

فصل پنجم

واحد یادگیری ۹ و ۱۰

حل مسائل شرطی

واحد یادگیری ۹

شایستگی کار با عملگرهای ریاضی، مقایسه‌ای و منطقی

مقدمات تدریس

الف) مفاهیم کلیدی

مفاهیم کلیدی			
عبارت	عملگر	عملوند	داده
عملگرهای کاهشی	عملگرهای افزایشی	اولویت عملگرهای ریاضی	عملگرهای ریاضی
عملگرهای منطقی	اولویت عملگرهای مقایسه‌ای	عملگرهای مقایسه‌ای	عملگرهای انتساب
خطاهای رایج	برنامه‌نویسی با عملگرها	اولویت عملگرها	عملگر الحاق رشته

ب) مراحل انجام کار واحد یادگیری

مراحل انجام کار	
۱	نوشتن عبارات ریاضی به زبان برنامه‌نویسی
۲	به‌کارگیری عملگرهای افزایشی، کاهشی و انتساب
۳	استفاده از عملگرهای مقایسه‌ای
۴	استفاده از عملگرهای منطقی

ج) تجهیزات لازم

مشخصات سخت‌افزاری برای نصب نرم‌افزار Visual Studio Express 2012

- پردازنده حداقل ۱/۶ گیگاهرتز
- حافظه RAM حداقل ۱ گیگابایت
- حداقل فضای موجود در دیسک سخت ۱ گیگابایت
- کارت ویدئویی متناسب با DirectX9 و قابل اجرا در رزولوشن ۷۶۸×۱۰۲۴ و بالاتر

د) بودجه‌بندی

برای بودجه‌بندی فصل پنجم می‌توانید از نمونه پیشنهادی استفاده کنید.

جلسه	واحد یادگیری	کارگاه (موضوع)	شماره صفحات	اهداف توانمندسازی	فعالیت‌های تکمیلی
۲۹	۹	مفهوم عبارت، عملوند و عملگر - عملگرهای ریاضی	۲۱۹-۲۱۴	تعریف و بیان مفهوم عملگر، عملوند و عبارت- بیان کاربرد عملگرها و عبارات ریاضی در زندگی روزمره - آشنایی با عملگرهای ریاضی برای محاسبه خارج قسمت و باقیمانده صحیح و اعشاری	- ارائه مثال‌های دلخواه از کاربرد عملیات ریاضی در زندگی روزمره - ایجاد یک پروژه برای یکی از کاربردهای عملیات ریاضی در زندگی روزمره - ایجاد برنامه برای محاسبه خالص حقوق دریافتی با کسر ۷ درصد مالیات و ۵ درصد بیمه
۳۰	۹	اولویت‌بندی عملگرهای ریاضی - عملگرهای انتساب - عملگرهای افزایشی و کاهشی	۲۲۰-۲۲۷	آشنایی با ترتیب اولویت عملگرهای ریاضی - آشنایی با نحوه تغییر اولویت عملگرها - آشنایی با عملگرهای انتساب - شناخت کاربرد عملگرهای انتساب معرفی عملگرهای کاهشی و افزایشی - آشنایی با شکل‌های پیشوندی و پسوندی عملگرهای افزایشی و کاهشی	- ایجاد یک پروژه برنامه‌نویسی برای حل معادله درجه ۲ - ارائه یک معمای ریاضی دلخواه و سپس ایجاد یک پروژه برنامه‌نویسی برای حل آن
۳۱	۹	عملگر الحاق دو رشته - عملگرهای مقایسه‌ای	۲۳۳-۲۲۸	آشنایی با عملگر الحاق دو رشته - شناخت عملگرهای رابطه‌ای و مقایسه‌ای - آشنایی با اولویت عملگرهای رابطه‌ای و مقایسه‌ای - بیان کاربرد عملگرهای مقایسه‌ای	- ایجاد یک پروژه برنامه‌نویسی برای نمایش سطح علمی یک هنرجو بر اساس معدل در چهار سطح ضعیف، متوسط، خوب و عالی
۳۲	۹	عملگرهای منطقی رفع خطاهای رایج در عملگرهای مقایسه‌ای و منطقی	۲۴۲-۲۳۴	آشنایی با انواع عملگرهای منطقی - شناسایی و رفع خطاهای رایج در عملگرهای منطقی و مقایسه‌ای	- ایجاد یک پروژه برنامه‌نویسی که نشان دهد معادله درجه ۲ ریشه مضاعف دارد یا ریشه ندارد.
	۱۰	کارگاه ۱	۲۴۷-۲۴۳	ساختار و عملکرد دستور شرطی if	مثال‌هایی ساده جهت استفاده از دستور if

جلسه	واحد یادگیری	کارگاه (موضوع)	شماره صفحات	اهداف توانمندسازی	فعالیت‌های تکمیلی
۳۳	۱۰	کارگاه ۲ الی ۶	۲۴۸ - ۲۵۴	ساختار و عملکرد دستور شرطی -if, else, بلاک در ساختار شرطی if, اجرای گام به گام if else, عملکرد سه تایی، عبارت شرطی تو در تو	برنامه‌ای که سه عدد را از ورودی گرفته و تشخیص دهد که اعداد وارد شده می‌توانند اضلاع یک مثلث باشند یا خیر. برنامه‌ای که سال میلادی را دریافت کرده و تشخیص دهد سال وارد شده کبیسه است یا خیر. برنامه دریافت دو عدد و بخش پذیر بودن یکی بر دیگری. برنامه‌ای که سال، ماه و روز را دریافت کرده و تاریخ فردای آن روز را مشخص کند.
۳۴	۱۰	کارگاه ۷ الی ۱۰	۲۵۵ - ۲۶۱	عبارات شرطی با عملگرهای منطقی، مرتب‌سازی، ترکیب دستورات منطقی به محاسباتی و شرطی، تبدیل if به switch	راه حل دوم سه ضلع تشکیل دهنده یک مثلث برنامه که ۷ رنگ را به کاربر نشان داده و کاربر با انتخاب رنگ، رنگ زمینه را تغییر دهد ابتدا با if نوشته شود و از دانش آموز خواسته شود که به Switch تبدیل کند.
۳۵	۱۰	کارگاه ۱۱ الی ۱۳	۲۶۲ - ۲۶۶	ساختار switch، ایجاد فهرست در کنسول، جزییات دستور switch و شرط منطقی	راه حل دوم و سوم مسئله سال کبیسه میلادی برنامه‌ای که ۷ رنگ را به کاربر نشان داده و کاربر با انتخاب رنگ، رنگ زمینه را تغییر دهد. برنامه طراحی ماشین حساب و انجام ۴ عمل اصلی توسط آن.

زمان‌بندی پیشنهادی واحد یادگیری:

ردیف	اهداف توانمندسازی	زمان تئوری	زمان تدریس عملی
۱	مفاهیم پایه مربوط به عبارت ریاضی	۲	
۲	عملگرهای ریاضی به ترتیب اولویت	۲	
۳	ایجاد عبارت ریاضی در برنامه‌نویسی		۲
۴	مفهوم خارج قسمت و باقیمانده صحیح و اعشاری	۱	
۵	ایجاد برنامه با عملگر تقسیم و باقیمانده صحیح و اعشاری		۲
۶	عملکرد عملگر کاهشی و افزایشی	۱	
۷	ایجاد برنامه با عملگرهای کاهشی و افزایشی		۲
۸	کاربرد انواع عملگرهای انتساب	۱	
۹	ایجاد برنامه با عملگر انتساب		۲
۱۰	عملکرد عملگرهای مقایسه‌ای	۱	
۱۱	ایجاد برنامه با عبارت ترکیبی مقایسه‌ای		۲
۱۲	کاربرد و ویژگی‌های نوع داده منطقی	۱	
۱۳	ایجاد برنامه با عبارت ترکیبی منطقی و مقایسه‌ای		۳
۱۴	رفع خطاهای رایج در عملگرهای مقایسه‌ای و منطقی		۲
۱۵	تعریف و نحوه محاسبه حاصل عبارت محاسباتی	۱	
۱۶	کاربرد عملگر +	۱	
۱۷	به‌کارگیری عبارت محاسباتی در برنامه		۳
۱۸	معرفی اصطلاحات انگلیسی رایج در استفاده از عبارت محاسباتی	۱	
	مجموع ساعات	۱۲	۱۸

طرح درس روزانه (هفتگی) پیشنهادی				
پایه: دهم			درس: حل مسئله	
پیام جلسه (هدف کلی): توانایی و شایستگی تعیین اجزا و حل مسئله				
اهداف یادگیری		فعالیت‌ها		زمان
طبقه هدف: حیطه عاطفی / شناختی / روانی - حرکتی	کار هنرآموز	کار هنرجو	مدت (دقیقه)	
سنجش میزان آگاهی هنرجو یان در خصوص حل مسئله	مطرح کردن سؤالات پیش‌آزمون جهت تعیین سطح هنرجویان از طریق بارش مغزی	مشارکت در پاسخگویی، تعامل و بیان مثال‌هایی از فعالیت‌های روزمره که انجام آن‌ها نیاز به محاسبات ریاضی دارد.	۱۰	
ایجاد انگیزه	ایجاد توجه و تمرکز برای ورود به بحث روش حل مسئله	بیان مثال‌هایی ملموس از فعالیت‌های روزمره که انجام آن‌ها نیاز به محاسبات ریاضی دارد. ارائه یک نرم‌افزار کاربردی مناسب	۲۰	
ارائه مفاهیم کلیدی (توضیح هنرآموز)	توضیح کامل مفاهیم کلیدی (دانشی) و ایجاد علاقه و انگیزه در هنرجویان (بینشی)	بیان مفاهیم عبارت، عملوند، عملگر و معرفی انواع عملگرها، معرفی عملگرهای محاسباتی و اولویت عملگرهای محاسباتی	۳۰	
فعالیت کارگاهی (تمرین هنرجویان)	نوشتن عبارت ریاضی در زبان برنامه‌نویسی، تغییر اولویت عملگرها، محاسبه نتیجه عبارت ریاضی، انتخاب نوع متغیر مناسب برای ذخیره نتیجه عبارت	ارائه و تشریح صورت مسائل و فعالیت‌های کارگاهی برای هنرجویان	۱۱۰	
ارزیابی فعالیت‌ها ارائه تمرین	توانایی نوشتن برنامه برای مسائل مطرح‌شده در فعالیت‌های کارگاهی	سرکشی و ارزیابی فعالیت‌های هنرجویان به دو صورت فردی و گروهی با توجه به شیوه‌های ارزشیابی از بخش ارزیابی فعالیت‌های کارگاهی. قبل از شروع فعالیت، ملاک‌های ارزیابی پروژه به هنرجویان اعلام شود. در انجام ارزیابی فعالیت	۶۰	

طرح درس روزانه (هفتگی) پیشنهادی				
درس: حل مسئله			پایه: دهم	
پیام جلسه (هدف کلی): توانایی و شایستگی تعیین اجزا و حل مسئله				
		گروه‌ها از هنجریان مستعد و سرگروه‌ها کمک بگیرند.		
۲۰	گوش دادن فعال، پرسش و پاسخ و برنامه‌ریزی به‌منظور انجام پژوهش	با توجه به نمودار مفهومی و ذهنی این مبحث از واحد یادگیری، مطالب گفته شده مرور و در مورد درس جلسه آینده حدود چند دقیقه به هنجرو اطلاعاتی ارائه دهد.	هنرجو با مفاهیم عبارت، عملوند، عملگر و انواع عملگرهای ریاضی و اولویت آن‌ها در مسئله آشنا شده و توانایی برنامه‌نویسی در این زمینه را داشته باشد.	ارائه نکات تکمیلی (جمع‌بندی)
۹۰	تکمیل فعالیت‌های کارگاهی و کارهای گروهی مربوط به مبحث درسی و ارائه توانمندی‌ها و شایستگی‌های خود.	ارزشیابی پایانی را با توجه به شیوه‌های ارزشیابی پایانی در بخش فعالیت‌های پس از تدریس برای هنجریان انجام دهد.	نوشتن عبارات ریاضی به زبان برنامه‌نویسی	ارزشیابی شایستگی (ارزشیابی پایانی)
۲۰	تقسیم‌بندی وظایف پروژه بین اعضای گروه و تحویل در زمان‌بندی تعیین شده	تعیین تکلیف جهت تمرین در منزل	انجام پروژه آموزشی به صورت گروهی یا فردی	تمرین در منزل
ویدئو پروژکتور، رایانه، تخته آموزشی، برنامه VisualStudio ، برنامه Netsupport				ابزارهای موردنیاز

ه) فیلم‌ها و پرونده‌های پیوست

شماره فیلم	هدف
102h47	عبارت محاسباتی، عملوند، عملگر
102h48	بررسی عملگرهای انتساب و عملگر الحاق رشته
102h49	بررسی عملگرهای مقایسه‌ای و منطقی
102h50	اشکال‌زدایی عبارات Debugging، اشکال‌زدایی عملگرهای رشته‌ای، اشکال‌زدایی خطای تقسیم بر صفر

نکته: برای انجام فعالیت‌های تکمیلی می‌توانید از تمرینات موجود در «پرونده پیوست فعالیت‌های تکمیلی» استفاده کنید.

و) ورود به بحث

انتظار می‌رود هنرجو با یادگیری این واحد، مهارت‌های زیر را کسب کند:

نوشتن عبارات ریاضی و محاسباتی به زبان برنامه نویسی سی شارپ



به کارگیری عملگرهای افزایشی، کاهشی و انتساب



استفاده از عملگرهای مقایسه‌ای و رابطه‌ای



استفاده از عملگرهای منطقی



به کارگیری انواع عبارت‌ها در برنامه



اولویت‌بندی عملگرها در عبارت‌های ترکیبی

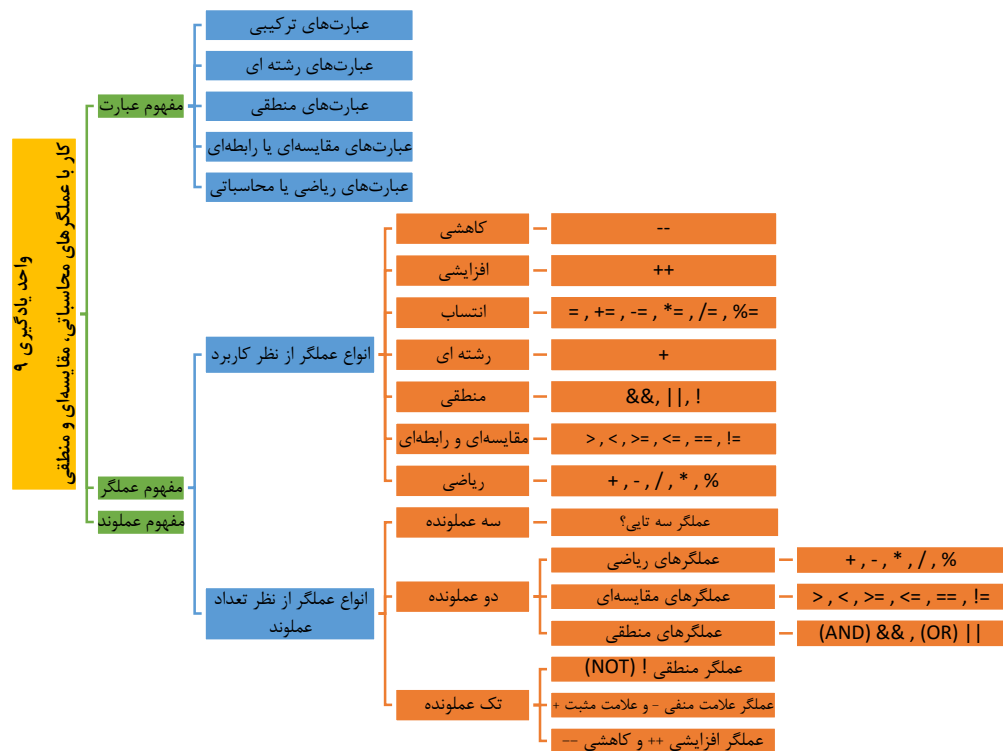


شناسایی و رفع خطاهای رایج برنامه‌نویسی در هنگام استفاده از انواع عملگرها



نمودار زیر نقشه مفهومی و ذهنی این واحد یادگیری را نمایش می‌دهد و شما می‌توانید با داشتن یک دورنما از مطالب و محتوای آموزشی، تدریس خود را به شکل بهتری مدیریت کنید.

نمودار ۱- نقشه مفهومی و ذهنی واحد یادگیری ۹



تذکر: در بخش عبارت‌های مقایسه‌ای و منطقی با مثال‌های ساده، هنجریان را در مسیر آمادگی برای درک مفهوم عبارت‌های شرط هدایت کرده ولی در این واحد یادگیری به بیان دستور if پرداخته نشود تا هنجرو جای خالی این دستور را در برنامه‌نویسی حس کرده و زمینه لازم را جهت تدریس واحد یادگیری بعدی مهیا کنید. شما می‌توانید از مثال‌هایی مانند موارد زیر استفاده کنید:

مثال ۱: تعداد هنجریان پایه دهم از تعداد هنجریان پایه دوازدهم بیشتر است.

مثال ۲: قیمت یک دیسک سخت خارجی قابل حمل (External Hard Disk) از یک دیسک سخت داخلی (Internal Hard Disk) بیشتر است.

مثال ۳: ظرفیت ذخیره‌سازی DVD از CD کمتر است.

مثال ۴: تعداد صفحات کتاب شما از تعداد صفحات کتاب دوستان بیشتر است و قیمت کتاب شما نیز از قیمت کتاب دوستان کمتر است.

مثال ۵: امروز هوا ابری است و باران می‌بارد.

مثال ۶: روز جمعه این هفته من به دیدن دوستم خواهیم رفت یا به تماشای تلویزیون خواهیم پرداخت.

تذکر: پیشنهاد می‌شود از بیان عبارت‌ها و مسائل ریاضی پیچیده در این مرحله خودداری کنید.

تدریس

عملگرهای محاسباتی

شرکت‌پذیری یک عملگر، یک ویژگی است که مشخص می‌کند در شرایط تقدم یکسان، عملیات از چه سمتی انجام می‌شود.

انواع شرکت‌پذیری:

- شرکت‌پذیری از راست یا راست به چپ
- شرکت‌پذیری از چپ یا چپ به راست
- ✓ عملگر انتساب دارای شرکت‌پذیری از راست است.
- ✓ عملگرهای ریاضی، شرکت‌پذیر از چپ هستند.
- ✓ در برخی زبان‌های برنامه‌نویسی عملگرهایی وجود دارد که شرکت‌ناپذیر هستند؛ مانند عملگر - در زبان پرولوگ

چند نکته:

- عملوندهای عبارت محاسباتی می‌توانند داده یا متغیر باشند.

$result = num1 + num2$

- اگر در یک عبارت محاسباتی چندین پرانتز تودرتو وجود داشته باشد؛ محاسبات از داخلی‌ترین پرانتز آغاز می‌شود.
- عملگرهای ++ و -- عملگرهای یکتایی هستند و فقط بر روی متغیرها عمل می‌کنند.
- عملگر + عملگر محاسباتی و عملگر الحاق دو رشته است و عملگر += به‌عنوان عملگر انتساب الحاق دو رشته به کار می‌رود.
- عملوند عملگرهای <، >، <=، >= نمی‌توانند از نوع string و bool باشند.
- عملوندهای یک عملگر منطقی می‌تواند متغیری از نوع bool، یک شرط و یا مقادیر true و یا false باشد.
- برنامه با استفاده از نتایج عملگرهای مقایسه‌ای و منطقی، برای انجام عملیات بعدی تصمیم‌گیری می‌کند.

مشکلات متداول در فرآیند یاددهی-یادگیری:

هنرجویان در انجام فعالیت‌های کارگاهی این واحد یادگیری ممکن است در هنگام کد نویسی، ترجمه کدها و اجرای برنامه با خطاهای متعددی مواجه شوند و لازم است هنرآموزان عزیز قبل از حضور در کلاس و تدریس این واحد یادگیری دانش لازم در خصوص انواع خطاها و چالش‌های حین تدریس این مبحث کسب کنند؛ بنابراین در ادامه به معرفی و شرح برخی خطاهای رایج و مرتبط با موضوع و مباحث این واحد یادگیری می‌پردازیم.

CS0019:Operator 'operator' cannot be applied to operands of type 'type' and 'type'A binary operator is applied to data types that do not support it

این خطا زمانی رخ می‌دهد که از عملگر دوتایی روی نوع داده‌ای استفاده شده باشد که مجاز نیست. برای مثال شما نمی‌توانید از عملگر || روی string و عملگرهای <، >، + یا > روی متغیرهای bool استفاده کنید مگر اینکه این عملگرها به صورت صریح در کلاس overload شده باشند. به مثال زیر توجه کنید.

```
namespace cs0019
{
    public class b
    {
        static void Main()
        {
            bool result = true;
            if (result > 0) //CS0019 bool روی متغیر > استفاده از عملگر
            {
                // Do something.
            }
        }
    }
}
```

```

}
int i = 1;
//
if (i == true) //CS0019 شما نمی‌توانید متغیر bool را با متغیر int مقایسه کنید
{
    //Do something...
}
string s = "Just try to subtract me.";
float f = 100 - s; // CS0019 استفاده از عملگر - روی متغیر رشته‌ای مجاز نیست
}
}
}

```

خطاهای این کد عبارت‌اند از:

```

Error 1      Operator '>' cannot be applied to operands of type
'bool' and 'int'
Error 2      Operator '==' cannot be applied to operands of
type 'int' and 'bool'
Error 3      Operator '-' cannot be applied to operands of type
'int' and 'string'

```

CS0023: Operator 'operator' cannot be applied to operand of type 'type' An attempt was made to apply an operator to a variable whose type was not designed to work with the operator.

این خطا زمانی رخ می‌دهد که از عملگرهای یکتایی مثل عملگر قرینه (-) و نقیض (!) روی عملوندی استفاده شوند که مجاز نیست و آن نوع متغیر برای کار با این عملگر طراحی نشده است. به مثالی که در ادامه آمده توجه کنید.

```

namespace x
{
    public class a
    {
        public static void Main()
        {
            string s = "hello";
            s = -s; // CS0023 عملگر قرینه روی رشته مجاز نیست
        }
    }
}

```

```
s != s; // CS0023 عملگر! روی رشته مجاز نیست
s = s - 9; //CS0019 خطای شماره ۱۹ روی عملگر دوتایی است
}
}
}
```

خطاهای این کد عبارت‌اند از:

Error	1	Operator '-' cannot be applied to operand of type 'string'
Error	2	Operator '!' cannot be applied to operand of type 'string'
Error	3	Operator '-' cannot be applied to operands of type 'string' and 'int'

CS0131: The left-hand side of an assignment must be a variable, property or indexer

این خطا زمانی رخ می‌دهد که در سمت چپ عملگر انتساب نام متغیر یا مشخصه نباشد. به مثال زیر توجه کنید.

```
using System;
namespace MyNamespace
{
    public class MyClass
    {
        public int i = 0;
        public void MyMethod()
        {
            i++ = 1; // CS0131
            // try the following line instead
            // i = 1;
        }
        public static void Main() { }
    }
}
```

خطای این کد عبارت است از:

Error	1	The left-hand side of an assignment must be a variable, property or indexer
-------	---	---

CS0201: Only assignment, call, increment, decrement, and new object expressions can be used as a statement

این خطا زمانی رخ می‌دهد که عبارتی به شکل صحیح استفاده نشده است. تنها دستورات انتساب، فراخوانی‌ها، عملگرهای افزایشی و کاهش‌ی و دستورات ایجاد یک شی می‌توانند به عنوان عبارت استفاده شوند. عبارت غیر معتبر عبارتی است در یک خط یا چند خط که در انتهای آخرین خط علامت؛ دارد و در آن علامت = برای دستور انتساب، () برای فراخوانی متد، new، عملگر ++ یا -- وجود ندارد. به مثال زیر توجه کنید:

```
namespace MyNamespace
{
    public class MainClass
    {
        public static void Main()
        {
            2 * 3; // CS0201
            // Try the following line instead.
            // int i = 2 * 3;
        }
    }
}
```

Error 1 Only assignment, call, increment, decrement, await, and new object expressions can be used as a statement

پاسخ به فعالیت‌ها

متغیر result می‌تواند از چه نوعی باشد ؟
پاسخ: نوع داده‌ای مناسب نوع int است. ولی می‌تواند از نوع float یا حتی byte نیز باشد.

کنجکاوی
ص ۲۱۵

در جدول زیر عملوندها و عملگرهای عبارات محاسباتی را مشخص کنید.

عملوند	عملگر	عبارت
۵	- (قرینه)	-۵
۷ ۱۳ ۱۲	- (تفریق) + (جمع)	۱۲-۱۳-۷
a b 2	- (تفریق) * (ضرب)	b*2-a

فعالیت کارگاهی
ص ۲۱۵

کنجکاوی
ص ۲۱۶

دستور `Console.WriteLine(۵+۶);` چه مقداری را روی صفحه نمایش نشان می‌دهد؟ پاسخ: ۱۱

فعالیت کارگاهی
ص ۲۱۶

جدول ۲ را تکمیل کنید.

عملگر	عملکرد	مثال	نتیجه
-	قرینه	-۵	-۵
*	ضرب	۲۰*۶	۱۲۰
/	تقسیم	۲۵/۴	۶
%	باقیمانده	۲۰٪۶	۲
+	جمع	۲۰+۳	۲۳
-	تفریق	۲۰-۳	۱۷

کارگاه ۱ – عملگر تقسیم

سؤال
ص ۲۱۷

پروژه جدیدی به نام `ConsloePopulation` ایجاد کنید و کد زیر را در آن وارد کنید. 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	۳	۳.۵۵
۹۰ (کاشان)	۳۲۳۳۷۱	۹۵۳۱۴	۳	۳.۳۹
۹۰ (فارس)	۹۳۹۴۱	۲۴۷۶۵	۳	۳.۷۹

- آیا خروجی برنامه با جواب شما برای بعد خانوار، یکسان است؟ پاسخ: خیر

فعالیت کارگاهی
ص ۲۱۸

برنامه‌ای بنویسید که حاصل عبارت $۱۰ * ۲ + ۳$ را نمایش دهد. کدامیک از نتایج بالا نمایش داده می‌شود؟
پاسخ: ۲۳

```
Console.WriteLine(۱۰ * ۲ + ۳);
```

- در برنامه بالا عبارت $۱۰ * (۳ + ۲)$ را جایگزین کنید و نتیجه را بنویسید.
پاسخ: ۵۰

```
Console.WriteLine(۱۰ * (۳ + ۲));
```

در تبدیل یک عبارت ریاضی به معادل آن در سی شارپ، درج پرانتز در چه مواقعی ضروری است؟
پاسخ: در عبارات ریاضی برای تغییر اولویت عملگرها و همچنین در عبارات کسری برای مشخص کردن صورت و مخرج از پرانتز استفاده می‌شود.

کارگاه ۲- تغییر اولویت عملگرها

پروژه جدیدی به نام ConsoleDiscount ایجاد کنید. دستورات را اضافه و برنامه را اجرا کنید. آیا خروجی برنامه با نتیجه‌ای که شما محاسبه کرده‌اید یکسان است؟
پاسخ: خیر

- نتیجه محاسبه اشتباه است. دلیل رخ دادن چنین خطایی چیست؟
پاسخ: از اولویت‌بندی پیش‌فرض زبان سی شارپ استفاده شده است.

- خطای منطقی، برنامه فوق را چگونه برطرف کنیم؟
پاسخ: با استفاده از پرانتز اولویت‌بندی دلخواه را پیاده‌سازی کنیم.

- پس از برطرف کردن خطای منطقی، برنامه را با مقادیر جدول زیر اجرا کنید.
کد کامل برنامه مربوط به این فعالیت کارگاه:

```
float familySize =(float) populate / numFamily;
static void Main(string[] args)
{
    Console.WriteLine(" ----- Final Price-----");
    //Get the First Price
    Console.WriteLine("\nEnter Price1: ");
    float price1 = float.Parse(Console.ReadLine());
    //Get the Second Price
    Console.WriteLine("\nEnter Price2: ");
    float price2 = float.Parse(Console.ReadLine());
    //Get the Third Price
    Console.WriteLine("\nEnter Price3: ");
    float price3 = float.Parse(Console.ReadLine());
    //Get the fourth Price
    Console.WriteLine("\nEnter Price4: ");
    float price4 = float.Parse(Console.ReadLine());
    //Calculate the Discount
```



```
float discount = (((price1 + price2 + price3) * 10) / 100) + ((price4
* 70) / 100);
Console.WriteLine("\n *** final Discount = " + discount);
Console.ReadKey();
}
```

قیمت کالا ۱	قیمت کالا ۲	قیمت کالا ۳	قیمت کالا ۴	میزان تخفیف	خرجی برنامه
۱۵۰	۱۰۰	۴۲۰	۲۳۰	۲۲۸	۲۲۸

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

پاسخ: کدهای زیر را به برنامه اضافه کنید.

```
float payment = (price1 + price2 + price3 + price4) - discount;
Console.WriteLine("\n *** final Payment = " + payment);
```

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

پاسخ: کد مربوط به محاسبه تخفیف زیر را در برنامه به صورت زیر اصلاح کنید.

```
//Calculate the Discount
float discount = (((price1+Price3)*10)/100) + ((price2*15)/100) +
((price4 * 65) / 100);
```

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

```
float familySize =(float) populate / numFamily;
static void Main(string[] args)
{
    Console.WriteLine(" ----- Final Price-----");
    //Get the First Price
    Console.WriteLine("\nEnter Price1: ");
    float price1 = float.Parse(Console.ReadLine());

    //Get the First Discount
    Console.WriteLine("\nEnter Discount1: ");
    float discount1 = float.Parse(Console.ReadLine());

    //Get the Second Price
    Console.WriteLine("\nEnter Price2: ");
```

```

float price2 = float.Parse(Console.ReadLine());

//Get the Second Discount
Console.WriteLine("\nEnter Discount2: ");
float discount2 = float.Parse(Console.ReadLine());

//Get the Third Price
Console.WriteLine("\nEnter Price3: ");
float price3 = float.Parse(Console.ReadLine());

//Get the Third Discount
Console.WriteLine("\nEnter Discount3: ");
float discount3 = float.Parse(Console.ReadLine());

//Get the fourth Price
Console.WriteLine("\nEnter Price4: ");
float price4 = float.Parse(Console.ReadLine());

//Get the Fourth Discount
Console.WriteLine("\nEnter Discount4: ");
float discount4 = float.Parse(Console.ReadLine());

//Calculate the Discount
float discount = ((price1*discount1)/100) +
((price2*discount2)/100) + ((price3*discount3)/100) + ((price4 *
discount4) / 100);
Console.WriteLine("\n *** final Discount = " + discount);
Console.ReadKey();
}

```

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 هرماه ۱۰ درصد مالیات و ۷ درصد بیمه کسر می‌شود، برنامه‌ای بنویسید که حقوق شخص را دریافت کرده، حقوق او را پس از کسر مالیات، بیمه و مبلغ اهدایی نمایش دهد.

پاسخ:

```
static void Main(string[] args)
```

فعالیت کارگاهی
ص ۲۲۱

```
{
    //Get the Salary
    Console.WriteLine("\nEnter Salary: ");
    float salary = float.Parse(Console.ReadLine());
    //Calculate the finalSalary
    float finalSalary = salary - (((salary * 10) / 100) + ((salary
* 7) / 100) + ((salary * 5) / 100));
    Console.WriteLine("\n *** Salary After Fractions = " +
finalSalary);
    Console.ReadKey();
}
```

کارگاه ۳ – عملگرهای افزایشی کاهشی

فعالیت گروهی

ص ۲۲۵

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);
```

پاسخ:

```
Console.WriteLine("count of students class 102={0} ",
count102);
Count102= count102+1;
Console.WriteLine("count of students class 102={0} ",
count102+1);
```

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);
```

پاسخ:

x	y	خروجی برنامه
۱۲	-۱۲	x=12 y=-11
۱۳	-۱۱	x=12 y=-10
۱۲	-۱۰	
۱۳	۱۲	x=13 y=12

نتیجه عبارت منطقی در چه نوع متغیری ذخیره می‌شود؟ متغیر flag چه نوعی است؟
پاسخ: bool

کنجکاوی
ص ۲۲۸

کارگاه ۵- عملگرهای منطقی

با توجه به مقادیر متغیرها جدول زیر را تکمیل کنید.
int a=5, failed=10, passed=15;

عبارت منطقی	نتیجه
passed >= 12	True
failed + 2 < passed - a	False
passed != 20	True
failed == passed - a	True

فعالیت کارگاهی
ص ۲۲۹

کارگاه ۶- اشکال زدایی (Debugging) عبارات

پروژه‌های با نام Debug1 ایجاد و کد زیر را وارد و با استفاده از کلید F6 ترجمه کنید.
خطاهای این برنامه و علت رخ دادن آن‌ها را در کادر زیر بنویسید.

سؤال
ص ۲۳۰

ردیف	خطا	خط	ستون	علت
۱	Only assignment, call, increment, decrement, and new object expressions can be used as a statement	۵	۱۳	num1*5; عبارات فقط می‌توانند شامل دستورات افزایشی، کاهش‌ی و انتساب باشند و استفاده از عبارت به این صورت در یک خط کد اشتباه است.

ردیف	خطا	خط	ستون	علت
۲	Only assignment, call, increment, decrement, and new object expressions can be used as a statemen	۷	۱۳	num1/2; عبارات فقط می‌توانند شامل دستورات افزایشی و کاهشی و انتساب باشند و استفاده از عبارت به این صورت در یک خط کد اشتباه است.

کارگاه ۷- اشکال زدایی در رشته‌ها

سؤال

ص ۲۳۲

پروژه‌ای با نام Debug2 ایجاد و کد زیر را وارد و با استفاده از کلید F6 ترجمه کنید. خطاهای این برنامه و علت رخ دادن آن‌ها را در جدول زیر بنویسید.

ردیف	خطا	خطا	ردیف	علت
۱	Operator '>' cannot be applied to operands of type 'bool' and 'int'	۲	۳۱	result > 0 عملگر > نمی‌تواند روی نوع داده‌ای bool اعمال شود.
۲	Operator '-' cannot be applied to operands of type 'string' and 'string'	۶	۱۸	s1 - s2 عملگر - نمی‌تواند روی نوع داده string اعمال شود.
۳	Operator '-' cannot be applied to operands of type 'int' and 'string'	۷	۲۳	100 - s1 عملگر - نمی‌تواند روی نوع داده string اعمال شود.
۴	Operator '-' cannot be applied to operand of type 'string'	۸	۱۸	-s1 عملگر کاهشی - نمی‌تواند روی نوع داده string اعمال شود.

کارگاه ۸- اشکال زدایی خطای تقسیم بر صفر

سؤال

ص ۲۳۱

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

تعداد دفترچه‌ها	تعداد بچه‌ها	خروجی
۵۰۰	۲۰۰	۲
۲۰۰	۰	خطای زمان اجرا

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

علت	خطا
تقسیم بر صفر	Attempted to divide by zero

- تفاوت خطای این کارگاه با خطای کارگاه ۷ و ۶ چیست؟
پاسخ: در برنامه قبل خطا از نوع خطای زمان کامپایل بود و برنامه اجرا نشد. ولی در این برنامه خطا از نوع خطای زمان اجراست و بعد از اجرای برنامه رخ می‌دهد.

کارگاه ۹- عملگر منطقی &&

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

x	y	خروجی
۳	5	False
۲	4	False
۲	5	True
۳	4	False

سؤال

ص ۲۳۳

کارگاه ۱۰- عملگر منطقی ||

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

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

سؤال

ص ۲۳۵

کارگاه ۱۱- عملگر منطقی !

پروژه‌ای با نام WeatherCamp ایجاد کنید. برنامه را اجرا و جدول را تکمیل کنید.

خروجی	وضعیت هوا
False	Cloudy
True	Sunny

سؤال

ص ۲۳۶

کنجاوی
ص ۲۳۷

کدام یک از عملگرهای منطقی یکتایی و کدام دوتایی هستند؟

عملگر NOT (!)

یکتایی

عملگرهای AND (&&) و عملگر OR (||)

دوتایی

الف) نمونه ارزشیابی پایانی

مرحله	سوال	شاخص	حد اقل نمره قبولی از ۳	نمره هر شاخص	نمره / وضعیت شایستگی
۱	می‌خواهیم برای یک فروشگاه که دارای اجناس تخفیف‌دار است، برنامه‌ای بنویسیم به صورتی که الف- نام ۴ کالا به همراه قیمت قبل از تخفیف آن‌ها، تعداد خریداری شده و درصد تخفیف هر کالا را از ورودی دریافت کرده، مبلغ تخفیف هر کدام را محاسبه کند ب- دستوراتی به برنامه اضافه کنید که مبلغ نهایی تخفیف و مبلغ قابل پرداخت را محاسبه کرده، نمایش دهد	نوشتن عبارت ریاضی در زبان برنامه‌نویسی	۱		نمره : غیر قابل قبول <input type="checkbox"/> قابل قبول <input type="checkbox"/> بالتر از حد انتظار <input type="checkbox"/>
		تغییر اولویت عملگرها و 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			
۲	ج- برنامه را به گونه‌ای تغییر دهید که ۱ واحد به درصد تخفیف کالاهایی که تعداد درخواستی آن‌ها بیش از ۵ است اضافه و ۱ واحد از درصد تخفیف کالاهایی که تعداد درخواستی آن‌ها کوچکتر یا مساوی ۲ است کم کند د- با فرض اینکه مدیر فروشگاه قصد دارد برای جذب مشتری بیشتر، ۵ درصد به همه تخفیف‌ها اضافه کند، تغییرات لازم را در برنامه اعمال نمایید	استفاده از عملگرهای کاهش، افزایش و انتساب در برنامه	۱		نمره : غیر قابل قبول <input type="checkbox"/> قابل قبول <input type="checkbox"/> بالتر از حد انتظار <input type="checkbox"/>
		محاسبه نتیجه عبارت شامل عملگرهای کاهش و افزایش و انتساب			
		محاسبه نتیجه عبارت شامل + و + = و انتخاب نوع متغیر مناسب برای ذخیره نتیجه عبارت			
۳	ه- به تصمیم مدیر فروشگاه به تمامی مشتریانی که مبلغ کل خریدشان بعد از کسر تخفیف بیشتر از ۳۰۰۰۰ تومان باشد جایزه‌ای تعلق خواهد گرفت. بر این اساس برنامه را تغییر دهید به طوری که این شرایط را نیز بررسی کند و نتیجه را نمایش دهد	بکارگیری عملگرهای مقایسه‌ای	۱		نمره : غیر قابل قبول <input type="checkbox"/> قابل قبول <input type="checkbox"/> بالتر از حد انتظار <input type="checkbox"/>
		تشخیص اولویت اجرای عملگرهای ریاضی و مقایسه‌ای و انتساب در یک عبارت و تعیین نتیجه عبارت منطقی			
		تشخیص و رفع خطای استفاده نادرست از عملگرهای ریاضی، مقایسه‌ای و انتساب			
۴	و- تعداد درخواستی از هر کالا نباید کمتر از صفر و مقدار تخفیف هر کالا فقط می‌تواند عددی از ۰ تا ۵۰ باشد. برنامه را به گونه‌ای تغییر دهید که این شرایط را چک کرده و نتیجه را نمایش دهد	استفاده از عملگرهای منطقی در ترکیب شرطها	۲		نمره : غیر قابل قبول <input type="checkbox"/> قابل قبول <input type="checkbox"/> بالتر از حد انتظار <input type="checkbox"/>
		تشخیص اولویت اجرای عملگرهای ریاضی، مقایسه‌ای و منطقی در یک عبارت ترکیبی منطقی و محاسبه نتیجه عبارت ترکیبی منطقی			
		انتخاب نوع متغیر برای خروجی عبارت منطقی و محاسبه نتیجه عبارت منطقی ترکیبی پیچیده و طولانی			
	میانگین		۲		غیر قابل قبول <input type="checkbox"/> قابل قبول <input type="checkbox"/> بالتر از حد انتظار <input type="checkbox"/>

ب) برای مطالعه بیشتر

هنرآموزان گرامی برای مطالعه بیشتر و کسب اطلاعات تکمیلی در خصوص این واحد یادگیری می‌توانند از منابع زیر استفاده کنند:

۱. <https://msdn.microsoft.com/en-us/library/ms173145.aspx>
۲. <https://msdn.microsoft.com/en-us/library/6a71f45d.aspx>
۳. <https://www.dotnetperls.com/operator>
۴. https://en.wikibooks.org/wiki/C_Sharp_Programming/Operators
۵. https://en.wikibooks.org/wiki/C_Sharp_Programming/Operators
۶. <http://www.blackwasp.co.uk/csharpbooleanoperators.aspx>
۷. <https://www.programtopia.net/csharp/docs/operator-types>
۸. <https://msdn.microsoft.com/en-us/library/sa7629ew.aspx>

واحد یادگیری ۱۰

شایستگی کار با ساختار شرطی

مقدمات تدریسی

الف) مفاهیم کلیدی

مفاهیم کلیدی			
دستور شرطی if	بلاک	دستور شرطی if - else	عملگر ternary (?:)
دستورات شرطی تودرتو	خطایابی دستور if	دستور شرطی switch	break
حالت default در دستور switch	خطایابی دستور switch		

ب) مراحل انجام کار واحد یادگیری

مراحل انجام 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

ج) بودجه بندی

زمان بندی پیشنهادی واحد یادگیری:

ردیف	اهداف توانمندسازی		زمان تدریس	
	تئوری	عملی		
۱	عملکرد دستور if و استفاده از آن	۲	۱	
۲	ساختار و عملکرد دستور شرطی if-else	۲	۲	
۳	عملگر?: در برنامه های شرطی	۱	۱	
۴	ساختار و عملکرد دستور شرطی if-else متداخل	۳	۲	
۵	تبدیل if متداخل به if ساده	-	۱	
۶	خطاهای دستور if	۲	۲	
۷	ساختار و عملکرد دستور شرطی switch	۲	۲	
۸	تبدیل دستور شرطی if به دستور switch	-	۲	
۹	خطایابی دستور switch	-	۲	
۱۰	تولید پروژه	-	۳	
جمع ساعات		۱۲	۱۸	

نمونه طرح درس روزانه:

طرح درس روزانه (هفتگی) پیشنهادی				
درس: دستورات شرطی			کلاس: دهم	
پیام جلسه (هدف کلی): ساختار و عملکرد دستور if-else را توضیح دهد و در برنامه به کار برد				
اهداف یادگیری		فعالیت‌ها		زمان فیزیکی
فعالیت	طبقه هدف: حیطه عاطفی / شناختی / روانی - حرکتی	کار هنرآموز	کار هنرجو	مدت (دقیقه)
ارزشیابی رفتار ورودی	سنجش میزان آشنایی هنرجو با مفهوم تصمیم‌گیری و شناخت دستور if ساده	طرح سؤال: اگر سیستم شما دارای کلمه عبور باشد و کلمه صحیح را وارد کنید چه اتفاقی می‌افتد؟ در صورت نادرست وارد کردن چه اتفاقی می‌افتد؟	مشارکت در پاسخ و تعامل	۳
ایجاد انگیزه	ایجاد توجه، تمرکز و دقت ذهن هنرجو	مثال‌هایی که برای دو حالت که در یک تصمیم‌گیری اتفاق می‌افتد نام ببرید؟	گوش دادن فعال و پاسخ و مشارکت	۲
ارائه مفاهیم کلیدی (توضیح هنرآموز)	ارائه مفاهیم کلیدی و ایجاد انگیزه	توضیح ساختار و عملکرد دستور if-else و نحوه به‌کارگیری آن در برنامه	مرور مطالب گفته‌شده در گروه و تمرین آن به صورت عملی	۲۰
فعالیت کارگاهی (تمرین هنرجویان)	تحقق اهداف توانمندسازی با انجام تمرینات کلاسی	هنرآموز توضیح کلی در مورد فعالیت کارگاهی به هنرجو داده و از آن‌ها می‌خواهد به‌صورت گروهی به حل آن‌ها بپردازند.	انجام تکالیف با خروجی صحیح.	۲۰
ارزیابی فعالیت‌ها (ارائه تمرین)	بررسی مهارت استفاده دستور if-else	هنرجویان در گروه‌های تقسیم‌شده به انجام برنامه می‌پردازند	انجام تکلیف با در نظر گرفتن ملاک‌های ارزیابی	۲۰

طرح درس روزانه (هفتگی) پیشنهادی				
درس: دستورات شرطی			کلاس: دهم	
پیام جلسه (هدف کلی): ساختار و عملکرد دستور if-else را توضیح دهد و در برنامه به کار برد				
اهداف یادگیری	فعالیت‌ها		زمان	فیزیکی
ارائه نکات تکمیلی (جمع‌بندی)	تکمیل مفاهیم کلیدی	مفاهیم کلیدی را به‌طور خلاصه توضیح داده و مرور کند به همراه نکات تکمیلی و پاسخ به سؤالات هنرجویان	توجه و یادداشت از نکات درس	۱۰
ارزشیابی شایستگی (ارزشیابی پایانی)	آگاهی از میزان درک و تفهیم کاربرد مبحث if-else هنرجو	آزمونی عملی دو نمره‌ای از مطالب گفته‌شده، گرفته شود.	پاسخ به سؤال هنرآموز	۱۰
تمرین در منزل (تعیین)	افزایش روحیه پژوهش	برنامه ۳ ضلع مثلث و توضیحی اجمالی از روند کار	گوش دادن به نکات و یادداشت‌برداری	۵
ابزارهای موردنیاز				ویدئو پروژکتور، رایانه مجهز به برنامه IDE، تخته آموزشی، برنامه Netsupport

د) فیلم‌ها و پرونده‌های پیوست

شماره فیلم	هدف
102h51	دستور شرطی ساده، ساختار شرطی if، ساختار شرطی if-else، بلاک در ساختار شرطی، عبارت شرطی تودرتو
102h52	عملگر سه‌تایی، مرتب‌سازی، دستور switch، ایجاد منو در کنسول

نکته: برای انجام فعالیت‌های تکمیلی می‌توانید از تمرینات موجود در «پرونده پیوست فعالیت‌های تکمیلی» استفاده کنید.

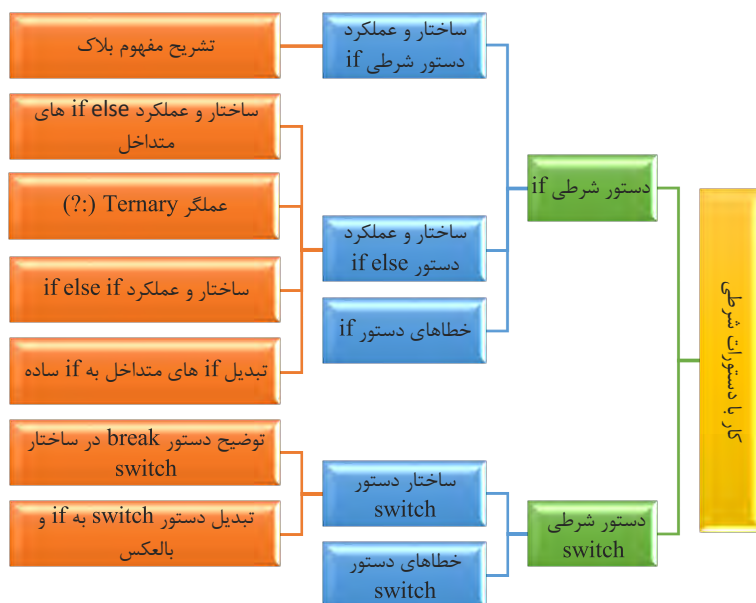
ه) ورود به بحث

هدف استفاده از دستورات شرطی کنترل منطق برنامه است، در صورت عدم وجود دستورات شرطی در برنامه دستورات نوشته‌شده یکی پس از دیگری به ترتیب اجرا می‌شوند تا برنامه خاتمه یابد، اما با استفاده از دستورات و عبارات شرطی برنامه‌نویس می‌تواند بر اساس داده‌های برنامه، تصمیمات مختلفی را گرفته و روند اجرای برنامه را بر اساس داده‌ها و ورودی‌های برنامه به نحو دلخواه مدیریت و پیاده‌سازی کند.

نتایج حاصل از یادگیری این فصل عبارت‌اند از:

کنترل داده‌های ثابت	
کنترل داده‌های متغیر	
کنترل ورودی‌ها	
مقایسه مقادیر موجود در برنامه	
شبیه‌سازی تصمیم‌گیری در محیط واقعی و انجام عملیات مناسب	

نمودار زیر نقشه مفهومی و ذهنی این واحد یادگیری را نمایش می‌دهد و شما می‌توانید با داشتن یک دورنما از مطالب و محتوای آموزشی، تدریس خود را به شکل بهتری مدیریت کنید.



کارگاه ۱ – دستور شرطی ساده

سؤال
ص ۲۴۴

پروژه‌ای با نام EPayment ایجاد کنید. برنامه را اجرا و نتایج آن را مشاهده کنید.
پاسخ: تصویر زیر خروجی برنامه در صورت وارد کردن عدد ۱ را نمایش می‌دهد.

```
C:\Windows\system32\cmd.exe
Enter your choice // 1 or 2
1
Pay bills
```

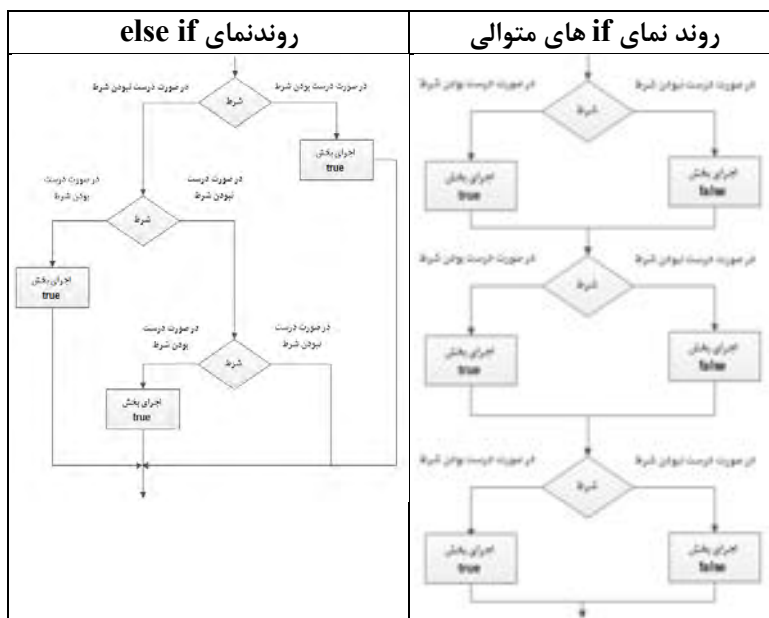
تصویر زیر خروجی برنامه در صورت وارد کردن عدد ۲ را نمایش می‌دهد.

```
C:\Windows\system32\cmd.exe
Enter your choice // 1 or 2
2
Pay bills
reduction of traffic
```

شرح: در صورتی که کاربر عدد ۱ را وارد کند به معنای پرداخت الکترونیک است و تنها عبارت Pay bills نمایش داده می‌شود اما در صورتی که عدد ۲ وارد شود پرداخت قبوض به صورت حضوری است و در اینجا عبارت Pay bills و reduction of traffic نمایش داده می‌شود؛ بنابراین این برنامه دارای دو شرط است که هر دو شرط بررسی می‌شود و هر کدام از شرط‌ها که با داده‌های وارد شده تطابق داشته باشد دستورات بدنه آن اجرا می‌شود.

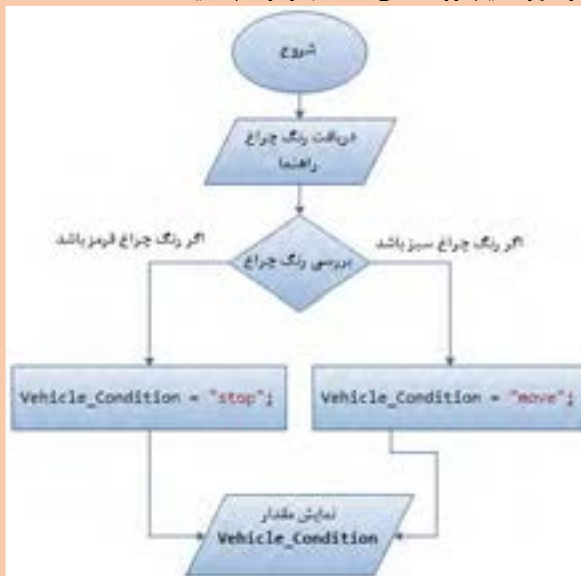
بعد از پرانتز if دوم یک؛ قرار داده، نتیجه را با نتیجه مرحله قبل مقایسه کنید. پاسخ: در این حالت به ازای ورود هر عدد خروجی Pay bills و reduction of traffic چاپ می‌شود، زیرا قرار دادن کاراکتر؛ بعد از دستور if به معنای تمام شدن شرط است و دستورات بعد از آن در کنترل دستور if نیستند.

کارگاه ۲- ساختار شرطی if else



سؤال
ص ۲۴۶

می‌خواهیم قسمتی از برنامه هدایت یک خودروی بدون سرنشین را برای عبور از چراغ‌قرمز طراحی کنیم. روندنمای مناسب را رسم کنید.



- برنامه را به‌صورت زیر تغییر دهید. اگر چراغ‌قرمز بود، ابتدا ماشین ترمز (break) کند، سپس متوقف شود؛ در غیر این صورت با زدن یک بوق حرکت کند.

```

string light, Vehicle_Condition;
light = Console.ReadLine();
if (light == "red")
{
    Vehicle_Condition = "break";
    Vehicle_Condition = "stop";
}
else
{
    Console.Beep(1000, 500);
    Vehicle_Condition = "move";
}
Console.WriteLine(Vehicle_Condition);
    
```

برنامه‌ای بنویسید که میزان سوخت موجود در یک خودرو را از ورودی خوانده، اگر کمتر از ۵ لیتر بود علامت قرمز رنگ و در غیر این صورت علامت سبز نمایش دهد.

پاسخ:

```
int amount;
Console.WriteLine("Please enter car amount fuel: ");
amount = int.Parse(Console.ReadLine());
if (amount < 5)
{
    Console.ForegroundColor = ConsoleColor.Red;
    Console.WriteLine("*");
}
else
{
    Console.ForegroundColor = ConsoleColor.Green;
    Console.WriteLine("*");
}
```

کارگاه ۳ – بلاک در ساختار شرطی if

پروژه جدیدی به نام week ایجاد و کد زیر را در متد main بنویسید. آیا برنامه خطا دارد؟

سؤال

ص ۲۴۷

پاسخ: برنامه خطای زمان اجرا یا کامپایلری ندارد اما خطای منطقی دارد زیرا بدنه شرط Even دارای بلاک نیست و برنامه‌های مربوط به روزهای زوج به جز studying در صورتی که کاربر Odd وارد کرده باشد نشان داده می‌شود.

– خروجی برنامه در صورتی که ورودی " Even " باشد چیست؟

```
studying
gym
say praying
c# programming Class
```

– خروجی برنامه در صورتی که ورودی " odd " باشد چیست؟

```
Studying
English class
Say praying
Visit Grand
```

gym
Say Praying
c# programming Class

- دلیل اشتراک پاسخ در موارد ۴ و ۵ چیست؟
پاسخ: شرط قسمت Even دارای بلاک نیست و برنامه‌های روزهای زوج به جز studying برای روزهای فرد نیز نمایش داده می‌شود.

- اگر خروجی برنامه نادرست است با توجه به آنچه قبلاً آموختید نوع خطا را مشخص کنید.
پاسخ: در صورت وارد کردن مقادیر غیر از Odd و Even خروجی مربوط به قسمت Even به جز studying نشان داده می‌شود و از نوع خطای منطقی است.

- مشکل برنامه را برطرف کنید.

```
static void Main(string[] args)
{
    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 ");
    }
}
```

- آیا می‌توان برنامه را با ساختار if-else نوشت؟
پاسخ: خیر نمی‌شود، زیرا در صورتی که کاربر مقداری غیر از Odd وارد کند دستورات قسمت Even اجرا می‌شود.

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

پاسخ: برای به دست آوردن تعداد افرادی که در صف انتظار هستند باید شماره آخرین نفر ورودی را از شماره آخرین نفر در حال انجام کار کسر کرد و با در نظر گرفتن اینکه به ازای هر بانه فعال نیز یک فرد در حال انجام کار هست و با در نظر گرفتن متغیرها به صورت زیر، تعداد افراد در حال انجام کار و صف انتظار از فرمول $a = (x - y) + z$ به دست می‌آید.

با در نظر گرفتن اینکه به تعداد بانه‌ها در آن واحد افراد در حال انجام کار هستند، $b = a/z$ و چون میانگین مدت‌زمان کار هر فرد در بانه معادل ۲ در نظر گرفته شده پس $c = b * 2$ مقدار c مدت‌زمان تقریبی انتظار در صف بانک است.

متغیر x : شماره آخرین نفر ورودی متغیر y : شماره آخرین نفر در حال انجام کار
متغیر z : تعداد بانه‌های فعال میانگین زمان انجام کار: ۲ دقیقه

```
static void Main(string[] args)
{
    int x, y, z;
    Console.WriteLine("Please enter x: ");
    x = int.Parse(Console.ReadLine());
    Console.WriteLine("Please enter y: ");
    y = int.Parse(Console.ReadLine());
    Console.WriteLine("Please enter z: ");
    z = int.Parse(Console.ReadLine());
    if (z > 0)
    {
        int a, b, c;
        a = (x - y) + z;
        b = a/z;
        c = b*2;
        Console.WriteLine("Average time is {0} minute.", c);
    }
    Else
    {
        Console.WriteLine("z value must be greater than zero.");
    }
}
```

کارگاه ۴ – اجرای گام به گام if – else

سؤال
ص ۲۴۸

پروژه جدیدی با نام ChElevator ایجاد و کد زیر را وارد کنید. اگر مراجعه کننده به طور سهوی عدد ۳۴ را وارد کند خروجی برنامه چیست؟
پاسخ: خروجی برنامه Elevator A است.
– برنامه را طوری تغییر دهید در صورتی که کاربر عددی خارج از محدوده تعداد طبقات وارد کرد پیام مناسبی چاپ شود.

```
Console.WriteLine("Enter the floor");
int floor = int.Parse(Console.ReadLine());
if (floor <= 24 && floor >= 1)
{
    if (floor%2 == 0)
        Console.WriteLine("Elevator A");
    else
        Console.WriteLine("Elevator B");
}
else
{
    Console.WriteLine("Floor number out of range");
}
```

خروجی برنامه را با مقادیر زیر Trace کنید.

floor	خروجی
10	Elevator A
25	Elevator B
-6	Floor number out of range
2.5	Input string was not in a correct format
"floor 13"	Input string was not in a correct format

کارگاه ۵ – عملگر سه تایی

سؤال

ص ۲۴۸

می‌خواهیم برنامه‌ای بنویسیم با دریافت سن فرد تعیین کند آیا صلاحیت دریافت گواهی‌نامه رانندگی دارد؟ کد زیر را تکمیل و در Main وارد کنید.

```
byte age;  
string status;  
Console.WriteLine("How old are you? ");  
age = byte.Parse(Console.ReadLine());  
if (age >= 18)  
    status = "Allowed";  
else  
    status = "Not Allowed";  
Console.WriteLine(status);
```

- خروجی برنامه را با مقادیر زیر Trace کنید.

age	خروجی
۲۰	Allowed
۱۸	Allowed
۱۰	Not Allowed

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

پاسخ: متغیر age از نوع byte - متغیر status از نوع string

- به جای کد قبلی، کد زیر را وارد کنید. خروجی برنامه را در دو حالت مقایسه کنید.

age	خروجی
۲۰	Allowed
۱۸	Allowed
۱۰	Not Allowed

۱- دو مورد از کارگاه‌های قبل که قابلیت استفاده از عملگر سه‌تایی را دارند، بازنویسی کنید.

۲- دستوراتی اضافه کنید که پیام‌ها با رنگ‌های متفاوت نمایش داده شوند.

پاسخ ۱: پیاده‌سازی کارگاه ۳ با استفاده از عملگر ۳ تایی

```
string day = Console.ReadLine();
string schedule = day == "odd"
? "Studying - English class - Say praing - Visit Grand"
: day == "Even"
? "Studying - gym - Say praing - c# programming Class"
: "String is not correct";
Console.WriteLine(schedule);
```

پاسخ ۲: در صورتی که سن بیشتر یا مساوی ۱۸ باشد پیغام بارنگ سبز نشان داده می‌شود در غیر این صورت بارنگ قرمز.

```
byte age;
string status;
Console.Write("How old are you?");
age = byte.Parse(Console.ReadLine());
if (age >= 18)
{
    Console.ForegroundColor = ConsoleColor.Green;
    status = "Allowed";
}
else
{
    Console.ForegroundColor = ConsoleColor.Red;
    status = "Not Allowed";
}
Console.WriteLine(status);
```

کارگاه ۶- عبارت شرطی تودرتو

پروژه جدیدی با نام UserPass ایجاد و کد زیر را وارد کنید. با توجه به برنامه جدول زیر را تکمیل کنید.

شرط

دستور

اگر نام کاربری و گذرواژه صحیح است؛	<pre> if (user == "reza") if (pass == "123456") Console.WriteLine("Welcome"+userna me); </pre>
اگر نام کاربری صحیح است و گذرواژه صحیح نیست.	<pre> Else Console.WriteLine("The password is incorrect"); </pre>
اگر نام کاربری صحیح نیست؛	<pre> Else Console.WriteLine("Unkown user"); </pre>

چرا هیچ کدام از ifها بلاک {} ندارند؟
پاسخ: دستورات بدنه هر شرط بیشتر از یکی نیست و می توان از بلاک استفاده نکرد.
- هر کدام از elseها مربوط به کدام if است؟ به کمک هنرآموز خود روش کلی برای این مسئله ارائه دهید.
پاسخ: بیرونی ترین else مربوط به (user == "reza") if و داخلی ترین else مربوط به (pass == "123456") if

کنجکاوی
ص ۲۵۱

کارگاه ۷- عبارات شرطی با عملگرهای منطقی

پروژه جدیدی بانام Weather ایجاد کنید. متغیرهای موردنیاز برنامه را اعلان کنید.

```

int temp
string Condition

```

سؤال
ص ۲۵۳

کد ردیف ۲ را با استفاده از عملگرهای منطقی بنویسید

```

else if ((temp > 19) && (Condition == "cloudy"))
Console.WriteLine("Good");

```

- به کمک هنرآموز خود کد این مرحله را با استفاده از عملگرهای منطقی بازنویسی کنید سپس نتیجه را با کد قبلی مقایسه کنید.

```

if (temp < 19 && (Condition == "sunny" || Condition == "partly
cloudy"))
Console.WriteLine("suitable");

```

کد ردیف ۴ را با عملگرهای منطقی به برنامه اضافه کنید.

```

if (temp < 19 && Condition == "rainy")
Console.WriteLine("impossible");

```


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

- دستوراتی اضافه کنید که پیام‌ها با رنگ‌های متفاوت نمایش داده شوند.

```
static void Main(string[] args)
{
    Console.ForegroundColor = ConsoleColor.Green;
    Console.WriteLine("Enter the air temperature");
    Console.ForegroundColor = ConsoleColor.Yellow;
    int temp = int.Parse(Console.ReadLine());
    Console.ForegroundColor = ConsoleColor.Green;
    Console.WriteLine("Enter the weather conditions:
    Sunny /Cloudy / partly cloudy/ rainy");
    Console.ForegroundColor = ConsoleColor.Yellow;
    string Condition = Console.ReadLine();
    if ((temp >= 8) && (temp < 19))
    {
        if (Condition == "sunny")
        {
            Console.ForegroundColor = ConsoleColor.Magenta;
            Console.WriteLine("suitable");
        }
        if (Condition == "partly cloudy")
        {
            Console.ForegroundColor = ConsoleColor.Blue;
            Console.WriteLine("suitable");
        }
    }
    else
    {
        Console.ForegroundColor = ConsoleColor.Red;
        Console.WriteLine("impossible ");
    }
    if (temp < 19 && temp >= 8 && Condition == "rainy")
    {
        Console.ForegroundColor = ConsoleColor.Red;
        Console.WriteLine("impossible");
    }
    Console.ForegroundColor = ConsoleColor.White;}
```

پروژه جدیدی با نام Sort ایجاد کنید. نوع مناسب برای متغیرهای a, b, c چیست؟
پاسخ: نوع مناسب برای متغیرها نوع عددی صحیح byte است.
- برنامه را طوری تغییر دهید تا صحت داده‌ها نیز بررسی شود.

```
byte a, b, c;
Console.WriteLine("Please enter a: ");
a = byte.Parse(Console.ReadLine());
Console.WriteLine("Please enter b: ");
b = byte.Parse(Console.ReadLine());
Console.WriteLine("Please enter c: ");
c = byte.Parse(Console.ReadLine());
if (a < 0 || a > 110)
    Console.WriteLine("a value not in range 0 to 110");
else if (b < 0 || b > 110)
    Console.WriteLine("b value not in range 0 to 110");
else if (c < 0 || c > 110)
    Console.WriteLine("c value not in range 0 to 110");
else
{
    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);
}
```

برنامه‌ای بنویسید که در انتخابات شورای دانش‌آموزی مدرسه، نفر برتر را از بین ۴ 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.

```
static void Main(string[] args)
{
    int vote1, vote2, vote3, vote4;
    int maxVote;
    Console.Write("Please enter vote 1: ");
    vote1 = int.Parse(Console.ReadLine());
    Console.Write("Please enter vote 2: ");
    vote2 = int.Parse(Console.ReadLine());
    Console.Write("Please enter vote 3: ");
    vote3 = int.Parse(Console.ReadLine());
    Console.Write("Please enter vote 4: ");
    vote4 = int.Parse(Console.ReadLine());
    maxVote = vote1;
    if (vote2 > maxVote)
        maxVote = vote2;
    if (vote3 > maxVote)
        maxVote = vote3;
    if (vote4 > maxVote)
        maxVote = vote4;
    Console.WriteLine("Max votes is {0}.", maxVote);
}
```

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.

```
static void Main(string[] args)
{
    int time1, time2, time3, time4;
    int firstRace;
    Console.Write("Please enter time1 (s): ");
    time1 = int.Parse(Console.ReadLine());
    Console.Write("Please enter time2 (s): ");
    time2 = int.Parse(Console.ReadLine());
    Console.Write("Please enter time3 (s): ");
```

```
time3 = int.Parse(Console.ReadLine());
Console.Write("Please enter time4 (s): ");
time4 = int.Parse(Console.ReadLine());
firstRace = time1;
if (time2 < firstRace)
firstRace = time2;
if (time3 < firstRace)
firstRace = time3;
if (time4 < firstRace)
firstRace = time4;
Console.WriteLine("first race time is {0} (s).", firstRace);
}
```

کارگاه ۹- ترکیب عبارات منطقی، محاسباتی و شرطی

پروژه جدیدی با نام Triangle ایجاد کنید. میتوانیم از شرط تودرتو استفاده کنیم. برنامه را اجرا کنید و نتایج را با کد قبلی مقایسه کنید.

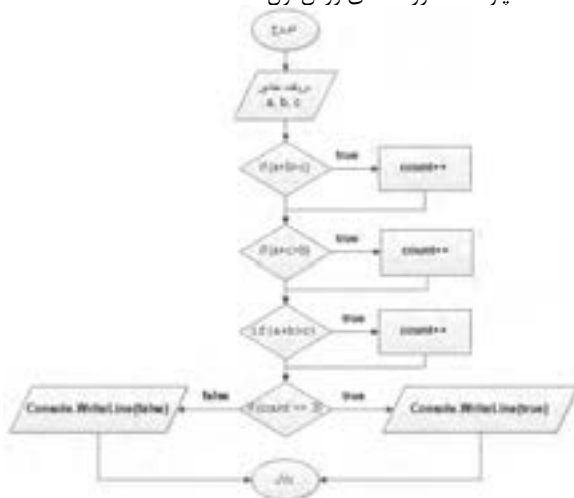
مقدار a	مقدار b	مقدار c	خروجی
۴	۵	۸	True
۵	۵	۵	True
۵	۹	۱	False

- از عملگرهای منطقی استفاده کنید. برنامه را اجرا کرده نتایج را دوباره با مراحل قبل مقایسه کنید.

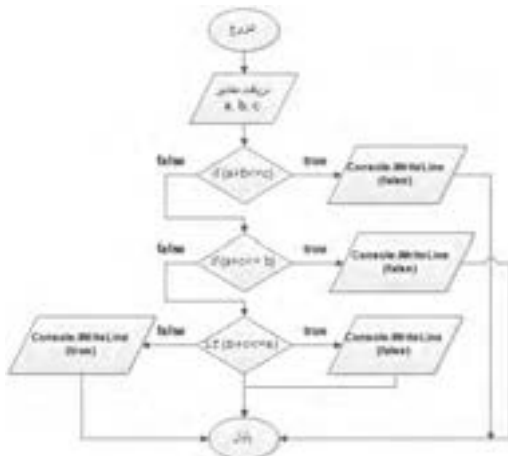
مقدار a	مقدار b	مقدار c	خروجی
۴	۵	۸	True
۵	۵	۵	True
۵	۹	۱	False

سؤال
ص ۲۵۷

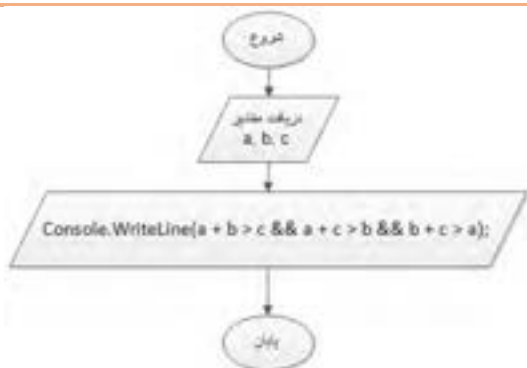
الگوریتم و روندنمای ۳ روش بالا را کشیده و باهم مقایسه کنید. به نظر شما کدام روش بهتر است چرا؟ روندنمای اول



روندنمای روش دوم



روندنمای روش سوم



روش سوم بهتر است زیرا کد برنامه در یک خط و به صورت خلاصه نوشته شده است.

- روش سوم را بازنویسی کنید و به جای `&&` از `||` استفاده کنید؛ و نتیجه آن‌ها را باهم مقایسه کنید.
پاسخ: پاسخ صحیح برگردانده نمی‌شود.

کارگاه ۱۰- تبدیل if به Switch

پروژه جدیدی با نام Season ایجاد کنید. برنامه زیر را جایگزین برنامه قبلی کرده، برخی دستورات ناقص را تکمیل کنید.

سؤال

ص ۲۵۹

```

string season = "bahar";
Console.Write("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":
    Console.WriteLine("Mehr, Aban, Azar");
    break;
case "Zemestan":

```

```
Console.WriteLine("Dey, Bahman, Esfand");
break;
default:
Console.WriteLine("Season Wrong");
break;
}
```

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

ورودی	خروجی
Bahar	Farvardin, Ordibehesht, Khordad
Tabestan	Tir, Mordad, Shahrivar
Paeiz	Mehr, Aban, Azar
Zemestan	Dey, Bahman, Esfand

کارگاه ۱۱ - ساختار Switch

سؤال

ص ۲۶۰

پروژه جدیدی با نام IdSudent ایجاد کنید. کد زیر را در متد Main وارد کنید. حالت‌های ۳ تا ۸ را مطابق جدول تکلیف، کدنویسی کنید.

```
static void Main(string[] args)
{
    long x = long.Parse(Console.ReadLine());
    x = x % 10;
    switch (x)
    {
        case 1:
            Console.WriteLine("5");
            break;
        case 2:
            Console.WriteLine("3");
            break;
        case 3:
            Console.WriteLine("11");
            break;
        case 4:
            Console.WriteLine("10");
```

```

break;
case 5:
    Console.WriteLine("4");
break;
case 6:
    Console.WriteLine("2");
break;
case 7:
    Console.WriteLine("8");
break;
case 9:
    Console.WriteLine("7");
break;
case 0:
    Console.WriteLine("9");
break;
}
}

```

- در برنامه بالا default استفاده نشده، آیا برنامه خطا دارد؟
 پاسخ: خیر، زیرا رقم آخر از سمت راست تمام اعداد از ۰ تا ۹ است و در دستور case
 برنامه تمام حالات در نظر گرفته شده است.
 - برنامه قبل را با ساختار if بازنویسی کنید و هر دو برنامه را باهم مقایسه کنید.

```

if (x == 1)
    Console.WriteLine("5");
else if (x == 2)
    Console.WriteLine("3");
else if (x == 3)
    Console.WriteLine("11");
else if (x == 4)
    Console.WriteLine("10");
else if (x == 5)
    Console.WriteLine("4");
else if (x == 6)
    Console.WriteLine("2");
else if (x == 7)
    Console.WriteLine("8");

```



```
else if (x == 9)
    Console.WriteLine("7");
else
    Console.WriteLine("9");
```

خروجی برنامه در هر دو حالت با استفاده از دستور switch و ifهای متداخل یکسان است.

- هنگام اجرا، به جای عدد، یک رشته وارد کنید. خطای حاصل را ترجمه کنید.
 پاسخ: در صورت وارد کردن رشته به جای عدد با خطای

Input string was not in a correct format

مواجه می‌شویم که به معنای "رشته وارد شده دارای فرمت صحیحی نیست"

فعالیت منزل
ص ۲۶۱

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

```
static void Main(string[] args)
{
    string fruit;
    Console.Write("Please enter fruit: ");
    fruit = Console.ReadLine();
    switch (fruit)
    {
        case "banana":
            Console.WriteLine(" anti-cancer, blood, weakness treatment ");
            break;
        case "Orange":
            Console.WriteLine("Blood purifier, anti-cancer ");
            break;
        case "Apricot":
            Console.WriteLine("Antipyretic, anti-rickets ");
            break;
        case "Pomegranate":
            Console.WriteLine("Reduce cholesterol, help to gingival health");
            break;
        case "Apple":
            Console.WriteLine("Blood pressure, brain tonic, skin ");
            break;
```

```

case "Figs":
    Console.WriteLine("Weight loss, improve headache, stomach pain");
break;
default:
    Console.WriteLine("Not found fruit in this list.");
break;
}
}

```

کارگاه ۱۲ – ایجاد فهرست در کنسول

پروژه جدیدی با نام MColor ایجاد کنید. برنامه زیر را در متد Main وارد و بخش‌های ناقص برنامه را تکمیل کنید.

```

static void Main(string[] args)
{
    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();

```

سؤال

ص ۲۶۲

```
break;
default:
Console.BackgroundColor = ConsoleColor.Black;
Console.Clear();
break;
}
}
```

- کد را طوری تغییر دهید که فهرست رنگ مانند شکل ۶ دیده شود.

```
static void Main(string[] args)
{
Console.WriteLine("What's your favorite color?");
Console.BackgroundColor = ConsoleColor.Black;
Console.Write(" r ");
Console.BackgroundColor = ConsoleColor.Red;
Console.ForegroundColor = ConsoleColor.Black;
Console.WriteLine(" The screen is red ");
Console.BackgroundColor = ConsoleColor.Black;
Console.ForegroundColor = ConsoleColor.White;
Console.Write(" g ");
Console.BackgroundColor = ConsoleColor.Green;
Console.ForegroundColor = ConsoleColor.Black;
Console.WriteLine(" The screen is green ");
Console.BackgroundColor = ConsoleColor.Black;
Console.ForegroundColor = ConsoleColor.White;
Console.Write(" b ");
Console.BackgroundColor = ConsoleColor.Blue;
Console.ForegroundColor = ConsoleColor.Black;
Console.WriteLine(" The screen is blue ");
Console.BackgroundColor = ConsoleColor.Black;
Console.ForegroundColor = ConsoleColor.White;
Console.Write(" w ");
Console.BackgroundColor = ConsoleColor.White;
Console.ForegroundColor = ConsoleColor.Black;
Console.WriteLine(" The screen is white ");
Console.BackgroundColor = ConsoleColor.Black;
Console.ForegroundColor = ConsoleColor.White;
```

```

Console.Write(" y ");
Console.BackgroundColor = ConsoleColor.Yellow;
Console.ForegroundColor = ConsoleColor.Black;
Console.WriteLine(" The screen is yellow ");
Console.ForegroundColor = ConsoleColor.White;
Console.BackgroundColor = ConsoleColor.Black;
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.Black;
Console.Clear();
break;
}
}

```

- یک ورودی خارج از فهرست وارد کنید. صفحه‌نمایش چه رنگی می‌شود؟
 پاسخ: در صورتی که ورودی خارج از فهرست باشد دستورات قسمت default اجرا شده و رنگ زمینه به رنگ مشکی و متن به رنگ سفید نشان داده می‌شود.

کارگاه ۱۳ – جزئیات دستور Switch و شرط منطقی ||

سؤال
ص ۲۶۳

پروژه جدیدی با نام IoDevice ایجاد کنید. برنامه زیر را در متد Main وارد کنید. قابلیت شناسایی دستگاه‌های بیشتری برای برنامه فراهم کنید.

```
static void Main(string[] args)
{
    Console.WriteLine("enter your device");
    string x = Console.ReadLine();
    switch (x)
    {
        case "keyborad":
        case "Mouse":
        case "Scanner":
        case "camera":
            Console.WriteLine("{0} is input device ", x);
            break;
        case "Monitor":
        case "Printer":
        case "Headphone":
        case "VideoProjection":
            Console.WriteLine("{0} is output device ", x);
            break;
        case "Modem":
        case "Network card":
        case "Audio Card":
        case "USB":
            Console.WriteLine("{0} is Both Input–OutPut Devices ", x);
            break;
        default:
            Console.WriteLine("unknown");
            break;
    }
}
```

الف) نمونه ارزشیابی پایانی

مکان : کارگاه رایانه مطابق استاندارد تجهیزات هنرستان‌ها
تجهیزات : رایانه‌ای که نرم‌افزار IDE برنامه‌نویسی روی آن نصب است- نوشت افزار - کاغذ
زمان: ۵۵ دقیقه (به‌کارگیری دستور if در برنامه ۱۰ دقیقه - به‌کارگیری دستور if-else در برنامه ۱۵ دقیقه - به‌کارگیری دستور if متداخل در برنامه ۱۵ دقیقه - به‌کارگیری دستور switch در برنامه ۱۵ دقیقه)

می‌خواهیم برنامه‌ای برای نوشتن تقویم سال جاری بنویسیم. باید تعیین کنیم که هر تاریخی در این سال چندشنبه است. با فرض این که روز اول سال دوشنبه است. برنامه‌ای بنویسید که عدد صحیح و مثبتی را به‌عنوان شماره یکی از روزهای سال خوانده و مشخص کند که عدد واردشده چه روزی از هفته است. (عدد ورودی باید در محدوده تعداد روزهای سال باشد - برنامه یک بار با if و بار دیگر با switch نوشته شود)

مرحله	شاخص	حداقل نمره قبولی	نمره	قبول رد
۱	- ارائه راه حل مساله	۲		
	- تشخیص وجود شرط در راه حل			
	- رسم روندنمای راه حل - انتخاب دستور شرطی - نوشتن برنامه C# مربوط به روندنما - تشخیص خطاهای if و رفع خطاهای if			
۲	- ارائه راه حل مساله	۲		
	- تشخیص وجود شرط ساده یا پیچیده در راه حل			
	- رسم روندنمای راه حل - انتخاب دستور شرطی - پیاده سازی برنامه C# روندنما - تشخیص بلاک دستورات بدنه if - تشخیص خطاهای if-else - رفع خطاهای if-else			
۳	- ارائه راه حل برنامه	۲		
	- تشخیص وجود شرط متداخل در راه حل برنامه - رسم روندنمای راه حل - انتخاب دستورات شرطی			

			<ul style="list-style-type: none"> - پیاده سازی برنامه C# روندنما - خطاهای احتمالی if متداخل - رفع خطاهای احتمالی if متداخل 	
		۲	<ul style="list-style-type: none"> - ارائه راه حل مساله - تشخیص وجود شرط پیچیده در راه حل 	۴
			<ul style="list-style-type: none"> - انتخاب دستور شرطی switch - نوشتن برنامه با دستور شرطی switch 	
			<ul style="list-style-type: none"> - تشخیص خطاهای به کارگیری دستور switch - رفع خطاهای به کارگیری دستور switch 	
			میانگین	

(ب) برای مطالعه بیشتر

۱. <https://msdn.microsoft.com/en-us/library/5011f09h.aspx>
۲. <https://msdn.microsoft.com/en-us/library/ty67wk28.aspx>
۳. <https://msdn.microsoft.com/en-us/library/06tc147t.aspx>
۴. <https://www.dotnetperls.com/if>
۵. <https://www.dotnetperls.com/ternary>
۶. <https://www.dotnetperls.com/switch>
۷. <https://www.dotnetperls.com/case>

نمونه های تکمیل شده از جدول ارزشیابی مبتنی بر شایستگی واحد یادگیری ۱:

نمونه ۱:

نمرات یک هنرجو برای این واحد یادگیری در جدول زیر آورده شده است. نمرات کسب شده در مراحل، بالاتر از حداقل قبولی مراحل و میانگین مراحل یادگیری ۲ (مساوی و یا بزرگتر از ۲) است. بنابراین نمره شایستگی همان نمره میانگین مراحل است. نمره این واحد یادگیری با فرض اینکه نمره مستمر هنرجو ۵ (از ۵ نمره) باشد به روش زیر محاسبه می شود. مراحل بحرانی با رنگ متمایز شده است.

$$۱۵ = ۵ + ۲ * ۵ = \text{نمره واحد یادگیری}$$

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	آماده سازی سند گرافیکی	۲	۲
۲	تصویرسازی با استفاده از اشکال هندسی	۲	۳
۳	تصویرسازی با استفاده از ابزارهای ترسیمی	۱	۳
۴	استفاده از متن	۱	۲
۵	استفاده از ماسک	۱	۲
	شایستگی های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش: ۱- کنترل کیفیت، نقش در گروه، نوآوری در طراحی و زبان فنی ۲- رعایت ارگونومی و کنترل حفاظتی ابزار و تجهیزات ۳- ایجاد اسناد الکترونیکی کاهش مصرف کاغذ ۴- به کارگیری خلاقیت در طراحی	۲	۲
	میانگین مراحل	۲	۲
	نمره شایستگی از ۳		۲
	نمره مستمر از ۵		۵
	نمره واحد یادگیری از ۲۰		۱۵

نمونه ۲:

نمرات یک هنرجو برای این واحد یادگیری در جدول زیر آورده شده است. نمره کسب شده در مرحله ۲ کمتر از نمره حداقل قبولی مرحله است و احراز شایستگی در این مرحله کاری دارای اهمیت است. بنابراین نمره شایستگی ۱ است نمره این واحد یادگیری با فرض اینکه نمره مستمر هنرجو ۳ (از ۵ نمره) باشد به روش زیر محاسبه می شود.

$$۸ = ۳ + ۵ \times ۱ = \text{نمره واحد یادگیری}$$

هنرجو در این واحد یادگیری شایستگی لازم را کسب نکرده و نمره قبولی (حداقل ۱۲) را بدست نیاورده است.

ردیف	مرحله کار	حداقل نمره قبولی از ۳	نمره هنرجو
۱	آماده سازی سند گرافیکی	۲	۲
۲	تصویر سازی با استفاده از اشکال هندسی	۲	۱
۳	تصویر سازی با استفاده از ابزارهای ترسیمی	۱	۳
۴	استفاده از متن	۱	۲
۵	استفاده از ماسک	۱	۲
	شایستگی های غیر فنی، ایمنی، بهداشت، توجهات زیست محیطی و نگرش: ۱- کنترل کیفیت، نقش در گروه، نوآوری در طراحی و زبان فنی ۲- رعایت ارگونومی و کنترل حفاظتی ابزار و تجهیزات ۳- ایجاد اسناد الکترونیکی کاهش مصرف کاغذ ۴- به کارگیری خلاقیت در طراحی	۲	۲
	میانگین مراحل	۲	۲
	نمره شایستگی از ۳		۱
	نمره مستمر از ۵		۳
	نمره واحد یادگیری از ۲۰		۸

نکته: در جدول ارزشیابی پایانی، برخی از مراحل کار بحرانی محسوب می شوند. در واقع از آنجا که حداقل نمره قبولی در اینگونه مراحل معادل ۲ است، لازم است هنرجو برای کسب شایستگی، نمره حد نصاب در این مراحل را کسب کند.

- ۱- شورای برنامه ریزی درسی گروه برق و رایانه (۱۳۹۴)، راهنمای برنامه درسی رشته شبکه و نرم افزار رایانه. سازمان پژوهش و برنامه ریزی آموزشی. دفتر تالیف کتاب های درسی فنی و حرفه ای و کاردانش.
- ۲- شورای برنامه ریزی درسی رشته شبکه و نرم افزار رایانه (۱۳۹۳)، استاندارد ارزشیابی حرفه فناوری اطلاعات و ارتباطات. سازمان پژوهش و برنامه ریزی آموزشی. دفتر تالیف کتاب های درسی فنی و حرفه ای و کاردانش.
- ۳- شورای برنامه ریزی درسی رشته شبکه و نرم افزار رایانه (۱۳۹۲)، استاندارد شایستگی حرفه فناوری اطلاعات و ارتباطات. سازمان پژوهش و برنامه ریزی آموزشی. دفتر تالیف کتاب های درسی فنی و حرفه ای و کاردانش.
- ۴- سند تحول بنیادین آموزش و پرورش، مصوب شورای عالی انقلاب فرهنگی، ۱۳۸۹.
- ۵- برنامه درسی ملی ایران، مصوب شورای عالی آموزش و پرورش، ۱۳۹۰.
- ۶- سند طراحی مفهومی آموزش های فنی و حرفه ای، دفتر تالیف کتاب های درسی فنی و حرفه ای و کاردانش، ۱۳۹۴.
- ۷- ناگل، کریستین. مرجع کامل C#.Net 2005. ترجمه : مهندس مهرداد توانا، مهندس سعید هراتیان (۱۳۸۶). انتشارات گروه مهندسی پژوهشی ساحر
- ۸- <https://msdn.microsoft.com/en-us/library/>
- ۹- <https://www.dotnetperls.com/>
- ۱۰- <https://www.tutorialspoint.com/csharp/>
- ۱۱- <https://helpx.adobe.com/support/>
- ۱۲- <https://www.lynda.com/Captivate-training-tutorials/>
- ۱۳- <http://elearningbrothers.com/adobe-captivate-resources/>
- ۱۴- <https://www.amazon.com/Adobe-Captivate-9-Essentials-Workbook>
- ۱۵- <http://kb.photodex.com/219/what-file-types-does-proshow-gold-support/>
- ۱۶- <https://www.dotnetperls.com>
- ۱۷- <https://www.fileinfo.com>
- ۱۸- <https://www.dotwhat.net>
- ۱۹- <http://www.lingoes.net>