



فصل پنجم

ساختار شرطی

استفاده از رایانه برای تولید برنامه و انجام محاسبات ریاضی و مسائل مختلف کاری به امرعادی بدل شده است. یک مسئله در دنیای واقعی دارای حالت‌های گوناگونی است که در صورت وقوع هر یک باید تصمیم و پاسخ مناسبی برای آن ایجاد شود. وقتی بخواهیم برای این مسائل مربوط به دنیای واقعی برنامه تولید کنیم، ضمن استفاده از زبان‌های برنامه نویسی باید شرایط و حالت‌های مختلف را نیز در برنامه در نظر بگیریم. برای بررسی این حالت‌ها در زبان‌های برنامه نویسی از ساختارهای شرطی و کنترل‌های خطا استفاده می‌شود که در ۲۴ کارگاه با آنها آشنا خواهید شد. در این فصل هرنرجو قادر خواهد بود با اتکا بردانش انواع عملگرها و ساختارهای شرط و مهارت به‌کارگیری این عملگرها و تشخیص وجود ساختار شرط در برنامه به‌وسیله تحلیل الگوریتم برنامه، برنامه‌های شرطی تولید نماید و خطاهای آن را برطرف کند.

واحد یادگیری ۹

■ شایستگی کار با عملگرهای ریاضی، مقایسه‌ای و منطقی

آیا تا به حال پی برده اید

- عملگرهای ریاضی در زبان برنامه‌نویسی سی‌شارپ چگونه نمایش داده می‌شوند؟
- چگونه عبارات ریاضی و مقایسه‌ای محاسبه می‌شوند؟
- حاصل یک عبارت مقایسه‌ای از چه نوعی است؟
- در یک عبارت محاسباتی اولویت انواع عملگرها نسبت به یکدیگر چگونه است؟
- کدام عملگر ریاضی می‌تواند روی داده‌های غیر عددی نیز عمل کند؟

هدف از این شایستگی به کارگیری انواع عملگرها برای ایجاد عبارات محاسباتی در برنامه و محاسبه مقدار یک عبارت محاسباتی با توجه به اولویت عملگرها است.

استاندارد عملکرد

با استفاده از دانش ریاضی و شناخت انواع عملگرها و اولویت آنها نسبت به یکدیگر در زبان سی‌شارپ، عبارات محاسباتی ایجاد و نتیجه آنها را محاسبه کرده، خطاهای احتمالی آن را برطرف کند.



یکی از کاربردهای اولیه رایانه، انجام عملیات ریاضی و مقایسه‌ای است. در زبان‌های برنامه‌نویسی برای انجام این عملیات از عبارت (expression) استفاده می‌شود. به عبارت $5 + 6 = \text{result}$ دقت کنید.

یک عبارت حداقل از یک عملگر (operator) و یک یا چند عملوند (operand) تشکیل شده است. به اعدادی مانند ۵ و ۶ و متغیر result که یک عملگر روی آنها عملی را انجام می‌دهد، عملوند می‌گویند. به علامت‌هایی مانند علامت + و = که بیانگر انجام یک عمل روی متغیرها و داده‌ها هستند، عملگر می‌گویند.



هر عبارت دارای حاصل و نتیجه‌ای است که می‌تواند به یک متغیر انتساب داده شود.

متغیر result از چه نوعی می‌تواند باشد؟

کنجکاوی



زبان C# از زبان‌های سطح بالاست و از عملگر + برای جمع و عملگر = برای انتساب استفاده می‌کند.

در جدول ۹-۱ عملوندها و عملگرهای عبارات محاسباتی را مشخص کنید.

فعالیت
کارگاهی



جدول ۹-۱: شناسایی عملگر و عملوند در عبارت

| عبارت | عملگر | عملوند |
|---------|-------|--------|
| -5 | | |
| 7-13+12 | | |
| a-b*2 | | |

می‌توان حاصل عبارت را با متد Write یا WriteLine به صورت مستقیم در خروجی نمایش داد.

`Console.WriteLine(6+5);`

دستور `Console.WriteLine(6+5);` چه مقداری را روی صفحه نمایش نشان می‌دهد؟

کنجکاوی



عملوندهای عبارت محاسباتی می‌توانند داده یا متغیر باشند. `result = num1 + num2;`

یادداشت



عملگرهای محاسباتی

در زبان‌های برنامه‌نویسی برای انجام عملیات ریاضی پایه (جمع، تفریق، ضرب و تقسیم) عملگرهایی وجود دارد. جدول ۹-۲ این عملگرها را در زبان سی‌شارپ نشان می‌دهد.

جدول ۹-۲ را تکمیل کنید.

فعالیت
کارگاهی



جدول ۹-۲: عملگرهای محاسباتی در سی‌شارپ

| عملگر | عملکرد | مثال | نتیجه |
|-------|-----------------|------|-------|
| - | قرینه | -5 | |
| * | ضرب | 20*6 | |
| / | تقسیم | 4/25 | |
| % | باقیمانده تقسیم | 6%20 | |
| + | جمع | 3+20 | |
| - | تفریق | 3-20 | |

به عملگرهایی مانند عملگر قرینه که به یک عملوند نیاز دارند، عملگرهای **یکتایی** (Unary) و به عملگرهایی مانند عملگرهای + و * و / که به دو عملوند نیاز دارند، عملگرهای **دوتایی** (Binary) می‌گویند.

کارگاه ۱ عملگر تقسیم /

بعد خانوار از جمله شاخص‌های با اهمیت یک کشور است که نشان‌دهنده میانگین تعداد افراد یک خانواده در آن کشور است. می‌خواهیم با توجه به آمار سال ۹۰ کشورمان، برنامه‌ای بنویسیم که جمعیت مردم ایران و تعداد خانوار را دریافت کرده، بعد خانوار را در ایران نمایش دهد.

۱ پروژه جدید به نام **ConsloePopulation** ایجاد کنید.

۲ کد زیر را درمتد **Main** بنویسید.

```
Console.WriteLine("----- family size in IRAN -----");
```

```
Console. WriteLine ("Enter Population:");
long populate = long.Parse(Console.ReadLine());
```

```
Console. WriteLine ("Enter number of family :");
numFamily = long.Parse(Console.ReadLine());
float familySize = populate / numFamily;
Console.WriteLine("family size in IRAN: "{0}", familySize);
```

۳ برنامه را با مقادیر داده شده در جدول زیر اجرا کرده، جدول را تکمیل کنید.

| سال | جمعیت | تعداد خانوار | خروجی | جواب شما |
|------------|----------|--------------|-------|----------|
| ۹۰ (کشور) | ۷۵۱۴۹۶۶۹ | ۲۱۱۸۵۶۴۷ | | |
| ۹۰ (کاشان) | ۳۲۳۳۷۱ | ۹۵۳۱۴ | | |
| ۹۰ (فارس) | ۹۳۹۴۱ | ۲۴۷۶۵ | | |

آیا خروجی برنامه با جواب شما برای بعد خانوار، یکسان است؟

نتیجه عملگر تقسیم با توجه به نوع عملوندهایش می‌تواند صحیح (بدون ممیز) و یا اعشاری (ممیزی) باشد. اگر حداقل یکی از عملوندهای عملگر تقسیم (/)، از نوع داده اعشاری باشد، نتیجه عبارت، عدد اعشاری و در غیر این صورت عدد صحیح است.

۴ برای حل مشکل برنامه، عبارت محاسبه **familySize** را به شکل زیر تغییر دهید:

```
float familySize = (float) populate/numFamily;
```

۵ برنامه را دوباره اجرا کرده، جدول زیر را تکمیل کنید.

| سال | جمعیت | تعداد خانوار | خروجی | جواب شما |
|-------------------|----------|--------------|-------|----------|
| ۹۰ (کشور) | ۷۵۱۴۹۶۶۹ | ۲۱۱۸۵۶۴۷ | | |
| ۹۰ (زادگاه شما) | | | | |
| ۹۰ (استان همسایه) | | | | |

۶ آمار جمعیت زادگاه خود و استان همسایه را از آمار سال ۹۰ در جدول بالا قرار داده، بعد خانوار را در دو شهر باهم مقایسه کنید.

برای حل مشکل برنامه، چه روش‌های دیگری پیشنهاد می‌کنید؟

کنجکاوی



فعالیت منزل



برداشت



- برنامه‌ای بنویسید که سن شما را از ورودی خوانده، تعیین کند چند ماه، چند روز و چند ساعت از عمر شما می‌گذرد؟
 - در برنامه بالا سن دریافتی را ۷ برابر کرده، حاصل را ۱۴۴۳ برابر کنید و نتیجه را نمایش دهید. نتیجه را با سن دریافتی مقایسه کنید.
 - برنامه را برای سن برادر، خواهر و یا همکلاسی خود دوباره اجرا کنید.
 - برنامه‌ای بنویسید که مدت زمان لازم برای انجام این فعالیت را بر حسب ثانیه دریافت کند و تعیین کند چند، دقیقه و ثانیه برای انجام این فعالیت فرصت دارید؟

آنچه آموختم:

۱.
۲.
۳.

اولویت بندی عملگرهای محاسباتی

$$3 + 2 * 10$$

نتیجه محاسبه عبارت روبرو چیست؟

این عبارت دارای دو عملگر + و * است. کدام عملیات اول انجام می‌شود؟ ۲ حالت وجود دارد.

الف) ابتدا جمع و بعد عملگر ضرب انجام شود که نتیجه ۵۰ خواهد بود.

$$1) 3 + 2 = 5$$

$$2) 5 * 10 = 50$$

$$3 + 2 * 10$$

ب) ابتدا ضرب و سپس جمع انجام شود که نتیجه ۲۳ خواهد بود.

$$1) 2 * 10 = 20$$

$$2) 3 + 20 = 23$$

$$3 + 2 * 10$$

- برنامه‌ای بنویسید که حاصل عبارت $3 + 2 * 10$ را نمایش دهد. کدام یک از نتایج بالا نمایش داده می‌شود؟

- در برنامه بالا عبارت $10 * (3 + 2)$ را جایگزین کنید و نتیجه را بنویسید.

فعالیت کارگاهی



جدول ۳-۹ اولویت عملگرهای محاسباتی را نشان می‌دهد. چنانچه چند عملگر اولویت یکسانی داشته باشند، عملگرها به ترتیب از چپ به راست انجام می‌شوند و به آن "شرکت پذیری چپ" گفته می‌شود.

جدول ۳-۹: اولویت عملگرهای محاسباتی

| اولویت | عملگر | نوع عملگر |
|--------|-------|-----------|
| ۱ | - | یکتایی |
| ۲ | * | دوتایی |
| | / | |
| | % | |
| ۳ | + | دوتایی |
| | - | |

با توجه به جدول اولویت‌ها در سی شارپ (جدول ۳-۹)، جدول زیر را تکمیل کرده، اولویت عملگرها را تعیین کنید.

فعالیت
کارگاهی



| شرح عبارت | عبارت ریاضی | معادل در C# | مقدار متغیرها | نتیجه عبارت |
|--|----------------------|---|-------------------|-------------|
| محاسبه دلتا برای حل معادله درجه ۲ | $\Delta = b^2 - 4ac$ | $\Delta = b * b - 4 * a * c$ <p>(۱) (۴) (۲) (۳)</p> | b=2 a=1 c=1 | |
| میانگین سه عدد | | $(a+b+c)/3$ | | |
| محاسبه حقوق خالص باکسر بیمه ۳ درصد | | | | |
| تعداد گروه‌های سه نفره در کارگاه با توجه به تعداد هنرجویان | | | | |
| محاسبه معدل هنرجو با توجه به تعداد واحدهای هر درس | | | | |

کنجکاوی



یادداشت



در تبدیل یک عبارت ریاضی به معادل آن درسی شارپ، درج پرانتز در چه مواقعی ضروری است؟

در صورتی که در یک عبارت محاسباتی چندین پرانتز تودرتو وجود داشته باشد؛ محاسبات از داخلی‌ترین پرانتز آغاز می‌شود.

کارگاه ۲ تغییر اولویت عملگرها

می‌خواهیم برای یک فروشگاه که دارای اجناس تخفیف‌دار است، برنامه‌ای بنویسیم. ۳ کالای اولی که خریداری شده، دارای تخفیف ۱۰ درصد و تخفیف کالای چهارم ۷۰ درصد است. برنامه‌ای بنویسید که قیمت قبل از تخفیف ۴ کالا را از ورودی دریافت کرده، مبلغ تخفیف را محاسبه کند.

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

۲ قطعه کد زیر برای دریافت قیمت کالاها از ورودی است. آن را تکمیل کنید.

```
Console.WriteLine (" ----- Final Price-----");
```

```
//Get the First Price
```

```
Console.Write ("\nEnter Price1:");
```

```
float price1= .....
```

```
//Get the Second Price
```

```
Console.Write ("\nEnter Price2:");
```

```
float price2= .....
```

```
//Get the Third Price
```

```
Console.Write ("\nEnter Price3:");
```

```
float price3 = .....
```

```
//Get the fourth Price
```

```
Console.Write ("\nEnter Price4:");
```

```
float price4 = .....
```

۳ دستورات زیر مبلغ نهایی را محاسبه کرده، نمایش می‌دهد. این دستورات را به کدهای قبلی اضافه کنید.

```
//Calculate the Discount
```

```
float discount = price1+ price2 + price3* 10 / 100 + price4* 70 / 100;
```

```
Console.Write("\n *** final Discount = "+ discount);
```

۴ با استفاده از مقادیر جدول زیر میزان تخفیف را محاسبه کنید، سپس برنامه را اجرا کرده و این مقادیر را به عنوان ورودی برنامه استفاده کنید.

| قیمت کالا ۱ | قیمت کالا ۲ | قیمت کالا ۳ | قیمت کالا ۴ | میزان تخفیف | خروجی برنامه |
|-------------|-------------|-------------|-------------|-------------|--------------|
| ۲۰۰ | ۳۰۰ | ۱۰۰ | ۵۰۰ | | |

آیا خروجی برنامه با نتیجه‌ای که شما محاسبه کرده‌اید، یکسان است؟

۵ همان‌طور که می‌بینید نتیجه محاسبه اشتباه است. دلیل رخ دادن چنین خطایی چیست؟
به خطاهایی که در اثر اشتباه برنامه‌نویس در طراحی الگوریتم ایجاد می‌شود، خطاهای منطقی (Logic Error) می‌گویند. در صورت وجود این خطاها برنامه اجرا می‌شود ولی نتیجه اجرای برنامه، نتیجه مطلوب نیست.

۶ خطای منطقی برنامه فوق را چگونه برطرف کنیم؟

۷ پس از برطرف کردن خطای منطقی، برنامه را با مقادیر جدول زیر اجرا کنید.

| قیمت کالا ۱ | قیمت کالا ۲ | قیمت کالا ۳ | قیمت کالا ۴ | میزان تخفیف | خروجی برنامه |
|-------------|-------------|-------------|-------------|-------------|--------------|
| ۱۵۰ | ۱۰۰ | ۴۲۰ | ۲۳۰ | | |

۸ برنامه را تغییر دهید تا علاوه بر میزان تخفیف، میزان مبلغی که باید پردازیم را هم نمایش دهد.

۹ برنامه بالا را برای درصدهای متفاوت تخفیف مطابق جدول زیر تغییر دهید و برنامه را اجرا کنید.

| تخفیف کالای اول | تخفیف کالای دوم | تخفیف کالای سوم | تخفیف کالای چهارم |
|-----------------|-----------------|-----------------|-------------------|
| ۱۰٪ | ۱۵٪ | ۱۰٪ | ۶۵٪ |

۱۰ کد برنامه را طوری تغییر دهید تا میزان تخفیف کالاها را نیز از ورودی دریافت کند.

۱۱ در دستورات **Console.Write** در کد برنامه از ابتدای رشته **n** را برداشته، برنامه را اجرا کرده، خروجی برنامه را با خروجی مرحله قبل مقایسه کنید.

درمورد انواع خطاها در برنامه‌نویسی تحقیق کنید.

پژوهش



فعالیت
کارگاهی



برداشت



کارمند اداره‌ای هر ماه ۵ درصد از حقوق خود را به سازمان حمایت از کودکان کار اهدا می‌کند. با فرض اینکه از حقوق این شخص هر ماه ۱۰ درصد مالیات و ۷ درصد بیمه کسر شود، برنامه‌ای بنویسید که حقوق شخص را دریافت کرده، حقوق او را پس از کسر مالیات، بیمه و مبلغ اهدایی نمایش دهد.

آنچه آموختم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۱

| نمره | استاندارد (شاخص‌ها/داوری/نمره‌دهی) | نتایج ممکن | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | مراحل کار |
|------|--|---------------------|---|---|
| ۲ | توجه به همه موارد | قابل قبول | تصمیم‌گیری - استدلال - شایستگی محاسبه و ریاضی - نقش در تیم - زبان فنی | شایستگی‌های غیرفنی |
| | | | رعایت ارگونومی، کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات) | ایمنی و بهداشت |
| ۱ | توجه به ایمنی و بهداشت | غیر قابل قبول | | توجهات زیست محیطی |
| | | | دقت در محاسبه نتیجه عبارت محاسباتی با توجه به اولویت انواع عملگرها | نگرش |
| ۳ | نوشتن عبارت ریاضی در زبان برنامه‌نویسی - تغییر اولویت عملگرها - محاسبه نتیجه عبارت ریاضی انتخاب نوع متغیر مناسب برای ذخیره نتیجه عبارت - نوشتن عبارت ریاضی پیچیده و نیازمند به پرانتزهای متداخل و تعیین نتیجه عبارت با مقادیر متفاوت متغیرها | بالاتر از حد انتظار | | نوشتن عبارات ریاضی به زبان برنامه نویسی |
| ۲ | نوشتن عبارت ریاضی در زبان برنامه نویسی - تغییر اولویت عملگرها - محاسبه نتیجه عبارت ریاضی | قابل قبول | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار ویژوال استودیو روی آن نصب باشد زمان: ۱۰ دقیقه | |
| ۱ | نوشتن عبارت ریاضی در زبان برنامه نویسی | غیر قابل قبول | | |

عملگرهای انتساب

از جمله عملگرهای زبان‌های برنامه‌نویسی عملگر انتساب است. به عبارات زیر توجه کنید:
 در این عبارت، مقدار سمت راست (Y) در متغیر سمت چپ (X) قرار می‌گیرد.
 در این عبارت مراحل زیر به ترتیب اجرا می‌شود:

$$X = Y;$$

$$X = Y = Z = 0;$$

- متغیر Z با 0 مقداردهی می‌شود.
- متغیر Y با مقدار متغیر Z مقداردهی می‌شود.
- متغیر X با مقدار متغیر Y مقداردهی می‌شود.

عملگرهای دو عبارت $X=Y=Z$ و $X+Y+Z$ را اولویت بندی کرده، با هم مقایسه کنید.

فعالیت
کارگاهی



عملگرهای ریاضی "شرکت پذیری چپ" دارند در حالی که عملگرهای انتساب "شرکت پذیری راست" دارند.

یادداشت



علاوه بر عملگر انتساب = عملگرهای انتساب دیگری نیز وجود دارند. با توجه به مقادیر متغیرهای X و Y جدول ۴-۹ را تکمیل کنید.

`int x = 10, y = 3;`

جدول ۴-۹: عملگرهای انتساب

| عبارت | معادل | نتیجه X |
|------------|---------------|-----------|
| $X += Y;$ | $X = X + Y;$ | 13 |
| $X -= Y;$ | $X = X - Y;$ | |
| $X *= Y;$ | $X = X * Y;$ | |
| $X /= Y;$ | $X = X / Y;$ | |
| $X \%= Y;$ | $X = X \% Y;$ | |

اولویت عملگرهای انتساب، از عملگرهای ریاضی کمتر است.

یادداشت



به کمک هم گروه خود برنامه‌ای بنویسید که عددی از ورودی دریافت کرده، به ترتیب عملیات زیر را انجام دهد و خروجی نهایی را چاپ کند.

فعالیت
گروهی



۱ عدد ورودی را در ۲ ضرب کند.

۲ حاصل را در ۵ ضرب کند.

۳ حاصل را با عدد ۱۰ جمع کند.

۴ حاصل را بر ۱۰ تقسیم کند.

۵ از حاصل عدد یک را کم کند.

۶ حاصل را نمایش دهد.

۷ برنامه را با ورودی‌های مختلف اجرا کنید. خروجی برنامه با عدد ورودی چه ارتباطی دارد؟

عملگرهای افزایشی - کاهششی

به عملگرهای یکتایی ++ (plus plus) و -- (minus minus) که به ترتیب برای افزایش و کاهش مقدار یک متغیر به اندازه یک واحد به کار می‌روند، عملگرهای **افزایشی - کاهششی** می‌گوییم.

جدول ۹-۵: عملگرهای افزایشی-کاهششی

| عبارت معادل | مثال | نام عملگر | عملگر |
|-------------|----------------|-----------|-------|
| $X = X + 1$ | $X++$ $++X$ | افزایشی | ++ |
| $X = X - 1$ | $--X$ $X--$ | کاهششی | -- |

عملگرهای افزایشی - کاهششی

کارگاه ۳

برنامه‌ای بنویسید که تعداد هنرجویان دو کلاس ۱۰۱ و ۱۰۲ را دریافت کند و پس از انتقال یکی از هنرجویان از کلاس ۱۰۱ به کلاس ۱۰۲ تعداد تغییر یافته را نمایش دهد.

۱ پروژ به جدیدی با نام **Students count** ایجاد کنید.

۲ دستورات زیر را در متد **Main** بنویسید.

```
int count101, count102;
```

```
Console.WriteLine("Enter count of student class 101:");
count101 = int.Parse(Console.ReadLine());
Console.WriteLine("Enter count of student class 102:");
count102 = int.Parse(Console.ReadLine());
Console.WriteLine("count of students class 101={0} ", count101--);
Console.WriteLine("count101={0} ", count101);
Console.WriteLine("count of students class 102={0} ", count102++);
Console.WriteLine("count102={0} ", count102);
```

۳ برنامه را اجرا کرده، خروجی برنامه را بنویسید.

آیا خروجی برنامه تعداد هنرجویان دو کلاس را به درستی نشان می‌دهد؟

۴ قسمت نمایش تعداد هنرجویان کلاس را مطابق کد زیر تغییر دهید و برنامه را دوباره اجرا کنید.

```
Console.WriteLine("count of students class 101={0} ", --count101);
Console.WriteLine("count of students class 102={0} ", ++count102);
```

آیا خروجی برنامه تعداد هنرجویان دو کلاس را به درستی نشان می‌دهد؟
مقدار متغیرها تغییر کرده است و به درستی نمایش داده می‌شود؟



عملگرهای ++ و -- عملگرهای یکتایی هستند و فقط بر روی متغیرها عمل می‌کنند.

۵ در برنامه به جای عملگرهای افزایشی-کاهشی، معادل آنها را قرار دهید و برنامه را اجرا کنید. اگر عملگر افزایشی-کاهشی در یک عبارت یا دستور بعد از متغیر به کار رود، ابتدا متغیر با مقدار فعلی در محاسبات شرکت می‌کند؛ سپس مقدار متغیر یک واحد اضافه - کم می‌شود.

```
Console.WriteLine ("count of students class 101={0} ", count101--);
```

دستور بالا معادل دو دستور زیر است.

```
Console.WriteLine ("count of students class 101={0} ", count101);
```

```
Count101--;
```

اگر عملگر افزایشی - کاهشی در یک عبارت یا دستور قبل از متغیر به کار رود، ابتدا مقدار متغیر یک واحد اضافه - کم شده و متغیر با مقدار جدید در محاسبات بعدی شرکت می‌کند.

```
Console.WriteLine ("count of students class 101={0} ", --count101);
```

دستور بالا معادل دو دستور زیر است.

```
--count101;
```

```
Console.WriteLine ("count of students class 101={0} ", count101);
```



- به کمک هم گروه خود معادل دستورات زیر را بنویسید.

```
Console.WriteLine("count of students class 102={0} ", count102++);
```

```
Console.WriteLine("count of students class 102={0} ", ++count102);
```

- با توجه به قطعه کد زیر و تغییرات متغیرهای x و y جدول را تکمیل کنید.

```
int x = 12, y = -12;
```

```
Console.WriteLine ("x={0} y={1}", x++, ++y);
```

```
Console.WriteLine ("x={0} y={1}", --x, ++y);
```

```
y = ++x;
```

```
x = y--;
```

```
Console.WriteLine ("x={0} y={1}", x, y);
```

| خروجی برنامه | y | x |
|--------------|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

عملگر الحاق دو رشته

به قطعه کد زیر توجه کنید. عملگر + علاوه بر اینکه یک عملگر محاسباتی است، عملگر رشته‌ای هم محسوب می‌شود. اگر حداقل یکی از عملوندهای عملگر + از نوع string باشد، این عملگر به عنوان عملگر الحاق دو رشته عمل می‌کند.

```
int a=5+7;
string s="ali"+"reza";
string count="count of pencil : "+100;
```

نتیجه این دستورها چیست؟

در یک برنامه قطعه کد بالا را نوشته، محتوای متغیرهای a، s و count را نمایش دهید.

فعالیت
کارگاهی



به قطعه کد زیر توجه کنید.

```
string s="ali";
s+="reza";
Console.WriteLine(s);
```

این قطعه کد را در متد Main بنویسید و برنامه را اجرا کنید. از خروجی برنامه چه نتیجه‌ای می‌گیرید؟

یادداشت



عملگر + عملگر محاسباتی و عملگر الحاق دو رشته است و عملگر += به عنوان عملگر انتساب الحاق دو رشته نیز به کار می‌رود.



برنامه‌ای بنویسید که نام و نام خانوادگی را به صورت جداگانه از ورودی دریافت کند، سپس نام و نام خانوادگی را با استفاده از:

- عملگر الحاق دو رشته در یک متغیر ذخیره کرده، نمایش دهد.
- عملگر انتساب الحاق دو رشته را در یک متغیر ذخیره کرده، نمایش دهد.



آنچه آموختم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۲

| مراحل کار | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | نتایج ممکن | استاندارد (شاخص‌ها/داوری/نمره دهی) | نمره |
|---|--|------------------------|--|------|
| به کارگیری عملگرهای افزایش و کاهش و انتساب | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار ویژوال استودیو روی آن نصب باشد زمان: ۱۰ دقیقه | بالاتر از حد انتظار | استفاده از عملگرهای کاهشی و افزایشی و انتساب در برنامه- محاسبه نتیجه عبارت شامل عملگرهای کاهشی و افزایشی و انتساب - محاسبه نتیجه عبارت شامل + و += ، انتخاب نوع متغیر مناسب برای ذخیره نتیجه عبارت | ۳ |
| | | قابل قبول | استفاده از عملگرهای کاهشی و افزایشی و انتساب در برنامه- محاسبه نتیجه عبارت شامل عملگرهای کاهشی و افزایشی و انتساب | ۲ |
| | | غیر قابل قبول | استفاده از عملگرهای کاهشی و افزایشی در برنامه | ۱ |

عملگرهای مقایسه‌ای

جدول ۹-۶: عملگرهای مقایسه‌ای

| عملکرد | معادل ریاضی | عملگر |
|---------------|-------------|-------|
| برابری | = | == |
| نامساوی | ≠ | != |
| کوچک‌تر | < | < |
| بزرگ‌تر | > | > |
| کوچک‌تر مساوی | ≤ | <= |
| بزرگ‌تر مساوی | ≥ | >= |

رتبه اول کلاس شما چه کسی است؟ بیشترین نمره فعالیت کارگاهی را چه کسی کسب کرده‌است؟ میزان سرانه آب مصرفی در ایران در مقایسه با سایر کشورها چگونه است؟ در زندگی روزمره با سؤالات بسیار زیادی از این قبیل روبه‌رو هستیم که پاسخ به آنها نیاز به انجام عمل مقایسه دارد. در زبان‌های برنامه‌نویسی از عملگرهای مقایسه‌ای برای انجام عمل مقایسه استفاده می‌شود (جدول ۹-۶).

به عبارتی که شامل عملگرهای مقایسه‌ای هستند یک عبارت منطقی گویند که بیان‌کننده یک شرط است. جواب شرط یا نتیجه یک عبارت منطقی، در صورت درستی true و در صورت نادرستی false است. حاصل عبارتهای منطقی را می‌توان در خروجی نمایش داد.

```
Console.WriteLine("flag = {0}", a > 10);
```

نتیجه عبارت منطقی در چه نوع متغیری ذخیره می‌شود؟
متغیر flag از چه نوعی است؟ flag = a > 10;

کنجکاوی



کارگاه ۴ عملگرهای مقایسه‌ای

در خبرنامه دانشجویان ایران به نقل از رئیس مرکز تحقیقات و ارتباط با صنعت شرکت آب و فاضلاب کشور آمده است: متوسط سرانه مصرف آب در بخش خانگی در ایران به طور متوسط ۱۹۰ لیتر در روز به ازای هر نفر و در دنیا ۱۳۵ تا ۱۴۰ لیتر است. می‌خواهیم برنامه‌ای بنویسیم که سرانه مصرف آب در ایران را با سرانه مصرف آب در دنیا مقایسه کند.

۱ پروژه جدیدی با نام WaterConsumption ایجاد کنید.

۲ دستورات زیر را در متد Main بنویسید.

```
int iranWater = 190;
Console.WriteLine(" The average consumption in IRAN -> " + iranWater);
int worldWater = 135;
Console.WriteLine(" The average consumption in WORLD -> " + worldWater);
// آیا سرانه مصرف آب در ایران بیشتر است؟
Console.WriteLine(" Is iranWater > worldWater: {0} ");
```

۳ برنامه را اجرا کرده، خروجی آن را بنویسید.

کارگاه ۵ عملگرهای منطقی

می‌خواهیم برنامه‌ای بنویسیم که عدم قبولی یک هنرجو را با نمایش false به علت گرفتن نمره ۷ نشان دهد.

۱ پروژه‌ای با نام Failure ایجاد کنید.

۲ کد زیر را در متد Main بنویسید.

```
bool flag=false;
int score=7;
Console.WriteLine("score = {0}",score);
Console.WriteLine("is student passed:{0}, is score=7:{1}",flag==true,score==7);
```

۳ با هم گروهی خود در مورد خروجی این برنامه بحث کنید.

۴ برنامه را اجرا کنید.

عملگر = عمل انتساب را انجام می‌دهد. عبارت $a=b$ برای مقداردهی متغیر a با مقدار متغیر b استفاده می‌شود، درحالی که عبارت $a==b$ بیان می‌کند که "آیا a و b مساوی هستند؟" و نتیجه این عبارت true یا false است.

اولویت عملگرهای محاسباتی بالاتر از عملگرهای مقایسه‌ای است.

یادداشت



فعالیت
کارگاهی



با توجه به مقادیر متغیرها جدول زیر را تکمیل کنید.

int a=5, failed=9, passed =15;

| عبارت منطقی | نتیجه |
|---------------------------|-------|
| $passed \geq 12$ | |
| $failed + 2 < passed - a$ | |
| $passed \neq 20$ | |
| $failed == passed - a$ | |

کارگاه ۶ اشکال زدایی (Debugging) عبارات

می‌خواهیم عددی را دریافت کرده، محاسبات زیر را انجام دهیم.

- افزایش یک واحد به ورودی
- ۵ برابر کردن نتیجه قبل
- کاهش یک واحد از نتیجه قبل
- نصف کردن نتیجه آخر

۱ پروژه‌ای با نام **Debug1** ایجاد کنید.

۲ کد زیر را در متد **Main** بنویسید.

```
int num1;
Console.WriteLine("Enter num1 : ");
num1=int.Parse(Console.ReadLine());
num1++;
num1*5;
--num1;
num1/2;
Console.WriteLine("num1={0} ", num1);
Console.ReadKey();
```

۳ برنامه را با استفاده از کلید **F6** ترجمه کنید.

۴ خطاهای این برنامه و علت رخ دادن آنها را در جدول زیر که مشابه پنجره لیست خطا در **VS** طراحی شده، بنویسید.

| شماره ردیف | خطا | شماره خط | شماره ستون | علت |
|------------|-----|----------|------------|-----|
| | | | | |
| | | | | |

کارگاه ۷ اشکال زدایی در رشته‌ها

۱ پروژه‌ای با نام **Debug2** ایجاد کنید.

۲ کد زیر را در متد **Main** بنویسید.

```
bool result = true;
Console.WriteLine(result > 0);
string s1 = "Ali";
string s2 = "Reza";
s1 += s2;
s1 = s1 - s2;
float f = 100 - s;
s1 = -s1;
```

۳ برنامه را با استفاده از کلید F6 ترجمه کنید.

۴ خطاهای این برنامه و علت رخ دادن آنها را در جدول زیر بنویسید.

| شماره ردیف | خطا | شماره خط | شماره ستون | علت |
|------------|-----|----------|------------|-----|
| | | | | |
| | | | | |
| | | | | |
| | | | | |

عملوند عملگرهای $>=$ و $<=$ و $>$ و $<$ نمی‌توانند از نوع `string` و `bool` باشند.

یادداشت



کارگاه ۸ اشکال زدایی خطای تقسیم بر صفر

فرد خیری تعدادی دفترچه به یک مؤسسه خیریه اهدا کرده است. می‌خواهیم برنامه‌ای بنویسیم که تعداد دفترچه‌ها و تعداد بچه‌های این مؤسسه را از ورودی دریافت کرده، تعیین کند به هریک از بچه‌ها چند دفترچه تعلق می‌گیرد.

۱ پروژه‌ای با نام `Debug3` ایجاد کنید.

۲ کد زیر را در متد `Main` بنویسید.

```
int booklet, children;
Console.WriteLine("Enter number of booklet : ");
booklet = int.Parse(Console.ReadLine());
Console.WriteLine("Enter number of children : ");
children = int.Parse(Console.ReadLine());
int count = booklet / children;
Console.WriteLine("count={0} ", count);
Console.ReadKey();
```

۳ برنامه را اجرا کنید.

تعداد دفترچه‌ها و بچه‌های مؤسسه را مطابق جدول زیر وارد کرده، جدول را تکمیل کنید.

| تعداد دفترچه‌ها | تعداد بچه‌ها | خروجی |
|-----------------|--------------|-------|
| ۵۰۰ | ۲۰۰ | |
| ۲۰۰ | ۰ | |

۴ خطاهای این برنامه و علت رخ دادن آن را در کادر زیر بنویسید.

| خطا | علت |
|-----|-----|
| | |

۵ تفاوت خطای این کارگاه با خطای کارگاه ۷ و ۶ چیست؟

آنچه آموختیم:

برداشت



۱.
۲.
۳.

جدول ارزشیابی مرحله ۳

| نمره | استاندارد (شاخص‌ها/داوری/نمره دهی) | نتایج ممکن | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | مراحل کار |
|------|---|------------------|---|-------------------------------|
| ۳ | به‌کارگیری عملگرهای رابطه‌ای و تعیین نتیجه عبارت منطقی- تشخیص و رفع خطای استفاده نادرست از عملگرهای ریاضی، مقایسه‌ای و انتساب | بالاتر از انتظار | مکان: کارگاه استاندارد رایانه زمان: ۱۰ دقیقه | استفاده از عملگرهای مقایسه‌ای |
| ۲ | به‌کارگیری عملگرهای رابطه‌ای و تعیین نتیجه عبارت منطقی | قابل قبول | | |
| ۱ | به‌کارگیری عملگرهای رابطه‌ای | غیرقابل قبول | | |

عملگرهای منطقی

آیا در بین دوستان شما افراد کمتر از ۱۵ سال و بزرگ‌تر از ۲۰ سال وجود دارد؟ آیا معدل سال گذشته شما بین ۱۲ و ۱۷ است؟ آیا شرایط لازم برای حضور هنرجویان در اردو فراهم است؟ در مسائلی از این قبیل با عملیات منطقی روبه‌رو هستیم و تصمیم نهایی ما به ترکیب شرط‌ها بستگی دارد. برای ترکیب عملگرهای مقایسه‌ای می‌توان از عملگرهای منطقی استفاده کرد. عبارتی که شامل عملگرهای منطقی باشد یک عبارت منطقی است.

پویانمایی ۰۲۲۵: آشنایی با عملگرهای منطقی

فیلم



کارگاه ۹ عملگر منطقی &&

رمز قفل یک چمدان مسافرتی 25 است. می‌خواهیم برنامه‌ای بنویسیم که باز شدن قفل را شبیه‌سازی کند. ارقام رمز را جداگانه از ورودی دریافت می‌کنیم.



۱ پروژه‌ای با نام Lock ایجاد کنید.

۲ قطعه کد زیر را در متد Main وارد کنید.

```
byte x, y;
y = byte.Parse(Console.ReadLine());
x = byte.Parse(Console.ReadLine());
Console.WriteLine((x==2) && (y==5));
```

در صورتی که خروجی true باشد، قفل باز می‌شود.

۲ برنامه را با مقادیر متفاوت برای x و y اجرا کرده، جدول را تکمیل کنید.

| خروجی | y | x |
|-------|---|---|
| | 5 | 3 |
| | 4 | 2 |
| True | | |
| False | | |

- عبارت منطقی بنویسید که درستی عدد ورودی که شماره یک ماه سال است را بررسی کند؟
- حاصل عبارت $X < 14 \ \&\& \ X > 10$ در چه صورتی true است؟

کنجکاوی



یادداشت



عملوندهای یک عملگر منطقی می‌تواند متغیری از نوع `bool`، یک شرط و یا مقادیر `true` و یا `false` باشد.

عملگر منطقی `&&` دو عملوند دارد. در جدول ۹-۷ نتیجه اجرای عملگر `&&` براساس حالت‌های مختلف عملوندهای آن آمده است.

جدول ۹-۷: عملگر منطقی `&&`

| p | q | p && q |
|-------|-------|--------|
| false | false | false |
| false | true | false |
| true | false | false |
| true | true | true |

جدول زیر را تکمیل کنید.

فعالیت
کارگاهی

| شرح | عبارت منطقی | اولویت بندی | مقدار | نتیجه |
|--|--|--|-----------------------|-------|
| نمره هنجرو بین ۰ تا ۲۰ است | <code>score >=0 && score <=20</code> | ^۱ ^۳ ^۲ <code>Score >=0 && score <=20</code> | <code>score=13</code> | |
| | | | <code>score=26</code> | |
| شرط ثبت نام در همایش نمره بالای ۱۲ و معدل بالای ۱۴ است | | | | |
| | | | | |
| هوا ابری نباشد و نمره هنجرو بالای ۱۵ باشد | | | | |
| | | | | |

یادداشت



اولویت عملگرهای مقایسه‌ای بالاتر از عملگرهای منطقی است.

کارگاه ۱۰ || عملگر منطقی

قفل چمدان مسافرتی کارگاه ۸ خراب شده است و در صورت درست بودن حداقل یکی از ارقام رمز هم باز می‌شود. می‌خواهیم برنامه‌ای بنویسیم که باز شدن قفل را در این وضعیت شبیه‌سازی کند. ارقام رمز را جداگانه از ورودی دریافت می‌کنیم.

۱ پروژه‌ای با نام Bad Lock ایجاد کنید توجه داشته باشید که رمز قفل همان 25 است.

۲ قطعه کد زیر را در متد Main بنویسید.

```
byte x, y;
y = byte.Parse(Console.ReadLine());
x = byte.Parse(Console.ReadLine());
Console.WriteLine((x==2) || (y==5));
```

۳ برنامه را با مقادیر متفاوت برای x و y اجرا کرده، جدول را تکمیل کنید.

| خروجی | y | x |
|-------|---|---|
| | 4 | 2 |
| | 4 | 3 |
| true | | |
| False | | |

- عبارت منطقی بنویسید که نشان دهد عدد ورودی در محدوده مجاز برای نمره هنرجو نیست؟
- حاصل عبارت $X < 10 \parallel X > 14$ در چه صورتی true است؟

کنجکاوی



عملگر منطقی || دو عملوند دارد. در جدول ۸-۹ نتیجه اجرای عملگر || براساس حالت‌های مختلف عملوندهای آن آمده است.

جدول ۸-۹ : عملگر منطقی ||

| p | q | p q |
|-------|-------|--------|
| false | false | false |
| false | true | true |
| true | false | true |
| true | true | true |

جدول زیر را تکمیل کنید.

فعالیت
کارگاهی

| نتیجه | مقدار | اولویت بندی | عبارت منطقی | شرح |
|-------|-----------|------------------------------------|---------------------------|--|
| | months=5 | ① ③ ② months < 1 months > 31 | months < 1 months > 31 | عدد ورودی در محدوده شماره روزهای یک ماه نیست. |
| | months=40 | | | |
| | | | | شرط ثبت نام در همایش معدل بالای ۱۷ یا قبولی در آزمون ورودی |
| | | | | هوا ابری باشد یا نمره هنرجو کمتر از ۱۵ باشد |

عملگر منطقی !

کارگاه ۱۱

مدیریت هنرستان تصمیم گرفته است اگر هوا مناسب باشد، هنرجویان را به اردو ببرد. می‌خواهیم برنامه‌ای بنویسیم که وضعیت هوا را از ورودی دریافت کند و وضعیت اردو را تعیین کند.

۱ پروژه‌ای با نام WeatherCamp ایجاد کنید.

۲ قطعه کد زیر را در متد Main بنویسید.

```
string weather;
```

```
Console.WriteLine(" weather:");
```

```
weather = Console.ReadLine();
```

```
Console.WriteLine("weather is not cloudy: {0}", weather!="cloudy");
```

۳ برنامه را اجرا کنید و جدول زیر را تکمیل کنید.

| خروجی | وضعیت هوا |
|-------|-----------|
| | cloudy |
| | sunny |

عملگر منطقی ! یک عملوند دارد. در جدول ۹-۹ نتیجه اجرای عملگر ! براساس حالت‌های مختلف عملوند آن آمده است.

جدول ۹-۹: عملگر منطقی!

| p | ! |
|-------|-------|
| false | true |
| true | false |

جدول زیر را تکمیل کنید.

فعالیت
کارگاهی



| نتیجه | مقدار | اولویت بندی | عبارت منطقی | شرح |
|-------|-------|---------------------------------|----------------------|--|
| | no=2 | ④ ① ③ ② !(no >= 1 && no <=4) | !(no >= 1 && no <=4) | شماره فصل نیست |
| | | | | هوا ابری است یا هنرجو نمره قبولی نگرفته است (بین ۱۲ تا ۲۰) |
| | | | | معادله درجه ۲ ریشه مضاعف دارد یا ریشه ندارد |

اولویت عملگرها در یک عبارت به ترتیب عبارتست از: محاسباتی، مقایسه‌ای و منطقی
برنامه با استفاده از نتایج عملگرهای مقایسه‌ای و منطقی، برای انجام عملیات بعدی تصمیم‌گیری
می‌کند.

یادداشت



کدام یک از عملگرهای منطقی یکتایی و کدام دوتایی هستند؟
یک تایی
دوتایی

کنجکاوی





– در برنامه‌ای سن و اندازه قد دو نفر دریافت می‌شود. برای هر کدام از مسایل زیر شرط مناسب بنویسید.

الف) آیا این دو نفر هم سن و هم قد هستند؟

ب) آیا این دو نفر هم سن یا هم قد هستند؟

– از کارمندی که بالای یک میلیون حقوق می‌گیرد و کمتر از ۳ فرزند دارد، مالیات کسر می‌شود. برنامه‌ای بنویسید که با دریافت حقوق و تعداد فرزندان با نمایش `true` یا `false` نشان دهد که آیا از این کارمند مالیات کسر می‌شود؟

– اگر کاربری با نام کاربری `"admin"` و گذرواژه `"Pass123"` داشته باشیم. برنامه‌ای بنویسید که نام کاربری و گذرواژه را دریافت کند و صحت آنها را با نمایش `true` یا `false` نمایش دهد.

۴ خروجی قطعه برنامه‌های زیر را بررسی کنید.

الف)

```
Console.WriteLine("Enter num1:");
int num1 = int.Parse(Console.ReadLine());
Console.WriteLine("Enter num2:");
int num2 = int.Parse(Console.ReadLine());
Console.WriteLine(" {0} greater than {1} -> {2}", num1, num2, num1 > num2);
Console.WriteLine(" {0} less than {1} -> {2}", num1, num2, num1 < num2);
Console.WriteLine(" {0} equal to {1} -> {2}", num1, num2, num1 == num2);
Console.WriteLine(" {0} not equal to {1} -> {2}", num1, num2, num1 != num2);
```

ب)

```
Console.WriteLine("Enter an integer:");
int myInt = int.Parse(Console.ReadLine());
bool isLessThan10 = myInt < 10;
bool isBetween0and5 = (0 <= myInt) && (myInt <= 5);
Console.WriteLine("Integer less than 10? {0}", isLessThan10);
Console.WriteLine("Integer between 0 and 5? {0}", isBetween0And5);
```



آنچه آموختیم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۴

| مراحل کار | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | نتایج ممکن | استاندارد (شاخص‌ها/داوری/نمره دهی) | نمره |
|--|--|------------------------------|---|------|
| استفاده از عملگرهای منطقی | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار ویژوال استودیو روی آن نصب باشد زمان: ۱۰ دقیقه | بالاتر از حد انتظار | استفاده از عملگرهای منطقی در ترکیب شرطها و تعیین نتیجه آنها - انتخاب نوع متغیر برای خروجی عبارت منطقی - محاسبه نتیجه عبارت منطقی ترکیبی پیچیده و طولانی | ۳ |
| | | قابل قبول | استفاده از عملگرهای منطقی در ترکیب شرطها و تعیین نتیجه آنها | ۲ |
| | | غیر قابل قبول | استفاده از عملگرهای منطقی در ترکیب شرطها | ۱ |
| | | | | |
| ارزشیابی کار (شایستگی انجام کار) | | <input type="checkbox"/> بلی | | |
| | | <input type="checkbox"/> خیر | | |
| معیار شایستگی انجام کار : کسب حداقل نمره ۲ از مرحله استفاده از عملگرهای منطقی 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:

- ۲- به کارگیری عملگرهای افزایش و کاهش و انتساب
۴- استفاده از عملگرهای منطقی

- ۱- نوشتن عبارات ریاضی به زبان برنامه‌نویسی
۳- استفاده از عملگرهای مقایسه‌ای

استاندارد عملکرد:

با استفاده از دانش ریاضی و شناخت انواع عملگرها و اولویت آنها نسبت به یکدیگر در زبان سی‌شارپ، عبارات محاسباتی ایجاد و نتیجه آنها را محاسبه کرده و خطاهای احتمالی آن را برطرف کند.

شاخص‌ها:

| شماره مرحله کار | شاخص‌های مرحله کار |
|-----------------|--|
| ۱ | انتخاب عملگر ریاضی - انتخاب محل قرار دادن پرانتز در عبارت ریاضی - انتخاب نوع متغیر برای نگهداری نتیجه عبارت ریاضی |
| ۲ | انتخاب عملگر انتساب - تشخیص محل استفاده از عملگرهای کاهشی یا افزایشی در عبارت - محاسبه نتیجه عبارت شامل عملگر + یا += - انتخاب نوع متغیر برای نگهداری نتیجه عبارت |
| ۳ | انتخاب عملگر مقایسه‌ای - تشخیص اولویت اجرای عملگرهای ریاضی و مقایسه‌ای و انتساب در یک عبارت - محاسبه نتیجه عبارات منطقی - تشخیص خطای عبارت منطقی |
| ۴ | انتخاب عملگر منطقی - ترکیب عبارت منطقی - تشخیص اولویت اجرای عملگرهای ریاضی و مقایسه‌ای و منطقی در یک عبارت ترکیبی منطقی - محاسبه نتیجه عبارت ترکیبی منطقی - انتخاب نوع متغیر برای نگهداری نتیجه عبارت منطقی ترکیبی |

شرایط انجام کار و ابزار و تجهیزات:

مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها

تجهیزات: رایانه‌ای که نرم‌افزار ویژوال استودیو روی آن نصب باشد

زمان: ۴۰ دقیقه نوشتن عبارات ریاضی به زبان برنامه‌نویسی ۱۰ دقیقه - به کارگیری عملگرهای افزایشی و کاهشی و انتساب ۱۰ دقیقه - استفاده از عملگرهای مقایسه‌ای ۱۰ دقیقه - استفاده از عملگرهای منطقی ۱۰ دقیقه

مشخصات فنی رایانه:

رایانه که بتواند ویندوز ۱۰ و نرم‌افزار ویژوال استودیو روی آن نصب شود.

معیار شایستگی:

| ردیف | مرحله کار | حداقل نمره قبولی از ۳ | نمره هنرجو |
|--|--|-----------------------|------------|
| ۱ | نوشتن عبارات ریاضی به زبان برنامه‌نویسی | ۱ | |
| ۲ | به کارگیری عملگرهای افزایش و کاهش و انتساب | ۱ | |
| ۳ | استفاده از عملگرهای مقایسه‌ای | ۱ | |
| ۴ | استفاده از عملگرهای منطقی | ۲ | |
| شایستگی‌های غیرفنی، ایمنی، بهداشت، توجهات زیست‌محیطی و نگرش: تصمیم‌گیری - استدلال - شایستگی محاسبه و ریاضی - نقش در تیم - زبان فنی دانش ارگونومیک (خود فرد) - کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات) دقت در محاسبه نتیجه عبارت محاسباتی با توجه به اولویت انواع عملگرها | | | |
| میانگین نمرات | | | |
| | | | * |

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

واحد یادگیری ۱۰

■ شایستگی کار با ساختار شرطی

آیا تابه حال پی برده اید

- در زبان سی شارپ چند نوع ساختار شرطی وجود دارد؟
- چگونه می توان ساختار if - else ساده را کدنویسی کرد؟
- چگونه می توان if های تودرتو را با دستورات کمتری بازنویسی کرد؟
- چگونه می توان برخی ساختارهای شرطی if را با switch بازنویسی کرد؟

هدف از این شایستگی تشخیص نیاز به استفاده از ساختار شرط در برنامه و به کارگیری ساختارهای شرطی و کنترل خطاهای آن است.

استاندارد عملکرد

با استفاده از دانش و شناخت اجزای محیط IDE و ویژوال استودیو، چگونگی عملکرد ساختارهای شرطی در الگوریتم برنامه را تحلیل و برنامه را با استفاده از دستورات شرطی ایجاد کند.



از دوران کودکی تاکنون سؤالات زیادی داشتیم که با **اگر** شروع می‌شد و مسیر زندگی انسان را مشخص می‌کرد.
اگر دوچرخه داشتم، چه می‌شد؟
اگر دیپلم گرفته بودم؛
اگر دانشجو بودم؛
 بعضی از این **اگرها** دور از دسترس و برخی دیگر دست‌یافتنی است و هنگامی که اتفاق می‌افتد؛ اکثر انسان‌ها در تعقیب **اگرهای** دیگرند.
 مثلاً **اگر** دوچرخه داشتم، چه می‌شد؟ **اگر** دوچرخه داشتم می‌گفتم اگر می‌توانستم برای دوچرخه‌ام زنگ بخرم چه می‌شد؟ بعد از آن در این فکر بودم که **اگر** در تیم دوچرخه سواری منطقه عضو می‌شدم و
 در اینجا می‌خواهیم برای **اگرها** برنامه بنویسیم.

به کمک هم‌گروهی خود **اگرهای** کتاب دانش فنی پایه را در جدول زیر وارد کنید.

فعالیت
گروهی



| ردیف | عنوان | شرط | درست بودن شرط | نادرست بودن شرط |
|------|--------------------|---------------------------|---------------------------------|-----------------|
| ۱ | تفکیک زباله | اگر زباله خشک باشد | در سطل مخصوص زباله خشک قرار بده | — |
| ۲ | خرید کالا | | | خرید نکردن |
| ۳ | | | | |
| ۴ | | | | |
| ۵ | بازی سنگ کاغذ قیچی | | | |

در برخی از مسائل اگرها ساده نیست و شامل چندین شرط است.
در این بخش می‌خواهیم یکی از اگرهای اولیه کودکی را تبدیل به یک قطعه کد کنیم.
«اگر دوچرخه داشتم بازی می‌کردم».

```
if (x=="bicycle")
    Console.WriteLine("Playing");
```

در قطعه کد بالا متغیر x نشان‌دهنده نوع وسیله‌ای است که داریم.
با همین اگر ساده توانستیم یک قطعه کد کوچک بنویسیم. همان‌طور که مشاهده می‌کنید واژه اگر در زبان برنامه نویسی به if تبدیل شد.

در قطعه کد بالا نوع متغیر x چیست؟

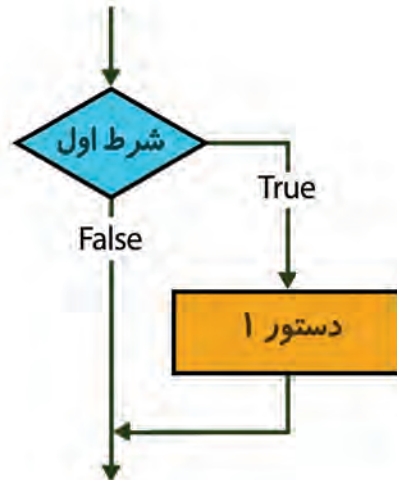
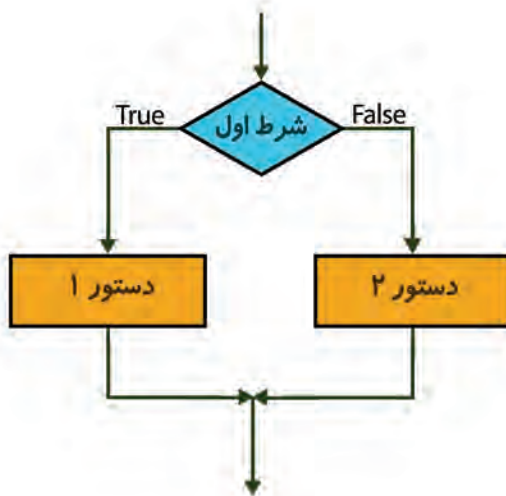
کنجکاوی



فعالیت
کارگاهی



دوروندنمای زیر را با هم مقایسه کرده و تعیین کنید کدام مناسب مسئله بالا است.



شکل ۱۰-۱ ساختار دستور if

شکل کلی if ساده:

```
If (عبارت شرطی)
    دستور;
```

دستور شرطی ساده

کارگاه ۱

می‌خواهیم برنامه‌ای بنویسیم که مزایای دو روش پرداخت قبوض آب و برق را مشخص کند:

✓ مراجعه به بانک: پرداخت قبوض

✓ استفاده از پرداخت الکترونیکی: پرداخت قبوض، کاهش ترافیک

۱ پروژه‌ای با نام EPayment ایجاد کنید.

۲ کد زیر را در متد Main وارد کنید.

```
Console.WriteLine("Enter your choice // 1 or 2");
byte x = byte.Parse(Console.ReadLine());
if (x==1)
Console.WriteLine("Pay bills");
if (x==2)
{
Console.WriteLine("Pay bills");
Console.WriteLine("reduction of traffic ");
}
```

۳ برنامه را اجرا کرده و نتایج آن را مشاهده کنید.

۴ بعد از پرانتز **if** دوم یک؛ قرار داده، نتیجه را با نتیجه مرحله قبل مقایسه کنید.

۵ با توجه به تصویر، مزایای دیگر پرداخت الکترونیکی را به دستورات **if** دوم اضافه کنید.



مجموعه دستوراتی که داخل {} قرار می‌گیرد. یک بلاک را تشکیل می‌دهند.

یادداشت





آنچه آموختیم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۱

| مرحله کار | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | نتایج ممکن | استاندارد (شاخص‌ها/داوری/نمره دهی) | نمره |
|-------------------------------------|--|--------------------------|--|------|
| شایستگی‌های غیرفنی | تصمیم‌گیری - استدلال - حل مسئله - تفکر خلاق - نقش در تیم - زبان فنی | قابل قبول | توجه به همه موارد | ۲ |
| ایمنی و بهداشت | رعایت ارگونومی، کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات) | | | |
| توجهات زیست‌محیطی | | غیرقابل قبول | توجه به ایمنی و بهداشت | ۱ |
| نگرش | دقت در تشخیص ساختار شرطی مناسب هنگام تبدیل الگوریتم به برنامه در مسئله مطرح شده | | | |
| به‌کارگیری دستور if در برنامه | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار ویژوال استودیو روی آن نصب باشد زمان: ۱۰ دقیقه | بالتر از حد انتظار | رسم روندنمای برنامه شرطی ساده و نوشتن کد آن - تشخیص و رفع خطاهای if | ۳ |
| | | قابل قبول | رسم روندنمای برنامه شرطی ساده و نوشتن کد آن | ۲ |
| | | غیرقابل قبول | رسم روندنمای برنامه شرطی ساده | ۱ |

کارگاه ۲ ساختار شرطی if _ else

می‌خواهیم قسمتی از برنامه هدایت یک خودروی بدون سرنشین را برای عبور از چراغ قرمز طراحی کنیم.

۱ پروژه جدیدی با نام **VCondition** ایجاد کنید.

۲ برنامه زیر را کامل کرده، در متر **Main** بنویسید.

```
if ( light=="red")
Vehicle_Condition = "stop";
else
Vehicle_Condition = "move";
Console.WriteLine(Vehicle_Condition );
```

۳ کدام یک از روندنماهای شکل ۱-۱۰ مناسب این برنامه است؟

۴ روندنمای این برنامه را رسم کنید.

۵ برنامه را به صورت زیر تغییر دهید.

اگر چراغ قرمز بود، ابتدا ماشین ترمز (break) کرده، سپس متوقف شود؛ در غیراین صورت با زدن یک بوق حرکت کند.

فعالیت منزل



برنامه‌ای بنویسید که میزان سوخت موجود در یک خودرو را از ورودی خوانده، اگر کمتر از ۵ لیتر بود پیام مناسب با رنگ قرمز و در غیراین صورت پیام مناسب دیگری با رنگ سبز نمایش دهد.

کارگاه ۳ بلاک در ساختار شرطی if

می‌خواهیم برنامه‌ای طراحی کنیم تا با دریافت یک رشته برای زوج یا فرد بودن روزهای هفته، برنامه بعد از ظهرهای علی را تعیین کند.

| روز | ۱۵ تا ۱۶:۳۰ | ۱۷ تا ۱۸:۳۰ | ۱۸:۴۰ تا ۱۹ | ۱۹:۱۵ تا ۲۰:۴۵ |
|------------------------------------|---------------------|-------------------|-------------|-------------------|
| روزهای شنبه – دوشنبه – چهارشنبه | انجام تکالیف روزانه | باشگاه ورزشی | فریضه نماز | کلاس C# |
| روزهای یکشنبه – سه شنبه – پنج شنبه | انجام تکالیف روزانه | کلاس زبان انگلیسی | فریضه نماز | سرزدن به پدر بزرگ |

۱ پروژه جدیدی با نام **Week** ایجاد کنید.

۲ کد زیر را در متد Main بنویسید.

```
string day = Console.ReadLine();
if (day == "odd")//condition true
{
    Console.WriteLine("Studying");
    Console.WriteLine("English class");
    Console.WriteLine("Say praing");
    Console.WriteLine("Visit Grand");
}
if (day == "Even") //condition true
{
    Console.WriteLine("Studying");
    Console.WriteLine("gym");
    Console.WriteLine("Say praing");
    Console.WriteLine("c# programming Class ");
}
```

۳ آیا برنامه خطا دارد؟

۴ خروجی برنامه در صورتی که ورودی «even» باشد، چیست؟

۵ خروجی برنامه در صورتی که ورودی «odd» باشد، چیست؟

دلیل اشتراک پاسخ در موارد ۴ و ۵ چیست؟

۶ اگر خروجی برنامه نادرست است با توجه به آنچه قبلاً آموختید نوع خطا را مشخص کنید و مشکل برنامه را برطرف کنید.

آیا می‌توان برنامه را با ساختار if - else نوشت؟

برنامه‌ای بنویسید مدت زمان تقریبی انتظار در صف بانک را با شرایط زیر به دست آورد.
برنامه باید ۳ عدد دریافت کند، شماره آخرین نفر ورودی به بانک، شماره آخرین نفری که در حال انجام کار است. و تعداد باجه‌های فعال، تعداد باجه‌های فعال، باید بیشتر از صفر باشد.
فرض کنید میانگین مدت زمان لازم برای انجام کار هر نفر در باجه معادل ۲ دقیقه باشد.

فعالیت منزل



کارگاه ۴ اجرای گام به گام if _ else

در یک اداره ۲۴ طبقه، ۲ دستگاه آسانسور، مخصوص طبقات زوج و فرد و یک نفر جهت راهنمایی مراجعان وجود دارد که با پرسش از افراد، آسانسور مناسب را مشخص می‌کند. می‌خواهیم با هدف مکانیزه کردن قسمتی از تشکیلات ساختمانی از یک برنامه کمک بگیریم تا با وارد کردن شماره طبقه، آسانسور مناسب را مشخص کند. (آسانسور A مخصوص طبقات فرد و آسانسور B مخصوص طبقات زوج است).

۱ پروژه جدیدی با نام ChElevator ایجاد کنید.

۲ کد زیر را در Main وارد کنید

```
Console.WriteLine("Enter the floor");
int floor = int.Parse(Console.ReadLine());
if (floor % 2 == 0)
    Console.WriteLine("Elevator A");
else
    Console.WriteLine("Elevator B");
```

۳ اگر مراجعه کننده به طور سهوی عدد ۳۴ را وارد کند خروجی برنامه چیست؟

۴ برنامه را طوری تغییر دهید در صورتی که کاربر عددی خارج از محدوده تعداد طبقات وارد کرد پیام مناسبی چاپ شود.

۵ خروجی برنامه را با مقادیر زیر Trace کنید.

| floor | خروجی |
|-----------|-------|
| 10 | |
| 25 | |
| -6 | |
| 5.2 | |
| "floor13" | |

کارگاه ۵ عملگر سه تایی

می خواهیم برنامه ای بنویسیم با دریافت سن فرد تعیین کند آیا صلاحیت دریافت گواهی نامه رانندگی دارد؟

۱ پروژه جدیدی با نام DLicence ایجاد کنید.

اگر سن فرد مساوی یا بزرگ تر از ۱۸ سال باشد می تواند در آزمون رانندگی شرکت کند و در غیر این صورت این امکان وجود ندارد. در نتیجه باید با استفاده از یک دستور شرطی، سن بررسی شود.

۲ کد زیر را تکمیل کنید و در متد Main وارد کنید.

```
if (age >= 18)
    status = "Allowed";
else
    status = "Not Allowed";
Console.WriteLine("status");
```

۳ خروجی برنامه را با مقادیر زیر Trace کنید.

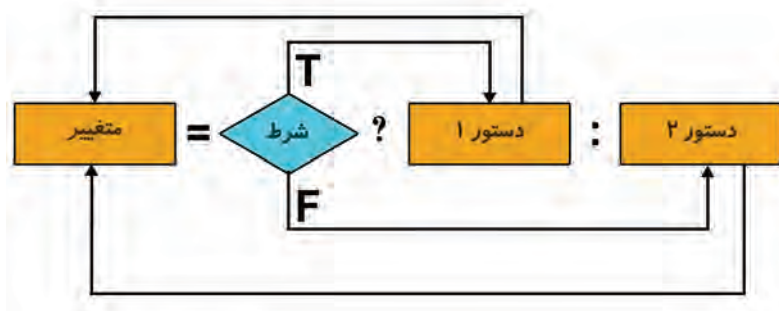
| خروجی | age |
|-------|-----|
| | 20 |
| | 18 |
| | 10 |

۴ متغیرهای age و status از چه نوعی باید تعریف شوند.

۵ به جای کد قبلی، کد زیر را وارد کنید.

```
Console.WriteLine("How old are you?");
int age = int.Parse(Console.ReadLine());
string s;
s = (age >= 18) ? "Allowed": " Not Allowed";
Console.WriteLine Status (s);
```

۶ خروجی برنامه را در دو حالت مقایسه کنید.



شکل ۱۰-۲- عملگر سه تایی

زمانی که دستور شرطی، از یک دستور، برای بخش if و else استفاده می کند؛ می توان از عملگر سه تایی (ternary) به صورت زیر استفاده کرد. هنگامی که شرط درست باشد خروجی این عملگر حاصل دستور 1 و هنگامی که شرط نادرست باشد، حاصل دستور 2 است.

دستور 2 : دستور 1 ؟ (عبارت شرطی) = خروجی

– دو مورد از کارگاه های قبل که قابلیت استفاده از عملگر سه تایی را دارند، بازنویسی کنید.
– دستوراتی اضافه کنید که پیام ها با رنگ های متفاوت نمایش داده شوند.

فعالیت
کارگاهی



برداشت



آنچه آموختیم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۲

| نمره | استاندارد (شاخص ها/داوری/نمره دهی) | نتایج ممکن | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | مراحل کار |
|------|---|---------------------|---|--------------------------------------|
| ۳ | رسم روندنمای برنامه if-else ساده و پیچیده و نوشتن کد آن - تشخیص بلاک دستورات بدنه if - استفاده از عملگر ؟ در برنامه شرطی - تشخیص و رفع خطاهای if-else | بالاتر از حد انتظار | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه ای که نرم افزار ویژوال استودیو روی آن نصب باشد زمان: ۱۰ دقیقه | به کارگیری دستور if - else در برنامه |
| ۲ | رسم روندنمای برنامه if-else ساده و پیچیده و نوشتن کد آن - تشخیص بلاک دستورات بدنه if - استفاده از عملگر ؟ در برنامه شرطی | قابل قبول | | |
| ۱ | رسم روندنمای برنامه if-else ساده و نوشتن کد آن | غیر قابل قبول | | |

کارگاه ۶ عبارت شرطی تودرتو

می خواهیم برنامه ای بنویسیم که کاربر معتبر برای ورود به سیستم را مشخص کند.

نام کاربری معتبر: reza

گذرواژه صحیح: 123456

۱ پروژه جدیدی با نام UserPass ایجاد کنید.

۲ کد زیر را در متد Main وارد کنید.

```
Console.WriteLine("Enter your user name");
string user = Console.ReadLine();
Console.WriteLine("Enter your password");
string pass = Console.ReadLine();
if (user == "reza")
if (pass == "123456")
Console.WriteLine("Welcome " + username);
else
Console.WriteLine("The password is incorrect");
else
Console.WriteLine("Unkown user");
Console.ReadKey();
```

۳ با توجه به برنامه جدول زیر را تکمیل کنید.

| شرط | دستور |
|--|-------|
| اگر نام کاربری و گذرواژه صحیح است؛ | |
| اگر نام کاربری صحیح است و گذرواژه صحیح نیست. | |
| اگر نام کاربری صحیح نیست؛ | |

1 123456
2 password
3 12345
4 12345678
5 qwerty
6 123456789
7 1234
8 baseball
9 dragon
10 football



شکل ۱۰-۳ در حال حاضر پرکاربردترین گذرواژه "123456" به دلیل سادگی لقب «بدترین گذر واژه دنیا» را دارد.

کنجکاوی



– چرا هیچ کدام از if ها بلاک {} ندارند؟
 – هر کدام از else ها مربوط به کدام if است؟ به کمک هنرآموز خود روش کلی برای این مسئله ارائه دهید.



شکل ۱۰-۴ ساختارهای شرطی در یک نگاه

یادداشت



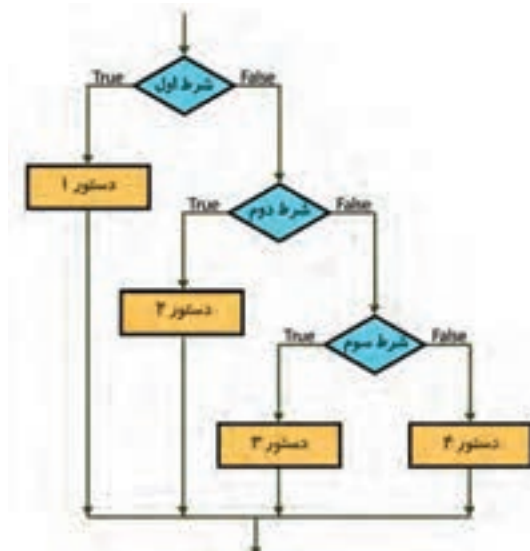
فعالیت گروهی



برای تقسیم‌بندی صحیح عبارات شرطی باید از بلاک {} استفاده کنید. بلاک تعیین‌کننده محدوده عبارت شرطی است که بیش از یک دستور دارد.

با کمک هم‌گروهی خود، برای روندنمای زیر یک مسئله طراحی کنید. می‌توانید از موارد زیر ایده بگیرید.

- ✓ داشتن شرایط مختلف برای استخدام در شرکت برنامه‌نویسی
- ✓ عیب‌یابی کامپیوتر
- ✓ ثبت نام دانشگاه
- ✓ کنترل کیفیت یک محصول غذایی



شکل ۱۰-۵ ساختار شرطی تودرتو



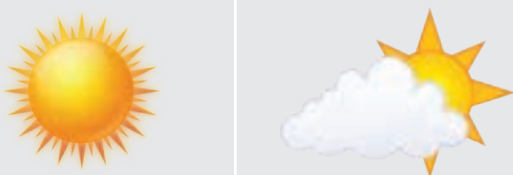



اگر درون یک ساختار شرطی، ساختار شرطی دیگری باشد؛ به آن عبارات شرطی تو در تو (Nested conditional statements) می‌گویند.

عبارات شرطی تو در تو و عملگرهای منطقی عبارات شرطی تو در تو را می‌توان با استفاده از عملگرهای منطقی ساده‌تر و خواناتر نوشت. اما به کار بردن آنها دقت زیادی لازم دارد.

کارگاه ۷ عبارات شرطی با عملگرهای منطقی

می‌خواهیم با استفاده از نرم‌افزار پیش‌بینی هوا امکان بازدید هنرجویان از نمایشگاه کتاب را در فصل بهار بررسی کنیم.

| ردیف | دما | هوا | امکان برگزاری |
|------|------------------|---|---------------|
| ۱ | بیشتر از ۱۹ درجه |  | عالی |
| ۲ | بیشتر از ۱۹ درجه |  | خوب |
| ۳ | کمتر از ۱۹ درجه |  | مناسب |
| ۴ | کمتر از ۱۹ درجه |  | غیر ممکن |

- ۱ پروژه جدیدی با نام Weather ایجاد کنید.
- ۲ متغیرهای مورد نیاز برنامه را اعلان کنید.

```
Console.WriteLine("Enter the air temperature");
int temp = int.Parse(Console.ReadLine());
Console.WriteLine("Enter the weather conditions: sunny /cloudy / partly cloudy/ rainy");
string Condition = Console.ReadLine();
```

- ۳ برای ردیف ۱ قطعه کد زیر را وارد کنید.

```
if (temp > 19)
if (Condition == "sunny")
Console.WriteLine("Excellent");
```

- ۴ برنامه را اجرا کرده، ورودی‌هایی به برنامه بدهید که پیام Excellent را نمایش دهد.
- ۵ کد زیر را جایگزین خطوط بالا کنید.

```
if ((temp > 19)&& (Condition == "sunny"))
Console.WriteLine("Excellent");
```

- ۶ برنامه را دوباره اجرا کرده، نتایج را مقایسه کنید.
- ۷ کد ردیف ۲ را با استفاده از عملگرهای منطقی بنویسید.
- ۸ کد زیر را برای ردیف ۳ به برنامه اضافه کرده، سپس برنامه را اجرا و نتیجه را یادداشت کنید.

```
if (temp < 19)
{
if (Condition == "sunny")
Console.WriteLine("suitable");
if (Condition == "partly cloudy ")
Console.WriteLine("suitable");
}
```

- ۹ به کمک هنرآموز خود کد این مرحله را با استفاده از عملگرهای منطقی بازنویسی کرده، سپس نتیجه را با کد قبلی مقایسه کنید.
- ۱۰ کد ردیف ۴ را با عملگرهای منطقی به برنامه اضافه کنید.



برنامه را طوری بازنویسی کنید که اگر کاربر در ردیف ۳ و ۴ دمای کمتر از ۸ درجه وارد کرد؛ برگزاری اردو غیرممکن شود.

کارگاه ۸ مرتب‌سازی

می‌خواهیم برای حفظ احترام به بزرگ‌ترهای فامیل، سن ۳ نفر را از ورودی بخوانیم و از بزرگ به کوچک مرتب کنیم.

۱ پروژه جدیدی با نام **Sort** ایجاد کنید.

۲ برنامه زیر را تکمیل کرده، در متد **Main** وارد کنید.

```
if ((a > b) && (a > c))
    if (b > c)
        Console.WriteLine("{0} {1} {2}", a, b, c);
    else
        Console.WriteLine("{0} {1} {2}", a, c, b);

else if ((b > a) && (b > c))
    if (a > c)
        Console.WriteLine("{0} {1} {2}", b, a, c);
    else
        Console.WriteLine("{0} {1} {2}", b, c, a);

else if ((c > a) && (c > b))
    if (a > b)
        Console.WriteLine("{0} {1} {2}", c, a, b);
    else
        Console.WriteLine("{0} {1} {2}", c, b, a);
```

۳ نوع مناسب برای متغیرهای **a**، **b**، **c** چیست؟

۴ برنامه را طوری تغییر دهید تا صحت داده‌ها نیز بررسی شود.

صحت داده‌ها: سن نباید منفی و از ۱۱۰ بیشتر باشد.

به کمک هم گروهی خود سعی کنید برنامه را با طرح و نقشه دیگری بنویسید.





– برنامه‌ای بنویسید که در انتخابات شورای دانش‌آموزی مدرسه، نفر برتر را از بین ۴ نفر شرکت‌کننده در انتخابات تعیین کند. برنامه باید با دریافت ۴ عدد که مشخص‌کننده تعداد آرای هر نفر است، نفری که بیشترین رأی را به خود اختصاص داده معین کند.

– برنامه‌ای بنویسید که نفر برتر مسابقات دوی ۱۰۰ متر کلاس را تعیین کند. در این برنامه باید با دریافت زمان ۴ شرکت‌کننده برحسب ثانیه، نفر اول مشخص شود.

کارگاه ۹ ترکیب عبارات منطقی، محاسباتی و شرطی

می‌خواهیم برنامه‌ای بنویسیم که با دریافت سه عدد a , b , c مشخص کند آیا این سه عدد می‌توانند اضلاع یک مثلث باشند؟

روش اول: برای حل مسئله، از قضیه‌ی همار در هندسه استفاده می‌کنیم. این قضیه بیان می‌کند که در هر مثلث مجموع هر دو ضلع از ضلع دیگر بزرگ‌تر است!



زمانی که داده‌های ورودی از سه شرط ($a + b > c$, $a + c > b$, $b + c > a$) به‌درستی عبور کنند به خروجی $true$ می‌رسیم و اگر در هر نقطه از مسیر منحرف شوند؛ خروجی $false$ خواهد بود. برای این کار به متغیری ($count$) نیاز داریم که در مسیر حرکت به سوی مقصد، تعداد شرط‌هایی را که به‌درستی از آن عبور می‌کنیم؛ بشمارد و هر وقت تعداد شمارش‌شده به عدد ۳ رسید؛ خروجی $true$ شود.

۱ پروژه جدیدی با نام **Triangle** ایجاد کنید.

۲ برنامه زیر را تکمیل کرده و در متد **Main** بنویسید.

```
int count = 0;
if (b + c > a) count++;
if (a + c > b) count++;
if (a + b > c) count++;
if (count == 3)
    Console.WriteLine(true);
else
    Console.WriteLine(false);
```

۳ برنامه را با ۳ مقدار ورودی اجرا کنید.

روش دوم: می‌توانیم از شرط تودرتو استفاده کنیم.

```

if (a + b <= c)
    Console.WriteLine(false);
else if (a + c <= b)
    Console.WriteLine(false);
else if (b + c <= a)
    Console.WriteLine(false);
else
    Console.WriteLine(true);

```

در این حالت true را در چهارمین دستور قرار می‌دهیم زیرا تنها زمانی دستور چهارم اجرا می‌شود که دستورهای یک تا سه اجرا نشده باشند. در این ساختار شرطی، دیگر نیازی به شمارنده نیست.

۴ برنامه را اجرا کنید و نتایج را با کد قبلی مقایسه کنید.

روش سوم: در این روش از عملگرهای منطقی استفاده می‌کنیم.

```
Console.WriteLine(a + b > c && a + c > b && b + c > a);
```

اگر به عملگر && دقت کنید؛ متوجه می‌شوید که خروجی این عملگر زمانی true است که همه عملوندهای آن true باشد.

۵ برنامه را اجرا کرده، نتایج را دوباره با مراحل قبل مقایسه کنید.

– الگوریتم و روندنمای ۳ روش بالا را کشیده و با هم مقایسه کنید. به نظر شما کدام روش بهتر است چرا؟

– روش سوم را با استفاده از عملگر || بنویسید. (از هنرآموز خود کمک بگیرید)

فعالیت
کارگاهی

برداشت



آنچه آموختم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۳

| نمره | استاندارد (شاخص ها/داوری/نمره دهی) | نتایج ممکن | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | مرا حل کار |
|------|---|------------------|--|--|
| ۳ | تشخیص ساختار شرطی مناسب برای برنامه - رسم روندنمای برنامه ifهای متداخل و نوشتن کد آن - تبدیل if متداخل به if ساده - تشخیص و رفع خطاهای احتمالی if متداخل | بالاتر از انتظار | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه ای که نرم افزار ویژوال استودیو روی آن نصب باشد. زمان: ۱۵ دقیقه | به کارگیری دستور if متداخل در برنامه |
| ۲ | تشخیص ساختار شرطی مناسب برای برنامه - رسم روندنمای برنامه ifهای متداخل و نوشتن کد آن | قابل قبول | | |
| ۱ | تشخیص ساختار شرطی مناسب برای برنامه - رسم روندنمای برنامه ifهای متداخل | غیر قابل قبول | | |

کارگاه ۱۰ تبدیل if به Switch

می خواهیم برنامه ای بنویسیم که با دریافت یک فصل، ماه های آن را تعیین کند.

۱ پروژهِ جدیدی با نام Season ایجاد کنید

۲ قطعه کد زیر را تکمیل کرده، در متد Main بنویسید.

```
if (season=="Bahar")
    Console.WriteLine("Farvardin, Ordibehesht, Khordad");
else if (season=="Tabestan")
    Console.WriteLine("Tir, Mordad, Shahrivar");
else if (season=="Paeiz")
    Console.WriteLine("Mehr, Aban, Azar");
else if (season=="Zemestan")
    Console.WriteLine("Dey, Bahman, Esfand");
else
    Console.WriteLine("Season Wrong");
```

۳ برنامه را اجرا و خروجی را مشاهده کنید.

برنامه ساده است اما نکته قابل توجه در این برنامه بررسی عبارت season در چندین حالت است. در زبان سی شارپ می توان برای بررسی حالت های مختلف یک عبارت از دستور switch استفاده کرد. ساختار این دستور به صورت زیر است:

شکل کلی دستور switch

switch (عبارت)

```
{
  case مقدار ۱:
    دستور ۱;
    break;
  case مقدار ۲:
    دستور ۲;
    break;
  case مقدار ۳:
    دستور ۳;
    break;
  default:
    دستور ۴;
    break;
}
```

اگر حاصل یا مقدار عبارت برابر مقدار ۱ بود دستور ۱ انجام شده و با دستور break از ساختار خارج می شود. اگر برابر مقدار ۲ بود دستور ۲ انجام شده، با دستور break از ساختار خارج می شود (برای بقیه مقادیر نیز همین روش اجرا می شود). در نهایت، در صورتی که حاصل برابر هیچ یک از مقادیر نباشد دستور ۴ (default) انجام می شود و با دستور break از ساختار خارج می شود.

۴ برنامه زیر را جایگزین برنامه قبلی کرده، برخی دستورات ناقص را تکمیل کنید.

```
string season ;
Console.WriteLine("Enter Season:");
season = Console.ReadLine();
switch (season)
{
  case "Bahar":
    Console.WriteLine("Farvardin, Ordibehesht, Khordad");
    break;
  case "Tabestan":
    Console.WriteLine("Tir, Mordad, Shahrivar");
    break;
  case "Paeiz ":
    _____
    _____
  case "Zemestan":
    _____
    break;
  default:
    Console.WriteLine("Season Wrong");
    _____
}
```

۵ برنامه را اجرا و نتایج را با کد قبلی مقایسه کنید.

در دستور switch برای قرار دادن مجموعه ای از دستورات در یک case یا default، نیازی به بلاک نیست.

یادداشت



ساختار switch کارگاه ۱۱

| شماره کارگاه | رقم سمت راست |
|--------------|--------------|
| ۱۱ | ۳ |
| ۱۰ | ۴ |
| ۴ | ۵ |
| ۲ | ۶ |
| ۸ | ۷ |

می‌خواهیم برنامه‌ای بنویسیم که با دریافت شماره دانش‌آموزی و تعیین رقم سمت راست آن، شماره یک کارگاه از کارگاه‌های قبلی را برای تکلیف منزل به او اختصاص دهد.

برای جدا کردن رقم سمت راست یک عدد صحیح کافی است باقی‌مانده تقسیم عدد بر ۱۰ را محاسبه کنیم.

تفاوت عدد و رقم چیست؟

پژوهش



۱ پروژه جدیدی با نام IdStudent ایجاد کنید.

۲ کد زیر را در متد Main وارد کنید. حالت‌های ۳ تا ۸ را مطابق جدول تکلیف، کدنویسی کنید.

```
long x = long.Parse(Console.ReadLine());
x = x % 10;
switch (x)
{
    case 1:
        Console.WriteLine("5");
        break;
    case 2:
        Console.WriteLine("3");
        break;
    case 9:
        Console.WriteLine("7");
        break;
    case 0:
        Console.WriteLine("9");
        break;
}
```

۳ در برنامه بالا default استفاده نشده است، آیا برنامه خطا دارد؟

۴ برنامه قبل را با ساختار if بازنویسی کنید و هر دو برنامه را با هم مقایسه کنید.

۵ هنگام اجرا، به جای عدد ورودی، یک رشته وارد کنید. خطای حاصل را ترجمه کنید.

۶ برنامه را اجرا و شماره تمرین خود را یادداشت کنید.

شماره تمرین خود را در کلاس اعلام کرده، هم گروهی احتمالی خود را پیدا کنید. گروه‌ها باید برنامه اختصاصی خود را در کلاس با شیوه‌ای خلاقانه شرح دهند.

وجود دستور **break** برای **case** الزامی است.

فیلم شماره ۱۰۲۲۶: اشکال زدایی **switch**

فیلم را مشاهده کرده و برداشت را تکمیل کنید.

آنچه آموختم:

۱.
۲.
۳.

برنامه‌ای بنویسید با انتخاب نام میوه‌های جدول زیر خواص آنها را نمایش دهد. اگر میوه در فهرست نبود پیام مناسبی نمایش دهد.

| ردیف | میوه | خواص |
|------|---|---------------------------------------|
| ۱ |  | ضد سرطان، خون‌ساز، درمان ضعف |
| ۲ |  | تصفیه‌کننده خون، ضد سرطان |
| ۳ |  | تب‌بر، ضد نرمی استخوان |
| ۴ |  | کاهش کلسترول خون، کمک به سلامت لثه |
| ۵ |  | تنظیم فشار خون، مقوی مغز، شادابی پوست |
| ۶ |  | کاهش وزن، بهبود سردرد و معده درد |

یادداشت



فیلم



برداشت



فعالیت منزل



کارگاه ۱۲ ایجاد فهرست در کنسول

می‌خواهیم برنامه‌ای بنویسیم که یک فهرست رنگ تولید کند، تا با انتخاب هر یک از نویسه‌های زیر به وسیله کاربر، صفحه نمایش تغییر رنگ دهد.

| | |
|---|----------------------|
| r | صفحه نمایش قرمز شود. |
| g | صفحه نمایش سبز شود |
| b | صفحه نمایش آبی شود |
| w | صفحه نمایش سفید شود |
| y | صفحه نمایش زرد شود |

شکل ۱۰-۶- فهرست رنگ‌ها بر اساس انتخاب نویسه

۱ پروژه جدیدی به نام MColor ایجاد کنید.

۲ برنامه زیر را در متد Main وارد کنید و بخش‌های ناقص برنامه را تکمیل کنید.

```
Console.WriteLine("What's your favorite color?");
Console.WriteLine("[r] red");
Console.WriteLine("[g] green");
Console.WriteLine("[b] blue");
Console.WriteLine("[w] white");
Console.WriteLine("[y] yellow");
Console.Write(" Enter your choice: ");

char choice = char.Parse(Console.ReadLine());

switch (choice)
{
case 'r':
    Console.BackgroundColor = ConsoleColor.Red;
    Console.Clear();
    break;
case 'g':
    Console.BackgroundColor = ConsoleColor.Green;
    Console.Clear();
    break;
case 'b':
    Console.BackgroundColor = ConsoleColor.Blue;
    Console.Clear();
    break;
default:
    Console.BackgroundColor = ConsoleColor.White;
    Console.Clear();
    break;
}
```

- ۳ کد برنامه را طوری تکمیل کنید که فهرست رنگ مانند شکل ۱۰-۶ در صفحه نمایش دیده شود.
- ۴ برای نویسه w و y برنامه را تکمیل کنید.
- ۵ با ورودی‌های مختلف برنامه را اجرا کنید.
- ۶ یک نویسه خارج از فهرست تعیین شده به عنوان ورودی بدهید. صفحه نمایش چه رنگی می‌شود؟
- ۷ به فهرست برنامه امکان انتخاب رنگ قلم صفحه نمایش را هم اضافه کنید.

کارگاه ۱۳ | جزئیات دستور switch و شرط منطقی ||

می‌خواهیم برنامه‌ای بنویسیم که با دریافت نام یک دستگاه کامپیوتری نوع ورودی، خروجی و یا ورودی و خروجی بودن آنها را اعلام کند.

- ۱ پروژه جدیدی با نام IoDevice ایجاد کنید.
- ۲ برنامه زیر را در متد Main وارد کنید.

```
Console.WriteLine("enter your device");
string x = Console.ReadLine();
switch (x)
{
case "keyboard":
case "Mouse":
case "Scanner":
    Console.WriteLine("{0} is input device ",x);
    break;
case "Monitor":
case "Printer":
case "Headphone":
    Console.WriteLine("{0} is output device ",x);
    break;
case "Modem":
case "Network card":
case "Audio Card":
    Console.WriteLine("{0} is Both Input-Output Devices ", x);
    break;
default:
    Console.WriteLine("unknown");
    break;
}
```

- ۳ برنامه را اجرا کنید.
- ۴ قابلیت شناسایی دستگاه‌های بیشتری برای برنامه فراهم کنید.

یک ساختار switch کوچک می‌تواند به ازای دو یا چند مقدار، یک مجموعه عملیات واحد را انجام دهد و در این حالت مقادیر با «یا» بررسی می‌شوند.

یادداشت



برداشت



آنچه آموختم:

۱.
۲.
۳.

جدول ارزشیابی مرحله ۴

| مرحله کار | شرایط عملکرد (ابزار، مواد، تجهیزات، زمان، مکان و ...) | نتایج ممکن | استاندارد (شاخص‌ها/داوری/نمره دهی) | نمره |
|---|--|------------------------|--|------|
| به‌کارگیری دستور switch در برنامه | مکان: کارگاه استاندارد رایانه تجهیزات: رایانه‌ای که نرم‌افزار ویژوال استودیو روی آن نصب باشد زمان: ۱۵ دقیقه | بالاتر از حد انتظار | تشخیص ساختار شرطی مناسب برای برنامه - ایجاد برنامه با استفاده از دستور switch با اجرای صحیح خطایابی دستور switch در برنامه - رفع انواع خطاهای مربوط به switch | ۳ |
| | | قابل قبول | تشخیص ساختار شرطی مناسب برای برنامه - ایجاد برنامه با استفاده از دستور switch با اجرای صحیح خطایابی دستور switch در برنامه | ۲ |
| | | غیر قابل قبول | تشخیص ساختار شرطی مناسب برای برنامه - ایجاد برنامه با استفاده از دستور switch با خطاهای syntax | ۱ |

ارزشیابی کار (شایستگی انجام کار)

☐ بلی☐ خیر

معیار شایستگی انجام کار :

کسب حداقل نمره ۲ از مراحل به‌کارگیری دستور if - else در برنامه

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

کسب حداقل میانگین ۲ از مراحل کار

جدول ارزشیابی پایانی

شرح کار:

- ۱- به کارگیری دستور if در برنامه
- ۲- به کارگیری دستور if - else در برنامه
- ۳- به کارگیری دستور if متداخل در برنامه
- ۴- به کارگیری دستور switch در برنامه

استاندارد عملکرد:

با استفاده از دانش و شناخت اجزای محیط IDE ویزوال استودیو، چگونگی عملکرد ساختارهای شرطی در الگوریتم برنامه را تحلیل و برنامه را با استفاده از دستورات شرطی ایجاد کند.

شاخص‌ها:

| شماره مرحله کار | شاخص‌های مرحله کار |
|-----------------|--|
| ۱ | ارائه راه حل مسئله - تشخیص وجود شرط در راه حل - رسم روندنمای راه حل - انتخاب دستور شرطی - نوشتن برنامه سی شارپ مربوط به روندنما - تشخیص خطاهای if رفع خطاهای if |
| ۲ | ارائه راه حل مسئله - تشخیص وجود شرط ساده یا پیچیده در راه حل - رسم روندنمای راه حل - انتخاب دستور شرطی - نوشتن برنامه سی شارپ روندنما - تشخیص بلاک دستورات بدنه if - تبدیل if-else به عملگر؟ - تشخیص خطاهای if-else - رفع خطاهای if-else |
| ۳ | ارائه راه حل مسئله - تشخیص وجود شرط متداخل در راه حل - رسم روندنمای راه حل - انتخاب دستورات شرطی نوشتن برنامه سی شارپ روندنما - انتخاب عملگر منطقی - ترکیب شرط‌های if متداخل - خطاهای احتمالی if متداخل - رفع خطاهای احتمالی if متداخل |
| ۴ | ارائه راه حل مسئله - تشخیص وجود شرط پیچیده در راه حل - انتخاب دستور شرطی - نوشتن برنامه با دستور شرطی switch - تشخیص خطاهای به کارگیری دستور switch - رفع خطاهای به کارگیری دستور switch |

شرایط انجام کار و ابزار و تجهیزات:

مکان: کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها
 تجهیزات: رایانه‌ای که نرم افزار ویزوال استودیو روی آن نصب باشد
 زمان: ۶۰ دقیقه به کارگیری دستور if در برنامه ۱۰ دقیقه - به کارگیری دستور if-else در برنامه ۱۵ دقیقه - به کارگیری دستور if متداخل در برنامه ۱۵ دقیقه - به کارگیری دستور switch در برنامه ۱۵ دقیقه
 مشخصات فنی رایانه:

معیار شایستگی:

| ردیف | مرحله کار | حدافل نمره قبولی از ۳ | نمره هنجار |
|---|--------------------------------------|-----------------------------|------------|
| ۱ | به کارگیری دستور if در برنامه | ۱ | |
| ۲ | به کارگیری دستور else-if در برنامه | ۲ | |
| ۳ | به کارگیری دستور if متداخل در برنامه | ۱ | |
| ۴ | به کارگیری دستور switch در برنامه | ۱ | |
| شایستگی‌های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش: | | | |
| تصمیم‌گیری - استدلال - حل مسئله - تفکر خلاق - نقش در تیم - زبان فنی | | | |
| دانش ارگونومیک (خود فرد) - کنترل حفاظتی الکتریکی و الکترونیکی (ابزار و تجهیزات) | | | |
| دقت در تشخیص ساختار شرطی مناسب هنگام تبدیل الگوریتم به برنامه در مسئله مطرح شده | | | |
| میانگین نمرات | | | |
| * | | | |

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

- ۱- یاقوتی، نیلوفر. (۱۳۹۵). پنج مزیت ارائه محتوای آموزشی آنلاین. <http://elearning.tebyan.net>، ۱۳۹۵/۷/۴
- ۲- حافظی نژاد، محمد. (۱۳۹۵). چهار گام اصلی تولید محتوای الکترونیکی. <http://raveshtadris.com>، ۱۳۹۵/۷/۴
- ۳- محمدی، محمدرضا و مینایی، غلامرضا. (۱۳۷۷). نرم افزارهای چندرسانه‌ای کد ۴۵۱/۲. تهران: شرکت چاپ و نشر کتاب‌های درسی ایران.
- ۴- کارکن، مهناز و دیگران، (۱۳۹۴)، کتاب معلم راهنمای تدریس برنامه‌سازی یک سی شارپ، تهران: شرکت چاپ و نشر کتاب‌های درسی ایران.
- ۵- کربلایی، مجید، (۱۳۹۳)، برنامه‌سازی ۱ کد ۳۵۸/۷۰، تهران: شرکت چاپ و نشر کتاب‌های درسی ایران.
- 6- Asadi, A (2015), Windows 10 for Beginners, Imagine Publishing Ltd, Volume(22), Issue 4, pages 28-33, 38-45
- 7- James Madison University-Official Site. www.jmu.edu/media-arts/anderson/old/4683/wwwnavi.html
- 8- Serif, Student Project Resources. http://educationresources.serif.com/creating_video/2_design_video_clip/part2-7.htm
- 9- Power ISO All in one solution. 2016, Help PowerISO .<http://www.poweriso.com/tutorials/index.htm>, 2016
- 10- Winrar help .2016, Winrar help .<http://www.rarlab.com/>
- 11- Windows 10 help. 2016, Windows 10 help.<https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/>
- 12- Rathbone.A (2015). Windows 10 For Dummies. new Jersey : John Wiley & Sons, Inc.
- 13- Windows 10 help.2016, Windows 10 help. <http://windows.microsoft.com/en-us/windows-10>, 2016
- 14- Adobe Captivate – Official Site, Blogs.adobe.com/captivate9
- 15- Multimedia Design: Navigation Maps and Storyboards. <http://www.docin.com/p-528258133.html> (2011)
- 16- Swish Max 4.o user guide
- 17- Proshow Gold 6.0 help documentation
- 18- Multimedia: Design: Methodology – storyboard. http://www.virtualmv.com/wiki/index.php?title=Multimedia%3ADesign%3AMethodology_-_storyboard
- 19- C# Programming Guide. <https://msdn.microsoft.com/>



هنرآموزان محترم، هنرجویان عزیز و اولیای آنان می‌توانند نظرهای اصلاحی خود را دربارهٔ مطالب این کتاب از طریق نامه بد نشانی تهران - صندوق پستی ۴۸۷۴ / ۱۵۸۷۵ - گروه درسی مربوط و یا پیام نگار tvoccd@roshd.ir ارسال نمایند.

وبگاه: www.tvoccd.medu.ir

دفترتالیف کتاب های درسی فنی و حرفه ای و کار دانش

همکاران هنرآموز که در فرایند اعتبارسنجی این کتاب مشارکت فعال داشته‌اند:

استان سمنان: اشراف‌السادات موسوی، سیمین معمارزاده

استان سیستان و بلوچستان: علیرضا حمیدی، هوشنگ ابراهیمی

استان فارس: بهاره بهره‌دار، سارا شیخی

استان کهگیلویه و بویراحمد: محمد فرجی‌زاده

استان گیلان: محمود باقری، حمیدرضا گل محمدی