با ایران توشه
یک قدم جلوتر
باش!

ایران توشه
توشه ای برای موفقیت



ایران توشه، مرجع دانلود فایل های کمک آموزشی دانش آموزان و کنکوری ها

همه چیز
برای موفقیت
تو!

- دانلود رایگان آزمون های آزمایشی قلمچی، خیلی سبز، گاج و ...
- دانلود رایگان گام به گام تمامی دروس
- دانلود رایگان نمونه سوالات امتحانی (نوبت اول و دوم) و امتحانات نهایی
- دانلود رایگان جزوه های کمک آموزشی و شب امتحانی

کافیست هر فایلی که نیاز دارید را در گوگل به این صورت جستجو کنید:

فایل مورد نیاز + ایران توشه

مثال:

دانلود گام به گام دوازدهم تجربی ایران توشه



irantooshe.ir

t.me/irantooshe

گام به گام توسعه برنامه‌سازی و پایگاه داده پایه یازدهم
پودمان دوم

عنوان فصل: مدیریت مجموعه داده

از صفحه‌ی ۵۷ تا صفحه‌ی ۹۸ کتاب

در این فایل پاسخ فعالیت‌ها، کارهای کلاسی، تمرین‌ها و پرسش‌های فصل مدیریت مجموعه داده آمده است.
مباحث فصل: ساختار تکرار (حلقه‌ی for و while، حلقه‌های متداخل)، آرایه و عملیات روی آن، جست‌وجوی خطی و دودویی، کلاس Array

تهیه‌شده در ایران‌توشه، توشه‌ای برای موفقیت

irantooshe.ir

فهرست پرسش‌ها

۱ صفحه ۵۹ · فعالیت گروهی — حلقه‌ی معین و نامعین ۸

۲ صفحه ۶۰ · فعالیت کارگاهی — تکمیل روندنمای ثبت فعالیت هنرجویان ۸

۳ صفحه ۶۱ · فعالیت کارگاهی — نوبت‌دهی مطب دندان‌پزشکی ۹

۴ صفحه ۶۲ · فعالیت کارگاهی — شمارش قبول‌شده‌ها ۹

۵ صفحه ۶۴ · کارگاه ۱ — تبدیل روندنما به برنامه ۱۰

۶ صفحه ۶۴ · کارگاه ۲ — بررسی شرایط حلقه‌ی تکرار معین ۱۰

۷ صفحه ۶۵ · کنجکاوی — نوع متغیر حلقه ۱۲

۸ صفحه ۶۵ · فعالیت گروهی — پیش‌بینی خروجی حلقه‌ها ۱۲

۹ صفحه ۶۵ · فعالیت منزل — گروه‌بندی هنرجویان ۱۲

ایران توشه
توشه ای برای موفقیت

۱۰ صفحه ۶۵ · کارگاه ۳ — کار با متغیر خارج از بلاک (بازی هپ) ۱۳
۱۱ صفحه ۶۶ · کارگاه ۴ — محاسبه‌ی مجموع زمان مطالعه ۱۳
۱۲ صفحه ۶۷ · کارگاه ۵ — خروج زودرس از حلقه (حدس ماه تولد) ۱۴
۱۳ صفحه ۶۸ · کارگاه ۵ — Trace و اصلاح برنامه‌ی حدس ماه ۱۵
۱۴ صفحه ۶۸ · فعالیت گروهی — پیش‌بینی خروجی حلقه‌های for ۱۶
۱۵ صفحه ۶۹ · فعالیت منزل — مسئله‌ی کاربردی برای حلقه‌ی گام ۳ ۱۷
۱۶ صفحه ۶۹ · فعالیت کارگاهی — اجرای گام‌به‌گام MonthGuess و پنجره‌ی Local ۱۷
۱۷ صفحه ۶۹ · فعالیت منزل — شمارش هنرجویان هم‌نام حضرت معصومه (س) ۱۸
۱۸ صفحه ۷۰ · فعالیت گروهی — مثال‌هایی از حلقه‌ی نامعین ۱۹
۱۹ صفحه ۷۰ · کارگاه ۶ — به‌کارگیری حلقه‌ی while (میانگین نمرات) ۱۹
۲۰ صفحه ۷۱ · فعالیت کارگاهی — Trace حلقه‌های while ۲۰

ایران توشه
توشه ای برای موفقیت

۲۱ صفحه ۷۲ · کارگاه ۷ — یافتن بزرگ‌ترین مقدار (روندنما) ۲۱

۲۲ صفحه ۷۳ · کارگاه ۷ — برنامه‌ی Max ۲۱

۲۳ صفحه ۷۳ · فعالیت کارگاهی — ماشین‌حساب کنسولی ۲۳

۲۴ صفحه ۷۴ · فعالیت گروهی — توصیف شکل حلقه‌های متداخل ۲۳

۲۵ صفحه ۷۵ · کارگاه ۸ — جدول‌ضرب ۵ در ۵ با حلقه‌ی متداخل ۲۴

۲۶ صفحه ۷۵ · فعالیت گروهی — پرداخت ۵۰ هزار تومان در خودپرداز ۲۵

۲۷ صفحه ۷۵ · کارگاه ۹ — توسعه‌ی نوبت‌دهی مطب دندان‌پزشکی ۲۵

۲۸ صفحه ۷۶ · فعالیت منزل — توسعه‌ی برنامه‌ی ATM ۲۶

۲۹ صفحه ۷۶ · فعالیت منزل — توسعه‌ی برنامه‌ی میانگین ۲۶

۳۰ صفحه ۷۹ · فعالیت کارگاهی — نمایش اسامی پنج هنرجو از آخر به اول ۲۷

۳۱ صفحه ۸۰ · فعالیت کارگاهی — اعلان آرایه و بررسی محتوای عناصر ۲۸

ایران توشه
توشه ای برای موفقیت

۳۲ صفحه ۸۱ · فعالیت کارگاهی — اعلان آرایه‌ها ۲۸
۳۳ صفحه ۸۱ · کنجکاوی — چرا اندیس ۲ برای خانه‌ی سوم ۲۹
۳۴ صفحه ۸۲ · کارگاه ۱ — تعریف و مقداردهی آرایه (Computer) ۲۹
۳۵ صفحه ۸۳ · کارگاه ۱ — روش‌های مقداردهی در هنگام اعلان ۳۰
۳۶ صفحه ۸۵ · کنجکاوی — نمایش عناصر آرایه ۳۰
۳۷ صفحه ۸۵ · کارگاه ۲ — نمایش نام روز هفته (Week) ۳۰
۳۸ صفحه ۸۶ · فعالیت کارگاهی — پیمایش آرایه‌ی week ۳۱
۳۹ صفحه ۸۷ · کارگاه ۳ — قیمت کل قطعات رایانه (PriceOfComputer) ۳۱
۴۰ صفحه ۸۸ · کارگاه ۴ — پیمایش نویسه‌های رشته (StringChar) ۳۲
۴۱ صفحه ۸۹ · کارگاه ۵ — ویرایش عناصر آرایه (ValueAddedTax) ۳۲
۴۲ صفحه ۸۹ · کنجکاوی — متغیر حلقه‌ی foreach ۳۳

ایران توشه
توشه ای برای موفقیت

۴۳ صفحه ۸۹ · فعالیت منزل — شمارش هنرجویان با سطح شایستگی ۱ ۳۴
۴۴ صفحه ۹۰ · فعالیت منزل — هزینه‌ی روزانه‌ی یک ماه ۳۴
۴۵ صفحه ۹۱ · فعالیت گروهی — جست‌وجوی خطی یا دودویی ۳۵
۴۶ صفحه ۹۱ · فعالیت گروهی — تعداد مقایسه در جست‌وجوی خطی (۱۰۰ هنرجو) ۳۵
۴۷ صفحه ۹۲ · کارگاه ۶ — پیاده‌سازی جست‌وجوی خطی (LinearSearch) ۳۵
۴۸ صفحه ۹۳ · فعالیت منزل — نمره‌ی ۲۰ هنرجویان ۳۷
۴۹ صفحه ۹۳ · فعالیت گروهی — بازی حدس عدد (جست‌وجوی دودویی) ۳۷
۵۰ صفحه ۹۳ · فعالیت منزل — تکمیل جدول جست‌وجوی دودویی با عدد ۱۲ ۳۸
۵۱ صفحه ۹۴ · کارگاه ۷ — متدهای کلاس Array (ArrayClass) ۳۸
۵۲ صفحه ۹۵ · فعالیت منزل — برنامه‌ی منودار عملیات آرایه ۳۹
۵۳ صفحه ۹۶ · پروژه فاز ۲ — برنامه‌ی کتابخانه‌ی هنرستان ۴۰

ایران توشه
توشه ای برای موفقیت

۱ صفحه ۵۹ فعالیت گروهی

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

نوع حلقه

مثال	نوع حلقه (معین یا نامعین)
یک نجار روزانه ۱۲ صندلی می‌سازد.	معین (تعداد تکرار ۱۲ است)
تا زمانی که دبیر ورزش سوت پایان را بزند همه‌ی هنرجویان چندین نرمش کششی انجام می‌دهند.	نامعین (زمان سوت معلوم نیست)
در یک نانوایی بربری، نانوا در هر تنور ۶۰ عدد نان می‌پزد.	معین (۶۰ بار)
یک نقاش ساختمان در یک روز تعدادی دیوار با ابعاد مختلف را رنگ می‌زند.	نامعین (تعداد دیوارها از پیش معلوم نیست)

۲ صفحه ۶۰ فعالیت کارگاهی

روندنمای بالا را با توجه به الگوریتم برنامه تکمیل کنید.

روندنما بر اساس مراحل الگوریتم به‌صورت زیر کامل می‌شود. بدنه‌ی حلقه (مراحل ۳ تا ۷) همان بلوک‌های «دریافت فعالیت»، «بررسی»، «ثبت گزارش» و «تحویل» به‌همراه افزایش i است.

```
Start
  |
  v
 i <- 1
  |
  v
<i <= 22 ?> --False--> End
  | True
  v
[Receive student i's activity]
[Check the activity]
[Record the report in the evaluation book]
[Return the activity to the student]
[i <- i + 1]
  |
  +--> back to the condition (i <= 22)
```

قسمت‌های پررنگ‌شونده‌ی حلقه: شرط i <= 22، چهار بلوک بدنه و بلوک i ← i + 1 همراه با پیکان بازگشت.

ایران توشه
توشه ای برای موفقیت

۳ صفحه ۶۱ فعالیت کارگاهی

ـ اگر تعداد بیماران ۳۰ نفر باشد، چه تغییری لازم است؟

ـ روندنما و الگوریتم را تغییر دهید تا تعداد بیماران را از ورودی دریافت کند.

با توجه به روندنما الگوریتم زیر را کامل کنید. (مراحل ۲ تا ۴)

تکمیل الگوریتم

1. اگر i <= 20 است برو به مرحله‌ی بعد، وگرنه برو به مرحله‌ی ۵.
2. مقدار i را نمایش بده (شماره‌ی نوبت).
3. به i یک واحد اضافه کن و به مرحله‌ی ۲ برو.

۳۰ بیمار

فقط شرط حلقه عوض می‌شود: به‌جای i <= 20 می‌نویسیم i <= 30 (در الگوریتم نیز در مرحله‌ی ۲).

دریافت تعداد بیماران از ورودی

یک متغیر n تعریف می‌کنیم و بلافاصله پس از شروع، مقدار آن را دریافت می‌کنیم؛ شرط حلقه به i <= n تغییر می‌کند.

1. شروع
2. مقدار n (تعداد بیماران) را دریافت کن.
3. مقدار i را برابر یک قرار بده.
4. اگر i <= n است برو به مرحله‌ی بعد، وگرنه برو به مرحله‌ی ۷.
5. i را نمایش بده.
6. به i یک واحد اضافه کن و به مرحله‌ی ۴ برو.
7. پایان

ایران توشه
توشه ای برای موفقیت

۴ صفحه ۶۲ فعالیت کارگاهی

مسئله: شرط قبولی در یک آزمون کسب حداقل نمره‌ی ۱۲ از ۲۰ است. می‌خواهیم نمرات هنرجویان یک کلاس ۲۰ نفره را دریافت کرده، تعداد قبول‌شده‌ها را مشخص کنیم.

الگوریتم را کامل کنید (مراحل ۳، ۴ و ۶).

1. شروع
2. در شمارنده (count) مقدار صفر قرار بده.
3. در i مقدار یک قرار بده.

4. اگر i <= 20 است برو به مرحله‌ی ۴ وگرنه برو به مرحله‌ی ۸. (مرحله‌ی ۳)
5. نمره‌ی هنرجوی nام را دریافت کن (mark). (مرحله‌ی ۴)
6. اگر نمره بزرگ‌تر یا مساوی ۱۲ بود، به شمارنده یکی اضافه کن.
7. به i یک واحد اضافه کن. (مرحله‌ی ۶)
8. برو به مرحله‌ی ۳.
9. شمارنده را نمایش بده.
10. پایان

در متن کتاب شماره‌ی مرحله‌ها هنگام استخراج جابه‌جا شده است؛ ترتیب بالا بر پایه‌ی روندنما نوشته شده است.

---

۵ صفحه ۶۴ کارگاه ۱

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

شماره‌های مورد نیاز از ۱۰۰ تا ۹۹۹ است.

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

مرحله‌ی ۳: شماره‌ها از ۱۰۰ تا ۹۹۹

```
for (int i = 100; i <= 999; i++)
    Console.WriteLine(i);
```

مرحله‌ی ۴: دریافت حداکثر شماره از ورودی

```
Console.Write("Max patient number: ");
int max = int.Parse(Console.ReadLine());
for (int i = 100; i <= max; i++)
    Console.Write(i + " ");
```

نمونه‌ی اجرا (ورودی ۱۱۰):

```
Max patient number: 110
100 101 102 103 104 105 106 107 108 109 110
```

ایران توشه
توشه ای برای موفقیت

---

۶ صفحه ۶۴ کارگاه ۲

۳ پس از پرانتز دستور for علامت ; قرار داده، خروجی را بررسی کنید. نتیجه را برای هم‌گروهی و هنرآموز خود توضیح دهید.

۴ قطعه کد مرحله‌ی ۲ را به‌صورت زیر تغییر داده، نتیجه را با مرحله‌ی قبل مقایسه کنید.

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

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

{

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

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

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

کد مرحله‌ی ۲ (شماره‌های زوج، مخصوص نظافت کلاس) این است: for (int i = 2; i <= 30; i += 2) Console.Write("{0,5}", i); و خروجی آن ۲، ۴، ...، ۳۰ است.

مرحله‌ی ۳: ; بعد از for

```
for (int i = 2; i <= 30; i += 2) ;
Console.Write("{0,5}", i);   // error CS0103: 'i' does not exist
```

; یک دستور خالی است و آن را به‌عنوان بدنه‌ی حلقه می‌گیرد؛ پس حلقه ۱۵ بار بدون هیچ کاری تکرار می‌شود و چاپ‌کردن بیرون از حلقه قرار می‌گیرد. چون i فقط درون for شناخته می‌شود، خطای کامپایل رخ می‌دهد. اگر i را پیش از حلقه تعریف کنیم، فقط یک بار و با مقدار نهایی (32) چاپ می‌شود.

مرحله‌ی ۴

```
    3    5    7    9   11   13   15   17   19   21   23   25   27   29   31
```

در هر دور i یک بار در سرصفحه‌ی for و یک بار در بدنه زیاد می‌شود (۲ واحد در هر دور)، پس بدنه روی اعداد فرد چاپ می‌کند. برای i=2 ابتدا i می‌شود ۳ و چاپ می‌شود؛ آخرین چاپ ۳۱ است (چون شرط فقط در ابتدای دور بررسی می‌شود).

مرحله‌ی ۵: حذف آکولادها

فقط ;i++ بدنه‌ی حلقه است و Console.Write بعد از حلقه می‌آید که i را نمی‌شناسد؛ برنامه خطای کامپایل CS0103 می‌دهد و چیزی چاپ نمی‌شود.

مرحله‌ی ۶: شماره‌های فرد (نظافت کارگاه)

```
for (int i = 1; i <= 30; i += 2)
    Console.Write("{0,5}", i);
//     1    3    5 ... 29
```

مرحله‌ی ۷: i >= 30

شرط از همان ابتدا (2 >= 30) نادرست است، پس بدنه‌ی حلقه هیچ‌گاه اجرا نمی‌شود و خروجی خالی است.

۷ صفحه ۶۵ کنجکاوی

آیا متغیر حلقه می‌تواند از نوع byte، float یا char باشد؟

بله؛ متغیر حلقه می‌تواند هر نوع عددی یا نویسه‌ای باشد که عملگرهای مقایسه و ++ یا =+ را بپذیرد. اجرای آزمایشی:

```
for (byte b = 250; b < 255; b++) Console.Write(b + " ");        // 250 251 252 253 254
for (float f = 0; f <= 1; f += 0.25f) Console.Write(f + " ");  // 0 0.25 0.5 0.75 1
for (char c = 'a'; c <= 'e'; c++) Console.Write(c + " ");       // a b c d e
```

دقت کنید در byte اگر شرط تا ۲۵۵ برود (مثلاً b <= 255) حلقه هرگز تمام نمی‌شود، چون b از ۲۵۵ به ۰ برمی‌گردد. در float هم به‌دلیل خطای گردکردن، شرط‌های برابری توصیه نمی‌شود.

۸ صفحه ۶۵ فعالیت گروهی

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

حلقه‌ها و خروجی آنها

حلقه	خروجی برنامه (پیش‌بینی و اجرا)
for (int i=5; i>=1; i--) Console.Write(i);	54321
for (int i=0; i<10; i+=3) Console.WriteLine(i);	0 3 6 9 (هر عدد در یک خط)
for (int i=10; i>=0; i-=2) Console.WriteLine(i);	10 8 6 4 2 0 (هر عدد در یک خط)

ایران توشه
توشه ای برای موفقیت

۹ صفحه ۶۵ فعالیت منزل

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

```
for (int i = 1; i <= 5; i++)
    Console.WriteLine("Group {0}: student {1} and student {2}", i, i, 11 - i);
```

```
Group 1: student 1 and student 10
Group 2: student 2 and student 9
Group 3: student 3 and student 8
Group 4: student 4 and student 7
Group 5: student 5 and student 6
```

۱۰ صفحه ۶۵ کارگاه ۳

برنامه را اجرا کنید. عملکرد برنامه چیست؟

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

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

۴ برنامه را برای مضارب ۳ بازنویسی کنید.

عملکرد برنامه

اعداد ۱ تا ۱۰۰ را چاپ می‌کند و به‌جای هر مضرب ۵ (۵، ۱۰، ۱۵، ...) کلمه‌ی Hop می‌نویسد: 1 2 3 4 Hop 6 7 8 9 Hop ...

دلیل خطا و راه‌حل

خطای CS0103: The name 'i' does not exist in the current context؛ چون i داخل پرانتز for تعریف شده و دامنه‌ی آن (بلاک) فقط خود حلقه است و بعد از پایان حلقه از بین می‌رود. راه‌حل: تعریف i پیش از حلقه.

```
int i;
for (i = 1; i <= 100; i++)
{
    if (i % 5 == 0) Console.WriteLine("Hop");
    else Console.WriteLine(i);
}
Console.Write("Final i :" + i);   // Final i :101
```

مرحله‌ی ۴: مضارب ۳

```
int i;
for (i = 1; i <= 100; i++)
    if (i % 3 == 0) Console.WriteLine("Hop");
    else Console.WriteLine(i);
```

۱۱ صفحه ۶۶ کارگاه ۴

۱ روندنمای مسئله را کامل کنید.

۳ قطعه کد زیر را در متد Main وارد کرده، برنامه را کامل کنید.

ایران توشه
توشه‌ای برای موفقیت

```
.........................
for (int i = 1; i < 8; i++)
{
    Console.WriteLine("Please rate the day {0} enter in minutes=", i);
    minute = int.Parse(Console.ReadLine());
    sum = sum + ........................;
}
Console.WriteLine("you studied for {0} minutes a week", sum);
```

۴ برنامه را طوری تغییر دهید که مجموع ساعات مطالعه‌ی ماهانه‌ی هنرجو را نمایش دهد.

۵ برنامه را توسعه دهید تا میانگین ساعت مطالعه‌ی ماهانه‌ی هنرجو را نمایش دهد.

مرحله‌ی ۱: روندنما

بلوک‌های خالی: داخل حلقه پس از «دریافت زمان مطالعه‌ی روزانه» بلوک sum ← sum + minute و سپس بلوک i ← i + 1 قرار می‌گیرد؛ خروجی مسیر «نادرست» به بلوک «نمایش sum» و سپس پایان می‌رود.

مرحله‌ی ۳: کد کامل

```
int minute, sum = 0;
for (int i = 1; i < 8; i++)
{
    Console.WriteLine("Please rate the day {0} enter in minutes=", i);
    minute = int.Parse(Console.ReadLine());
    sum = sum + minute;
}
Console.WriteLine("you studied for {0} minutes a week", sum);
```

با ورودی‌های ۳۰، ۴۵، ۶۰، ۰، ۹۰، ۱۲۰، ۱۵ خروجی نهایی: you studied for 360 minutes a week

مرحله‌ی ۴ و ۵: ماهانه (۳۰ روز)

```
int minute, sum = 0;
for (int i = 1; i <= 30; i++)
{
    Console.Write("Day {0} (minutes)= ", i);
    minute = int.Parse(Console.ReadLine());
    sum += minute;
}
Console.WriteLine("monthly study: {0:F1} hours", sum / 60.0);
Console.WriteLine("average per day: {0:F2} hours", sum / 60.0 / 30);
```

اگر هر روز ۶۰ دقیقه وارد شود: monthly study: 30.0 hours و average per day: 1.00 hours . تقسیم بر 60.0 (اعشاری) لازم است تا بخش اعشاری حذف نشود.

۱۲ صفحه ۶۷ کارگاه ۵

۲ کد زیر را تکمیل کرده، در متد Main وارد کنید و برنامه را اجرا کنید. در کد زیر متغیرهای مورد نیاز برنامه را اعلان کرده، دستورات دریافت آنها از ورودی را تکمیل کنید.

```
.............................
int score = 5;
Console.Write("Enter number your month:");
month = ....................................
Console.Clear();
for (int i = 1; i <= 5; i++)
{
    Console.Write("your guess?");
    guess = ....................................
    ...
```

{

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

هر دو متغیر month و guess عددی صحیح‌اند و ورودی کنسول رشته است، پس با int.Parse تبدیل می‌شود. break به‌محض حدس درست، حلقه را رها می‌کند.

ایران توشه
توشه ای برای موفقیت

۱۳ صفحه ۶۸ کارگاه ۵

۳ خروجی برنامه را با مقادیر زیر Trace کنید. (ماه ۱۱ با حدس‌های ۳، ۸، ۲، ۴، ۵ و ماه ۵ با حدس‌های ۳، ۸، ۵)

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

Trace با month = 11 (مقدار score پس از پایان هر دور)

ماه ۱۱

خروجی	score	i	guess
! try again	4	1	3
! try again	3	2	8
! try again	2	3	2
! try again	1	4	4
! try again	0	5	5

Trace با month = 5

ماه ۵

خروجی	score	i	guess
! try again	4	1	3
! try again	3	2	8
you win / your score: 3	3	3	5

مرحله‌ی ۴: اصلاح

```
else
{
    score -= 1;
    if (i < 5)
        Console.WriteLine("try again !");
    else
        Console.WriteLine("you lose! The month number was {0}", month);
}
```

اجرای واقعی با ماه ۱۱ و حدس‌های بالا، در دور پنجم پیام you lose! The month number was 11 چاپ می‌شود و امتیاز نهایی ۰ است.

۱۴ صفحه ۶۸ فعالیت گروهی

جدول زیر را به کمک هم‌گروهی خود تکمیل کنید.

قطعه کدها و خروجی

قطعه کد	خروجی برنامه پس از اجرا
for (int k = 10; k <= 15;) { Console.WriteLine(k); k++; }	10 11 12 13 14 15 (هر عدد در یک خط)
= int i = 1; Console.ForegroundColor ConsoleColor.Cyan; for (; i<=10; i++) { ;++Console.WriteLine("Iran"); i }	کلمه‌ی Iran پنج بار (با رنگ فیروزه‌ای)؛ i در هر دور دو واحد زیاد می‌شود (۱، ۳، ۵، ۷، ۹) و در پایان ۱۱ است.
(;;) Console.ForegroundColor = ConsoleColor.Cyan; for { ;Console.WriteLine("Iran") }	حلقه‌ی بی‌پایان: Iran بی‌نهایت بار چاپ می‌شود (بدون شرط، حلقه هرگز تمام نمی‌شود؛ با Ctrl+C متوقف می‌شود).

هر سه بخش سرصفحه‌ی for (مقداردهی اولیه، شرط، تغییر) اختیاری است و در صورت حذف، ; باید بماند. نبودن شرط یعنی شرط همواره درست است.

۱۵ صفحه ۶۹ فعالیت منزل

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

```
for (int i = 0; i < 100; i += 3)
    Console.WriteLine("{0,3}", i);
```

خروجی: اعداد ۰، ۳، ۶، ...، ۹۹ (۳۴ عدد) هر کدام در یک خط.

مسئله‌ی نمونه: یک کلاس ۱۰۰ نفره از هنرجویان داریم که از نفر اول، هر سومین نفر برای شرکت در مسابقه انتخاب می‌شود. برنامه‌ای بنویسید که شماره‌ی هنرجویان انتخاب‌شده (با شمارش از صفر) را چاپ کند.

۱۶ صفحه ۶۹ فعالیت کارگاهی

پس از مشاهده‌ی فیلم، برنامه‌ی MonthGuess را با کلید F10 گام به گام اجرا کنید. تغییرات متغیرها را در جدول زیر یادداشت کنید. (month، guess، i، score، خروجی)

برنامه‌ی کارگاه ۳ را با استفاده از پنجره‌ی Local گام به گام اجرا کنید. چرا پس از اجرا و اتمام کار حلقه، متغیر i از پنجره‌ی Local حذف می‌شود؟

نمونه برای month = 11 و حدس‌های ۳، ۸، ۲، ۴، ۵ (مقدارها پس از اجرای هر دور):

گام‌به‌گام

ایران توشه
توشه‌ای برای موفقیت

خروجی	score	i	guess	month
! try again	4	1	3	11
! try again	3	2	8	11
! try again	2	3	2	11
! try again	1	4	4	11
! try again	0	5	5	11

حذف i از پنجره‌ی Local

متغیر i در سرصفحه‌ی for تعریف شده، پس دامنه‌ی آن (scope) فقط همان حلقه است. با خروج از حلقه، حافظه‌ی آن آزاد می‌شود و دیگر در پنجره‌ی Local دیده نمی‌شود (دلیل خطای CS0103 هم همین است).

این فایل توسط سایت ایران‌توشه تهیه شده است.
irantooshe.ir

۱۷ صفحه ۶۹ فعالیت منزل

برنامه‌ای طراحی کنید که با دریافت نام هنرجویان یک هنرستان تعداد هنرجویانی که هم‌نام حضرت معصومه (س) هستند را نمایش دهد. تعداد کل هنرجویان هنرستان در ابتدای برنامه از کاربر دریافت شود.
برنامه را با کمک کلید F10 گام به گام اجرا کرده، با واردکردن نام ۵ هنرجو مقدار متغیرهای ورودی و خروجی را مشاهده کرده، در جدول زیر وارد کنید. (مقدار متغیر حلقه i / name / count / خروجی)

```
Console.Write("Number of students: ");
int n = int.Parse(Console.ReadLine());
int count = 0;
for (int i = 1; i <= n; i++)
{
    Console.Write("Enter name {0}: ", i);
    string name = Console.ReadLine();
    if (name == "Masoumeh")
        count++;
}
Console.WriteLine("Number of students named Masoumeh: {0}", count);
```

ورودی نمونه: Masoumeh، Zahra، Masoumeh، Ali، Maryam (نتیجه‌ی اجرای واقعی: ۲)

گام‌به‌گام

خروجی	count	name	(i) مقدار متغیر حلقه
—	1	Masoumeh	1
—	1	Zahra	2
—	2	Masoumeh	3
—	2	Ali	4
Number of students named Masoumeh: 2	2	Maryam	5

۱۸ صفحه ۷۰ فعالیت گروهی

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

- وارد کردن رمز عبور تا زمانی که رمز درست وارد شود (تعداد تلاش‌ها معلوم نیست).
- پرداخت قبوض یا خرید در فروشگاه تا زمانی که موجودی کارت تمام شود.
- خواندن داده‌ها از یک فایل تا رسیدن به انتهای آن.

۱۹ صفحه ۷۰ کارگاه ۶

۱ روندنمای مسئله را رسم کنید.

۳ قطعه کد زیر را در متد Main وارد کنید و برنامه را تکمیل کنید.

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

{

کاربرد متغیر count چیست؟

۴ دستورات لازم برای محاسبه و نمایش میانگین را به برنامه اضافه کنید.

مرحله‌ی ۱: روندنما

ایران توشه
توشه ای برای موفقیت

```
Start
  sum <- 0 , count <- 0
  Receive mark
  <mark >= 0 ?> --False--> avg <- sum / count ; Show avg ; End
     | True
  sum <- sum + mark
  count <- count + 1
  Receive mark
  +--> back to the condition
```

مرحله‌ی ۳ و ۴: برنامه‌ی کامل

```
float sum = 0, avg;
int count = 0;
Console.Write("Enter mark {0} =", count + 1);
float mark = float.Parse(Console.ReadLine());
while (mark >= 0)
{
    sum = sum + mark;
    count++;
    Console.Write("Enter mark {0} =", count + 1);
    mark = float.Parse(Console.ReadLine());
}
if (count > 0)
{
    avg = sum / count;
    Console.WriteLine("\nCount = {0}, Average = {1:F2}", count, avg);
}
```

با ورودی ۱۸، ۱۵٫۵، ۱۲، ۲۰ و ۱-: Count = 4, Average = 16.38

شرط حلقه mark >= 0 است (ورود عدد منفی یعنی پایان). متغیر count تعداد نمرات واردشده را می‌شمارد و هم برای نمایش شماره‌ی نمره‌ی بعدی (count + 1) و هم برای مخرج میانگین به کار می‌رود. شرط count > 0 از تقسیم بر صفر در حالتی که هیچ نمره‌ای وارد نشده جلوگیری می‌کند.

ایران توشه
توشه ای برای موفقیت

۲۰ صفحه ۷۱ فعالیت کارگاهی

قطعه کدهای زیر را Trace کرده، اولین و آخرین مقدار چاپ‌شده را در جدول زیر بنویسید.

```
int i = 99;
while (i >= 1)
    Console.WriteLine(--i);
int i = 100;
while (i <= 1000)
    Console.WriteLine(++i);
```

Trace

دستور	اولین مقدار در خروجی	آخرین مقدار در خروجی
;int i = 99; while (i >= 1) Console.WriteLine(--i)	98	0
+)int i = 100; while (i <= 1000) Console.WriteLine ;(i+	101	1001

در i-- ابتدا کاهش و سپس چاپ انجام می‌شود؛ وقتی i برابر ۱ است شرط درست است، کاهش به ۰ می‌رسد و ۰ چاپ می‌شود. در حالت دوم هنگامی که i = 1000 است حلقه یک بار دیگر اجرا شده، ۱۰۰۱ چاپ می‌شود. هر دو نتیجه با اجرای واقعی تأیید شد.

۲۱ صفحه ۷۲ **کارگاه ۷**

در روندنما قسمت‌های مربوط به حلقه را پررنگ کنید.

در الگوریتم کارگاه، مراحل ۳ تا ۶ (بررسی شرط hour > 0، دریافت ساعت کاری نفر بعدی، مقایسه با max و بازگشت به مرحله‌ی ۳) بدنه‌ی حلقه هستند. در روندنما باید این بخش‌ها پررنگ شوند:

- لوزی شرط hour > 0 (در صورت «نادرست» خروج از حلقه و رفتن به نمایش max)،
- متوازی‌الاضلاع «دریافت ساعت کاری نفر بعدی (hour)»،
- لوزی شرط hour > max و مستطیل max ← hour (در صورت «درست»)،
- پیکان برگشت از انتهای بدنه به لوزی شرط hour > 0.

بخش‌های «شروع»، «دریافت ساعت کاری نفر اول»، «max ← hour» اولیه، «نمایش max» و «پایان» خارج از حلقه‌اند و پررنگ نمی‌شوند.

ایران توشه
توشه ای برای موفقیت

۲۲ صفحه ۷۳ **کارگاه ۷**

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

شرط خروج از حلقه در این برنامه چیست؟

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

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

تکمیل برنامه

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

با ورودی ۱۶۰، ۱۷۵، ۱۴۰ و ۰ خروجی max= 175 است.

شرط خروج از حلقه

حلقه تا وقتی ادامه می‌یابد که hour > 0 درست باشد؛ بنابراین با وارد کردن عدد صفر (یا عدد منفی) شرط نادرست شده، حلقه پایان می‌یابد.

کمترین ساعت کاری

فقط مقدار اولیه و شرط مقایسه تغییر می‌کند (برای همان ورودی‌ها خروجی ۱۴۰ است):

```
min = hour;
while (hour > 0)
{
    if (hour < min)
        min = hour;
    Console.Write("Enter hour (0 for Exit): ");
    hour = int.Parse(Console.ReadLine());
}
Console.WriteLine("min= {0}", min);
```

کمترین و بیشترین با هم

```
int hour, max, min;
Console.Write("Enter hour: ");
hour = int.Parse(Console.ReadLine());
max = min = hour;
while (hour > 0)
{
    if (hour > max) max = hour;
    if (hour < min) min = hour;
    Console.Write("Enter hour (0 for Exit): ");
    hour = int.Parse(Console.ReadLine());
}
Console.WriteLine("max= {0}, min= {1}", max, min);
```

ایران توشه
توشه ای برای موفقیت

۲۳ صفحه ۷۳ فعالیت کارگاهی

ماشین‌حساب کنسولی همانند شکل روبه‌رو ایجاد کنید.

برنامه دو عدد و یک عملگر را از ورودی می‌گیرد، نتیجه را چاپ می‌کند و تا زمانی که کاربر پایان را انتخاب نکرده است تکرار می‌شود (شکل کتاب در دسترس نبود):

```
char op = '+';
while (op != 'e')
{
    Console.Write("Enter first number: ");
    double a = double.Parse(Console.ReadLine());
    Console.Write("Enter operator (+ - * /): ");
    op = char.Parse(Console.ReadLine());
    Console.Write("Enter second number: ");
    double b = double.Parse(Console.ReadLine());
    switch (op)
    {
        case '+': Console.WriteLine("= {0}", a + b); break;
        case '-': Console.WriteLine("= {0}", a - b); break;
        case '*': Console.WriteLine("= {0}", a * b); break;
        case '/':
            if (b != 0) Console.WriteLine("= {0}", a / b);
            else Console.WriteLine("Division by zero!");
            break;
    }
    Console.Write("Press e to exit, any other key to continue: ");
    op = char.Parse(Console.ReadLine());
}
```

این جواب ممکن است دقیق نباشد.

۲۴ صفحه ۷۴ فعالیت گروهی

به کمک هم‌گروهی خود شکل را توصیف کنید.

در این مثال‌ها یک کار تکراری (حلقه‌ی بیرونی) انجام می‌شود و در هر بار تکرار آن، کار تکرارشونده‌ی دیگری (حلقه‌ی درونی) به‌طور کامل اجرا می‌شود؛ مانند عقربه‌های ساعت: به ازای هر یک حرکت عقربه‌ی کند، عقربه‌ی تند چندین بار می‌چرخد.

- حلقه‌ی بیرونی: ۵ دور دویدن دور حیاط، یا ۱۰ طبقه‌ی مجتمع.
- حلقه‌ی درونی: ۳ حرکت بارفیکس در پایان هر دور، یا ۸ واحد هر طبقه.
- تعداد کل تکرار کارِ درونی = تعداد تکرار حلقه‌ی بیرونی × تعداد تکرار حلقه‌ی درونی (مثلاً ۵×۳ = ۱۵ حرکت بارفیکس و ۱۰×۸ = ۸۰ غذای نذری).

ایران توشه
توشه‌ای برای موفقیت

این جواب ممکن است دقیق نباشد.

۲۵ صفحه ۷۵ کارگاه ۸

۲ ـ ... برنامه را اجرا کنید. آیا خروجی یک جدول‌ضرب را نشان می‌دهد؟

۳ ـ با توجه به اعداد مشخص‌شده یک تا پنج، در کد مرحله ۲ با استفاده از دستور حلقه‌ی for دستورهای Console.Write را به یک دستور تبدیل کنید.

۴ ـ برای تنظیم نمایش جدول‌ضرب به جای جانگهدار از دستور SetCursorPosition استفاده کنید.

۵ ـ برنامه را با کمک کلید F10 خط به خط اجرا کنید.

بله؛ هر بار اجرای حلقه‌ی for، یک سطر جدول (ضرب‌های ۱ تا ۵ برای i) را در ستون‌هایی با عرض ۴ می‌نویسد و Console.WriteLine() به سطر بعد می‌رود. خروجی (اجرای واقعی):

```
   1   2   3   4   5
   2   4   6   8  10
   3   6   9  12  15
   4   8  12  16  20
   5  10  15  20  25
```

تبدیل پنج دستور Write به یک حلقه (حلقه‌ی متداخل)

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

استفاده از SetCursorPosition

```
for (int i = 1; i <= 5; i++)
    for (int j = 1; j <= 5; j++)
    {
        Console.SetCursorPosition(j * 4, i);   // ستون، سطر
        Console.Write(i * j);
    }
Console.SetCursorPosition(0, 7);
```

در SetCursorPosition(left, top) پارامتر اول شماره‌ی ستون و دومی شماره‌ی سطر است و مکان‌نما را قبل از چاپ هر عدد جابه‌جا می‌کند.

با F10 هر بار یک دستور اجرا می‌شود؛ مشاهده می‌شود به ازای هر مقدار i، حلقه‌ی درونی j پنج بار (۱ تا ۵) کامل اجرا شده و سپس i یک واحد زیاد می‌شود.

ایران توشه
توشه ای برای موفقیت

۲۶ صفحه ۷۵ **فعالیت گروهی**

اگر خودپردازی دارای اسکناس‌های ۲ و ۵ هزار تومانی باشد و شما درخواست ۵۰ هزار تومان وجه نقد کنید، چگونه پرداخت را انجام می‌دهد؟ چه روش‌های پرداختی می‌تواند داشته باشد؟ جدول روبه‌رو را تکمیل کنید.

اگر x تعداد اسکناس ۲ هزار تومانی و y تعداد اسکناس ۵ هزار تومانی باشد، باید 2000x + 5000y = 50000 شود. همه‌ی حالت‌ها:

جدول پرداخت ۵۰ هزار تومان

تعداد اسکناس ۵ هزار تومانی	تعداد اسکناس ۲ هزار تومانی
۱۰	۰
۸	۵
۶	۱۰
۴	۱۵
۲	۲۰
۰	۲۵

خودپرداز معمولاً با توجه به موجودی اسکناس‌ها، ترجیحاً کمترین تعداد اسکناس (۱۰ اسکناس ۵ هزار تومانی) را می‌پردازد.

۲۷ صفحه ۷۵ **کارگاه ۹**

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

کافی است نوبت‌دهی یک روز را داخل یک حلقه‌ی بیرونی با شش تکرار (شش روز کاری) قرار دهیم:

```
string[] days = {"Saturday", "Sunday", "Monday", "Tuesday", "Wednesday", "Thursday"};
for (int d = 0; d < 6; d++)
{
    Console.WriteLine("--- {0} ---", days[d]);
    for (int hour = 16; hour < 20; hour++)          // ساعت‌های کاری
        for (int min = 0; min < 60; min += 30)       // هر نوبت ۳۰ دقیقه
            Console.WriteLine("{0:00}:{1:00}", hour, min);
}
```

این جواب ممکن است دقیق نباشد.

ایران توشه
توشه ای برای موفقیت

۲۸ صفحه ۷۶ فعالیت منزل

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

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

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

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

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

خروجی برنامه (اجرای واقعی): X=0,Y=10، X=5,Y=8، X=10,Y=6، X=15,Y=4، X=20,Y=2 و X=25,Y=0 است که دقیقاً با شش سطر جدول بالا یکی است. خودپرداز معمولاً حالتی را انتخاب می‌کند که تعداد اسکناس کمتر باشد، یعنی X=0 و Y=10.

بیشترین تعداد اسکناس ۲ هزار تومانی برای ۵۰ هزار تومان ۵۰,۰۰۰ ÷ ۲,۰۰۰ = ۲۵ و بیشترین تعداد اسکناس ۵ هزار تومانی ۵۰,۰۰۰ ÷ ۵,۰۰۰ = ۱۰ است؛ بنابراین حدّ بالای حلقه‌ها همین مقادیر است.

حلقه‌ی بیرونی ۲۶ بار (X از ۰ تا ۲۵) و درونی برای هر بار ۱۱ بار (Y از ۰ تا ۱۰) اجرا می‌شود؛ یعنی ۲۶ × ۱۱ = ۲۸۶ حالت آزمایش می‌شود (با اجرای واقعی تأیید شد).

با اسکناس ۱۰ هزار تومانی (Z) و گرفتن مبلغ از ورودی

```
Console.Write("Enter amount: ");
int amount = int.Parse(Console.ReadLine());
for (int X = 0; X <= amount / 2000; X++)
    for (int Y = 0; Y <= amount / 5000; Y++)
        for (int Z = 0; Z <= amount / 10000; Z++)
            if (2000 * X + 5000 * Y + 10000 * Z == amount)
                Console.WriteLine("2000:{0}  5000:{1}  10000:{2}", X, Y, Z);
```

ایران توشه
توشه ای برای موفقیت

۲۹ صفحه ۷۶ فعالیت منزل

ـ برنامه‌ی کارگاه ۶ را توسعه دهید تا با دریافت نمرات بقیه‌ی دروس پایه‌ی یازدهم، میانگین آنها را محاسبه و نمایش دهد.

ـ کمترین و بیشترین میانگین نمره را نمایش دهید.

برای هر درس میانگین نمرات محاسبه می‌شود (پایان ورود نمره با عدد منفی) و میانگین‌ها با کمترین و بیشترین مقایسه می‌شوند: حلقه‌ی بیرونی روی دروس و حلقه‌ی درونی (while) برای ورود نمرات هر درس.

```
Console.Write("Number of courses: ");
int n = int.Parse(Console.ReadLine());
float max = float.MinValue, min = float.MaxValue;
for (int c = 1; c <= n; c++)
{
    float sum = 0; int count = 0;
    Console.Write("Course {0}, enter mark (negative to stop): ", c);
    float mark = float.Parse(Console.ReadLine());
    while (mark >= 0)
    {
        sum += mark; count++;
        Console.Write("Enter mark: ");
        mark = float.Parse(Console.ReadLine());
    }
    if (count == 0) continue;
    float avg = sum / count;
    Console.WriteLine("Average of course {0} = {1:F2}", c, avg);
    if (avg > max) max = avg;
    if (avg < min) min = avg;
}
Console.WriteLine("Max average = {0:F2}, Min average = {1:F2}", max, min);
```

۳۰ صفحه ۷۹ فعالیت کارگاهی

روندنمایی رسم کنید که اسامی پنج هنرجو را دریافت کرده، سپس اسامی را از آخر به اول نمایش دهد.

برنامه‌ی این الگوریتم را به زبان سی‌شارپ بنویسید.

در این مثال، اگر تعداد اسامی ۳۰ نفر باشد: چه تعداد دستور خواندن از ورودی و نوشتن روی صفحه‌نمایش لازم است؟ آیا می‌توان از دستور for برای خواندن ورودی‌ها و نوشتن اسامی استفاده کرد؟ برای چنین مسئله‌هایی که با داده‌های زیاد سروکار دارند، چه روشی را پیشنهاد می‌کنید؟

ایران توشه
توشه ای برای موفقیت

روندنما (شرح)

1. شروع
2. دریافت نام اول تا پنجم و قرار دادن در متغیرهای name1 تا name5 (پنج متوازی‌الاضلاع ورودی)
3. نمایش name5، سپس name2، name3، name4 و name1 (پنج متوازی‌الاضلاع خروجی)
4. پایان

برنامه‌ی C#

```
string name1, name2, name3, name4, name5;
name1 = Console.ReadLine();
name2 = Console.ReadLine();
name3 = Console.ReadLine();
name4 = Console.ReadLine();
name5 = Console.ReadLine();
Console.WriteLine(name5);
Console.WriteLine(name4);
Console.WriteLine(name3);
Console.WriteLine(name2);
Console.WriteLine(name1);
```

برای ۳۰ نفر به ۳۰ متغیر، ۳۰ دستور خواندن و ۳۰ دستور نوشتن (جمعاً ۶۰ دستور) نیاز است. چون هر نام در یک متغیر جدا با نام متفاوت قرار دارد نمی‌توان با for به متغیرها دسترسی یافت. راه‌حل مناسب استفاده از آرایه است: یک نام مشترک و شماره‌ی اندیس برای هر عنصر؛ در این صورت حلقه‌ی for با اندیس به همه‌ی عناصر دسترسی می‌یابد.

۳۱ صفحه ۸۰ **فعالیت کارگاهی**

یک پروژه‌ی جدید ایجاد کنید و دستور اعلان آرایه و دستور تخصیص حافظه به آرایه‌ی name را بنویسید.

بررسی کنید محتوای هر عنصر آرایه چیست؟

```
string[] name;          // اعلان
name = new string[5];   // تخصیص حافظه برای ۵ عنصر
```

با اجرای برنامه با F10 و دیدن محتوای آرایه، هر پنج عنصر (name[0] تا name[4]) برابر null هستند؛ زیرا حافظه تخصیص یافته ولی هنوز مقداری در عناصر قرار نگرفته است (مقدار پیش‌فرض عناصر رشته‌ای null است).

ایران توشه
توشه ای برای موفقیت

۳۲ صفحه ۸۱ **فعالیت کارگاهی**

ـ آرایه‌ای به نام salary برای نگهداری حقوق ۳۰۰ کارمند اعلان کنید.

ـ آرایه‌ای به نام vowels برای نگهداری حروف صدادار انگلیسی اعلان کنید.

ـ آرایه‌ای به نام lamp برای نگهداری وضعیت روشن و خاموش بودن پنج لامپ اعلان کنید.

ـ آرایه‌ای با نام دلخواه برای نگهداری اسامی ماه‌های سال اعلان کنید.

ـ آرایه‌ای با نام دلخواه برای نگهداری معدل هنرجویان کلاس اعلان کنید.

```
long[] salary = new long[300];
char[] vowels = {'a', 'e', 'i', 'o', 'u'};
bool[] lamp = new bool[5];                  // true = روشن ، false = خاموش
string[] months = new string[12];           // اسامی ماه‌های سال
float[] gpa = new float[30];                // معدل هنرجویان (مثلاً ۳۰ نفر)
```

نوع داده‌ی آرایه باید با نوع داده‌ی عناصر آن هم‌خوان باشد: حقوق عدد صحیح بزرگ، حروف char، وضعیت دوحالته bool، نام ماه string و معدل عدد اعشاری.

۳۳ صفحه ۸۱ کنجکاوی

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

چون در سی‌شارپ (و بیشتر زبان‌های برنامه‌نویسی) اندیس اولین عنصر آرایه صفر است. پس عنصر اول اندیس ۰، عنصر دوم اندیس ۱ و عنصر سوم اندیس ۲ دارد؛ یعنی اندیس هر عنصر یک واحد کمتر از شماره‌ی ترتیبی آن است.

۳۴ صفحه ۸۲ کارگاه ۱

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

چرا مقدار برخی از عناصر برابر null است؟

۶ ـ برای نگهداری قیمت قطعات رایانه، آرایه‌ی price را متناظر با آرایه‌ی نام قطعات اعلان کنید.

۷ ـ عناصر آرایه‌ی price را مقداردهی کنید.

۸ ـ محتوای آرایه‌ی price چیست؟ چرا؟

ایران توشه
توشه ای برای موفقیت

برای ذخیره‌ی نام قطعات بیشتر باید اندازه‌ی آرایه را بزرگ‌تر کنیم (مثلاً new string[8]) و سپس به عناصر جدید مقدار بدهیم؛ زیرا اندازه‌ی آرایه پس از ساخته شدن ثابت است. عناصری که مقداردهی نشده‌اند مقدار null دارند، چون هنوز مقداری در آنها قرار نگرفته است.

```
long[] price = new long[5];
price[0] = 200000;
price[1] = 150000;
price[2] = 300000;
price[3] = 450000;
price[4] = 100000;
```

هنگام اجرا با F10 تا رسیدن به دستور مقداردهی price[2]: price[0] برابر ۲۰۰۰۰۰ و price[1] برابر ۱۵۰۰۰۰ است و عناصر ۲ تا ۴ برابر ۰ هستند؛ زیرا مقدار پیش‌فرض عناصر عددی صفر است و هنوز مقداردهی نشده‌اند.

۳۵ صفحه ۸۳ کارگاه ۱

۱۰ ـ دستور زیر را جایگزین کد مرحله ۹ کنید ... به‌جای عدد ۵ عدد ۴ را قرار دهید. خطای رخ‌داده چیست؟

۱۲ ـ دستوری برای تغییر مقدار عنصر شماره‌ی ۳ آرایه‌ی computer به "Keyboard" به کد اضافه کنید.

۱۳ ـ دستوری بنویسید که قیمت Keyboard را از ورودی دریافت کرده و در عنصر شماره‌ی ۳ آرایه‌ی price قرار دهد.

اگر در new string[4]{"CPU", "MainBoard", "RAM", "HDD", "DVD"} اندازه ۴ باشد ولی ۵ مقدار داده شود، خطای زمان کامپایل رخ می‌دهد: تعداد مقادیر با اندازه‌ی آرایه برابر نیست (Array initializers must have exactly the declared number of elements). برای جلوگیری از این خطا بهتر است آرایه بدون new و اندازه تعریف شود.

منظور از «عنصر شماره‌ی ۳» اندیس ۳ است (چهارمین عنصر، HDD):

```
computer[3] = "Keyboard";
price[3] = long.Parse(Console.ReadLine());
```

۳۶ صفحه ۸۵ کنجکاوی

آیا می‌توان با یک متد WriteLine() تمام عناصر آرایه را نمایش داد؟ حاصل اجرای دستور Console.WriteLine(price); چیست؟

خیر؛ Console.WriteLine(price) عناصر را چاپ نمی‌کند، بلکه نام نوع آرایه را نمایش می‌دهد؛ برای مثال برای آرایه‌ی long[] خروجی System.Int64[] است. برای نمایش عناصر باید از حلقه (یا string.Join) استفاده کرد.

۳۷ صفحه ۸۵ کارگاه ۲

۲ ـ اعلان آرایه‌ی week را تکمیل کنید.

۴ ـ برنامه را با عدد ۷ اجرا کنید. چه خطایی رخ می‌دهد؟ علت آن چیست؟

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

ایران توشه
توشه‌ای برای موفقیت

۶ ـ دستوراتی به برنامه اضافه کنید که اگر شماره‌ی واردشده در محدوده‌ی مجاز نبود، پیام مناسب نمایش داده شود.

```
string[] week = {"saturday", "sunday", "monday", "tuesday", "wednesday", "thursday",
"friday"};
Console.Write("Enter the number one day of week (0-6): ");
byte num = byte.Parse(Console.ReadLine());
Console.WriteLine(week[num]);
```

با عدد ۷ خطای زمان اجرا IndexOutOfRangeException (Index was outside the bounds of the array) رخ می‌دهد؛ زیرا آرایه‌ی week هفت عنصر با اندیس‌های ۰ تا ۶ دارد و اندیس ۷ وجود ندارد.

برای دریافت عدد ۱ تا ۷، از شماره‌ی واردشده یک واحد کم می‌کنیم: week[num – 1].

```
Console.Write("Enter the number of day (1-7): ");
byte num = byte.Parse(Console.ReadLine());
if (num >= 1 && num <= 7)
    Console.WriteLine(week[num - 1]);
else
    Console.WriteLine("Invalid number! Enter a number between 1 and 7.");
```

۳۸ صفحه ۸۶ فعالیت کارگاهی

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

منظور روزهایی است که اندیس آنها زوج است (۰، ۲، ۴، ۶): شنبه، دوشنبه، چهارشنبه و جمعه.

```
string[] week = {"saturday", "sunday", "monday", "tuesday", "wednesday", "thursday",
"friday"};
for (int i = 0; i < week.Length; i += 2)
    Console.WriteLine(week[i]);
```

خروجی: saturday، monday، wednesday، friday. برای آرایه‌ی ۱۰۰ عنصری هم کافی است حد حلقه ۱۰۰ باشد: for (int i = 0; i < 100; i++).

۳۹ صفحه ۸۷ کارگاه ۳

۲ ـ تعداد قطعات رایانه را از ورودی دریافت کنید و دو آرایه با نام‌های computer و price اعلان کنید.

۴ ـ کد مرحله‌ی قبل را تغییر دهید تا علاوه بر عناصر computer، قیمت قطعات را هم دریافت کنید.

ایران توشه
توشه‌ای برای موفقیت

۶ ـ برنامه را طوری تکمیل کنید که نام، قیمت قطعات و قیمت کل را نمایش دهد.

```
Console.Write("Number of parts: ");
int n = int.Parse(Console.ReadLine());
string[] computer = new string[n];
long[] price = new long[n];
long sumPrice = 0;
for (int i = 0; i < computer.Length; i++)
{
    Console.Write("Enter the name of a computer part: ");
    computer[i] = Console.ReadLine();
    Console.Write("Enter its price: ");
    price[i] = long.Parse(Console.ReadLine());
    sumPrice += price[i];
}
for (int i = 0; i < computer.Length; i++)
    Console.WriteLine("{0,-12}{1,10}", computer[i], price[i]);
Console.WriteLine("Total price = {0}", sumPrice);
```

۴۰ صفحه ۸۸ کارگاه ۴

۳ ـ ... برنامه را اجرا کنید. خروجی آن چیست؟

۴ ـ کد زیر را جایگزین مرحله‌ی ۳ کنید (foreach) ... نتیجه و کد را با مرحله‌ی ۳ مقایسه کنید.

۵ ـ برای نمایش معکوس رشته، کد زیر را تکمیل کنید. آیا می‌توان این کد را با دستور foreach نوشت؟

```
Console.Write("Enter a string: ");
string myStr = Console.ReadLine();
for (int i = 0; i < myStr.Length; i++)
    Console.WriteLine(myStr[i]);
```

با ورودی Ali نویسه‌های A، l و i هر یک در یک سطر چاپ می‌شوند. کد foreach (char c in myStr) Console.WriteLine(c); نیز دقیقاً همان خروجی را می‌دهد، ولی بدون استفاده از اندیس و با کد کوتاه‌تر.

```
for (int i = myStr.Length - 1; i >= 0; i--)
    Console.Write(myStr[i]);
```

خیر؛ foreach همیشه عناصر را از اول تا آخر پیمایش می‌کند و امکان پیمایش از آخر به اول ندارد.

۴۱ صفحه ۸۹ کارگاه ۵

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

۳ ـ دستورات محاسبه‌ی مالیات ارزش‌افزوده را بنویسید ... در این کد نقش متغیر i و j چیست؟

ایران توشه
توشه‌ای برای موفقیت

۴ ـ در قطعه کد مرحله‌ی ۳ تبدیل صریح به int را حذف کنید. چه خطایی رخ می‌دهد؟ چرا؟

```
int[] priceList = new int[50];
int i, temp;
for (i = 0; i < 50; i++)
{
    Console.Write("Enter the price of {0} the product: ", i + 1);
    temp = int.Parse(Console.ReadLine());
    if (temp > 0)
        priceList[i] = temp;
    else
        break;
}
Console.WriteLine("Product price plus VAT:");
for (int j = 0; j < i; j++)
{
    priceList[j] = (int)(priceList[j] + 0.09 * priceList[j]);
    Console.WriteLine("Product{0}: {1}", j, priceList[j]);
}
```

متغیر i تعداد قیمت‌های معتبر واردشده را نگه می‌دارد (در حلقه‌ی اول شمارنده‌ی ورودی‌ها و پس از خروج با break تعداد محصولات است). متغیر j اندیس پیمایش عناصر پُر آرایه (از ۰ تا i–1) برای محاسبه‌ی قیمت با مالیات است. مثلاً برای قیمت‌های ۱۰۰۰، ۲۰۰۰ و ۳۵۰۰ نتیجه ۱۰۹۰، ۲۱۸۰ و ۳۸۱۵ می‌شود (اجرای واقعی).

اگر (int) حذف شود، خطای زمان کامپایل Cannot implicitly convert type 'double' to 'int' رخ می‌دهد؛ چون عبارت priceList[j] * 0.09 + priceList[j] نتیجه‌ی double دارد و تبدیل ضمنی double به int چون ممکن است بخش اعشاری از بین برود مجاز نیست؛ باید تبدیل صریح انجام شود.

۴۲ صفحه ۸۹ کنجکاوی

ـ در دستور foreach منظور از متغیر حلقه چیست؟

ـ آیا می‌توان به‌وسیله‌ی عملگر انتساب (=) مقداری را داخل متغیر حلقه‌ی foreach قرار داد؟

متغیر حلقه‌ی foreach در هر بار تکرار یک عنصر آرایه را (به ترتیب از اولین تا آخرین عنصر) نگه می‌دارد و نوع آن باید با نوع عناصر آرایه یکی باشد.

خیر؛ متغیر حلقه‌ی foreach فقط‌خواندنی است و انتساب به آن خطای کامپایل می‌دهد (Cannot assign to ... because it is a 'foreach iteration variable'). بنابراین با foreach نمی‌توان عناصر آرایه را تغییر داد.

ایران توشه
توشه‌ای برای موفقیت

۴۳ صفحه ۸۹ فعالیت منزل

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

```
Console.Write("Number of students: ");
int n = int.Parse(Console.ReadLine());
int[] level = new int[n];
for (int i = 0; i < n; i++)
{
    do
    {
        Console.Write("Level of student {0} (1-3): ", i + 1);
        level[i] = int.Parse(Console.ReadLine());
    } while (level[i] < 1 || level[i] > 3);   // فقط سطح ۱، ۲ یا ۳ پذیرفته می‌شود
}
int count = 0;
foreach (int l in level)
    if (l == 1) count++;
Console.WriteLine("Not competent: {0}", count);
```

۴۴ صفحه ۹۰ فعالیت منزل

برنامه‌ای بنویسید که هزینه‌ی روزانه‌ی یک ماه شما را دریافت کرده، در یک آرایه ذخیره کند. سپس هزینه‌ی ماهانه و میانگین هزینه‌ی روزانه‌ی شما را محاسبه کرده، نمایش دهد. سپس تعداد روزهایی که هزینه‌ی روزانه بیشتر از میانگین شده‌است را محاسبه کرده و نمایش دهد.

```
long[] cost = new long[30];
long sum = 0;
for (int i = 0; i < cost.Length; i++)
{
    Console.Write("Cost of day {0}: ", i + 1);
    cost[i] = long.Parse(Console.ReadLine());
    sum += cost[i];
}
double avg = (double)sum / cost.Length;
int above = 0;
foreach (long c in cost)
    if (c > avg) above++;
Console.WriteLine("Monthly cost = {0}", sum);
Console.WriteLine("Average daily cost = {0:F1}", avg);
Console.WriteLine("Days above average = {0}", above);
```

ایران توشه
توشه ای برای موفقیت

۴۵ صفحه ۹۱ فعالیت گروهی

در هر یک از موارد زیر تعیین کنید از روش جست‌وجوی خطی یا دودویی استفاده می‌شود؟ انتخاب روش بر اساس چه معیاری است؟

نوع جست‌وجو

مورد	روش	دلیل
۱ ـ یافتن کارنامه‌ی یک هنرجو بین کارنامه‌های مرتب‌شده بر اساس نام	دودویی	داده‌ها بر اساس نام مرتب‌اند
۲ ـ پیدا کردن یک کتاب بر اساس نام آن در کتابخانه	دودویی	کتاب‌ها معمولاً بر اساس حروف الفبا مرتب‌اند
۳ ـ پیدا کردن یک کتاب بر اساس کد کتاب در کتابخانه	دودویی	کدها مرتب (ترتیبی) هستند
۴ ـ پیدا کردن نام هنرجویی که بالاترین معدل را دارد، اگر اسامی بر اساس معدل مرتب باشد	نیازی به جست‌وجو نیست	بالاترین معدل در ابتدا یا انتهای فهرست مرتب قرار دارد؛ در حالت کلی دودویی

معیار انتخاب: اگر داده‌ها مرتب باشند، جست‌وجوی دودویی سریع‌تر است؛ اگر مرتب نباشند فقط جست‌وجوی خطی ممکن است.

۴۶ صفحه ۹۱ فعالیت گروهی

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

تعداد مقایسه

مکان عنصر موردنظر در آرایه	اول آرایه	وسط آرایه	آخر آرایه	در آرایه نباشد
تعداد مقایسه	۱	۵۰	۱۰۰	۱۰۰

۴۷ صفحه ۹۲ کارگاه ۶

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

۳ ـ ... برنامه را اجرا کنید. خروجی برنامه چیست؟

۴ ـ برنامه را طوری تغییر دهید تا قیمت خودروی مورد جست‌وجو در خروجی نمایش داده شود.

ایران توشه
توشه ای برای موفقیت

۵ ـ مقدار item را برابر Hyundai قرار دهید. خروجی چیست؟

نقش متغیر found در این کد چیست؟ به‌جای شرط found==false عبارت !found را قرار دهید و برنامه را اجرا کنید.

۷ ـ برنامه را طوری تغییر دهید که نام خودرو را از ورودی دریافت کند. با دو مقدار Renault و Hyundai نتیجه‌ی برنامه را مشاهده کنید.

```
string[] car = {"Pride", "Mazda", "Renault", "Peugeot", "Toyota"};
long[] price = {2500, 6000, 4500, 5000, 7000};   // قیمت‌ها (فرضی)
Console.Write("Enter car name: ");
string item = Console.ReadLine();
bool found = false;
for (int i = 0; i < car.Length; i++)
    if (item == car[i])
    {
        found = true;
        Console.WriteLine("price {0}={1}", item, price[i]);
        break;
    }
if (!found)
    Console.WriteLine("item Not Found");
```

با item = "Pride" خروجی مرحله‌ی ۳ Pride Found in 0 است (اولین عنصر آرایه) و در مرحله‌ی ۴ قیمت آن (price Pride=2500) چاپ می‌شود. با Hyundai چون در آرایه نیست در مرحله‌ی ۵ هیچ خروجی‌ای چاپ نمی‌شود و در مرحله‌ی ۶ پیام item Not Found نمایش داده می‌شود. با Renault خروجی price Renault=4500 است.

متغیر found پرچم (flag) است: مقدار اولیه‌اش false است و فقط وقتی عنصر پیدا شود true می‌شود تا بعد از حلقه معلوم باشد جست‌وجو موفق بوده یا نه. عبارت !found دقیقاً همان معنی found == false را دارد و خروجی یکسان است.

قیمت‌های آرایه‌ی price در کتاب نیامده و فرضی هستند؛ هر مقدار دلخواه پذیرفته است.

ایران توشه
توشه ای برای موفقیت

۴۸ صفحه ۹۳ فعالیت منزل

برنامه‌ای بنویسید که نام و نمره‌ی هنرجویان را از ورودی دریافت کرده، در آرایه‌ی مناسب ذخیره کند. سپس نام هنرجویانی را نمایش دهد که نمره‌ی ۲۰ گرفته‌اند.

```
Console.Write("Number of students: ");
int n = int.Parse(Console.ReadLine());
string[] name = new string[n];
float[] mark = new float[n];
for (int i = 0; i < n; i++)
{
    Console.Write("Name: ");
    name[i] = Console.ReadLine();
    Console.Write("Mark: ");
    mark[i] = float.Parse(Console.ReadLine());
}
Console.WriteLine("Students with mark 20:");
for (int i = 0; i < n; i++)
    if (mark[i] == 20)
        Console.WriteLine(name[i]);
```

۴۹ صفحه ۹۳ فعالیت گروهی

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

ـ هنرجویی عدد ۵۹ را یادداشت کرده است. در روش جست‌وجوی دودویی حداکثر با چند مقایسه عدد مورد نظر حدس زده خواهد شد؟ جدول زیر را کامل کنید.

ـ در جست‌وجوی دودویی اگر عنصر مورد جست‌وجو در آرایه نباشد، الگوریتم چگونه پایان می‌یابد؟

بهترین الگوریتم، حدس زدن عدد وسط بازه و نصف کردن بازه‌ی باقی‌مانده بر اساس پاسخ «بزرگ‌تر/کوچک‌تر» است. اولین حدس باید وسط بازه‌ی ۱ تا ۱۰۰ یعنی 50 باشد. برای عدد ۵۹ پنج حدس لازم است:

جدول حدس‌ها

مرحله	اولین عدد	آخرین عدد	عدد حدس‌زده‌شده (وسط)	نتیجه‌ی مقایسه
۱	۱	۱۰۰	۵۰	عدد بزرگ‌تر است
۲	۵۱	۱۰۰	۷۵	عدد کوچک‌تر است
۳	۵۱	۷۴	۶۲	عدد کوچک‌تر است
۴	۵۱	۶۱	۵۶	عدد بزرگ‌تر است
۵	۵۷	۶۱	۵۹	برابر است (پیدا شد)

ایران توشه
توشه ای برای موفقیت

حداکثر تعداد مقایسه برای بازه‌ی ۱۰۰ عددی حدود ۷ است (چون ۲ به توان ۷ برابر ۱۲۸ است). اگر عدد در آرایه نباشد، با هر مقایسه بازه نصف می‌شود تا اولین اندیس از آخرین اندیس بزرگ‌تر شود (بازه خالی)؛ آنگاه الگوریتم با پیام «پیدا نشد» پایان می‌یابد.

۵۰ صفحه ۹۳ فعالیت منزل

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

جست‌وجوی دودویی عدد ۱۲

مرحله	اولین عدد	آخرین عدد	عدد حدس‌زده‌شده (وسط)	نتیجه‌ی مقایسه
۱	۱	۱۰۰	۵۰	عدد کوچک‌تر است
۲	۱	۴۹	۲۵	عدد کوچک‌تر است
۳	۱	۲۴	۱۲	برابر است (پیدا شد)

۵۱ صفحه ۹۴ کارگاه ۷

مقدار متغیر found پس از اجرای برنامه چیست؟ عملکرد متد IndexOf چیست؟

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

مقدار متغیر found پس از اجرای این دستور چیست؟ تفاوت عملکرد متد LastIndexOf با متد IndexOf چیست؟

۷ ـ با دستور foreach عناصر آرایه‌ی a را در خروجی نمایش دهید. عملکرد متد Sort چیست؟

۸ ـ مقدار متغیر found چیست؟ (BinarySearch)

۹ ـ ... عملکرد متد Reverse چیست؟

ایران توشه
توشه ای برای موفقیت

۱۰ ـ مقدار متغیر found چیست؟

```
int[] a = {20, 18, 10, 15, 50, 80, 10, 37};
int x = 10;
int found = Array.IndexOf(a, x);
Console.WriteLine(found);          // 2
found = Array.LastIndexOf(a, x);
Console.WriteLine(found);          // 6
Array.Sort(a);
foreach (int v in a) Console.Write(v + " ");   // 10 10 15 18 20 37 50 80
found = Array.BinarySearch(a, x);
Console.WriteLine(found);          // 1
Array.Reverse(a);
foreach (int v in a) Console.Write(v + " ");   // 80 50 37 20 18 15 10 10
found = Array.BinarySearch(a, x);
Console.WriteLine(found);          // -1
```

نتایج اجرای واقعی

دستور	نتیجه	توضیح
Array.IndexOf(a, 10)	۲	اندیس اولین عنصر برابر ۱۰ را برمی‌گرداند
Array.IndexOf(a, 60)	۱-	عنصر وجود ندارد؛ مقدار منفی ۱ برمی‌گردد
Array.LastIndexOf(a, 10)	۶	اندیس آخرین عنصر برابر ۱۰ (IndexOf از ابتدا و LastIndexOf از انتها جست‌وجو می‌کند)
Array.Sort(a)	10 10 15 18 20 37 50 80	عناصر را به‌صورت صعودی مرتب می‌کند
Array.BinarySearch(a, 10)	۱	روی آرایه‌ی مرتب‌شده‌ی صعودی؛ اندیس یکی از عنصرهای ۱۰ (با وجود تکرار)
Array.Reverse(a)	80 50 37 20 18 15 10 10	ترتیب عناصر را وارونه می‌کند
Array.BinarySearch(a, 10)	۱-	آرایه دیگر صعودی نیست؛ جست‌وجوی دودویی نتیجه‌ی درست نمی‌دهد (عنصر پیدا نمی‌شود)

جست‌وجوی دودویی فقط روی آرایه‌ی مرتب‌شده‌ی صعودی درست کار می‌کند.

۵۲ صفحه ۹۵ **فعالیت منزل**

می‌خواهیم برنامه‌ای بنویسیم که با استفاده از یک فهرست انتخاب (منو) عملیات زیر را روی آرایه انجام دهد و برنامه تا زمانی که کاربر گزینه‌ی خروج را انتخاب نکرده است، ادامه یابد. گزینه‌های منو به رنگ سفید روی

ایران توشه
توشه ای برای موفقیت

زمینه‌ی آبی نمایش داده شوند. ۱ دریافت عناصر آرایه ۲ نمایش محتوای عناصر آرایه ۳ مرتب کردن ۴ معکوس کردن ۵ جست‌وجوی خطی ۶ جست‌وجوی دودویی ۷ خروج

```
int[] a = new int[0];
int choice = 0;
while (choice != 7)
{
    Console.BackgroundColor = ConsoleColor.Blue;
    Console.ForegroundColor = ConsoleColor.White;
    Console.Clear();
    Console.WriteLine("1 Input   2 Show   3 Sort   4 Reverse");
    Console.WriteLine("5 Linear Search   6 Binary Search   7 Exit");
    Console.Write("Your choice: ");
    choice = int.Parse(Console.ReadLine());
    switch (choice)
    {
        case 1:
            Console.Write("Number of elements: ");
            a = new int[int.Parse(Console.ReadLine())];
            for (int i = 0; i < a.Length; i++) a[i] = int.Parse(Console.ReadLine());
            break;
        case 2: foreach (int v in a) Console.Write(v + " "); break;
        case 3: Array.Sort(a); break;
        case 4: Array.Reverse(a); break;
        case 5:
            Console.Write("Value: ");
            Console.WriteLine("Index = " + Array.IndexOf(a,
int.Parse(Console.ReadLine())));
            break;
        case 6:
            Console.Write("Value (array must be sorted): ");
            Console.WriteLine("Index = " + Array.BinarySearch(a,
int.Parse(Console.ReadLine())));
            break;
    }
    if (choice != 7) { Console.WriteLine("\nPress Enter..."); Console.ReadLine(); }
}
```

ایران توشه
توشه ای برای موفقیت

این فایل توسط سایت ایران‌توشه تهیه شده است.

irantooshe.ir

۵۳ صفحه ۹۶ پروژه فاز ۲

برنامه‌ی کتابخانه‌ی هنرستان را با استفاده از دستوراتی که در این پودمان آموخته‌اید بنویسید. تعداد کتاب‌ها و اعضا را از ورودی دریافت کرده، آرایه‌های متناظر برای آنها اعلان کنید ... منوی ۱ Input book ۲ Input user

۳ Exit ۷ EditUser ۶ Subject Book ۵ Show Book ۴ Show User ... یک نوع داده‌ی شمارشی به نام MenuItem برای تشخیص منوی انتخاب‌شده تعریف کنید.

```
enum MenuItem { InputUser = 1, InputBook, ShowUser, ShowBook, SubjectBook, EditUser,
Exit }
// داخل Main:
Console.Write("Number of members: ");
int u = int.Parse(Console.ReadLine());
Console.Write("Number of books: ");
int b = int.Parse(Console.ReadLine());
string[] firstName = new string[u], lastName = new string[u];
string[] title = new string[b], subject = new string[b];
MenuItem m = 0;
while (m != MenuItem.Exit)
{
    Console.WriteLine("\n1 Input user  2 Input book  3 Show User  4 Show Book");
    Console.WriteLine("5 Subject Book  6 EditUser  7 Exit");
    m = (MenuItem)int.Parse(Console.ReadLine());
    switch (m)
    {
        case MenuItem.InputUser:
            for (int i = 0; i < u; i++)
            { Console.Write("First, Last name of member {0}: ", i);
              firstName[i] = Console.ReadLine(); lastName[i] = Console.ReadLine(); }
            break;
        case MenuItem.InputBook:
            for (int i = 0; i < b; i++)
            { Console.Write("Title, Subject of book {0}: ", i);
              title[i] = Console.ReadLine(); subject[i] = Console.ReadLine(); }
            break;
```

ایران توشه
توشه ای برای موفقیت

```
        case MenuItem.ShowUser:
            for (int i = 0; i < u; i++)
                Console.WriteLine("{0}: {1} {2}", i, firstName[i], lastName[i]);
            break;
        case MenuItem.ShowBook:
            for (int i = 0; i < b; i++)
                Console.WriteLine("{0}: {1} ({2})", i, title[i], subject[i]);
            break;
        case MenuItem.SubjectBook:
            Console.Write("Subject: ");
            string s = Console.ReadLine();
            for (int i = 0; i < b; i++)
                if (subject[i] == s) Console.WriteLine(title[i]);
            break;
        case MenuItem.EditUser:
            Console.Write("Membership code: ");
            int code = int.Parse(Console.ReadLine());
            if (code >= 0 && code < u)
            { firstName[code] = Console.ReadLine(); lastName[code] =
Console.ReadLine(); }
            else Console.WriteLine("Invalid code");
            break;
    }
}
```

اندیس آرایه‌ها همان کد عضویت (و کد کتاب) است.

تکمیل پروژه: مقایسه‌ی کد ۱ و ۲

کد شماره‌ی ۱ (Thread.Sleep(50) + foreach) هر نویسه‌ی رشته‌ی C# Programing را با ۵۰ میلی‌ثانیه توقف جداگانه چاپ می‌کند، بنابراین متن به‌صورت «تایپ‌شونده» و تدریجی دیده می‌شود؛ کد شماره‌ی ۲ کل رشته را یک‌جا و بی‌درنگ چاپ می‌کند. برای زیباسازی پروژه (مثلاً چاپ نام برنامه در ابتدای اجرا) از کدی مشابه کد ۱ استفاده می‌کنیم و برای استفاده از Sleep باید using System.Threading; اضافه شود.

این فایل توسط سایت ایران‌توشه تهیه شده است.
irantooshe.ir