یک **switch** یا **if** دیگر باشد مثلاً

```
case 2:
    if (b < 3 && b > 2)
        Console.WriteLine("test");
    else
        Console.WriteLine("Good");
    break;
```

نکته) عبارت جلوی **switch** نمی تواند اعشاری باشد اما می تواند حرفی، رشته ای و صحیح باشد. دستور زیر خطا است.

```
switch( n/10.0)
```

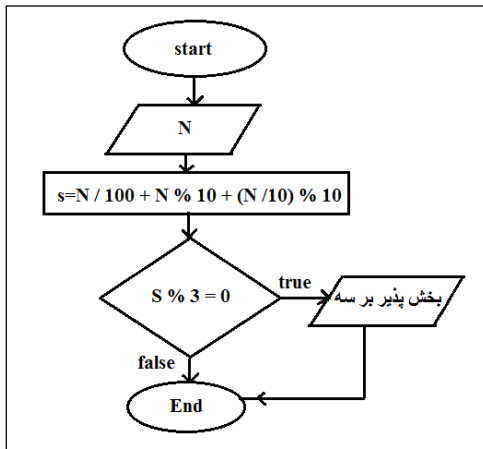
تمرین) برنامه ای بنویسید که یک عدد به عنوان شماره ی ماه سال دریافت کند و فصل آن را تعیین کند.

```
int b = int.Parse(Console.ReadLine());
switch (b)
{
    case 1:
    case 2:
    case 3:
        Console.WriteLine("Spring");
        break;
    case 4:
    case 5:
    case 6:
        Console.WriteLine("Summer");
        break;
    case 7:
    case 8:
    case 9:
        Console.WriteLine("Fall");
        break;
    case 10:
    case 11:
    case 12:
        Console.WriteLine("Winter");
        break;
}
```

تمرین) برنامه ای بنویسید که یک عدد دریافت کند و زوج یا فرد بودن آن را مشخص کند ( با استفاده از رقم یکان عدد ورودی)

```
int n = int.Parse(Console.ReadLine());
int r = n % 10;
switch (r)
{
    case 0:
    case 2:
    case 4:
    case 6:
    case 8:
        Console.WriteLine("Is Even");
        break;
    default :
        Console.WriteLine("Is Odd");
        break;
}
```

تمرین) برنامه مربوط به فلوجارت روبرو را بنویسید؟ ( فرض کنید عدد ورودی سه رقمی است)



**فصل ۷) دستورات تکرار حلقه :**

برای تکرار دستورات از دستورات while یا do-while یا for استفاده می کنیم. در دستورات تکرار، تعدادی دستور(بدنه ی حلقه) بر اساس درستی یک شرط تکرار می شوند. دستورات تکرار به دو گروه معین و نامعین دسته بندی می شوند. دستورات while و do-while نامعین هستند یعنی تعداد تکرار دستورات نامشخص است و بستگی به شرط حلقه دارد. اما دستور for معین است و تعداد مشخصی تکرار می شود.

**دستور while :**

**(عبارت منطقی) while**  
 یک دستور

در ابتدا شرط بررسی می شود اگر درست باشد دستور اجرا می شود و کنترل برنامه دوباره به دستور while باز می گردد و شرط مجددا بررسی می شود. اگر شرط نادرست باشد حلقه دیگر تکرار نمی شود.  
 در فلوچارت روبرو ابتدا شرط بررسی اگر درست بود دستور انجام می گردد و مجددا به بالای فلوچارت بر می گردد و دوباره شرط بررسی می گردد. این کار ادامه پیدا می کند تا زمانی که شرط نادرست شود.

مثال :

```
int a=1;
while (a < 5)
{
    Console.WriteLine( a);
    a++;
}
Console.WriteLine("End");
```

ا	خروجی
1	1
2	2
3	3
4	4
5	End

در این مثال اعداد یک تا ۴ نمایش داده می شود هنگامی که a=5 می شود شرط حلقه نادرست می شود و از حلقه بیرون می آید و عبارت End را نمایش می دهد. نکته مهم این که همانند دستور شرطی if چون می خواهیم دو دستور تکرار شوند آنها را داخل { } قرار می دهیم.  
 مثال دیگر :

```
int a=1;
while (a < 10)
    Console.WriteLine(a++);
Console.WriteLine("End");
```

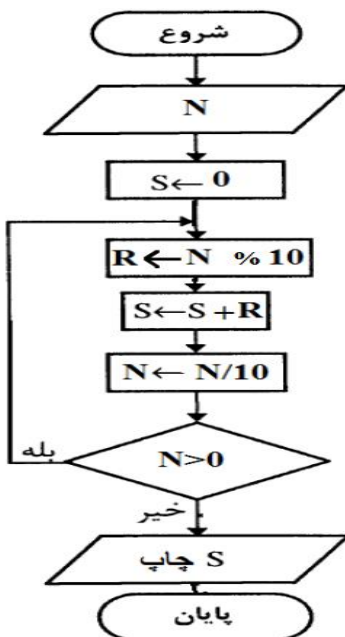
نکته) اگر در همان ابتدای حلقه، شرط نادرست باشد حلقه تکرار نمی شود.

نکته) اگر بعد از while علامت ; قرار گیرد در صورتی که شرط برقرار باشد عبارت ; تکرار می شود و حلقه بی نهایت می شود(خطای منطقی).

نکته) در صورتی که بدنه ی حلقه فقط یک دستور باشد نیازی به استفاده از بلاک نیست. اما اگر بدنه ی حلقه بیشتر از یک دستور باشد باید از بلاک استفاده شود.

تمرین) برنامه ای بنویسید که اعداد زوج دو رقمی را نمایش دهد؟

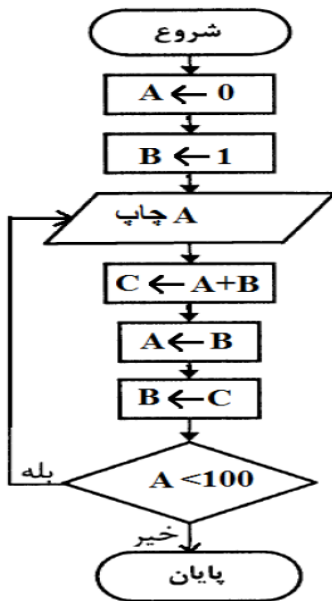
تمرین) برنامه ای بنویسید که یک عدد صحیح دریافت کند و مجموع ارقام آن را نمایش دهد؟



تمرین) برنامه زیر چه عملی را انجام می دهد؟

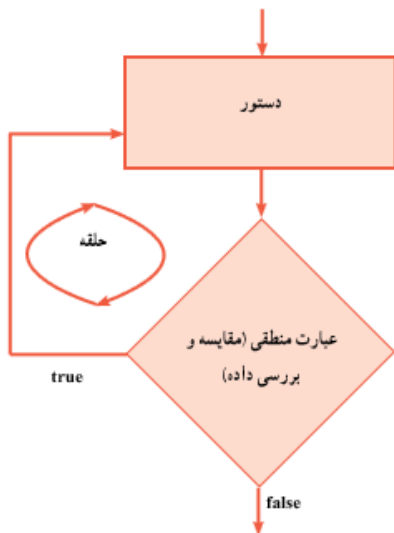
```
int n = int.Parse(Console.ReadLine());
long f = 1;
int c = 1;
while(c<=n)
{
    f *= c;
    ++c;
}
Console.WriteLine(f);
Console.ReadKey();
```

تمرین) برنامه مربوط به فلوجارت روبرو را بنویسید؟



تمرین) برنامه ای بنویسید که یک عدد دریافت کند مغلوب عدد را نمایش دهد؟ 5241 → 1425

**دستور do-while**: این دستور مانند دستور while است با این تفاوت که شرط حلقه بعد از اجرای بدنه ی حلقه بررسی می شود. لذا حداقل یکبار اجرا می شود. شکل کلی دستور:



```
do
    دستور;
while(عبارت منطقی);
```

```
int a=5;
do
    Console.WriteLine(a);
while (a > 7);
Console.WriteLine("End");
```

در برنامه فوق، با توجه به اینکه ارزش شرط نادرست می باشد اما حداقل یکبار اجرا می شود. ۵ را نمایش می دهد سپس شرط را بررسی می کند و چون false می باشد از حلقه خارج شده و عبارت End را نمایش می دهد.

نکته) بین do و while باید فقط یک دستور باشد و اگر بخواهیم بیش از یک دستور را تکرار کنیم آنها را بین {} قرار می دهیم.

نکته) در دستور do - while چون انتهای دستور while می باشد لذا بعد از while حتما باید ; قرار دهیم.

```
int a=5;
do
    Console.WriteLine(a);
    a++;
while (a > 7);
Console.WriteLine("End");
```

تمرین) برنامه روبرو خطا دارد؟ علت آن را نوشته و در برنامه تغییری دهید تا خطا رخ ندهد.

تمرین) برنامه ای بنویسید که تعدادی عدد از کاربر دریافت کند و سپس میانگین اعداد وارد شده را محاسبه و نمایش دهد. (پایان بخش اعداد، صفر است)

```
int number, sum = 0, tedad = 0;
do
{
    number = int.Parse(Console.ReadLine());
    sum += number;
    tedad++;
} while (number != 0);
float avrg = sum / (tedad - 1);
Console.WriteLine(avrg);
Console.ReadKey();
```

تمرین

۱) خروجی دستورات روبرو را بنویسید؟

```
static void Main()
{
    int a = 1, b = 1;
    if (! (a==1 && b==2))
        Console.WriteLine(true);
    if (a !=1 || b!=2)
        Console.WriteLine(false);
}
```

۲) جاخالی های زیر را با عبارات مناسب پر کنید؟

الف) مترجم نمی تواند نوع داده ..... را به طور ضمنی و خودکار به نوع float تبدیل کند.

ب) در زبان برنامه نویسی C#، عملگرهای یکتایی ++ و -- به ترتیب برای ..... و ..... مقدار یک متغیر به اندازه ی ..... واحد در نظر گرفته شده است.

ج) آکولاد های باز و بسته محدوده عملیات ..... و ..... دستور switch را معین می کند.

د) اگر در برنامه ای با عبارت ها و اعداد اعشاری با دقت حداکثر ۷ رقم سر و کار دارید از متغیرهای نوع ..... استفاده می شود.

۳) دستوری صحیح برای نمایش رشته زیر به زبان C# بنویسید.

I study "computer".

۴) اگر  $y=2$  ,  $x=7$  int نتیجه کدام عبارت، عدد ۴ نمی شود؟

الف)  $y+x/2-1$       ب)  $(x+y)\%2+2$       ج)  $x-2+(-y)$       د)  $-y*2+4*2$

۵) برنامه روبرو چه عملی را انجام می دهد؟

```
char ch, up;
ch = char.Parse(Console.ReadLine());
if (ch >= 'a' && ch <= 'z')
    up = (char) (ch - 32);
else
    up = ch;
Console.WriteLine(up);
Console.ReadKey();
```

۶) خروجی برنامه روبرو را بدست آورید؟

```
int a, b, c;
a=b=1;
while(a<100) {
    Console.WriteLine(a);
    c = a + b;
    a = b;
    b = c;
}
```

۷) برنامه روبرو را بررسی کنید و خروجی آن چیست؟

```
int n = 1;
while (n < 10);
{
    Console.WriteLine(n);
    ++n;
}
```

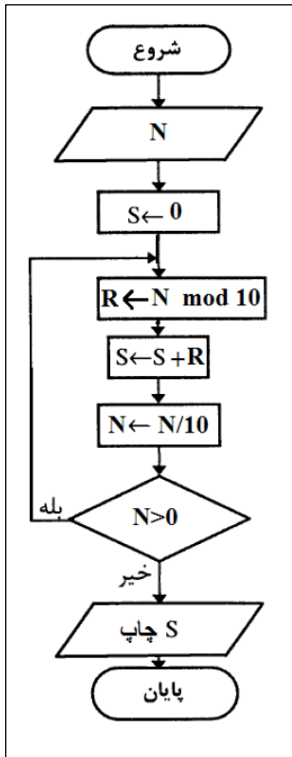
۸) برنامه مربوط به فلوجارت روبرو را بنویسید.

برنامه ایی که یک عدد دریافت می کند و مجموع ارقام آن را محاسبه و چاپ می کند.

۹) خروجی برنامه روبرو کدام گزینه است؟

الف) SalamBye      ب) Salam

ج) Bye      د) هیچ چیزی چاپ نمی شود



```
int s = 0x12;
if (s == 18)
    Console.Write("Salam");
if (s == 12);
    Console.WriteLine("Bye");
```



**دستور حلقه for :**

در صورتی که تعداد تکرار مشخص باشد از حلقه ی for استفاده می شود. شکل کلی این دستور بدین صورت است:

(گام حلقه یا تغییر مقدار متغیر ؛ عبارت منطقی ؛ بخش مقداردهی اولیه به متغیر) for

و یک دستور

نکته) ابتدا مقدار دهی اولیه انجام می شود سپس در صورت true بودن شرط حلقه، بدنه ی آن اجرا می شود و در نهایت دستور تغییر اجرا می شود و دوباره شرط بررسی می شود.

مثال : مراحل اجرای دستور روبرو به صورت زیر است:

```
for(int i=5 ; i<=10; i++)
    Console.WriteLine(i);
```

۱- دستور int i=5 اجرا می شود.

۲- شرط بررسی می شود. اگر درست بود مرحله ۳ اجرا می شود در غیراینصورت حلقه پایان می یابد.

۳- دستور بدنه for یعنی Console اجرا می شود.

۴- دستور i++ اجرا می شود. و دوباره به مرحله ۲ می رود.

برنامه بالا را می توان به صورت زیر نیز نوشت

```
for (int i = 5; i <= 10; Console.WriteLine(i), i++) ;
یا
for (int i = 5; i <= 10; Console.WriteLine(i++)) ;
```

نکته) اگر دستورات داخل حلقه بخواهیم بیشتر از یکی باشد از بلاک استفاده می کنیم. مثال ها) هر کدام از دستورات تکرار زیر را بررسی کنید:

```
int i;
for (i = 1; i < 100; i *= 2)
    Console.WriteLine(i);
```

```
int a,b;
for (a = 1, b = 10; a < b; ++a , --b )
    Console.WriteLine("{0} {1}" , a,b);
```

```
int c, f, n;
n = int.Parse(Console.ReadLine());
for (c = f = 1; c <= n; c++)
    f = f * c;
Console.WriteLine(f);
```

```
int c=1;
for ( ; c<10 ; )
{
    Console.WriteLine(c);
    c++;
}
```

```
int a;
for (int i = 5; i > 0; i-- )
{
    int x = i + 10;
    a = i + 10;
}
Console.WriteLine(a + i + x);
```

نکته) اگر متغیری در داخل یک دستور ( مانند if یا for یا while ) یا در یک بلاک تعریف شود، آن متغیر فقط در همان دستور یا بلاک قابل استفاده است و خارج از آن در دسترس نیست. در برنامه روبرو متغیرهای i و x در داخل دستور for تعریف شده اند لذا بعد از اتمام اجرای حلقه ی for دیگر نمی توان از آن استفاده کرد(خطا).

تمرین ۱) برنامه ای بنویسید که یک عدد دریافت کند و مقسوم علیه های آن را نمایش دهد؟

تمرین ۲) برنامه ای بنویسید که یک عدد صحیح دریافت کند و تشخیص دهد که آیا عدد اول است یا نه؟

تمرین ۳) برنامه ای بنویسید که دو عدد صحیح X و Y را از کاربر دریافت کند و سپس Y \* X را بدون استفاده از عملگر \* محاسبه و نمایش دهد؟

تمرین ۴) خروجی دستورات زیر را بدست آورید؟

```
for (int i = 1; i < 10; i++) ;
    Console.WriteLine(i);
```

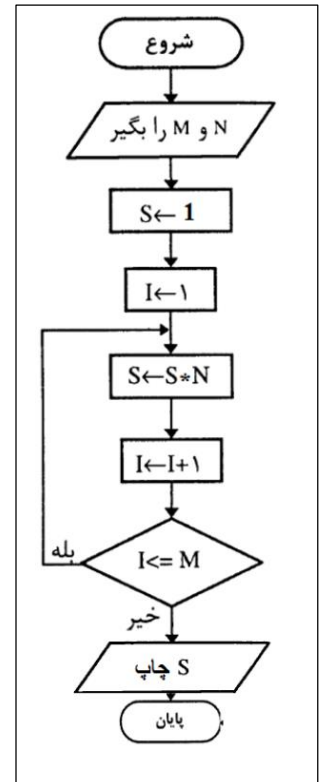
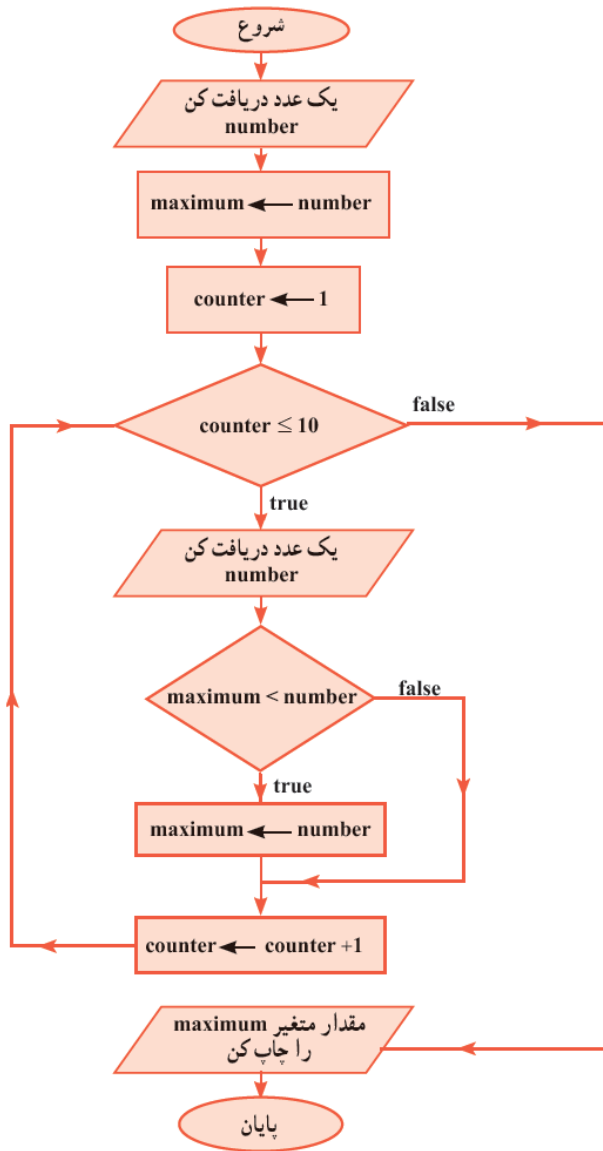
```
int i;
for ( i = 1; i < 10; i++) ;
    Console.WriteLine(i);
```

```
for (int i = 1; i < 20; i++)
    if (i % 2 == 0)
        Console.WriteLine(i);
```

```
for (int a = 0, b = 1, c; a < 1000; )
{
    Console.WriteLine(a);
    c = a + b;
    a = b;
    b = c;
}
```

تمرین) برنامه ای بنویسید که ۱۰ عدد دریافت کند و ماکزیمم آنها را نمایش دهد؟

تمرین) با توجه به فلوجارت زیر برنامه ای بنویسید که اعداد صحیح M, N را دریافت کند و  $M^N$  را محاسبه و نمایش دهد؟



```
int a = 3;
for (a = 20; a > 100; a++)
    Console.WriteLine(a);
    Console.WriteLine(a * 2);
```

```
for ( ; ; )
    Console.WriteLine("*");
```

تست) تعداد تکرار حلقه روبرو کدام گزینه است؟ for(int i=3; i<8 ;i++)

الف) ۸ بار      ب) ۵ بار      ج) ۴ بار      د) ۶ بار

تست) خروجی دستورات روبرو کدام گزینه است؟

الف) اعداد ۲۰ تا ۹۹      ب) اعداد ۲۰ تا ۱۰۰

ج) عدد ۳      د) عدد ۴۰

نکته) در دستور for اگر شرط حلقه مشخص نشود حلقه بی نهایت بار تکرار می شود.

## دستور های break و continue

**دستور break:** هرگاه که در ساختارهای while یا do/while یا for یا switch اجرا گردد، باعث خروج فوری برنامه از آن ساختار خواهد شد و برنامه

اولین دستور بعد از آن ساختار را اجرا خواهد کرد. به برنامه زیر توجه کنید:

خروجی برنامه بدین صورت می باشد

10, 9, 8, 7, 6, 5, 4, 3, countdown aborted!

```
for (int n = 10; n > 0; n--)
{
    Console.WriteLine(n + ",");
    if (n == 3)
    {
        Console.WriteLine("countdown aborted!");
        break;
    }
}
```

تمرین) خروجی برنامه روبرو را بدست آورید؟

```
int n=1;
while(true)
{
    Console.WriteLine(n);
    if (n >= 10)
        break;
    Console.WriteLine('+');
    n++;
}
```

تمرین) برنامه بنویسید که تعدادی عدد از کاربر دریافت کند و میانگین آنها را نمایش دهد؟ ( داده های ورودی با ورود صفر به اتمام می رسند)

**دستور continue:** هرگاه در ساختارهای while یا do/while یا for اجرا گردد دستورات بعدی آن ساختار نادیده گرفته می شود و کنترل برنامه به

ابتدای حلقه می رود. در دو ساختار while و do/while پس از اجرای دستور continue شرط حلقه مورد بررسی قرار می گیرد، اما در ساختار for

ابتدا مقدار شمارنده افزایش یا کاهش می یابد، سپس شرط حلقه بررسی می شود. به عنوان مثال برنامه زیر مجموع اعداد ۱ تا ۲۰ به جز ۱۰ را محاسبه می کند.

```
int n = 0, sum = 0;
while (n < 20)
{
    ++n;
    if (n == 10) continue;
    sum += n; // sum = sum + n;
}
Console.WriteLine("1+2+ ...(except 10)...+20=" + sum);
```

## حلقه های تودر تو:

ساختارهای تکرار می تواند به صورت تو در تو استفاده شوند.

به برنامه ی روبرو توجه کنید:

```
int a, b;
for (a = 1; a < 10; a++)
    for (b = 1; b < 6; b++)
        Console.WriteLine("*");
```

حلقه دوم به عنوان بدنه ی حلقه ی اول است. هنگامی که متغیر  $a=1$  می شود و شرط حلقه ی اول true است حلقه دوم اجرا می شود و ۵ بار تکرار می شود. سپس

$a=2$  می شود و چون  $a < 10$  است دوباره حلقه ی دوم از ابتدا اجرا می شود و پنج \* چاپ می شود. در کل ۴۵ ستاره چاپ می شود.

تمرین) برنامه ای بنویسید که جدول ضرب را نمایش دهد؟

```
int i, j;
for (i = 1; i <= 10; ++i)
{
    for (j = 1; j <=10; j++)
        Console.Write("{0} ",i*j);
    Console.WriteLine();
}
```

تمرین) خروجی برنامه روبرو را بدست آورید؟

```
int a, b;
for (a = 1; a <= 5; a++ )
{
    Console.WriteLine();
    for (b = 1; b <= a; b++)
        Console.Write("*");
}
```

تمرین) خروجی برنامه روبرو را بدست آورید؟

```
for (int x = 1; x <= 5; x++ )
    for (int y = 5; y >8; y++)
        Console.Write("*");
```

تمرین) برنامه ای بنویسید که همه ی اعداد اول بین ۱۰۰ تا ۲۰۰ را نمایش دهد؟

### تست و تمرین

۱) کدام یک از گزینه های زیر داده نیست؟

الف) اسامی دانش آموزان      ب) معدل دانش آموزان      ج) نام شهر      د) محل سکونت

۲) اندازه دمای محیط کدام گزینه زیر است؟

الف) داده      ب) اطلاعات      ج) الگوریتم      د) پردازش

۳) وظیفه ی کامپایلر چیست؟

الف) تبدیل الگوریتم به فلوجارت      ب) ترجمه برنامه نوشته شده به کدهای زبان ماشین

ج) تبدیل الگوریتم به برنامه      د) اجرای برنامه

۴) کدامیک از گزینه های زیر زبانی است که دستورات برای دسترسی راحت تر سخت افزار پیش بینی شده و همچنین به زبان عامیانه نزدیک است؟

الف) Java      ب) C#      ج) C      د) ویژوال بیسیک

۵) اینکه در بین نمرات درس یک دانش آموز نمره کمتر از 0 و بیشتر از 20 نداشته باشیم، نشان دهنده کدام مورد از فرایند رسیدن داده به اطلاعات

است؟

الف) درستی انجام محاسبات      ب) روش انجام پردازش      ج) صحت داده ها      د) درستی برنامه

۶) خروجی برنامه روبرو کدام گزینه است؟

```
int n=12,s=13;
if (++n == s--)
    Console.WriteLine(n + s);
else
    Console.WriteLine(n - s);
```

الف) 0      ب) 25      ج) 26      د) 1

۷) خروجی برنامه روبرو کدام گزینه است؟

```
int a=5,b=13,c=7;
switch(c>a || b<a && c>b )
{
    case true:
        Console.WriteLine(a++); break;
    case false :
        Console.WriteLine(--b);break;
}
```

- الف) 5
- ب) 6
- ج) 12
- د) 13

۸) خروجی قطعه برنامه روبرو کدام گزینه است؟

```
char a = '\u000a';
Console.Write("C#" + a + "94");
```

```
C#
a
94
```

```
C#
94
```

```
C#94
```

```
C#a94
```

۹) خروجی برنامه روبرو کدام گزینه است؟

```
int b,a = 'A';
b = a + 2;
Console.WriteLine("{1}{0}", a, b);
```

- الف) AC
- ب) CA
- ج) 6567
- د) 6765

۱۰) خروجی برنامه روبرو کدام است؟

```
int m=18, n=-4, k=1;
k += m % n;
if (k == -1)
    Console.WriteLine("A");
else if (k == 2)
    Console.WriteLine("B");
else if (k == 3)
    Console.WriteLine("C");
else
    Console.WriteLine("D");
```

- الف) A
- ب) B
- ج) C
- د) D

۱۱) کدامیک از گزینه‌های زیر از ویژگی‌های زبان C# نیست؟

- الف) شباهت زیادی به زبان‌های C و Java دارد (ب) شیء‌گرا است
- ج) سطح بالا است
- د) متن باز است

۱۲) به محیطی که در آن می‌توان تمام مراحل برنامه‌نویسی، ترجمه، اشکال‌یابی و اجرا را انجام داد چه می‌گویند؟

- الف) Visual
- ب) IDE
- ج) Framework
- د) Class

۱۳) نام پیش فرض پروژه هنگام ایجاد یک پروژه جدید چیست؟

- الف) Program1
- ب) App1
- ج) ConsoleApplication1
- د) Untitled

۱۴) به صورت پیش‌فرض پروژه‌های ایجاد شده در برنامه Visual Studio در کدام مسیر ذخیره می‌شود؟

- الف) My Documents → Visual Studio → Projects
- ب) C:\ProgramFiles → Windows → System32
- ج) My Documents → Projects
- د) C:\ → Projects

۱۵) مکانی از حافظه RAM کامپیوتر که برای نگهداری موقتی داده یا نتایج حاصل از پردازش مورد استفاده قرار می‌گیرد و می‌توان محتوای آن را در طول اجرای برنامه تغییر داد، چه می‌گویند؟

- الف) Constant
- ب) Variable
- ج) Static
- د) Class

۱۶) در متغیر از نوع داده sbyte چه اعدادی می‌توان قرار داد؟

- الف) ۰ تا ۲۵۵
- ب) ۰ تا ۶۵۵۳۶
- ج) -۱۲۸ تا +۱۲۷
- د) -۱۲۷ تا +۱۲۸

۱۷) دستور زیر چه عملی انجام می‌دهد؟

- الف) مقدار 0x1B به صورت رشته در متغیر Value قرار می‌گیرد.
- ب) مقدار 1B در متغیر Value ذخیره می‌شود.

ج) مقدار ۲۷ در متغیر Value ذخیره می‌شود. د) دستور اشتباه است و نمی‌توان چنین دستوری را نوشت.

۱۸) در زبان برنامه نویسی سی شارپ، برای تبدیل رشته دریافتی به عدد از کدام متد استفاده می‌شود؟

الف) Read() ب) ReadLine() ج) Parse() د) Val()

۱۹) در زبان سی شارپ در نامگذاری متغیر کدام مورد صحیح است؟

الف) نام متغیر نمی‌تواند با عدد شروع شود. ب) استفاده از کلمات کلیدی و یا رزرو مجاز است.

ج) استفاده از علامت فاصله و خط تیره مجاز است. د) استفاده از اعداد و کاراکتر زیر خط مجاز نیست.

۲۰) اگر در برنامه یا عبارتی بخواهیم عدد اعشاری با دقت حداکثر ۷ رقم ذخیره کنیم پس از هر عدد اعشاری چه حرفی باید قرار دهیم؟

الف) F یا f ب) H یا h ج) U یا u د) D یا d

۲۱) در دستور switch نوع داده عبارتی که داخل پرانتز نوشته می‌شود، کدام نوع نمی‌تواند باشد؟

الف) اعشاری ب) حرفی ج) رشته‌ای د) صحیح

```
int a = 9, c = 5, b;
b = a >> 2 << 2;
b *= a + c;
Console.WriteLine(b);
```

۲۲) خروجی دستورات روبرو کدام گزینه است؟

الف) 126 ب) 112 ج) 77 د) 86

۲۳) مقدار a کدام گزینه باشد تا \$ چاپ شود؟

الف) 9 ب) 25 ج) 14 د) 30

```
int a = 30;
if (a > 12 ^ a < 25)
    Console.WriteLine("*");
else
    Console.WriteLine("$");
```

۲۴) خروجی برنامه روبرو کدام گزینه است؟

الف) \* ب) \$ ج) چیزی چاپ نمی‌شود د) این برنامه خطا دارد

```
int M = 5, m = 5;
if (M++ == ++m)
    Console.WriteLine("*");
else
    Console.WriteLine("$");
```

۲۵) خروجی برنامه روبرو کدام گزینه است؟

الف) 1 2 3 4 5 6 7 8 9 ب) 2 3 4 5 6 7 8 9

ج) 1 2 3 4 5 6 7 8 9 10 د) 2 3 4 5 6 7 8 9 10

```
int n = 1;
while (n++ < 10)
    Console.WriteLine(n);
```

۲۶) خروجی برنامه روبرو کدام گزینه است؟

الف) رقم یکان عدد ورودی ب) مقلوب عدد ورودی

ج) عدد صفر چاپ می‌شود د) رقم پرارزش عدد ورودی

```
int n = int.Parse(Console.ReadLine());
int s = 0, r;
do
{
    r = n % 10;
    s = s * 10 + r;
    n /= 10;
} while (n != 0);
Console.WriteLine(s);
```

۲۷) استفاده از روش های بهینه و الگوریتم های مناسب در عمل پردازش بیانگر کدام فرآیند رسیدن داده به اطلاعات می باشد؟

الف) صحت و اطلاعات ب) درستی انجام محاسبات ج) روش انجام پردازش د) ب و ج

۲۸) نتیجه و تفسیر اطلاعات که می‌تواند مبنای تصمیم گیری برای انجام کاری شود چه نام دارد؟

الف) داده ها (ب) پردازش داده ها (ج) نتیجه حاصل از پردازش (د) دانش (۲۹) کدام گزینه صحیح است؟

الف) زبان سی شارپ یک زبان سطح بالا، شی گرا، متن باز و همه منظوره است  
 ب) در زبان سی بر خلاف سی شارپ در هنگام ترجمه و همچنین اجرای برنامه دقت زیادی بر روی تطبیق داده ها صورت می گیرد تا از اشتباهات دستوری برنامه نویسی یا کاربر جلوگیری نماید.

ج) برای اجرای برنامه دات نت حتما باید فریم ورک روی سیستم نصب باشد (د) اجرای برنامه سی شارپ همیشه از کلاس Main شروع می شود (۳۰) برای سازماندهی کتابخانه کلاس ها و قراردادن آنها در یک آرایش مخصوص جهت دسترسی آسان از ..... استفاده می شود.

الف) کلاس (ب) شی (ج) فضای نام (د) متد (۳۱) خروجی کدام دستور با بقیه متفاوت است؟

الف) Console.WriteLine("12" + 13 + 2);  
 ب) Console.WriteLine(12 + "13" + 2);  
 ج) Console.WriteLine(12 + 13 + "2");  
 د) Console.WriteLine("12" + "13" + 2);

(۳۲) کدام گزینه می تواند نام متغیر در سی شارپ باشد؟

الف) aa2b (ب) \_b\_ (ج) Main (د) همه گزینه ها (۳۳) کدام نوع داده فقط برای ذخیره کردن اعداد مثبت استفاده می شود

الف) short (ب) double (ج) Byte (د) int

```
float f=1.23456789f;
System.Console.WriteLine(f);
```

(۳۴) خروجی دستور زیر چیست؟  
 الف) 1.2e+9 (ب) 1.23456789 (ج) 1.234567 (د) 1.234568

(۳۵) کدام گزینه نمی تواند یک ثابت کراکتی باشد؟

الف) '\u0043' (ب) '2' (ج) '\x0013' (د) '\u23'

(۳۶) کدام گزینه صحیح می باشد؟

الف) حداکثر تعداد ارقام غیر صفر و با معنی مانعیت عدد رادقت عدد می نامند.

ب) اگر در برنامه، یک عدد اعشاری بدون نشانه بنویسید این عدد به عنوان عدد اعشاری با دقت زیاد در نظر گرفته می شود.

ج) دستور decimal x = 3.432; خطای گرامری دارد.

د) اگر double d=1.2; و int x=35000; آنگاه دستور d=x; خطای گرامر دارد زیرا نمی توان متغیر صحیح را در متغیر اعشاری قرار داد.

(۳۷) در سوالات زیر خروجی دستورات را بنویسید؟

```
int a=3,b=3;
if (a > 2)
    if (b > 2)
        Console.WriteLine("###");
else
    Console.WriteLine("***");
```

الف) ### (ب) \*\*\*

ج) دستورات خطا دارد (د) برنامه خروجی ندارد

```
int a = 1, b = 3, c = 4;
int d = a++ + --b * c + b++ - a;
Console.WriteLine(d);
```

(۳۸)

الف) ۹ (ب) ۸ (ج) ۱۰ (د) ۷

```
int a = 1, b = 3, c = 0;
if (a > 2 && b / c > 1)
    Console.WriteLine("###");
else
    Console.WriteLine("***");
```

الف) خطای زمان اجرا- تقسیم بر صفر (ب) خطای منطقی تقسیم بر صفر

ج) ### (د) \*\*\*

(۴۰)

```
int a = 1, b = 3, c = 0;
if (a > 2 || b / c > 1)
    Console.WriteLine("###");
else
    Console.WriteLine("***");
```

الف) خطای زمان اجرا- تقسیم بر صفر (ب) خطای منطقی تقسیم بر صفر

ج) ### (د) \*\*\*

```
int f = int.Parse(Console.ReadLine());
int sum=f%10 + f/10;
switch(sum)
{
    case 3:
    case 6:
    case 9:
    case 12:
    case 15:
    case 18:
        Console.WriteLine("Divisible 3"); break;

    default :
        Console.WriteLine("Non Divisible 3"); break;
}
```

۴۱) برنامه ی روبرو چه عملی انجام می دهد؟

```
int x = 10, y = 19;
Console.WriteLine(x | y);
Console.WriteLine(x>y || y>=19);
```

۴۲) خروجی برنامه روبرو را بدست آورید؟

```
int s=1;
do
{
    Console.WriteLine(s);
    s <<= 1;
} while (s < 100);
```

۴۳) خروجی برنامه روبرو را بنویسید؟

```
int n = int.Parse(Console.ReadLine());
string hex = ""; int r;
while(n>0)
{
    r = n % 16;
    if (r < 10)
        hex = r + hex;
    else
        switch (r)
        {
            case 10: hex = 'A' + hex; break;
            case 11: hex = 'B' + hex; break;
            case 12: hex = 'C' + hex; break;
            case 13: hex = 'D' + hex; break;
            case 14: hex = 'E' + hex; break;
            case 15: hex = 'F' + hex; break;
        }
    n /= 16;
}
Console.WriteLine(hex);
```

۴۴) برنامه روبرو را بررسی و تحلیل کنید. چه عملی را انجام می دهد؟

```
Console.WriteLine(A && true ^ B);
```

۴۵) به ازای کدام مقادیر برای A و B، خروجی دستور مقابل True خواهد بود؟

- الف) A=true , B=true
- ب) A=false , B=true
- ج) A=false , B=false
- د) A=true , B=false

۴۶) خروجی برنامه مقابل چیست؟

- الف) HamidAli
- ب) Saeid
- ج) SaeidAli
- د) Hamid

```
int a = 5;
if (a!=2+4)
    Console.Write("Hamid");
else
    Console.Write("Saeid");
    Console.Write("Ali");
```



۴۷) کدام نوع داده بازه ی بیشتری دارد؟

الف) uint      ب) long      ج) ushort      د) byte

۴۸) کدام روش مقدار دهی صحیح است؟

الف) char a="@";      ب) char a='\u41';  
ج) char a='\x240D';      د) char a='AB';

۴۹) کدام گزینه عدد ۱۵ را در متغیر Z قرار می دهد؟

الف) int z=\x15;      ب) int z=0xf;  
ج) int z=0x15;      د) int z=\xf;

۵۰) خروجی برنامه مقابل چیست؟

الف) B  
ب) C  
ج) D  
د) A

```
switch (5 - 4 * 5 / 3 + 8 % 5)
{
    case 1:
        Console.WriteLine("A");
        break;
    case 2:
        Console.WriteLine("B");
        break;
    case 3:
        Console.WriteLine("C");
        break;
    default:
        Console.WriteLine("D");
        break;
}
```

۵۱) کدام گزینه رنگ قرمز را به عنوان رنگ زمینه پنجره خروجی کنسول در نظر می گیرد؟

الف) Console.ForegroundColor=CosoleColor.Red;      ب) Console.BackgroundColor=CosoleColor.Red;

ج) Console.ForegroundColor(CosoleColor.Red);      د) Console.BackgroundColor(CosoleColor.Red);

۵۲) کدام روش استفاده از دستور using صحیح است؟

الف) نام کلاس . فضای نام using      ب) نام متد . نام کلاس using      ج) فضای نام using      د) نام کلاس using

۵۳) کدام نوع داده غیر عددی است؟

الف) ushort      ب) int      ج) long      د) bool

۵۴) خروجی برنامه روبرو کدام گزینه است؟

```
int m = 1;
while (++m < 10)
    Console.WriteLine(m);
```

الف) ۱ تا ۱۰      ب) ۲ تا ۱۰      ج) ۲ تا ۹      د) ۱ تا ۹

۵۵) نوشتن دستورات لازم برای کنترل نحوه کار کامپیوتر، به طوری که کامپیوتر بتواند یک کار مشخص را انجام دهد ..... نامیده می شود.

الف) مترجم      ب) برنامه نویسی      ج) برنامه نویس      د) پردازش

۵۶) سیستم عامل LINUX با زبان ..... طراحی شده است.

الف) C      ب) C#      ج) زبان ماشین      د) JAVA

۵۷) نام فایل مترجم در .NET FrameWork. کدام گزینه است؟

الف) CSS      ب) SCS      ج) CSC      د) CMD

۵۸) نام هایی که برنامه نویس باید انتخاب کند(مانند نام یک کلاس) ..... گفته می شوند.

الف) متغیر      ب) کلاس      ج) شناسه      د) متد

۵۹) کدامیک از گزینه های زیر کلمات رزرو شده هستند؟

الف) Void, Class, CSharp      ب) C#, Class, Static      ج) Main, CSharp, Static      د) Void, Class, Static

۶۰) کدامیک از گزینه های زیر مجموعه ای از دستورات است که برای انجام یک کار لازم است؟

الف) متغیر      ب) کلاس      ج) متد      د) شناسه

۶۱) بالاترین سطح دسته بندی در محیط ویژوال استودیو دات نت، ..... است که وظیفه آن نگهداری روابط بین پروژه ها است.

الف) Class      ب) NameSpace      ج) Solution      د) Console

۶۲) برای ترجمه یک برنامه از کدام کلید صفحه کلید استفاده می شود؟

F5 (الف) خروجی دستور روبرو کدام است؟  
 ۴ (الف) 20/5 (ب)  
 ۶۳ خروجی دستور روبرو کدام است؟

F6 (ب) long (ب) int (الف)  
 ۶۴ کدام نوع داده از بقیه فضای بیشتری اشغال می کند؟

F2 (ج) uint(ج) float (د)  
 ۶۵ کدام گزینه برای تعریف سه متغیر صحیح نیست؟  
 int a, b, c = 5;(ب) int a, b = 5, c = 5;(الف)  
 ۶۶ اگر متغیرهای X,Y,Z,K را به صورت روبرو تعریف کنیم کدام انتساب زیر صحیح نیست؟  
 int a = b = c = 5;(د) int x; long y; byte k; short z;  
 x = z;(ج) k = z;(ب) y = x;(الف)  
 z = k;(د)

```
float s = 25.54352368f;
Console.WriteLine(s);
```

۶۷ خروجی دستورات روبرو کدام است؟  
 25.5435237 (ب) 25.54352 (الف)  
 25.5435 (د) 25.54352368 (ج)

۶۸ خروجی دستور روبرو کدام گزینه است؟  
 Console.WriteLine(35 % (-28 + 45 / 7 \* 2) + 5 << 1);  
 -10 (د) 4(ج) 30 (ب) 16 (الف)

```
if (a > b ^ b < 20)
    Console.WriteLine("#");
else
    if (a < b && b > 15)
        Console.WriteLine("*");
    else
        Console.WriteLine("$");
```

۶۹ در برنامه ی روبرو مقدار a , b کدام گزینه باشد تا کاراکتر \$ چاپ شود؟  
 a=10 , b=16 (الف) a=10 , b=25 (ب)  
 a=7 , b=8(د) a=30 , b=8 (ج)

۷۰ حاصل اجرای دستور روبرو کدام گزینه است؟  
 Console.Write('\u0041' + 2);  
 A2 (الف) 412(ب) 67(ج) (د) خطا دارد

```
switch (27 >> 1 << 1 )
{
    case 27:
        Console.WriteLine("A");break;
    case 26:
        Console.WriteLine("B");break;
    case 29:
        Console.WriteLine("C");break;
    default :
        Console.WriteLine("D");break;
}
```

۷۱ خروجی کدام دستور m2k خواهد بود؟  
 Console.WriteLine("m\u0032k");(الف)  
 Console.WriteLine("m" + "2" + "k"); (ب)  
 Console.WriteLine("m" + '\x0032' + "k");(ج)  
 ۷۲ خروجی برنامه روبرو کدام گزینه است؟  
 A (الف) B (ب) C (ج) D (د)

```
int c;
for (c = 10; c > 0; --c);
    Console.WriteLine(c);
```

۷۳ خروجی برنامه روبرو کدام گزینه است؟  
 الف) اعداد 10 تا 1  
 ب) اعداد 10 تا 0  
 ج) عدد 0  
 د) برنامه خطا دارد.

```
int a,tedad=0;
for ( ; ; )
{
    a = int.Parse(Console.ReadLine());
    if (a % 2 == 0 && a % 5 == 0)
        break;
    if (a % 2 == 0 || a % 3 == 0)
        continue;
    tedad++;
}
Console.WriteLine(tedad);
```

۷۴ در برنامه روبرو اگر ورودی به ترتیب زیر باشد خروجی را محاسبه کنید؟(چپ به راست)  
 8 25 14 9 17 30

جاهای خالی را پر کنید:

• به منظور کمپایل نمودن برنامه نوشته شده به زبان C# توسط Net Framework. از فایل ..... استفاده می کنیم.

- به کمک متد ..... می توان پنجره خروجی را پاک نموده و توسط متد ..... می توان حالت وقفه در اجرای برنامه ایجاد نمود.
- متد پیش فرض و آغازگر هر پروژه در C# ..... نام دارد.
- کلاس ها در داخل ..... دسته بندی شده و هر کلاس شامل تعدادی ..... می باشد.
- به محیطی که توسط یک زبان برنامه نویسی به منظور نوشتن دستورات، خطایابی و اجرای برنامه فراهم شده است، ..... گویند.

### سوالات برنامه نویسی:

- برنامه ای بنویسید عددی را بگیرد (چهار رقمی) مجموع ارقام زوج آن را نمایش دهد.
- برنامه ای بنویسید عددی اعشاری از نوع float بگیرد سپس بخش صحیح و اعشار آن را جداگانه نمایش دهد.
- برنامه ای بنویسید ۳ عدد بگیرد آنها را از کوچک به بزرگ بصورت مرتب شده نمایش دهد.
- برنامه ای بنویسید دو عدد a و b را به همراه یکی از عملگرهای % / \* - + بگیرد حاصل عبارت محاسباتی را نمایش دهد.
- برنامه ای بنویسید A , B را بگیرد سپس عدد کوچکتر را به توان عدد بزرگتر رسانیده و حاصل را نمایش دهد.
- برنامه ای بنویسید حقوق یک کارمند را بگیرد سپس براساس اطلاعات زیر مالیات را محاسبه و به همراه حقوق دریافتی نمایش دهد
  - اگر حقوق کمتر از ۸۰۰۰۰۰ تومان باشد معاف از مالیات
  - حقوق بین ۸۰۰۰۰۰ تا ۱۲۰۰۰۰۰ هفت درصد مالیات
  - حقوق بیش از ۱۲۰۰۰۰۰ ده درصد مالیات

$$s=1-2+3-4+5-6+\dots+N$$

- برنامه ای بنویسید n را بگیرد حاصل عبارت روبرو را نمایش دهد.
- برنامه ای بنویسید که به تعداد نامشخص از کاربر عدد دریافت کرده (خاتمه دریافت عدد، وارد نکردن هیچ مقداری در ورودی می باشد)، سپس در انتها اعداد دریافت شده را به همراه بیشترین و میانگین آنها نمایش دهد.
- برنامه ای بنویسید که عدد n را از کاربر دریافت کند و سپس اول بودن آن را تشخیص دهد. ( به عددی اول گفته می شود که تعداد مقسوم علیه های آن فقط دو تا باشد)

$$\frac{1}{2} - \frac{1}{4} + \frac{1}{6} - \frac{1}{8} + \dots + \frac{1}{2N}$$

۱۰) برنامه ای بنویسید که عدد N را دریافت کنید و حاصل سری زیر را محاسبه و نمایش دهد؟

### فعالیت کارگاهی (مطالعه آزاد):

`Console.WriteLine("This is\na test! ");`

۱) دستور روبرو را در سی شارپ اجرا کنید و نتیجه را مشاهده کنید:

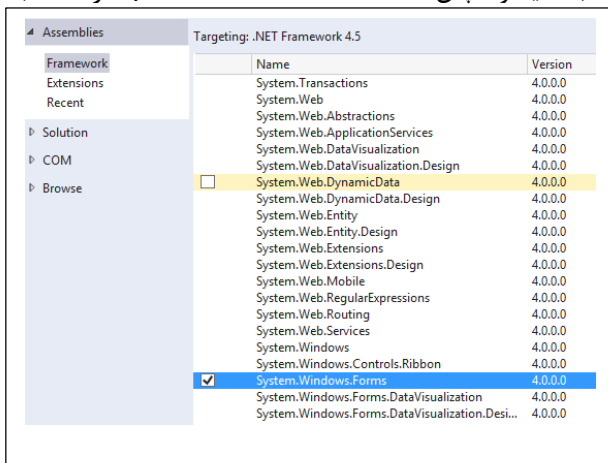
اگر در یک رشته کاراکتر n وجود داشته آنگاه .....

در دستور فوق بجای n کاراکتر b را قرار دهید و نتیجه را مشاهده کنید. چه نتیجه ای می گیرید؟

`Console.WriteLine("This is\\a test! ");`

دستور روبرو را نیز اجرا کنید و نتیجه گیری کنید؟

۲) نمایش کادر پیام: از منوی Projects گزینه Add Reference را انتخاب کنید و سپس System Windows Forms را انتخاب کنید.



سپس در سی شارپ using زیر را اضافه کنید:

`using System.Windows.Forms;`

اکنون برنامه زیر را بنویسید و اجرا کنید؟

```
MessageBox.Show("سلام");
MessageBox.Show("1394", "سال نو مبارک");
```

۳) تولید اعداد تصادفی: در زبان سی شارپ امکان تولید اعداد تصادفی وجود دارد. برای اینکار از کلاس random به صورت زیر استفاده می شود. برنامه زیر سه عدد تصادفی بین 0 تا 1000 تولید می کند.

```
Random R = new Random() ;
Console.WriteLine(R.Next(1000));
Console.WriteLine(R.Next(1000));
Console.WriteLine(R.Next(1000));
```

سوال) خروجی دستورات روبرو را بدست آورید؟

```
int a=5, b=3, c;
c = a++ - b-- + a * b;
Console.WriteLine(c);
```