

پودمان دوم

مدیریت مجموعه داده

هر مسئله دارای بخش‌های مختلفی، شامل داده، عملیات، حالت‌های شرط و تکرار است. تشخیص هرکدام از این بخش‌ها به حل درست مسئله منجر می‌شود. یک‌زبان برنامه‌نویسی برای پیاده‌سازی مسئله برای هر یک از این موارد، دستوراتی دارد که باید از خوانایی و سهولت استفاده برخوردار باشند. در برخی از مسئله‌ها نیاز به ایجاد ساختار تکرار و یا تعدادی داده هم‌نوع داریم که نمایش آنها در ساختار آرایه بسیار مناسب و کارا است. آرایه‌ها شامل داده‌هایی هم‌نوع و هم‌نام هستند که در برخی مسائل نظیر جست‌وجو و مرتب‌سازی استفاده از آنها ضروری است. زبان برنامه‌نویسی دارای امکانات بسیار وسیع و متنوعی برای کار با انواع دستورات شرط، تکرار و آرایه است. در این فصل معرفی و کاربرد انواع ساختار تکرار، آرایه، فنون جست‌وجو و مرتب‌سازی داده‌ها را فرا خواهید گرفت.

واحد یادگیری ۳

■ شایستگی کار با ساختار تکرار

آیا تا به حال پی برده اید

- چرا در برخی برنامه‌ها، دستورات تکرار می‌شوند؟
 - چگونه می‌توان در برنامه، اجرای چند دستور را به تعداد معین تکرار کرد؟
 - چگونه می‌توان تعیین کرد تا زمان برقراری یک شرط اجرای چندین دستور تکرار شود؟
 - ساختار تکرار چگونه به بهینه‌سازی مسئله کمک می‌کند؟
- هدف از این واحد شایستگی به‌کارگیری ساختار تکرار در برنامه‌های کنسول است.

استاندارد عملکرد

با استفاده از دانش ساختار تکرار در برنامه‌نویسی، مسئله را تحلیل و در صورت نیاز برنامه را با استفاده از ساختار تکرار کدنویسی و خطایابی کند.

حلقه‌ها

- آیا معلم از شما خواسته است تا جمع نمرات یک کلاس ۱۵ نفره را محاسبه کنید؟
 - آیا در زنگ ورزش ۱۰ بار دور حیاط مدرسه دویده‌اید؟
 - آیا در یک مسابقه فوتبال به عنوان یک گروه، فرصت زدن ۵ پنالتی به گروه مقابل را داشته‌اید؟
- همه اینها شما را به یاد کارهای تکراری می‌اندازد. هرگاه یک یا چند عمل تکراری داشته باشیم، می‌توانیم از ساختاری به نام حلقه استفاده کنیم. این ساختار به شما کمک می‌کند تا یک یا چند دستور را یکبار بنویسید؛ ولی چندین بار اجرا کنید.
- هنگامی که تعداد تکرار دستورات مشخص باشد حلقه معین است و در مواردی که از ابتدا تعداد تکرار دستورات مشخص نیست حلقه نامعین است.

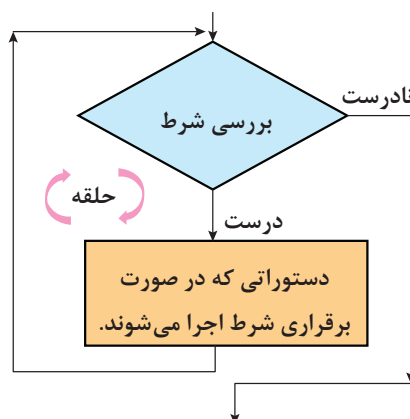
به مثال‌های زیر توجه کرده و با کمک هنرآموز خود، مشخص کنید کدام یک از آنها معین یا نامعین هستند.

مثال	نوع حلقه (معین یا نامعین)
یک نجار روزانه ۱۲ صندلی می‌سازد.	
تا زمانی که دبیر ورزش سوت پایان را بزند همه هنرجویان چندین نرمش کششی انجام می‌دهند.	
در یک ناوایی بربری، ناوا در هر تنور ۶۰ عدد نان می‌پزد.	
یک نقاش ساختمان در یک روز تعدادی دیوار با ابعاد مختلف را رنگ می‌زند.	

فعالیت گروهی



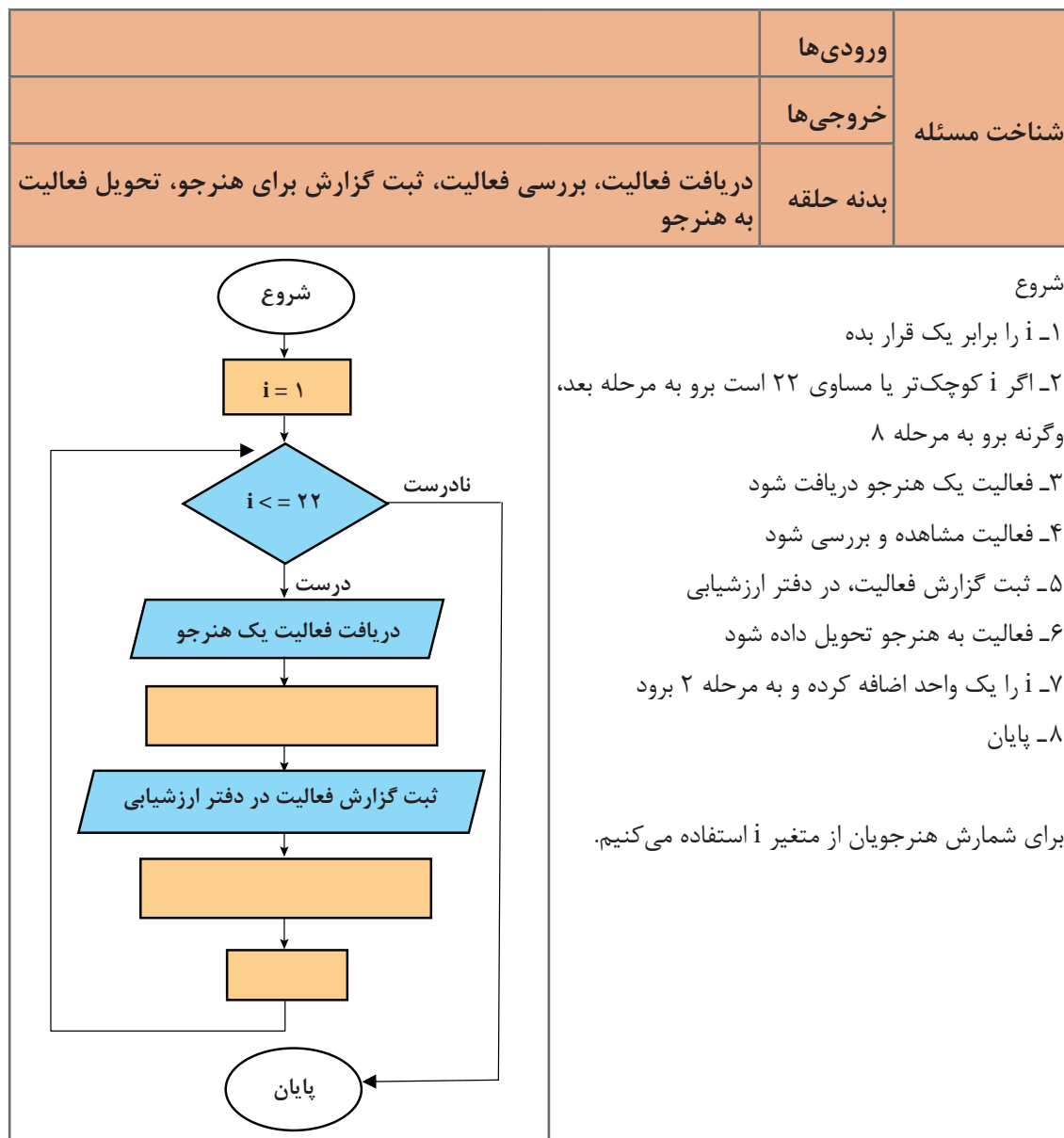
روندنامای حلقه:



برای آشنایی بیشتر با حلقه به مثال زیر توجه کنید:

مثال: یک هنرآموز، فعالیت منزل هنرجویان را تحویل گرفته، پس از بررسی گزارش هر کدام را در دفتر ارزشیابی خود وارد می‌کند (تعداد هنرجویان ۲۲ نفر است).

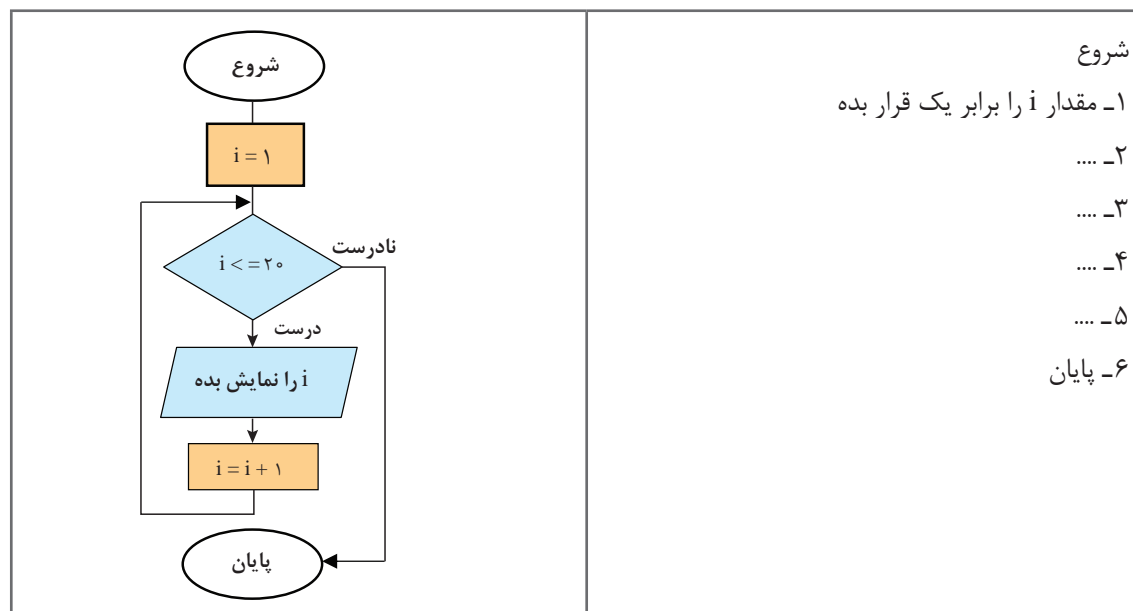
- دریافت فعالیت یک هنرجو
- مشاهده و بررسی فعالیت
- ثبت گزارش فعالیت، در دفتر ارزشیابی
- تحویل فعالیت بررسی شده به هنرجو
- تکرار مراحل بالا



روندنمای بالا را با توجه به الگوریتم برنامه تکمیل کنید.



مسئله: می‌خواهیم سیستم نوبت‌دهی یک مطب دندانپزشکی را مکانیزه کنیم، به‌طوری‌که برای نوبت‌دهی به ۲۰ بیمار خود از شماره‌های چاپ‌شده ۱ تا ۲۰ استفاده کند.
با توجه به روندنما الگوریتم زیر را کامل کنید.



- اگر تعداد بیماران ۳۰ نفر باشد، چه تغییری لازم است؟
- روندنما و الگوریتم را تغییر دهید تا تعداد بیماران را از ورودی دریافت کند.

فعالیت کارگاهی



جدول ارزشیابی شایستگی‌های غیر فنی، ایمنی و بهداشت و توجهات زیست محیطی



شایستگی‌ها	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
شایستگی‌های غیر فنی	مسئولیت‌پذیری - تصمیم‌گیری - استدلال - حل مسئله - تفکر خلاق - نقش در گروه - زبان فنی	قابل قبول	انجام وظایف محوله در گروه - بازگرداندن تنظیمات IDE به تنظیمات اولیه پس از استفاده از آن - ارائه دلیل برای انتخاب ساختار تکرار و دستورات مورد استفاده در برنامه - ارائه راه‌حل خلاقانه برای مسئله	۲
	رعایت ارگونومی - کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات)			
توجهات زیست محیطی	حفاظت از تجهیزات کارگاه			
نگرش	دقت در تشخیص نوع حلقه مورد نیاز برای برنامه - استفاده از ابزارهای خطیابی برای رفع خطای برنامه	غیر قابل قبول	توجه به ایمنی و بهداشت محیط کارگاه	۱

● این شایستگی‌ها در ارزشیابی پایانی واحد یادگیری باید مورد توجه قرار گیرند.

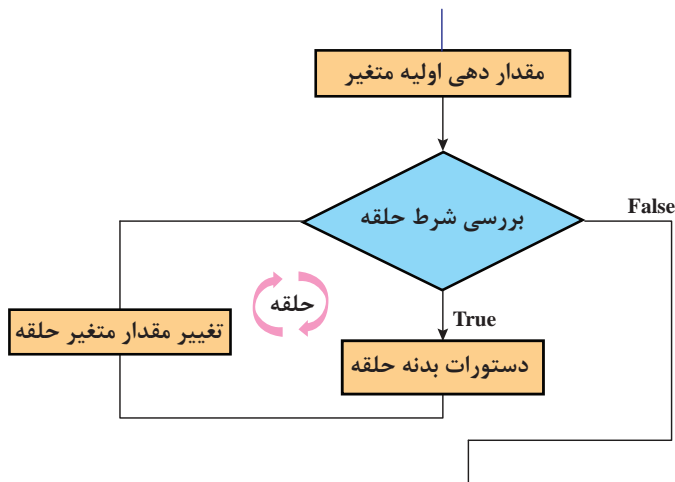
ارزشیابی مرحله ۱



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
حل مسئله تکرار	مکان: کارگاه استاندارد رایانه تجهیزات: کاغذ - نوشت افزار زمان: ۱۵ دقیقه	بالاتر از حد انتظار	تعیین داده، اطلاعات، ورودی و خروجی در مسئله - تعیین نوع تکرار در الگوریتم مسئله - ارائه راه حل برای مسئله - رسم روندنمای مسئله - توسعه روندنما	۳
		در حد انتظار	تعیین داده، اطلاعات، ورودی و خروجی در مسئله - تعیین نوع تکرار در الگوریتم مسئله - ارائه راه حل برای مسئله	۲
		پایین تر از حد انتظار	تعیین داده، اطلاعات، ورودی و خروجی در مسئله	۱

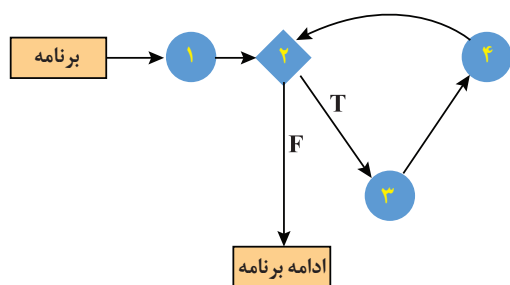
حلقه for

ساختار تکرار for برای پیاده‌سازی حلقه معین مناسب است.
روندنمای حلقه for



شکل کلی for در سی شارپ

۱ {
 ۲ {
 ۳ {
 ۴ {
 for (تغییر مقدار متغیر حلقه ; شرط حلقه ; مقداردهی اولیه متغیر)
 {
 دستوراتی که باید در حلقه تکرار شود
 }
 }
 }
 }



شکل ۱- ترتیب اجرای دستورات حلقه for

ابتدا مقداره‌ی اولیه متغیر انجام‌شده (۱)، در صورت صحیح بودن شرط حلقه (۲) دستورات بدنه حلقه (۳) اجرا می‌شود، پس از اجرای دستورات بدنه حلقه، مقدار متغیر حلقه تغییر می‌کند (۴) و دوباره شرط حلقه بررسی می‌شود. با نادرست شدن شرط حلقه کنترل برنامه به دستورات پس از حلقه منتقل می‌شود. مطابق شکل ۱ دستورات مراحل ۲، ۳ و ۴ تا برقراری شرط تکرار می‌شوند.

کارگاه ۱ تبدیل روندنما به برنامه

می‌خواهیم روندنمایی که برای مطب دندانپزشکی طراحی کردیم را به برنامه تبدیل کنیم.

۱ یک پروژه جدید به نام Counter ایجاد کنید.

۲ قطعه کد زیر را در متد Main() وارد کرده، برنامه را اجرا کنید.

```
int i;
```

```
for (i=1; i<=20 ; i++)
```

```
    Console.WriteLine (i);
```

۳ برنامه را طوری تغییر دهید که برای یک درمانگاه خیریه با تعداد بیماران بیشتر قابل استفاده باشد.

شماره‌های مورد نیاز از ۱۰۰ تا ۹۹۹ است.

۴ برنامه را طوری تغییر دهید که حداکثر تعداد بیماران را از ورودی دریافت کند.

کارگاه ۲ بررسی شرایط حلقه تکرار معین

برای نظافت کارگاه و کلاس، هنرجویان یک کلاس ۳۰ نفره را به دو دسته با شماره‌های زوج و فرد تقسیم کرده‌ایم. هنرجویانی که در دفتر هنرآموز با شماره زوج ثبت شده‌اند برای نظافت کلاس و هنرجویان با شماره فرد برای نظافت کارگاه انتخاب شده‌اند. می‌خواهیم برنامه‌ای بنویسیم که شماره هنرجویانی که برای نظافت کلاس انتخاب شده‌اند را به ترتیب در خط‌های جداگانه چاپ کند.

۱ یک پروژه جدید به نام Even ایجاد کنید.

۲ قطعه کد زیر را در متد Main() وارد کرده، خروجی را بررسی کنید.

برای سرعت برنامه‌نویسی در VS، پس از نوشتن دستور for دو بار کلید tab را فشار دهید و دستور را ویرایش کنید.

```
byte i;
```

```
for (i = 2; i <= 30; i += 2)
```

```
    Console.WriteLine (" {0,5}", i);
```

یادداشت



۳ پس از پرانتز دستور **for** علامت ; قرار داده، خروجی را بررسی کنید.

نتیجه را برای هم‌گروهی و هنرآموز خود توضیح دهید.

۴ قطعه کد مرحله ۲ را به صورت زیر تغییر داده، نتیجه را با مرحله قبل مقایسه کنید.

```
byte i;  
for (i = 2; i <= 30; i++)  
{  
    i++;  
    Console.WriteLine ("{0,5}", i);  
}
```

۵ بررسی کنید در صورتی که آکولادهای قطعه کد بالا حذف شوند خروجی چه تغییری می‌کند؟

۶ قطعه کد مرحله ۲ را طوری تغییر دهید تا شماره هنرجویانی که باید کارگاه را نظافت کنند چاپ کند.

۷ در عبارت منطقی، علامت <= را به >= تبدیل کنید، خروجی چه تغییری می‌کند؟

آیا در VS با دو بار فشردن کلید tab پس از دستور **if** و **switch**، شکل کامل دستور نوشته می‌شود؟

کنجکاو



فعالیت کارگاهی



خروجی قطعه کدهای زیر را با کمک هم‌گروهی خود و بدون استفاده از رایانه بنویسید، سپس آن را اجرا و نتیجه را مقایسه کنید.

حلقه	خروجی برنامه از نظر شما	خروجی برنامه پس از اجرا
<code>int i; for (i=5;i >=1;i --) Console. Write (i);</code>		
<code>int i; for (i=0;i<10;i + =3) Console. WriteLine (i);</code>		
<code>int i; for (i=10;i >=0;i- = 2) Console. WriteLine (i);</code>		

آیا متغیر حلقه می‌تواند از نوع **float** یا **char** باشد؟

کنجکاو





هنرآموزی می‌خواهد ده نفر از هنرجویان یک کلاس را به گروه‌های ۲ نفره تقسیم کند، او قصد دارد در هر گروه به ترتیب یک نفر از اول فهرست و یک نفر از آخر فهرست را قرار دهد. برنامه‌ای بنویسید که شماره هنرجویان هر گروه را در خروجی نمایش دهد.

کارگاه ۳ کار با متغیر خارج بلاک

- ۱ می‌خواهیم برنامه‌ای بنویسیم که بازی هُپ را برای مضارب ۵ شبیه‌سازی کند. در بازی هُپ، هر جا که به مضرب عدد تعیین شده می‌رسیم باید پیام هُپ را چاپ کنیم. یک پروژه جدید به نام Game ایجاد کنید.
- ۲ قطعه کد زیر را در متد Main() وارد کرده، خروجی را بررسی کنید.

```
int i;
for (i = 1; i <= 100; i++)
    if (i % 5 == 0)
        Console.WriteLine ("Hop");
    else
        Console.WriteLine (i);
```

- برنامه را اجرا کنید. عملکرد برنامه چیست؟
- ۳ متغیر شمارنده حلقه را در داخل حلقه اعلان کنید.

```
for (int i = 0; i <= 100; i++)
```

برنامه را اجرا کنید.

- ۴ دستور زیر را به انتهای برنامه اضافه کنید تا مقدار نهایی متغیر i پس از اتمام حلقه چاپ شود. دلیل خطای زیر چیست؟ به کمک هنرآموز خود راه‌حل آن را پیدا کنید.

```
Console.WriteLine ("Final i :"+i);
```

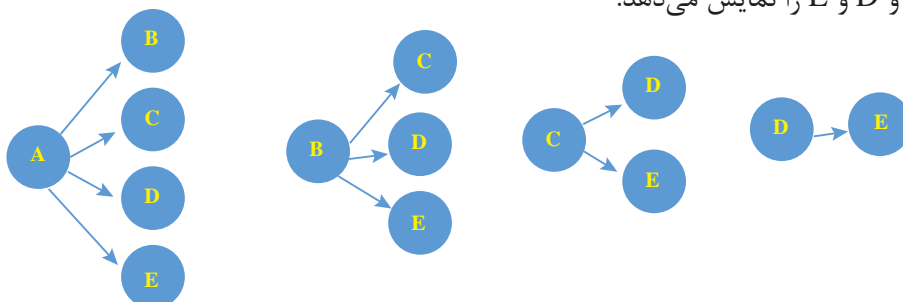
The name 'i' does not exist in the current context

- ۵ برنامه را برای مضارب ۳ بازنویسی کنید.

کارگاه ۴ محاسبه مجموع

دبیر تربیت‌بدنی هنرستان جابر بن حیان قصد دارد به مناسبت دهه فجر مسابقات فوتبال دوره‌ای بین کلاسی برگزار کند. این هنرستان ۵ کلاس درس دارد. می‌خواهیم برنامه‌ای بنویسیم که تعداد بازی‌های دوره‌ای برای این ۵ گروه را محاسبه کند.

الگوریتم برنامه: در بازی دوره‌ای همه گروه‌ها باهم بازی می‌کنند. شکل ۲ نمودار نحوه بازی پنج گروه A و B و C و D و E را نمایش می‌دهد.



شکل ۲- نمودار بازی‌های دوره‌ای

طبق نمودار بازی‌های دوره‌ای، تعداد بازی برابر با مجموع $4+3+2+1$ است که با یک حلقه `for` می‌توان آن را محاسبه کرد.

❶ پروژه به نام `NumberOfGames` ایجاد کنید.

❷ قطعه کد زیر را در متد `Main` وارد کنید.

```
int numberOfGame = 0;
for (int i = 4; i >= 1; i--)
    numberOfGame += i;
Console.WriteLine("Number Of Games:" + numberOfGame);
```

برای جمع کردن مقادیر متغیر حلقه از متغیر `numberOfGame` استفاده می‌کنیم.

برنامه را اجرا کنید. چه مقداری در خروجی نمایش داده می‌شود؟

❸ برنامه را به صورت زیر تغییر دهید و برنامه را اجرا کنید.

```
int numberOfGame = 0;
for (int i = 4; i >= 1; numberOfGame += i, i--);
Console.WriteLine("Number Of Games:" + numberOfGame);
```

آیا در خروجی تغییری صورت گرفته است؟

❹ برنامه را طوری تغییر دهید که تعداد کل بازی‌های لیگ برتر را نمایش دهد.

تعداد گروه‌های لیگ برتر را ۱۶ در نظر بگیرید.

❺ برنامه را طوری تغییر دهید که تعداد گروه‌ها را از ورودی دریافت کند.

کارگاه ۵ خروج زودرس از حلقه

می‌خواهیم بازی حدس ماه تولد را بنویسیم. کاربر اول شماره ماه تولد خود را وارد کرده، کاربر دوم باید شماره ماه را حدس بزند. با هر حدس اشتباه یک امتیاز از کاربر دوم کم می‌شود و حداکثر ۵ بار اجازه دارد تا شماره ماه را حدس بزند. در آخر برنامه امتیاز بازیکن دوم نمایش داده می‌شود. امتیاز اولیه ۵ است.

❶ یک پروژه جدید به نام `MonthGuess` ایجاد کنید.

❷ کد صفحه بعد را تکمیل کرده، در متد `Main()` وارد کنید و برنامه را اجرا کنید.

در کد زیر متغیرهای موردنیاز برنامه را اعلان کرده، دستورات دریافت آنها از ورودی را تکمیل کنید.

```
.....
int score= 5;
Console. Write ("Enter number your month:");
month = .....
Console. Clear();
for (byte i = 1; i <= 5; i++)
{
    Console. Write ("your guess?");
    guess = .....
    if (guess == month)
    {
        Console. ForegroundColor= ConsoleColor. Green;
        Console. WriteLine ("you win");
        Console. WriteLine ("your score: {0}", score);
        break;
    }
    else
    {
        score -= 1;
        Console. WriteLine ("try again !");
    }
}
```

در مواقعی که نیاز دارید برنامه زودتر از موعد از حلقه خارج شود می‌توانید از دستور **break** استفاده کنید. **۳ خروجی برنامه را با مقادیر زیر Trace کنید.**

مقدار month را یازده وارد کنید.			
i	guess	score	خروجی
۱	۳		
۲	۸		
۳	۲		
۴	۴		
۵	۵		

مقدار month را پنج وارد کنید.			
i	guess	score	خروجی
۱	۳		
۲	۸		
۳	۵		

۴ برنامه را اصلاح کنید.

پس از اجرای برنامه و Trace متوجه شدید که با حدس اشتباه در بار پنجم مشابه دفعات قبل پیام **try again!** دیده می‌شود، برنامه را طوری تغییر دهید که پس از ۵ بار حدس اشتباه کاربر دوم، پیام مناسب دیگری چاپ شده، شماره ماه تولد را نمایش دهد.

۵ قطعه کد زیر را در متد **Main()** وارد کرده، نتیجه را با قطعه کد مرحله ۲ مقایسه کنید.

```
Console.WriteLine(" Enter number your month");
string month = Console.ReadLine();
Console.Clear();
for (string guess = ""; guess!=month; guess= Console.ReadLine())
    Console.WriteLine ("type your guess");
    Console.WriteLine ("you win, month is" + month);
```

۶ برنامه مرحله ۵ را طوری تغییر دهید که پس از موفق شدن کاربر دوم، تعداد حدس‌های زده شده را با پیام مناسب چاپ کند.

جدول زیر را به کمک هم‌گروهی خود تکمیل کنید.

فعالیت گروهی



خروجی برنامه پس از اجرا	خروجی برنامه از نظر شما	قطعه کد
		<pre>for (int k = 10; k <= 15;) { Console.WriteLine(k); K++; }</pre>
		<pre>int i = 1; for (; i<=10; i++) { Console.ForegroundColor = ConsoleColor.Cyan; Console.WriteLine("Iran"); i++; }</pre>
		<pre>for (;;) { Console.ForegroundColor = ConsoleColor.Cyan; Console.WriteLine("Iran"); }</pre>



برای قطعه کد زیر پس از اجرا و بررسی نتیجه، یک مسئله کاربردی بنویسید.

```
for (int i = 0; i <= 99; i += 3)
    Console.WriteLine("{0,3}", i);
```

ارزشیابی مرحله ۲



نمره	استاندارد (شاخص‌ها/داوری / نمره‌دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	مراحل کار
۳	تبدیل روندنمای مسئله تکرار به برنامه - trace کردن حلقه for در برنامه - خطایابی و رفع خطای برنامه حلقه for - خروج زودرس از حلقه	بالاتر از حد انتظار	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	ایجاد برنامه با حلقه for
۲	تبدیل روندنمای مسئله تکرار به برنامه - trace کردن حلقه for در برنامه	در حد انتظار		
۱	تعیین اجزای ساختار حلقه for	پایین‌تر از حد انتظار		

فیلم شماره ۱۱۱۰۲: اجرای گام به گام برنامه

فیلم



فعالیت کارگاهی



- پس از مشاهده فیلم، برنامه MonthGuess را با کمک
کلید F10 گام به گام اجرا کنید. تغییرات متغیرها را در
جدول زیر یادداشت کنید.

month			
i	guess	score	خروجی

- برنامه کارگاه ۳ را با استفاده از پنجره local گام به گام اجرا کنید. چرا پس از اجرا و اتمام کار حلقه متغیر i
از پنجره local حذف می‌شود؟



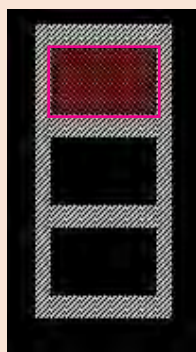
- هنرستان دخترانه پروین اعتصامی هر سال به مناسبت میلاد حضرت معصومه (س) و روز دختر به هنرجویانی که نام آنها معصومه است هدیه می‌دهد. برنامه‌ای طراحی کنید که با دریافت نام هنرجویان یک هنرستان تعداد هنرجویانی که هم‌نام حضرت معصومه (س) هستند را نمایش دهد. تعداد کل هنرجویان هنرستان در ابتدای برنامه از کاربر دریافت شود.

- برنامه را با کمک کلید F10 گام به گام اجرا کرده، با وارد کردن نام ۵ هنرجو مقدار متغیرهای ورودی و خروجی را مشاهده کرده، در جدول زیر وارد کنید.

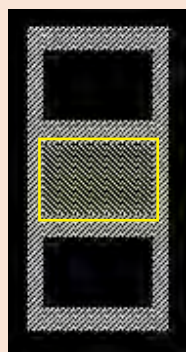
خروجی	count	name	مقدار متغیر حلقه i /
			۱
			۲
			۳
			۴
			۵

- برنامه‌ای بنویسید که چراغ راهنمایی و رانندگی را شبیه‌سازی کند. چراغ راهنمایی و رانندگی ۵ بار از ابتدا تا انتها تکرار شود.

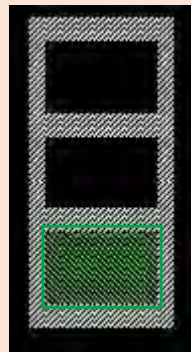
راهنمایی: از کدهای اسکی و متد SetCursorPosition برای ترسیم و از متد Beep برای ایجاد تأخیر بین تغییر رنگ‌ها استفاده کنید.



۳ ثانیه



۲ ثانیه



۷ ثانیه

ارزشیابی مرحله ۳



مرحل کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
خطایابی با استفاده از ابزارها	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۱۰ دقیقه	بالا تر از حد انتظار	خطایابی با استفاده از نقطه قطع، پنجره watch و پنجره Local	۳
		در حد انتظار	خطایابی با استفاده از نقطه قطع و پنجره watch	۲
		پایین تر از حد انتظار	درج و حذف نقطه قطع در برنامه - فعال کردن ابزارهای خطایابی	۱

حلقه while

آیا تا به حال بازی حدس عدد را با دوستان خود انجام داده‌اید؟ در این بازی بازیکن اول عددی را در ذهن خود انتخاب کرده و بازیکن دوم عددهایی را حدس می‌زند تا وقتی که عدد را پیدا کند. به نظر شما آیا از ابتدا مشخص است که بازیکن دوم با چند حدس به جواب می‌رسد؟

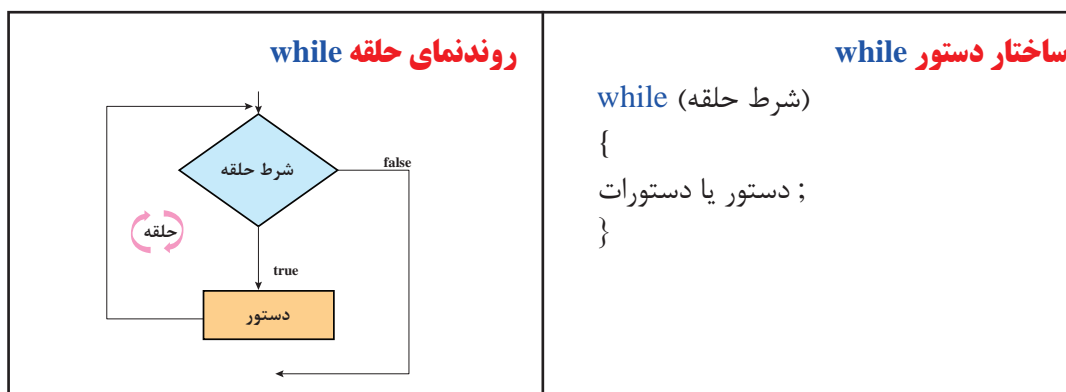
آیا میزان مسافتی که خودرو با یک باک پر از بنزین طی می‌کند، همیشه یکسان است؟

آیا می‌دانید در زمانی که چراغ راهنمایی یک تقاطع سبز است، چه تعدادی خودرو از تقاطع عبور می‌کند؟ تعیین تعداد خودروهایی که از چهارراه تا زمانی که چراغ راهنمایی سبز است، عبور می‌کنند یک حلقه نامعین است. چون تعداد خودروهایی که می‌توانند در آن زمان از چهارراه عبور کنند مشخص نیست.

آیا مشخص است که یک فرد برای دریافت گواهینامه راهنمایی و رانندگی چند بار آزمون می‌دهد؟ همه مثال‌های بالا یک نوع حلقه نامعین هستند، که از ابتدا تعداد تکرار مشخص نیست. در این گونه موارد اغلب از دستور **while** استفاده می‌شود.

با کمک هم‌گروهی و هنرآموز خود دو مثال دیگر برای حلقه نامعین بزنید.

فعالیت گروهی



در صورت درست بودن شرط حلقه، دستور یا دستورات بدنه حلقه اجرا می‌شود و شرط خروج از حلقه **while** نادرست بودن شرط حلقه است.

کارگاه ۶ به کار گیری حلقه while

می‌خواهیم برنامه‌ای که برای مطب دندانپزشکی با حلقه **for** طراحی کرده بودیم را این بار با حلقه **while** شبیه‌سازی کنیم.

۱ یک پروژه جدید به نام **CounterWhile** ایجاد کنید.

۲ قطعه کد زیر را در متد **Main()** وارد کنید.

```
int i = 1;
while (i <= 20)
{
    Console.WriteLine(i);
    i++;
}
```

برنامه را اجرا کرده، نتیجه را بررسی کنید.

۳ آکولادهای حلقه را حذف کرده، برنامه را اجرا کنید.

خروجی را با مرحله قبل مقایسه کنید.

۴ دستورات داخل حلقه را به یک دستور تبدیل کنید.

۵ پس از شرط دستور **while** سمی کالن قرار دهید. چه تغییری در نتیجه رخ می‌دهد؟

۶ دستور مربوط به افزایش شمارنده حلقه (**i++**) را در قطعه کد مرحله ۳ حذف کرده، برنامه را اجرا کنید.

۷ برنامه را طوری تغییر دهید تا تعداد بیماران از ورودی دریافت شود.

با کمک هم گروهی خود معادل هر یک از قطعه کدها را با دستور خواسته شده بنویسید و آنها را اجرا کنید.

نتیجه	قطعه برنامه با حلقه while	قطعه برنامه حلقه for
	<pre>int i = 100; while (i <= 1000) { Console.WriteLine(i); i++; }</pre>	
		<pre>for (int i = 20; i >= 0; i-=2) Console.WriteLine(i);</pre>

فعالیت کارگاهی





- عملکرد قطعه کد زیر چیست؟ این برنامه را با دستور `while` بنویسید.

```
float avg, sum=0;
int count=1;
Console. Write ("Enter mark {0} =", count);
float mark = float.Parse (Console.ReadLine());
for (; mark >= 0; count++)
{
    sum = sum + mark;
    Console. Write ("Enter mark {0} =", count + 1);
    mark = float.Parse (Console.ReadLine());
}
avg = sum / (count - 1);
Console. WriteLine("avg" + avg);
```

- قطعه کدهای زیر را Trace کرده، اولین و آخرین مقدار چاپ شده را در جدول زیر بنویسید.

دستور	اولین مقدار در خروجی	آخرین مقدار در خروجی
<pre>int i = 99; while (i >= 1) Console.WriteLine (- i);</pre>		
<pre>int i = 100; while (i <= 1000) Console.WriteLine (++i);</pre>		

کارگاه ۷ یافتن بزرگ‌ترین مقدار

یک شرکت خدماتی با هدف تشویق کارمندان خود می‌خواهد به فردی که بیشترین ساعت کاری را در ماه اسفند داشته است، مبلغ ۲۰۰۰۰۰ تومان بن خرید هدیه دهد. می‌خواهیم برنامه‌ای بنویسیم که در انتهای ماه ساعت‌های کاری هر کارمند را دریافت کرده، بیشترین ساعت کاری را چاپ کند.

۱ یک پروژه جدید به نام `Max` ایجاد کنید.

۲ متغیرهای استفاده شده در قطعه کد زیر را اعلان کرده، قطعه کد زیر را تکمیل و برنامه را اجرا کنید.

```
.....
Console. ForegroundColor = ConsoleColor.Blue;
Console. WriteLine("\nEnter hour");
```

```
.....
hour = .....
max=hour;
```

```
while (hour > 0)
{
    if (hour > max)
        max = hour;
    Console.WriteLine ("\nEnter 0 for Exit");
    Console.Write ("enter hour: ");
    hour = int.Parse (Console.ReadLine());
}
Console.WriteLine("max= {0}", max);
```

شرط خروج از حلقه در این برنامه چیست؟

۳ برنامه را طوری تغییر دهید که کمترین ساعت کاری را نمایش دهد.

۴ برنامه را طوری تغییر دهید که بیشترین و بیشترین ساعت کاری را نمایش دهد.

- برنامه نوشته شده در کارگاه ۳ را با استفاده از حلقه **while** نوشته، اجرا کنید.
- ماشین حساب کنسولی همانند شکل روبه‌رو ایجاد کنید.

فعالیت کارگاهی



آنچه آموختیم:

برداشت



ارزشیابی مرحله ۴

مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
ایجاد برنامه با حلقه while	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالا تر از حد انتظار	تبدیل روندنمای مسئله تکرار به برنامه - trace کردن حلقه while در برنامه - خطایابی و رفع خطای برنامه حلقه while - تبدیل حلقه while for به while	۳
		در حد انتظار	تبدیل روندنمای مسئله تکرار به برنامه - trace کردن حلقه while در برنامه	۲
		پایین تر از حد انتظار	تعیین اجزای ساختار حلقه while	۱



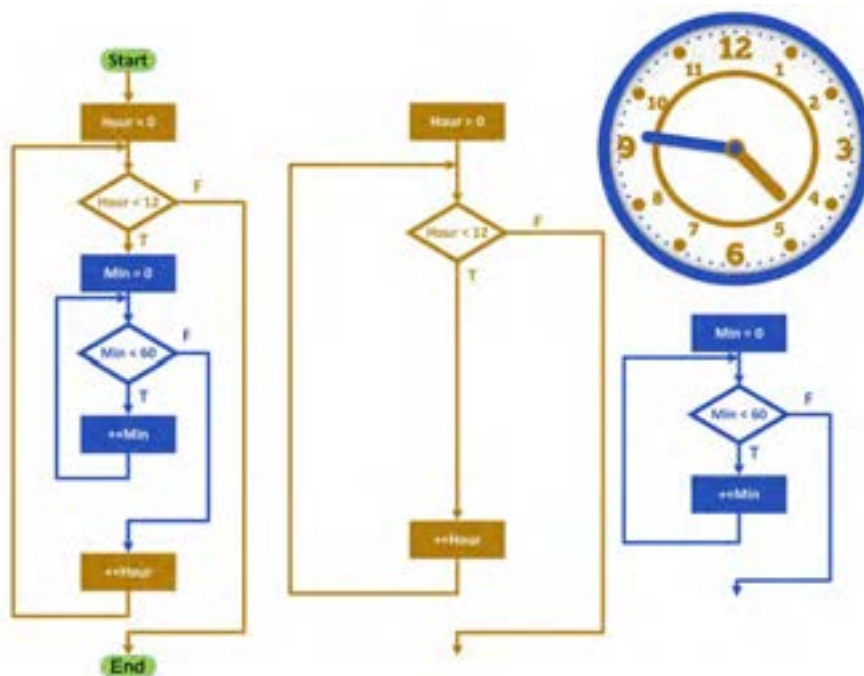
حلقه‌های متداخل

در برخی از مواقع، یک دستور تکرار را در داخل دستور تکرار دیگری به کار می‌بریم. به عبارت دیگر هنگامی که در داخل یک حلقه، حلقه دیگری قرار داشته باشد، حلقه‌های تودرتو یا متداخل نامیده می‌شوند.

دبیر ورزش از هنرجویان خود خواسته است تا پنج بار دور حیاط مدرسه دویده، در پایان هر دور سه حرکت بارفیکس انجام دهند.

می‌خواهیم در یک مجتمع ۱۰ طبقه‌ای که هر طبقه دارای ۸ واحد است، غذای نذری پخش کنیم. چند غذا برای این مجتمع نیاز است؟

همه این مثال‌ها کارهای تکراری است که در درون کار تکراری دیگری قرار دارند و کاربرد حلقه‌های متداخل را بیان می‌کنند.



به کمک هم گروهی خود شکل را توصیف کنید.

فعالیت گروهی



کارگاه ۸ کاربرد حلقه‌های متداخل

می‌خواهیم برای دانش‌آموزان کلاس سوم برنامه‌ای بنویسیم که جدول ضرب ۵ در ۵ را نمایش دهد.

۱ یک پروژه جدید به نام **Multiple** ایجاد کنید.

۲ قطعه کد زیر را در متد **Main()** وارد کنید.

```

for (int i = 1; i <= 5; i++)
{
    Console.WriteLine("{0,4}", i * 1);
    Console.WriteLine("{0,4}", i * 2);
}
  
```

```

Console.Write ("{0,4}", i * 3);
Console.Write ("{0,4}", i * 4);
Console.Write ("{0,4}", i * 5);
Console.WriteLine();
}

```

- برنامه را اجرا کنید. آیا خروجی یک جدول ضرب را نشان می‌دهد؟
- ۳ با توجه به اعداد مشخص شده یک تا پنج، در کد مرحله ۲ با استفاده از دستور حلقه **for** دستورهای **Console.Write** را به یک دستور تبدیل کنید.
- دستور **Console.Write** چند بار اجرا می‌شود؟
- ۴ برای تنظیم نمایش جدول ضرب به جای جا نگهدار از دستور **SetCursorPosition** استفاده کنید.
- ۵ برنامه را با کمک کلید **F10** خط به خط اجرا کنید.

فعالیت گروهی



با کمک هم گروهی خود بررسی کنید دستور **Console.WriteLine** در قطعه کد زیر چند بار اجرا می‌شود؟

```

for (int j = 1; j <= 4; j++)
    for (int k = 1; k <= j; k++)
        Console.WriteLine ("Iran");

```

فعالیت منزل



سیستم نوبت‌دهی مطب دندانپزشکی را طوری تغییر دهید که برای یک هفته که شامل شش روز کاری است نوبت‌دهی انجام دهد.

کارگاه ۹ توسعه حلقه متداخل

تعداد اسکناس ۵ هزار تومانی	تعداد اسکناس ۲ هزار تومانی
۸	۵
	۱۰

آیا به خودپرداز جهت دریافت وجه نقد مراجعه کرده‌اید؟ فکر می‌کنید خودپرداز چگونه پرداخت انواع اسکناس را انجام می‌دهد؟ اگر خودپردازی دارای اسکناس‌های ۲ و ۵ هزار تومانی باشد و شما درخواست ۵۰ هزار تومان وجه نقد کنید، چگونه پرداخت را انجام می‌دهد؟ چه روش‌های پرداختی می‌تواند داشته باشد؟ جدول رو به رو را تکمیل کنید.

خودپرداز از شمارش برای پرداخت وجه نقد استفاده می‌کند و با توجه به موجودی اسکناس‌ها پرداخت را تعیین می‌کند.

الگوریتم: اگر X تعداد اسکناس‌های ۲ هزار تومانی و Y تعداد اسکناس‌های ۵ هزار تومانی باشد برای اینکه پرداخت معادل ۵۰ هزار تومانی باشد شرط روبه‌رو باید برقرار باشد:

$$2000X + 5000Y = 50000$$

به کمک حلقه متداخل همه مقادیر X و Y را بررسی می‌کنیم هرکدام در شرط بالا صدق کرد، می‌تواند یک پرداخت برای ۵۰ هزار تومانی باشد.

۱ یک پروژه به نام ATM ایجاد کنید.

۲ قطعه کد زیر را در بخش Main بنویسید.

```
for (int X = 1; X <= 25; X++)
    for (int Y = 1; Y <= 10; Y++)
        if (2000 * X + 5000 * Y == 50000)
            Console.WriteLine ("X={0},Y={1}", X, Y);
```

برنامه را اجرا کنید. آیا با جدولی که شما کامل کرده‌اید مطابقت دارد؟ فکر می‌کنید خودپرداز کدام پرداخت را انجام می‌دهد؟

چرا مقدار نهایی X مقدار ۲۵ و مقدار نهایی Y مقدار ۱۰ در نظر گرفته شده است؟

۳ برنامه چند حالت را با مقادیر مختلف (X,Y) آزمایش می‌کند تا به جواب برسد؟

۴ برنامه را با فرض اینکه خودپرداز اسکناس ۱۰ هزار تومانی هم دارد، بنویسید.

۵ برنامه را طوری تغییر دهید که وجه نقد درخواستی کاربر را از ورودی دریافت کنیم.

- خودپرداز پرداخت اسکناس را بر اساس کمترین تعداد اسکناس انجام می‌دهد. برنامه‌ای بنویسید که وجه درخواستی کاربر را بگیرد و پرداخت خودپرداز را تعیین کند. برای مثال اگر کاربر ۲۰۰ هزار تومان درخواست کرد سه چک پول ۵۰۰۰۰ و چهار اسکناس ۱۰۰۰۰ و دو اسکناس ۵۰۰۰ پرداخت کند.

فعالیت منزل



- می‌خواهیم برنامه‌ای بنویسیم که تعداد گروه‌های حاضر در بازی دوره‌ای را بگیرد و جدول بازی‌های این گروه‌ها را در خروجی نمایش دهد. نام گروه‌ها را با A و B و C و... نام‌گذاری کنید. برای مثال اگر تعداد گروه‌ها ۵ باشد خروجی به صورت زیر است. (متغیرهای حلقه را از نوع داده char تعریف کنید)

AB - AC - AD - AE - BC - BD - BE - CD - CE - DE

ارزشیابی مرحله ۵



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
ایجاد برنامه با حلقه‌های متداخل	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	نوشتن روندنمای مسئله تکرار متداخل - تبدیل روندنما به برنامه - اجرای گام‌به‌گام حلقه متداخل - خطایابی و رفع خطای حلقه متداخل	۳
		در حد انتظار	نوشتن روندنمای مسئله تکرار متداخل - تبدیل روندنما به برنامه - اجرای گام‌به‌گام حلقه متداخل	۲
		پایین تر از حد انتظار	تعیین نیاز به حلقه متداخل	۱

معیار شایستگی انجام کار:

کسب حداقل نمره ۲ از مراحل حل مسئله تکرار، ایجاد برنامه با حلقه for و ایجاد برنامه با حلقه‌های متداخل
کسب حداقل نمره ۲ از بخش شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش
کسب حداقل میانگین ۲ از مراحل کار

جدول ارزشیابی پایانی

شرح کار:

- ۱ حل مسئله تکرار
۲ ایجاد برنامه با حلقه for
۳ خطایابی با استفاده از ابزارها
۴ ایجاد برنامه با حلقه while
۵ ایجاد برنامه با حلقه‌های متداخل

استاندارد عملکرد:

با استفاده از دانش ساختار تکرار در برنامه‌نویسی، مسئله را تحلیل و در صورت نیاز برنامه را با استفاده از ساختار تکرار کدنویسی و خطایابی کند.

شاخص‌ها:

شماره مرحله کار	شاخص‌های مرحله کار
۱	تعیین داده، اطلاعات، ورودی و خروجی در مسئله - تعیین نوع تکرار در الگوریتم مسئله - تعیین نوع حلقه - ارائه راه‌حل برای مسئله - رسم روندنمای مسئله - توسعه روندنما
۲	انتخاب ساختار تکرار موردنیاز - تعیین و مقاردهی اجزای ساختار تکرار for - تعیین بلاک حلقه for در برنامه - تبدیل روندنمای مسئله تکرار به برنامه - trace کردن حلقه for در برنامه - خطایابی و رفع خطای برنامه حلقه for - خروج زودرس از حلقه
۳	انتخاب ابزار خطایابی - درج و حذف نقطه قطع در برنامه - فعال کردن ابزارهای خطایابی - اجرای گام‌به‌گام برنامه با ابزار خطایابی تعیین‌شده
۴	انتخاب ساختار تکرار موردنیاز - تعیین اجزای ساختار حلقه while - تعیین حداقل تعداد تکرار حلقه while - تبدیل روندنمای مسئله تکرار به برنامه - trace کردن حلقه while در برنامه - خطایابی و رفع خطای برنامه حلقه while - تبدیل حلقه for به while
۵	تشخیص نیاز به حلقه متداخل - نوشتن روندنمای مسئله تکرار متداخل - تعیین نوع حلقه داخلی و خارجی - تبدیل روندنمای مسئله تکرار متداخل به برنامه - خطایابی و رفع خطای حلقه متداخل - اجرای گام‌به‌گام حلقه متداخل

شرایط انجام کار و ابزار و تجهیزات:

مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها

تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است - نوشت افزار - کاغذ

زمان: ۹۰ دقیقه (حل مسئله تکرار ۱۵ دقیقه - ایجاد برنامه با حلقه for ۲۰ دقیقه - خطایابی با استفاده از ابزارها ۱۰ دقیقه - ایجاد برنامه با حلقه while ۲۰ دقیقه - ایجاد برنامه با حلقه‌های متداخل ۲۰ دقیقه)

معیار شایستگی:

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	حل مسئله تکرار	۲	
۲	ایجاد برنامه با حلقه for	۲	
۳	خطایابی با استفاده از ابزارها	۱	
۴	ایجاد برنامه با حلقه while	۱	
۵	ایجاد برنامه با حلقه‌های متداخل	۲	
<p>شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش:</p> <p>مسئولیت‌پذیری - تصمیم‌گیری - استدلال - حل مسئله - تفکر خلاق - نقش در گروه - زبان فنی رعایت ارگونومی - کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات) حفاظت از تجهیزات کارگاه</p> <p>دقت در تشخیص نوع حلقه موردنیاز برای برنامه - استفاده از ابزارهای خطایابی برای رفع خطای برنامه</p>			
میانگین نمرات			
			*

* حداقل میانگین نمرات هنرجو برای قبولی و کسب شایستگی، ۲ است.

واحد یادگیری ۴

■ شایستگی کار با آرایه و داده‌های شمارشی

آیا تا به حال پی برده اید

- اسامی هنرجویان یک کلاس و نمرات آنها چگونه در حافظه نگه‌داری و پردازش می‌شود؟
- چگونه می‌توان مجموعه‌ای از واژه‌ها و معانی آنها را بر اساس حروف الفبا مرتب کرد؟
- چگونه می‌توان معنی یک واژه را در بین واژه‌های مرتب‌شده بر اساس حروف الفبا جست‌وجو کرد؟
- چگونه می‌توان نمرات هنرجویان را همراه با نام آنها به صورت نزولی مرتب کرد؟
- اسامی روزهای هفته، نام کلیدهای صفحه‌کلید، اسامی رنگ‌ها و انواع مدارک تحصیلی را چگونه نگه‌داری کنیم؟

هدف از این واحد شایستگی استفاده از آرایه و داده شمارشی است.

استاندارد عملکرد

با استفاده از دانش ساختار آرایه و داده‌های شمارشی مسئله را تحلیل کرده، برنامه را ایجاد و خطایابی کند.

آرایه

می‌دانیم برای نگه‌داری داده‌ها در حافظه، از متغیر استفاده می‌شود. برای مثال برای نگه‌داری نام یک هنرجو، متغیری از نوع رشته تعریف می‌شود.

سعید
مجید
محمد
علی
حسین

فهرست اسامی هنرجویان کلاس را در نظر بگیرید. اگر در برنامه‌ای نیاز باشد که نام تمام هم‌کلاسی‌های خود را دریافت و نگه‌داری کنید، چه خواهید کرد؟ آیا به تعداد هم‌کلاسی‌هایتان متغیر تعریف می‌کنید؟ در این مثال، به جای اینکه اسامی هر یک از هنرجویان را در متغیرهای جدا ذخیره کنیم، می‌توانیم جدولی به شکل روبه‌رو تعریف کرده، اسامی را در آن نگه‌داری کنیم:

شماره	نام
۱	سعید
۲	مجید
۳	محمد
۴	علی
۵	حسین

این جدول شامل نام پنج هنرجو است. اسامی هنرجویان در ردیف‌های مجزا نوشته شده است. برای این جدول، یک نام مناسب انتخاب می‌کنیم. همچنین برای دسترسی به نام هنرجویان، در کنار هر نام، یک شماره قرار می‌دهیم. برای مثال، نام نفر سوم، محمد است.

فعالیت کارگاهی



- روندنمایی رسم کنید که اسامی پنج هنرجو را دریافت کرده، سپس اسامی را از آخر به اول نمایش دهد. **روش انجام کار:** اگر برای هر هنرجو متغیر جداگانه‌ای تعریف کنیم، نیاز به تعریف پنج متغیر است. دریافت مقدار از ورودی و قرار دادن در متغیر باید برای هر پنج اسم انجام شود؛ یعنی نام اول را دریافت و در متغیر اول قرار دهیم، سپس نام دوم را دریافت و در متغیر دوم قرار دهیم و این روال را تا متغیر پنجم ادامه دهیم. در پایان متغیر پنجم، چهارم، سوم، دوم و اول را نمایش دهیم. - برنامه این الگوریتم را به زبان سی‌شارپ بنویسید.

در این مثال، اگر تعداد اسامی ۳۰ نفر باشد:

چه تعداد دستور خواندن از ورودی و نوشتن روی صفحه نمایش لازم است؟ آیا می‌توان از دستور for برای خواندن ورودی‌ها و نوشتن اسامی استفاده کرد؟ برای چنین مسئله‌هایی که با داده‌های زیاد سروکار دارند، چه روشی را پیشنهاد می‌کنید؟

در زبان‌های برنامه‌نویسی مانند سی‌شارپ برای نگهداری تعدادی داده هم نوع، از مفهوم آرایه استفاده می‌شود. آرایه، فهرستی از عناصر هم نوع است که تحت یک نام مشترک کنار هم در حافظه ذخیره می‌شوند. هر خانه آرایه، یک **عنصر آرایه** نامیده می‌شود. برای تفکیک و دسترسی به هر عنصر، از یک عدد صحیح به نام **اندیس** استفاده می‌شود. در زبان سی‌شارپ، اندیس اولین عنصر آرایه صفر است، اندیس عنصر بعدی یک و به همین ترتیب اندیس عناصر یکی‌یکی اضافه می‌شوند.

فهرست اسامی ۵ هنرجو را می‌توان به شکل زیر در نظر گرفت:

اولین اندیس	۰	۱	۲	۳	۴	آخرین اندیس
عنصر	سعید	مجید	محمد	علی	حسین	تعداد عناصر ۵ عدد است

تعداد عناصر آرایه را طول آرایه می‌نامند. آرایه بالا دارای طول ۵ است و عناصر آن از ۰ تا ۴ شماره‌گذاری شده است.

تعریف آرایه

به یاد دارید برای تعریف یک متغیر از نوع رشته‌ای به شکل زیر عمل می‌کردیم:

```
string name;
```

در زبان سی‌شارپ می‌توان تعریف و ایجاد آرایه را در دو مرحله انجام داد. برای تعریف یک آرایه که از نوع رشته‌ای است، در مرحله اول، مشابه تعریف متغیر رشته‌ای به شکل زیر عمل می‌شود:

```
string[] name;
```

در این دستور متغیر `name`، آرایه‌ای از نوع رشته‌ای اعلان شده است؛ اما تعداد خانه‌های آرایه مشخص نیست و حافظه‌ای برای آرایه تخصیص نیافته است.

در مرحله دوم با استفاده از عملگر `new` و مشخص کردن اندازه آرایه، حافظه مناسب به آرایه اختصاص داده‌شده، آرایه ایجاد می‌شود.

```
name = new string[5];
```

در این دستور، `name` نام آرایه‌ای است که در مرحله قبل اعلان شده بود. `string` نوع عناصر و ۵ تعداد عناصر آرایه است. اندیس عناصر آرایه از ۰ تا ۴ است.

یک پروژه جدید ایجاد کنید و دستور اعلان آرایه و دستور ایجاد آرایه `name` را بنویسید. بررسی کنید محتوای هر عنصر آرایه چیست؟

```
string[] name;
name = new string[5];
```

```
name {string[5]}
```

برای دیدن محتوای عناصر آرایه، برنامه را با کلید `F10` اجرا کنید. سپس با قرار دادن اشاره‌گر ماوس روی نام آرایه و کلیک روی علامت + مقادیر آرایه را مشاهده کنید.

فعالیت کارگاهی



مرحله تعریف آرایه و ایجاد آرایه را می‌توانیم در یک دستور و به‌صورت زیر بنویسیم:

```
string[] name = new string[5];
```

شکل کلی تعریف و ایجاد آرایه

[تعداد عناصر آرایه] نوع داده new = نام آرایه [نوع داده];

فعالیت کارگاهی



- آرایه‌ای به نام months برای نگهداری اسامی ماه‌های سال تعریف کنید.
- آرایه‌ای به نام salary برای نگهداری حقوق ۳۰۰ کارمند تعریف کنید.
- آرایه‌ای به نام vowels برای نگهداری حروف صدادار انگلیسی تعریف کنید.
- آرایه‌ای به نام average برای نگهداری معدل هنرجویان کلاس تعریف کنید.
- آرایه‌ای به نام lamp برای نگهداری وضعیت روشن و خاموش بودن پنج لامپ تعریف کنید.

مقداردهی عناصر آرایه

برای مقداردهی عناصر آرایه روش‌های مختلفی وجود دارد. یک روش مقداردهی، استفاده از دستور انتساب است. برای مثال دستور زیر مقدار maryam را در خانه سوم آرایه رشته‌ای name قرار می‌دهد:

```
name[2] = "maryam";
```

کنجکاوی



چرا برای مراجعه به خانه سوم آرایه، از اندیس ۲ استفاده شده است؟

شکل کلی مقداردهی عناصر آرایه با دستور انتساب

مقدار = [اندیس] نام آرایه

برداشت



آنچه آموختم:

-
-
-

کارگاه ۱ تعریف و مقداردهی آرایه

هنرجویی قصد دارد که قطعات رایانه را جداگانه خریداری کرده، در منزل آن را مونتاژ کند. می‌خواهیم برنامه‌ای بنویسیم که یک آرایه تعریف کرده، اسامی قطعات خریداری‌شده را در آن قرار دهیم.

۱ پروژه جدیدی به نام **Computer** ایجاد کنید.

۲ کد زیر را در متد **Main()** بنویسید.

```
string[] computer = new string[5];
```

آرایه **computer** دارای ۵ خانه از نوع رشته‌ای است.

۳ اسامی قطعات رایانه را در خانه‌های اول، دوم، سوم، چهارم و پنجم آرایه قرار دهید.

```
computer[0] = "CPU";
```

```
computer[1] = "MainBoard";
```

```
computer[2] = "RAM";
```

```
computer[3] = "HDD";
```


```
computer[4] = "DVD";
```

۴ برای ذخیره کردن اسامی قطعات دیگر، چه تغییری در کد باید بدهیم؟

۵ برنامه را با فشردن کلید **F10** یا از طریق منوی **Debug** گزینه **Step Over** اجرا کنید.

مشاهده خواهید کرد با هر بار فشردن کلید **F10**، یک دستور با رنگ زرد مشخص می‌شود. این دستور، دستوری است که اجرا خواهد شد. تا دستور `computer[2] = "RAM";` را اجرا کرده، اشاره‌گر ماوس را روی دستور قرار دهید. در کادر ظاهر شده، روی علامت + کنار نام آرایه کلیک کنید.

```
static void Main(string[] args)
{
    string[] computer = new string[5];
    computer[0] = "CPU";
    computer[1] = "MainBoard";
    computer[2] = "RAM";
    computer[3] = "HDD";
    computer[4] = "DVD";
}
```



کنید شکل ۳ مشاهده خواهد شد.

شکل ۳- مشاهده محتوای عناصر آرایه در زمان اجرا

چرا مقدار برخی از عناصر برابر **null** است؟

۶ برای نگه‌داری قیمت قطعات رایانه، آرایه **price** را متناظر با آرایه نام قطعات اعلان کنید.

۷ عناصر آرایه **price** را مقداردهی کنید.

کد زیر را تکمیل کرده، به متد **Main** اضافه کنید.

```
price[0] = 200000;
```

```
price[1] = 1500000;
```

```
price[2] = .....
```

```
.....
```

```
.....
```

۸ برنامه را با کلید **F10** تا رسیدن به دستور مقداردهی **price[2]** اجرا کنید و محتوای آرایه **price** را مشاهده کنید.

محتوای آرایه **price** چیست؟ چرا؟

۹ دستور زیر را جایگزین کد مرحله ۲ و ۳ کنید.

```
string[] computer= new string[] {"CPU", "MainBoard", "RAM", "HDD", "DVD"};
```

روش دیگر مقداردهی عناصر آرایه، مقداردهی در هنگام اعلان و ایجاد آرایه است در این حالت باید مقادیر اولیه آرایه مشخص باشد. در این روش آرایه مانند قبل اعلان می شود ولی در انتهای اعلان آرایه و در بین علامت های { } مقدار عناصر آرایه به ترتیب معین می شود.

شکل کلی تعریف و مقداردهی آرایه در یک دستور

{مقدار آخرین عنصر, ..., مقدار دوم, مقدار اول} [] نوع داده new = نام آرایه [] نوع داده

۱۰ دستور زیر را جایگزین کد مرحله ۹ کنید.

```
string[] computer= new string[4] {"CPU", "MainBoard", "RAM", "HDD", "DVD"};
```

به جای عدد ۴ عدد ۵ را قرار دهید. خطای رخ داده چیست؟

۱۱ دستور زیر را جایگزین کد مرحله ۹ کنید.

```
string[] computer= {"CPU", "MainBoard", "RAM", "HDD", "DVD"};
```

در حالتی که آرایه دارای مقادیر اولیه مشخص است، می توان بدون استفاده از عملگر new آرایه را به صورت زیر ایجاد کرد:

شکل کلی تعریف و مقداردهی آرایه بدون استفاده از عملگر new

{مقدار آخرین عنصر, ..., مقدار دوم, مقدار اول} = نام آرایه [] نوع داده

برنامه را اجرا کرده، نتیجه را با مراحل قبل مقایسه کنید.

۱۲ دستوری برای تغییر مقدار عنصر شماره ۳ آرایه computer به "Keyboard" به کد اضافه کنید.

برنامه را با F10 اجرا کرده، محتوای عناصر آرایه computer را مشاهده کنید.

یادداشت



برای مقداردهی عناصر آرایه در دستور اعلان آرایه، برای جلوگیری از خطای حاصل از کم یا زیاد بودن تعداد مقادیر انتسابی به آرایه، بهتر است در بین علامت { } تعداد عناصر آرایه را ننویسیم.

کارگاه ۲ دریافت عناصر آرایه از کاربر

می خواهیم برنامه ای بنویسیم که قیمت قطعات رایانه را از ورودی دریافت کند.

۱ پروژه جدیدی به نام ComputerPrice ایجاد کنید.

۲ آرایه price را برای نگه داری قیمت قطعات رایانه اعلان کنید.

۳ دستور زیر را برای دریافت قیمت CPU بنویسید.

```
Console.WriteLine ("Enter Price");
```

```
price[0] = int.Parse (Console.ReadLine());
```

۴ برای دریافت قیمت قطعات دیگر برنامه را تکمیل کنید.

۵ برنامه را تغییر دهید تا در ابتدا نام قطعه (عناصر آرایه computer) و سپس قیمت آن از ورودی دریافت شود.

- برنامه را با کلید F10 اجرا کنید و نحوه پر شدن خانه‌های آرایه را مشاهده کنید.
- ۶ برنامه را برای دریافت نام و قیمت ۱۰ قطعه تغییر دهید.
- ۷ تعداد قطعات رایانه را از ورودی دریافت کنید.
- در این صورت تعداد عناصر آرایه را کاربر تعیین خواهد کرد. کد زیر را تکمیل کنید.

```
string[] computer;
int[] price;
int count = .....; // دریافت تعداد قطعات
computer = new string[count];
price = .....; // ایجاد آرایه
.....
.....
.....
.....
```

آنچه آموختیم:

۱.
۲.
۳.

برداشت



جدول ارزشیابی شایستگی‌های غیر فنی، ایمنی و بهداشت و توجهات زیست محیطی

شایستگی‌ها	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
شایستگی‌های غیر فنی	مسئولیت‌پذیری - تصمیم‌گیری - استدلال - حل مسئله - تفکر خلاق - نقش در گروه - زبان فنی	قابل قبول	انجام وظایف محوله در گروه - بازگرداندن تنظیمات IDE به تنظیمات اولیه پس از استفاده از آن - ارائه دلیل برای انتخاب نوع داده آرایه و دستورات مورد استفاده در برنامه - ارائه راه حل خلاقانه برای مسئله	۲
ایمنی و بهداشت	رعایت ارگونومی - کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات)	غیر قابل قبول	توجه به ایمنی و بهداشت محیط کارگاه	۱
توجهات زیست محیطی	حفاظت از تجهیزات کارگاه			
نگرش	دقت در تشخیص ساختار مناسب داده هنگام تعریف آرایه در برنامه			

• این شایستگی‌ها در ارزشیابی پایانی واحد یادگیری باید مورد توجه قرار گیرند.



ارزشیابی مرحله ۱



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
اعلان آرایه	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE روی آن نصب است زمان: ۱۰ دقیقه	بالاتر از حد انتظار	اعلان آرایه - ایجاد آرایه - مقداردهی عناصر آرایه - تعیین نوع آرایه با توجه به داده‌ها - رفع خطای اندیس خارج از محدوده	۳
		در حد انتظار	اعلان آرایه - ایجاد آرایه - مقداردهی عناصر آرایه	۲
		پایین تر از حد انتظار	اعلان آرایه	۱

دسترسی به عناصر آرایه

در فهرست اسامی هنجریان، با دانستن شماره هنجرو در فهرست اسامی، به راحتی می‌توان به نام وی دسترسی داشت.

برای دسترسی به خانه‌های آرایه، از نام آرایه به همراه اندیس عنصر به صورت زیر استفاده می‌کنیم:
[اندیس] نام آرایه

برای مثال `price[0]` اولین عنصر آرایه و `price[2]` سومین عنصر آرایه است.
هر متغیر در هر لحظه می‌تواند فقط یک مقدار داشته باشد و با انتساب مقدار جدید، مقدار قبلی آن از بین می‌رود.
در آرایه نیز با انتساب یک مقدار جدید در هر عنصر، مقدار قبلی از بین رفته، مقدار جدید جایگزین خواهد شد.
به دستور زیر توجه کنید:

```
price[2] = 250000;
```

پس از اجرای این دستور، محتوای سومین عنصر آرایه `price` از بین رفته، مقدار 250000 جایگزین خواهد شد.

نمایش عناصر آرایه

برای نمایش محتوای عناصر آرایه می‌توان از متدهای `Write()` یا `WriteLine()` استفاده کرد. برای مثال برای نمایش محتوای عناصر اول و دوم آرایه `price` خواهیم داشت:

```
Console.WriteLine(price[0]);
```

```
Console.WriteLine(price[1]);
```

آیا می‌توان با یک متد `WriteLine()` تمام عناصر آرایه را نمایش داد؟
حاصل اجرای دستور زیر چیست؟

```
Console.WriteLine(price);
```

کنجکاوی



کارگاه ۳ نمایش عناصر آرایه

می‌خواهیم برنامه‌ای بنویسیم که با دریافت شماره یک روز هفته، نام آن را نمایش دهد.

۱ پروژه جدیدی به نام Week ایجاد کنید.

۲ اعلان آرایه week را تکمیل کنید.

```
string[] week = { "saturday", ....., "friday" };
```

آرایه week دارای ۷ عنصر از نوع رشته‌ای برای نگه‌داری اسامی روزهای هفته است.

۳ اعلان آرایه week و کد زیر را در متد Main بنویسید.

```
Console.WriteLine ("Enter the number one day of week (0 - 6): ");
```

```
byte num = byte.Parse (Console.ReadLine ());
```

```
Console.WriteLine (week[num]);
```

برنامه را با اعداد مختلف اجرا کنید.

۴ برنامه را با عدد ۷ اجرا کنید.

چه خطایی رخ می‌دهد؟ علت آن چیست؟

۵ اگر بخواهیم به جای اعداد ۰ تا ۶، اعداد ۱ تا ۷ دریافت شود، چه تغییری باید در برنامه ایجاد کنیم؟

۶ دستوراتی به برنامه اضافه کنید که اگر شماره وارد شده در محدوده مجاز نبود، پیام مناسب نمایش داده شود.

دسترسی به عنصری از آرایه که وجود ندارد، باعث بروز خطا می‌شود.

یادداشت



پیمایش آرایه

گاهی لازم است به تمام عناصر آرایه مراجعه شود. برای مثال معلم هنگام حضور و غیاب، اسامی را از ابتدا تا انتهای فهرست هنرجویان می‌خواند. به این عمل پیمایش یا مرور آرایه می‌گوییم.

فرض کنید می‌خواهیم اسامی روزهای هفته را نمایش دهیم. یک روش آن است که عناصر آرایه را از ابتدا تا انتها یکی‌یکی نمایش دهیم. به دستورات زیر توجه کنید:

```
Console.WriteLine (week[0]);
```

```
Console.WriteLine (week[1]);
```

```
Console.WriteLine (week[2]);
```

```
Console.WriteLine (week[3]);
```

```
Console.WriteLine (week[4]);
```

```
Console.WriteLine (week[5]);
```

```
Console.WriteLine (week[6]);
```

در این روش باید به تعداد عناصر آرایه از متد WriteLine استفاده شود.

اگر تعداد عناصر آرایه زیاد باشد و بخواهیم تمام عناصر آن را پیمایش کنیم، چه روشی پیشنهاد می‌کنید؟ مقادیر خانه‌های یک آرایه دارای ۱۰۰ عنصر را چگونه نمایش می‌دهید؟



برای پیمایش آرایه‌های بزرگ، از حلقه تکرار استفاده می‌شود. برای مثال دستورات نمایش اسامی روزهای هفته را می‌توان به صورت زیر نوشت:

```
for(int i = 0; i < 7; i++)
    Console.WriteLine(week[i]);
```

برنامه‌ای بنویسید که اسامی روزهای زوج آرایه week را نمایش دهد.

کارگاه ۴ پیمایش آرایه

می‌خواهیم برنامه‌ای بنویسیم که قیمت قطعات رایانه را از ورودی دریافت کند و قیمت کل رایانه را به همراه نام و قیمت قطعات در خروجی نمایش دهد.

- ۱ پروژه جدیدی به نام PriceOfComputer ایجاد کنید.
- ۲ دو آرایه با نام‌های computer و price برای قطعات و قیمت آنها اعلان کنید.
- ۳ با یک حلقه for هم‌زمان دو آرایه computer و price را از ورودی دریافت کنید.
- ۴ کد محاسبه قیمت رایانه را بنویسید.

برای محاسبه قیمت کل قطعات رایانه از متغیر sumPrice استفاده می‌کنیم:

```
long sumPrice = 0;
for (int i = 0; i < price.Length; i++)
    sumPrice += price[i];
```

هر آرایه دارای ویژگی Length است که تعداد عناصر آرایه را مشخص می‌کند. price.Length تعداد عناصر آرایه price است.

- ۵ برنامه را طوری تکمیل کنید که نام، قیمت قطعات و قیمت کل را نمایش دهد.

کارگاه ۵ پیمایش نویسه‌های رشته

می‌توان از رشته به صورت آرایه‌ای از نویسه‌ها استفاده کرد. می‌خواهیم برنامه‌ای بنویسیم که یک مقدار رشته‌ای از ورودی گرفته، نویسه‌های آن را جدا کرده، هر کدام را در یک خط بنویسد و معکوس رشته را نمایش دهد.

- ۱ پروژه جدیدی به نام StringChar ایجاد کنید.
- ۲ دستورات لازم را برای دریافت یک رشته از کاربر بنویسید.
- نام متغیر رشته‌ای را myStr در نظر بگیرید.
- ۳ دستورات زیر را به کد اضافه کنید.

```
for (int i = 0; i < myStr.Length; i++)
    Console.WriteLine (myStr[i]);
```

ویژگی Length طول متغیر رشته‌ای یعنی تعداد نویسه‌های رشته و همچنین اندازه آرایه را تعیین می‌کند. برنامه را اجرا کنید. خروجی آن چیست؟

۴ کد زیر را جایگزین مرحله ۳ کنید.

```
foreach (char c in myStr)
```

```
    Console.WriteLine(c);
```

برنامه را اجرا کنید. نتیجه و کد را با مرحله ۳ مقایسه کنید.

اگر بخواهیم به محتوای تمام عناصر آرایه دسترسی داشته باشیم و آرایه را پیمایش کنیم، می‌توان به‌جای دستور `for`، از دستور `foreach` استفاده کرد.

شکل کلی دستور foreach (نام آرایه `in` متغیر حلقه نوع داده) `foreach` ; دستور

در این دستور از اندیس استفاده نمی‌شود و متغیر حلقه، نقش یک عنصر از آرایه را به عهده دارد؛ بنابراین باید از نوع عناصر آرایه باشد.

پس از کلیدواژه `foreach` در VS دو بار کلید Tab را فشار دهید تا شکل کامل دستور نوشته شود.

یادداشت



۵ برای نمایش معکوس رشته، کد زیر را تکمیل کنید.

```
for (int i = myStr.Length - 1; i >= ..... ; .....)
```

.....

آیا می‌توان این کد را با دستور `foreach` نوشت؟

– در دستور `foreach` منظور از متغیر حلقه چیست؟
– آیا می‌توان به‌وسیله عملگر انتساب (=) مقداری را داخل متغیر حلقه `foreach` قرار داد؟

کنجکاوی



فعالیت منزل



برنامه‌ای بنویسید که سطح شایستگی هنرجویان یک کلاس را به صورت عددی دریافت کرده، به‌وسیله دستور `foreach` تعداد هنرجویانی را که سطح شایستگی لازم را کسب نکرده‌اند یعنی دارای سطح شایستگی ۱ هستند نمایش دهد. بررسی شود که سطح شایستگی دریافت شده یکی از سه سطح ۱، ۲ و ۳ باشد.

ویرایش عناصر آرایه

آرایه متغیری است که می‌تواند تعداد زیادی داده هم‌نوع را نگهداری کند. به‌وسیله حلقه‌های تکرار می‌توانیم عملیات یکسانی روی تمام یا برخی از عناصر آرایه انجام دهیم.

کارگاه ۶ ویرایش عناصر آرایه

می‌خواهیم قیمت محصولات یک فروشنده را از ورودی دریافت کرده، در آرایه ذخیره کنیم. سپس مبلغ مالیات ارزش‌افزوده (VAT) را به قیمت محصولات اضافه کرده، ضمن نمایش قیمت جدید هر محصول، آن را در همان خانه آرایه ذخیره کنیم. تعداد محصولات در ابتدا مشخص نیست اما حداکثر ۵۰ عدد است. با ورود عدد صفر یا عدد منفی به‌عنوان قیمت محصول، دریافت داده پایان می‌یابد.

۱ پروژه جدیدی به نام `ValueAddedTax` ایجاد کنید.

۲ با اعلان آرایه و متغیرهای مناسب، کد زیر را برای دریافت قیمت محصولات تکمیل کنید.

```
for (i = 0; i < 50; i++)
{
    Console.WriteLine ("Enter the price of {0} th product: ", i+1);
    temp = int.Parse (Console.ReadLine());
    if (temp > 0)
        priceList[i] = temp;
    else
        break;
}
```

به وسیله این حلقه، حداکثر ۵۰ عدد به عنوان قیمت محصول دریافت می شود. عدد دریافتی ابتدا بررسی می شود. اگر مثبت بود در آرایه قرار می گیرد و اگر صفر یا منفی بود، دستور `break` باعث خروج از حلقه می شود. تعداد قیمت محصولات وارد شده، به وسیله متغیر `i` نگهداری می شود.

۳ دستورات محاسبه مالیات ارزش افزوده را بنویسید.

```
Console.WriteLine ("Product price plus VAT:");
for (j = 0; j < i; j++)
{
    priceList[j] = (int)(priceList[j] + 0.09 * priceList[j]);
    Console.WriteLine ("Product{0} : {1}", j, priceList[j]);
}
```

در این کد نقش متغیر `i` و `j` چیست؟

در این قطعه کد با نمایش پیام مناسب، مالیات ارزش افزوده ۹ درصد به قیمت هر محصول اضافه شده، در همان خانه آرایه ذخیره می شود.

۴ در قطعه کد ۳ تبدیل صریح به `int` را حذف کنید. چه خطایی رخ می دهد؟ چرا؟

برنامه ای بنویسید که هزینه روزانه یک ماه شما را دریافت کرده، در یک آرایه ذخیره کند. سپس هزینه ماهانه و میانگین هزینه روزانه شما را محاسبه کرده، نمایش دهد.

فعالیت منزل



برداشت



آنچه آموختم:

۱.
۲.
۳.

ارزشیابی مرحله ۲



مرحله کار	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	نتایج ممکن	استاندارد (شاخص‌ها/داوری/نمره دهی)	نمره
دسترسی به عناصر آرایه	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE روی آن نصب است زمان: ۲۰ دقیقه	بالاتر از حد انتظار	نمایش عناصر آرایه با دستور تکرار - ویرایش عناصر آرایه - foreach	۳
		در حد انتظار	نمایش عناصر آرایه با دستور تکرار - ویرایش عناصر آرایه	۲
		پایین‌تر از حد انتظار	نمایش عناصر آرایه	۱

جست‌وجو در آرایه

فرض کنید در دفتر تلفن می‌خواهیم شماره تلفن یک شخص خاص را پیدا کنیم. اگر اسامی دفتر تلفن مرتب نباشد، لازم است از ابتدای دفتر تلفن نام مورد نظر را به ترتیب با تمام اسامی موجود در دفتر تلفن مقایسه کنیم. نام مورد نظر در هر کجای دفتر تلفن می‌تواند باشد، بنابراین عمل مقایسه تا یافتن نام مورد نظر و یا رسیدن به انتهای دفتر تلفن ادامه می‌یابد. به این روش **جست‌وجو**، **جست‌وجوی خطی** یا **ترتیبی** (Liner Search) می‌گویند. جست‌وجو یکی از عملیات رایج و پرکاربرد روی آرایه‌ها است. در جست‌وجو، شماره مکان عنصر مورد جست‌وجو مشخص می‌شود.

پویانمایی شماره ۱۱۱۰۳: جست‌وجوی خطی

فیلم



فعالیت گروهی



یک آرایه نامرتب شامل اسامی ۱۰۰ هنرجو داریم. برای پیدا کردن نام یک هنرجو در این آرایه به روش جست‌وجوی خطی چند مقایسه نیاز است؟ جدول زیر را کامل کنید.

مکان عنصر موردنظر	اول آرایه	وسط آرایه	آخر آرایه	در آرایه نباشد
تعداد مقایسه				



اگر اسامی دفتر تلفن بر اساس حروف الفبا مرتب باشند، به سرعت می توان نام موردنظر را پیدا کرد. در این حالت وسط دفتر را باز می کنیم، اگر نام مورد نظر را دیدیم، جست و جو پایان یافته است، در غیر این صورت با توجه به ترتیب الفبایی و نام موردنظر، در اسامی قبلی یا بعدی دفتر تلفن جست و جو را ادامه می دهیم. به این روش جست و جو، **جست و جوی دودویی (Binary Search)** می گویند.

در هر یک از موارد زیر تعیین کنید از روش جست و جوی خطی یا دودویی استفاده می شود؟ انتخاب روش براساس چه معیاری است؟

- ۱ یافتن کارنامه تحصیلی یک هنرجو در بین سایر کارنامه های مرتب شده براساس نام هنرجویان.
- ۲ پیدا کردن یک کتاب براساس نام آن کتاب در کتابخانه.
- ۳ پیدا کردن یک کتاب براساس کد کتاب در کتابخانه.
- ۴ پیدا کردن نام هنرجویی که در کلاس بالاترین معدل را دارد، اگر اسامی براساس معدل مرتب شده باشد.

کارگاه ۷ پیاده سازی جست و جوی خطی

می خواهیم برنامه ای برای نمایشگاه خودرو بنویسیم که نام خودرو را از ورودی خوانده، قیمت آن را در خروجی نمایش دهد.

۱ پروژه جدیدی به نام **LinearSearch** ایجاد کنید.

۲ برای خودروها و قیمت آنها دو آرایه متناظر اعلان کرده، مقداردهی کنید.

```
string[] car = {"Prid", "Mazda", "Renault", "Peugeot", "Toyota"};
```

```
long[] price = .....
```

۳ برای جست و جو در آرایه **car** کدهای زیر را بنویسید.

```
string item = "Prid";
```

```
for (int i = 0; i < car.Length; i++)
```

```
if (item==car[i])
```

```
{
```

```
Console.WriteLine ("{0} Found in {1}", item, i);
```

```
break;
```

```
}
```

متغیر **item** عنصر مورد جست و جو را نگه داری می کند.

برنامه را اجرا کنید. خروجی برنامه چیست؟

۴ برنامه را طوری تغییر دهید تا قیمت خودروی مورد جست و جو در خروجی نمایش داده شود.

۵ مقدار **item** را برابر **Hyundai** قرار دهید.

برنامه را اجرا کنید. خروجی چیست؟

۶ برای تشخیص یافتن و یا عدم یافتن عنصر، کد را به صورت زیر تغییر دهید.

```
string item = "Prid";
```

```
bool found = false;
```

```
for (int i = 0; i < car.Length; i++)
    if (item==car[i])
    {
        found = true;
        Console.WriteLine ("price {0}={1}", item,price[i]);
        break;
    }
if (found == false)
    Console.WriteLine ("item Not Found");
```

نقش متغیر found در این کد چیست؟

۷ برنامه را طوری تغییر دهید که نام خودرو را از ورودی دریافت کند.

به کمک کلید F10 برنامه را اجرا کنید. با دو مقدار Renault و Hyundai نتیجه برنامه را مشاهده کنید.

برنامه‌ای بنویسید که نام و نمره هنجویان را از ورودی دریافت کرده، در آرایه مناسب ذخیره کند. سپس نام هنجویانی را نمایش دهد که نمره ۲۰ گرفته‌اند.

فعالیت منزل



فیلم



فعالیت گروهی



پویانمایی شماره ۱۱۱۰۴: جست و جوی دودویی

بازی حدس عدد

هنرجویی یک عدد بین ۱ تا ۱۰۰ انتخاب کرده، آن را یادداشت می‌کند. برنده شخصی است که با کمترین تعداد، عدد مورد نظر را حدس بزند.

– شما چه الگوریتمی برای برنده شدن پیشنهاد می‌دهید؟ بهتر است اولین حدس چه عددی باشد؟

– هنرجویی عدد ۵۹ را یادداشت کرده است. در روش جست و جوی دودویی حداکثر با چند مقایسه عدد مورد نظر حدس زده خواهد شد. جدول زیر را کامل کنید.

مقایسه	نتیجه مقایسه	عدد مورد نظر	عدد حدس زده شده (وسط آرایه)	آخرین عدد آرایه	اولین عدد آرایه
۱	عدد بزرگ تر است	۵۹	۵۰	۱۰۰	۱
۲					
۳					
۴					
۵					

– در جست و جوی دودویی اگر عنصر مورد جست و جو در آرایه نباشد، الگوریتم چگونه پایان می‌یابد؟



با در نظر گرفتن عدد انتخابی ۱۲، جدول صفحه قبل را پر کنید.

کلاس Array

کتابخانه غنی NET Framework دارای کلاس‌های آماده فراوانی است. این کلاس‌های آماده و از قبل ایجادشده (Built-in) دارای ویژگی‌ها و متدهایی هستند که انجام عملیات را آسان‌تر کرده، به توسعه سریع‌تر نرم‌افزار کمک می‌کنند. با کلاس Console و برخی متدهای آن مثل WriteLine و ReadLine آشنا هستیم. برای عملیات مختلف روی آرایه‌ها از کلاس آماده Array استفاده می‌کنیم.



هر کلاس (Class) شامل دستوراتی برای انجام کارها است که به آنها متد (Method) می‌گویند. همچنین کلاس دارای داده‌هایی است که به آنها ویژگی (Property) می‌گویند.

کارگاه ۸ متدهای کلاس Array

می‌خواهیم با یک مثال عملکرد متدهای کلاس Array را بررسی کنیم.

۱ یک پروژه جدید به نام ArrayClass ایجاد کنید.

۲ یک آرایه مفروض به صورت زیر اعلان و مقداردهی کنید.

```
int[] a = {20, 18, 10, 15, 50, 80, 10};
```

۳ کدهای زیر را پس از اعلان وارد کنید.

```
int x = 10;
```

```
int find = Array.IndexOf(a, x);
```

```
Console.WriteLine(find);
```

مقدار متغیر find پس از اجرای برنامه چیست؟

عملکرد متد IndexOf چیست؟

۴ اگر $x=60$ قرار دهیم خروجی چه می‌شود؟

۵ کد زیر را به برنامه اضافه کنید.

```
int find = Array.LastIndexOf(a, x);
```

```
Console.WriteLine(find);
```

مقدار متغیر find پس از اجرای این دستور چیست؟

تفاوت عملکرد متد LastIndexOf با متد IndexOf چیست؟

۶ کد زیر را به برنامه اضافه کنید.

```
Array.Sort(a);
```

۷ با دستور foreach عناصر آرایه a را در خروجی نمایش دهید.

برنامه را اجرا کنید. عملکرد متد Sort چیست؟

۸ کد زیر را به برنامه اضافه کرده، برنامه را اجرا کنید.

```
int x = 10;
int find = Array.BinarySearch (a, x);
Console.WriteLine (find);
```

مقدار متغیر find چیست؟

۹ کد زیر را به برنامه اضافه کنید و دوباره با دستور foreach عناصر آرایه a را در خروجی نمایش دهید.

```
Array.Reverse (a);
```

برنامه را اجرا کنید. عملکرد متد Reverse چیست؟

۱۰ کد زیر را اضافه کرده، برنامه را اجرا کنید.

```
find=Array.BinarySearch (a, x);
Console.WriteLine (find);
```

مقدار متغیر find چیست؟

۱۱ جدول زیر را کامل کنید.

جدول ۱- متدهای پرکاربرد کلاس Array

نام متد	کاربرد	شکل کلی
Sort		Array.Sort (نام آرایه)
Reverse		Array.Reverse (نام آرایه)
IndexOf		Array.IndexOf (مقدار، نام آرایه)
LastIndexOf		Array.LastIndexOf (مقدار، نام آرایه)
BinarySearch	یک مقدار را در آرایه صعودی مرتب شده جست و جو کرده، مکان اولین مورد پیدا شده را برمی گرداند. اگر پیدا نشود عدد منفی برمی گرداند.	Array.BinarySearch (مقدار، نام آرایه)

- می خواهیم برنامه ای بنویسیم که با استفاده از یک فهرست انتخاب (منو) عملیات زیر را روی آرایه انجام دهد و برنامه تا زمانی که کاربر گزینه خروج را انتخاب نکرده است، ادامه یابد. گزینه های منو به رنگ سفید روی زمینه آبی نمایش داده شوند.

فعالیت منزل



- | | |
|--|---|
| ۱ دریافت عناصر آرایه Input | ۲ نمایش محتوای عناصر آرایه Show |
| ۳ مرتب کردن عناصر آرایه Sort | ۴ معکوس کردن عناصر آرایه Reverse |
| ۵ جست و جوی خطی یک عنصر در آرایه Linear Search | ۶ جست و جوی دودویی یک عنصر در آرایه Binary Search |
| ۷ خروج Exit | |

ارزشیابی مرحله ۳



نمره	استاندارد (شاخص‌ها/داوری/نمره دهی)	نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	مراحل کار
۳	جست‌وجوی خطی یک عنصر در آرایه - تعیین نتیجه اجرای الگوریتم - جست‌وجوی دودویی روی آرایه - استفاده از متدهای جست‌وجوی خطی و دودویی - معکوس کردن و مرتب‌سازی آرایه - استفاده ترکیبی از متدها - تعیین خروجی متد	بالاتر از حد انتظار	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE روی آن نصب است زمان: ۳۰ دقیقه	به کارگیری متدهای آماده روی آرایه
۲	جست‌وجوی خطی یک عنصر در آرایه - استفاده از متدهای جست‌وجوی خطی و دودویی - معکوس کردن و مرتب‌سازی آرایه - تعیین خروجی متد	در حد انتظار		
۱	استفاده از متدهای جست‌وجوی خطی و دودویی - تعیین خروجی متد	پایین‌تر از حد انتظار		

آنچه آموختم:

برداشت



۱.
۲.
۳.

داده شمارشی (Enumeration)

می‌خواهیم برنامه‌ای بنویسیم که روزهای هفته را بر اساس شماره روز نگه‌داری کند؛ بنابراین لازم است تعداد هفت ثابت به صورت زیر تعریف کنیم:

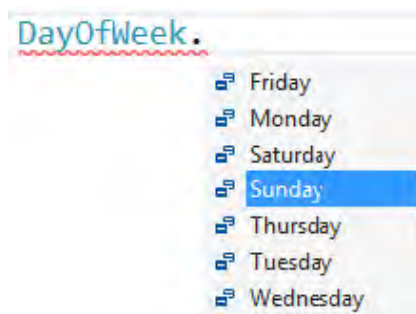
```
Const int Sunday =0;
Const int Monday =1;
Const int Tuesday =2;
Const int Wednesday =3;
Const int Thursday =4;
Const int Friday =5;
Const int Saturday =6;
```

آیا در این روش امکان انجام عملیات محاسباتی روی ثابت‌ها برای به دست آوردن نام دیگر وجود دارد؟ برای مثال می‌توان به یک ثابت دو واحد اضافه کرد و نام دو روز بعد را به دست آورد؟ برای بالا بردن خوانایی برنامه می‌توانیم از نوع داده شمارشی استفاده کنیم. نوع داده شمارشی مجموعه‌ای از چند نام دلخواه است که اجازه می‌دهد به جای اعداد ثابت و بی‌معنی، از واژه‌های بامعنی استفاده کنیم.

در زبان سی شارپ داده‌های شمارشی آماده‌ی بسیاری تعریف شده است (جدول ۲).
جدول ۲- برخی از داده‌های شمارشی آماده‌ی سی شارپ

داده شمارشی	شرح
DayOfWeek	روزهای هفته
ConsoleColor	۱۶ رنگ اصلی
ConsoleKey	کلیدهای صفحه کلید

نوع داده شمارشی **DayOfWeek** شامل اسامی هفت روز هفته با شماره ۰ تا ۶ است.



راهنمای ConsoleKey Enumeration را مشاهده کنید. جدول نمایش داده‌شده شامل چه ستون‌هایی است؟

فعالیت منزل



دسترسی به اعضای داده شمارشی

نام عضو. نوع داده شمارشی

DayOfWeek. Monday

به کد تغییر رنگ زمینه کنسول توجه کنید:

```
Console.BackgroundColor = ConsoleColor.Blue;
```

برای ذخیره اعضای داده شمارشی لازم است که متغیری از نوع داده شمارشی به صورت زیر اعلان کنیم.

نام متغیر نوع داده شمارشی

```
DayOfWeek day;
```

متغیر **day** از نوع داده شمارشی **DayOfWeek** است و می‌تواند با یکی از اعضای نوع داده شمارشی به صورت زیر مقداردهی شود.

; مقدار = نام متغیر

```
day = DayOfWeek.Friday;
```

کارگاه ۹ داده شمارشی رنگ‌ها

می‌خواهیم برنامه‌ای بنویسیم که اسامی ۱۶ رنگ اصلی را نمایش دهد.

۱ پروژه جدیدی به نام **Colors** ایجاد کنید.

۲ دستورات زیر را در متد **Main()** بنویسید:

```
ConsoleColor c;  
for (c = ConsoleColor.Black; c <= ConsoleColor.White; c++)  
    Console.WriteLine(c);
```

۳ برنامه را اجرا کرده، خروجی را مشاهده کنید.

۴ دستوری اضافه کنید که نام هر رنگ را با همان رنگ زمینه نمایش دهد.

۵ کد زیر را به برنامه اضافه کنید.

```
c=0;  
Console.BackgroundColor = c+3;
```

برنامه را اجرا کنید.

با استفاده از نوع داده شمارشی **DayOfWeek** برنامه‌ای بنویسید که اسامی روزهای هفته را همراه با شماره روز نمایش دهد.

فعالیت منزل



کارگاه ۱۰ داده شمارشی کلیدهای صفحه کلید

می‌خواهیم برنامه‌ای بنویسیم که اسامی کلیدهای صفحه کلید را به همراه کد هر یک نمایش دهد.

۱ پروژه جدیدی به نام **KeyBoard** ایجاد کنید.

۲ کد زیر را در متد **Main()** بنویسید.

```
for(int i =0; i < 128; i++)  
    Console.WriteLine (i+": "+(ConsoleKey) i);
```

برنامه را اجرا کرده، خروجی را مشاهده کنید.

۳ کد را تغییر دهید به صورتی که برای نمایش خروجی از جا نگه‌دار استفاده کند.

برای نمایش کد هر کلید، یک فضای پنج ستونی در نظر بگیرید.

برنامه‌ای بنویسید که با دریافت هر کد، کلید معادل آن را نمایش دهد. برنامه با ورود عدد صفر پایان یابد.

فعالیت منزل



تعریف داده شمارشی

در زبان سی شارپ، برنامه‌نویس می‌تواند نوع داده شمارشی جدیدی به‌وسیلهٔ کلیدواژه `enum` تعریف کند.

نام دلخواه `enum`

```
{
    فهرستی از نام‌ها
}
```

شکل کلی تعریف داده شمارشی

محل تعریف داده شمارشی معمولاً خارج از کلاس و در ابتدای برنامه است. روش نام‌گذاری داده شمارشی روش پاسکال است. در فهرست نام‌ها، هر نام با علامت کاما (,) از نام دیگر جدا می‌شود. در مثال زیر نوع داده شمارشی `MonthOfYear` برای نام ماه‌های سال میلادی به‌صورت خلاصه تعریف شده است:

`enum MonthOfYear`

```
{
    Jan, Feb, Mar, Apr, May, Jun, Jul, Aug, Sep, Oct, Nov, Dec
}
```

هر یک از اعضای نوع داده شمارشی معادل یک عدد صحیح ثابت است. این اعداد به‌طور پیش‌فرض از عدد صفر شروع می‌شوند و به ترتیب یک واحد اضافه می‌شوند. البته می‌توان عدد دیگری را به نام‌ها اختصاص داد. در نوع داده شمارشی برای دسترسی به عدد صحیح هر یک از عضوهای آن، به‌صورت زیر عمل می‌کنیم:

نام عضو. نوع داده شمارشی (`int`)

در زیر نوع داده شمارشی `Fall` برای اسامی فصل‌های سال تعریف شده است:
در این تعریف `Winter` معادل با عدد یک و `Spring = 2` است.

`enum Season`

```
{
    Winter = 1,
    Spring,
    Summer,
    Fall
}
```

در نوع داده شمارشی `Season` نام `Fall` معادل با چه عددی است؟ برای جواب به این سؤال، می‌توان از دستور زیر کمک گرفت.

```
Console.WriteLine((int) Season.Fall);
```

کنجکاوی



برداشت



آنچه آموختیم:

۱.
۲.
۳.

می‌خواهیم برنامه‌ای بنویسیم که نمره شایستگی یک واحد یادگیری هنرجو را به صورت عددی گرفته، معادل آن را نمایش دهد.

۱ پروژۀ جدیدی به نام **Assessment** ایجاد کنید.

۲ داده شمارشی **Results** را قبل از متد **Main()** اعلان کنید.

```
enum Results
{
    LowerThanExpected = 1,
    Acceptable = 2,
    HigherThanExpected = 3
}
```

۳ دستورات زیر را در متد **Main()** بنویسید.

```
Console.Write ("Enter Result(1/2/3): ");
int result = int.Parse (Console.ReadLine());
switch(result)
{
    case 1: Console.WriteLine("Result is: {0}",Results.LowerThanExpected);
        break;
    case 2: Console.WriteLine("Result is: {0}",Results.Acceptable);
        break;
    case 3: Console.WriteLine("Result is: {0}",Results.HigherThanExpected);
        break;
    default:
        break;
}
```

برنامه را اجرا کنید.

۴ کد زیر را به جای کد مرحله ۳ قرار دهید و برنامه را اجرا کنید.

```
Console.Write ("Enter Result(1/2/3): ");
int result = int.Parse (Console.ReadLine());
switch (result)
{
    case 1:
    case 2:
    case 3:
        Console.WriteLine ("Result is: {0}",(Results)result);
        break;
    default:
        break;
}
```

برنامه را اجرا کنید. کد و نتیجه را با مرحله ۳ مقایسه کنید.

۵ دستوراتی به برنامه اضافه کنید که اگر شماره وارد شده در محدوده مجاز نباشد، پیام مناسب نمایش داده شود.

برنامه را تعمیم دهید به صورتی که نام هنجویان یک کلاس و نمره یک واحد یادگیری آنها را دریافت کرده، سپس سطح شایستگی همه هنجویان کلاس را نمایش دهد.

فعالیت منزل



فعالیت کارگاهی



– کدام یک از دستورات مقاردهی قطعه کد زیر نادرست است؟ خطای آنها چیست؟

DayOfWeek day;

day = Monday;

day = DayOfWeek.Wednesday;

Console.WriteLine(day);

day = 5;

day = (DayOfWeek) 0;

Console.WriteLine(day);

Console.WriteLine((int)day);

– دستوراتی که خطا دارد را حذف کرده، کد را در متد Main نوشته و اجرا کنید.

ارزشیابی مرحله ۴

نمره	استاندارد (شاخص‌ها/داوری/نمره دهی)		نتایج ممکن	شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...)	مراحل کار
۳	دسترسی به اعضای داده شمارشی آماده - تعریف داده شمارشی - اعلان و مقداردهی متغیر شمارشی - رفع خطاهای استفاده از داده شمارشی		بالا تر از حد انتظار	مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار IDE روی آن نصب است زمان: ۲۰ دقیقه	به کارگیری داده‌های شمارشی
	دسترسی به اعضای داده شمارشی آماده - تعریف داده شمارشی - اعلان و مقداردهی متغیر شمارشی		در حد انتظار		
	دسترسی به اعضای داده شمارشی آماده		پایین تر از حد انتظار		

معیار شایستگی انجام کار:

کسب حداقل نمره ۲ از مراحل اعلان آرایه، دسترسی به عناصر آرایه و به کارگیری متدهای آماده روی آرایه

کسب حداقل نمره ۲ از بخش شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش

کسب حداقل میانگین ۲ از مراحل کار



جدول ارزشیابی پایانی

شرح کار:

۱ اعلان آرایه

۲ دسترسی به عناصر آرایه

۳ به کارگیری متدهای آماده روی آرایه

۴ به کارگیری داده‌های شمارشی

استاندارد عملکرد:

با استفاده از دانش ساختار آرایه و داده‌های شمارشی مسئله را تحلیل کرده، برنامه را ایجاد و خطایابی کند.

شاخص‌ها:

شماره مرحله کار	شاخص‌های مرحله کار
۱	انتخاب نوع آرایه - تعریف آرایه - مقداردهی عناصر آرایه پس از اعلان - مقداردهی عناصر آرایه هنگام اعلان آرایه - تشخیص خطای اندیس خارج از محدوده - رفع خطای اندیس خارج از محدوده
۲	دسترسی به عنصر آرایه تعیین‌شده - دریافت مقدار عناصر آرایه از ورودی - نمایش همه عناصر آرایه - ویرایش عناصر تعیین‌شده - پیمایش آرایه با حلقه for - پیمایش آرایه با حلقه foreach
۳	انتخاب روش جست‌وجو - جست‌وجوی خطی یک عنصر - استفاده از مکانیزم flag در جست‌وجوی خطی - تعیین محل عنصر مورد جست‌وجو در آرایه - جست‌وجوی دودویی عنصر - انتخاب متد - استفاده از متدهای جست‌وجوی خطی و دودویی - معکوس کردن و مرتب‌سازی آرایه - تعیین خروجی متد - استفاده ترکیبی از متدهای آماده روی آرایه
۴	دسترسی به اعضای داده شمارشی آماده ConsoleColor, ConsoleKey و DayOfWeek - اعلان متغیر شمارشی - مقداردهی متغیر شمارشی - تبدیل نوع عدد به عضو داده شمارشی هنگام مقداردهی متغیر شمارشی - تعیین عدد معادل عضو داده شمارشی - تعیین محل تعریف داده شمارشی در برنامه - تعریف داده شمارشی - رفع خطای استفاده از داده شمارشی

شرایط انجام کار و ابزار و تجهیزات:

مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها

تجهیزات: رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است.

زمان: ۸۰ دقیقه (اعلان آرایه ۱۰ دقیقه - دسترسی به عناصر آرایه ۲۰ دقیقه - به کارگیری متدهای آماده روی آرایه ۳۰ دقیقه - به کارگیری داده‌های شمارشی ۲۰ دقیقه)

معیار شایستگی:

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	اعلان آرایه	۲	
۲	دسترسی به عناصر آرایه	۲	
۳	به کارگیری متدهای آماده روی آرایه	۲	
۴	به کارگیری داده‌های شمارشی	۱	
<p>شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش:</p> <p>مسئولیت‌پذیری - تصمیم‌گیری - استدلال - حل مسئله - تفکر خلاق - نقش در گروه - زبان فنی رعایت ارگونومی - کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات)</p> <p>حفاظت از تجهیزات کارگاه</p> <p>دقت در تشخیص ساختار مناسب داده هنگام تعریف آرایه در برنامه</p>			
میانگین نمرات			
		۲	*

* حداقل میانگین نمرات هنرجو برای قبولی و کسب شایستگی، ۲ است.