

جزوه کارگاه برنامه نویسی

شامل:

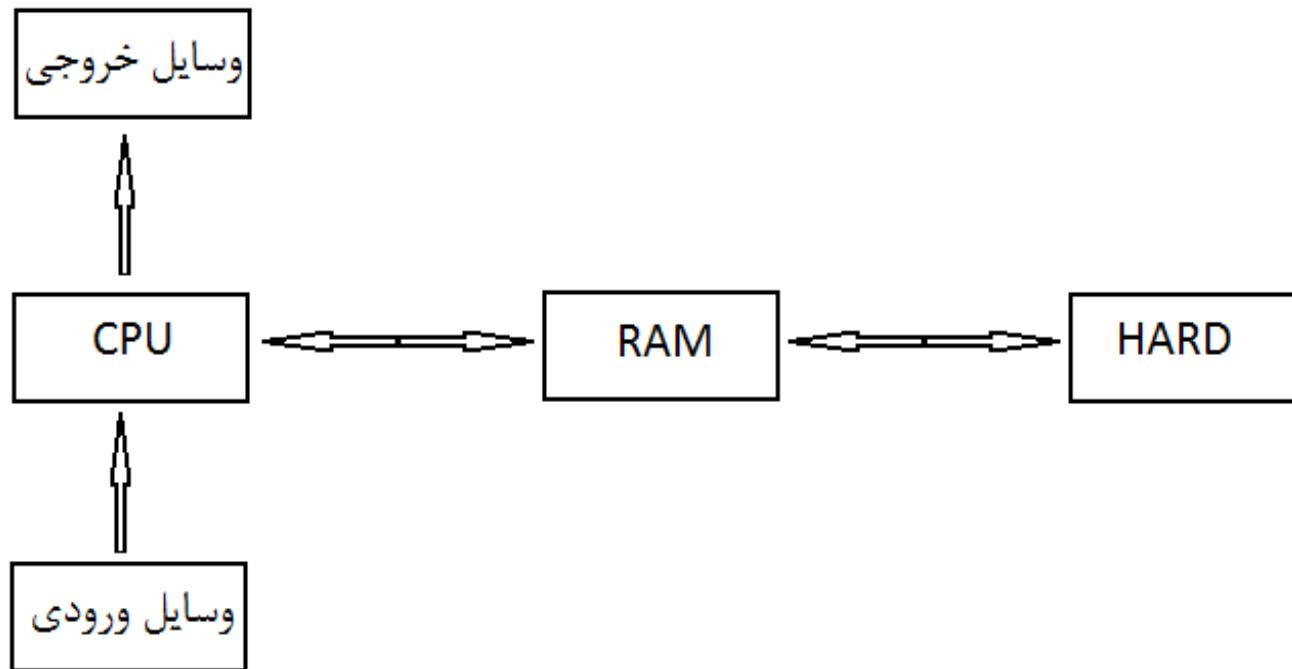
- ✓ توضیح و حل برخی تمرینات آخر فصل کتاب الگوریتم و فلوچارت
- ایرج صادقی
- ✓ گریزی به نرم افزار visual studio 2012

تهیه و تنظیم:

محمد ربانی

۱) مقدمه ای بر نحوه برنامه نویسی:

✓ دیدگاه برنامه نویس از سخت افزار رایانه:



✓ HARD محل ذخیره دائمی اطلاعات، فضای ذخیره سازی بالا، سرعت دستیابی اطلاعات کم، درایوهای رایانه فایل ها و پوشه های آنها محل هایی از هارد هستند.

✓ RAM محل ذخیره موقت اطلاعات، فضای ذخیره سازی کم، سرعت دستیابی اطلاعات بالا، اطلاعات برنامه های باز شده در رم قرار می گیرد و در هنگام ذخیره تغییرات این تغییرات در هارد ذخیره خواهد شد. علت و جود رم در رایانه جبران سرعت پایین دستیابی اطلاعات در هارد است.

✓ CPU محل پردازش اطلاعات

✓ کارهایی که برنامه نویس می تواند در محیط برنامه نویسی انجام دهد:

(۱) تعریف متغیر

(۲) تخصیص متغیر

(۳) گرفتن متغیر از ورودی و نمایش آن در خروجی

(۴) مقایسه بین متغیرها

(۵) کنترل، کنترل برنامه

✓ (۱) **تعریف متغیر** : منظور از تعریف متغیر در زبان برنامه نویسی این است که فضایی در ram رایانه برای یک متغیر در نظر گرفته شود.
در زبان VB تعریف متغیر به صورت زیر است:

نوع متغیر as نام متغیر Dim

نام متغیر : نام متغیر را برای آن تعیین می کنیم که بعدا بتوانیم به آن اشاره کنیم که میتواند ترکیبی از حروف و اعداد باشد اما نباید با عدد شروع شود مثلا نام متغیر می تواند a1 باشد یا s و یا ali و هر چیز دیگری.

نوع متغیر: نوع متغیر را برای آن تعیین می کنیم که کامپیوتر بداند چه اندازه فضا باید برای متغیر ما در ram در نظر بگیرد که برای اعداد صحیح به جای نوع متغیر می نویسیم integer ، برای اعداد اعشاری می نویسیم double و برای متغیر از نوع حروف (کاراکتر) می نویسیم string
برای مثال عبارت `dim k1 as double` زمانی که در برنامه اجرا شود کامپیوتر فضایی به اندازه ی double در ram خود برای متغیر اعشاری k1 در نظر می گیرد.

✓ (۲) تخصیص متغیر

منظور از تخصیص متغیر قرار دادن مقدار در متغیر است مثلاً اگر ما در یک برنامه بنویسیم:

```
Dim a as double
```

```
A=3.14
```

در زمان اجرای برنامه در خط اول متغیر a از نوع اعشاری تعریف میشود در زمان تعریف یک متغیر مقدار اولیه که برای آن در نظر گرفته می شود مقدار صفر است یعنی وقتی خط اول اجرا میشود کامپیوتر یک فضا به اندازه ی اعداد اعشاری (double) در ram در نظر می گیرد که مقدار اولیه ی آن صفر است حال زمانی که خط دوم اجرا می شود رایانه عدد 3.14 را در متغیر a قرار میدهد. پس برای انجام عمل تخصیص یا مقدار دهی متغیر ما از علامت = در زبان برنامه نویسی استفاده می کنیم این علامت با علامت تساوی که در زبان ریاضی استفاده می شود متفاوت است که در ادامه متوجه می شوید.

طرف دوم علامت = در زبان برنامه نویسی می تواند یک محاسبه ی ریاضی بین یک یا چند متغیر باشد مثلاً عبارت $a=a+1$ در زبان برنامه نویسی یعنی حاصل $a+1$ را در a قرار بده اما از نظر ریاضی عبارت $a=a+1$ اشتباه است.

مثال دیگر: عبارت $b = (a1 * b2 + 4) / 2$ در زبان برنامه نویسی یعنی متغیر $a1$ را در متغیر $b2$ ضرب کن و بعد با عدد ۴ جمع کرده سپس حاصل را بر ۲ تقسیم کرده جواب را در متغیر b بریز. دقت کنیم که در زبان برنامه نویسی $b2$ به معنای $b \times 2$ نیست $b2$ کلاً نام یک متغیر است و اگر در زبان برنامه نویسی بخواهیم $b \times 2$ اجرا شود باید بنویسیم $b * 2$ و نه $b2$

✓ (۳) گرفتن متغیر از ورودی و نمایش آن در خروجی:

در زبان VB و در پلت فرم console گرفتن متغیر از ورودی با عبارت

`Console.ReadLine()` = نام متغیر

و نمایش متغیر در خروجی با عبارت

`Console.WriteLine(نام متغیر)`

نوشته می شود. مثلا برنامه ای بنویسید که یک عدد اعشاری را از ورودی بگیرد و به توان خودش برساند و در خروجی نمایش دهد.
حل:

```
Dim a as double
```

```
A = Console.ReadLine()
```

```
A = a ^ a
```

```
Console.WriteLine(a)
```

توضیح برنامه: خط اول متغیر `a` را تعریف می کند تا فضایی به اندازه ی `double` برای عددی که کاربر وارد می کند در `ram` در نظر گرفته شود خط دوم عددی که کاربر وارد می کند را از ورودی می گیرد خط سوم آن عدد را به توان خودش میرساند و در خودش می ریزد خط چهارم حاصل را نمایش می دهد.

✓ (۴) مقایسه ی بین متغیر ها:

وقتی ما دو مقدار را با هم مقایسه می کنیم حاصل به صورت صحیح یا غلط است مثلاً وقتی ما وزن فیل را با وزن شیر مقایسه می کنیم اگر بگوییم وزن فیل کمتر از وزن شیر است غلط است و اگر بگوییم وزن شیر کمتر از وزن فیل است صحیح است کامپیوتر هم می تواند مقدار دو یا چند متغیر را با هم مقایسه کند و اگر حاصل این مقایسه درست باشد کارهایی را انجام دهد و اگر حاصل این مقایسه غلط باشد کارهای دیگری را انجام دهد. مقایسه در زبان vb با دستور شرطی if صورت می گیرد که حالت کلی آن را با یک مثال شرح می دهیم.

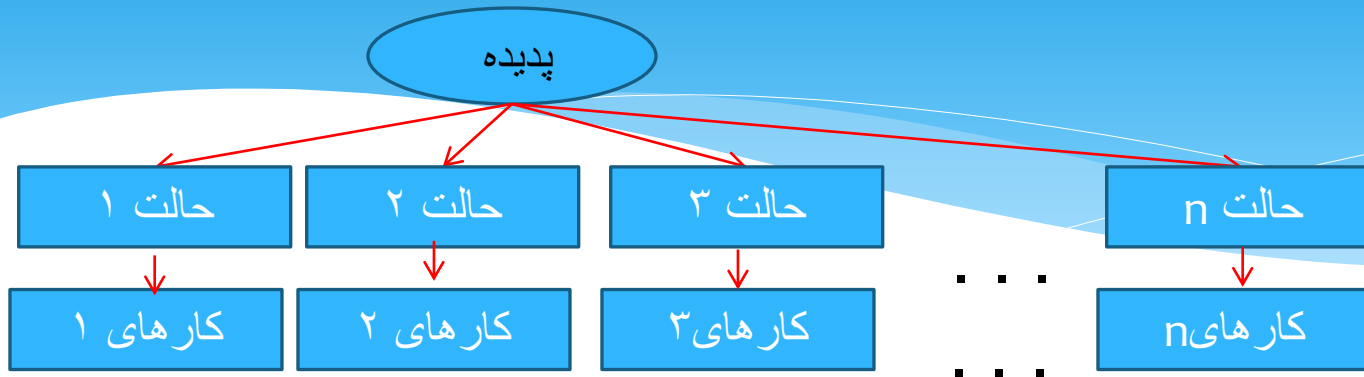
مثال: برنامه ای بنویسید که عدد صحیحی را از ورودی دریافت کند اگر این عدد برابر ۱ باشد عبارت one را چاپ کند برای عدد ۲ عبارت tow ، برای عدد ۳ عبارت three و برای عدد بیشتر از ۳ عبارت more than three را چاپ کند.

```
Dim a As Integer
a = Console.ReadLine()
If a = 1 Then
    Console.WriteLine("one")
ElseIf a = 2 Then
    Console.WriteLine("tow")
ElseIf a = 3 Then
    Console.WriteLine("three")
ElseIf a > 3 Then
    Console.WriteLine("more than tree")
End If
```

توضیح: خط اول متغیر ش را تعریف می کند. خط دوم متغیر a را از ورودی دریافت می کند خط سوم میگوید اگر متغیر a برابر ۱ باشد سپس (انگاه) خط چهارم اجرا شود خط چهارم عبارت one را چاپ می کند خط پنجم می گوید در غیر اینصورت (یعنی a=1 نبود) اگر a=2 بود خط شش اجرا شود و ... خط آخر end if پایان دستور if را اعلام می کند.

دستور if در حالت کلی:

اگر برای یک پدیده که دارای n حالت است بخواهیم برای هر حالت کارهایی صورت گیرد از این دستور استفاده می کنیم.



If حالت ۱ Then

کارهای ۱

ElseIf حالت ۲ then

کارهای ۲

ElseIf حالت ۳ then

کارهای ۳

.

.

.

ElseIf حالت n-1 then

کارهای n-1

Else

کارهای n

End if

برای این کار در VB می نویسیم:

اگر تنها یک حالت داشته باشیم دستور if را میتوانیم به صورت زیر بنویسیم.

If حالت ۱ Then

کارهای ۱

End if

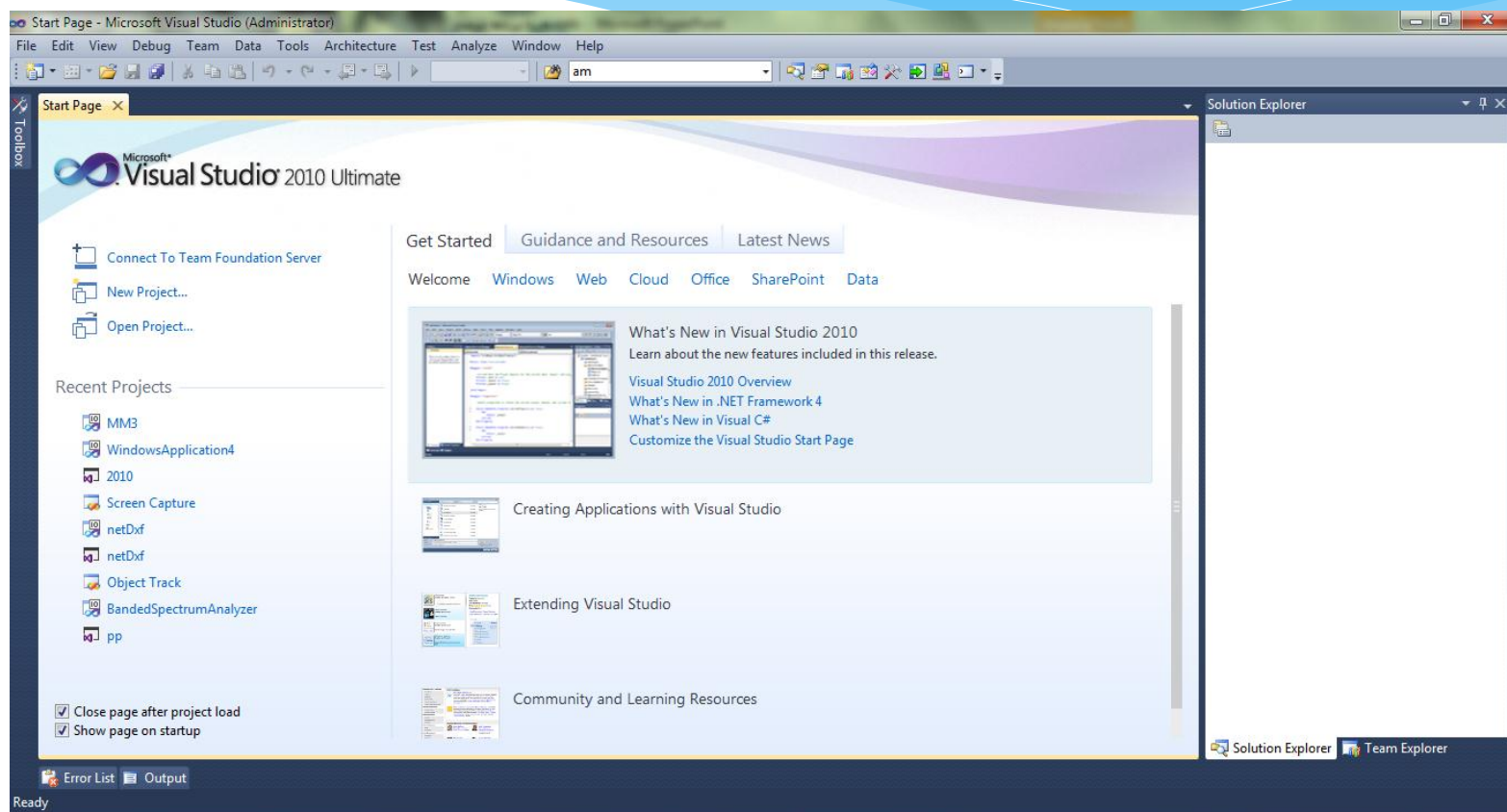
و یا اینکه با قرار دادن علامت : بین کارها دستور if را در یک خط بنویسیم.

If کار ۱ : کار ۲ Then حالت ۱

✓ (۵) کنترل، کنترل برنامه :

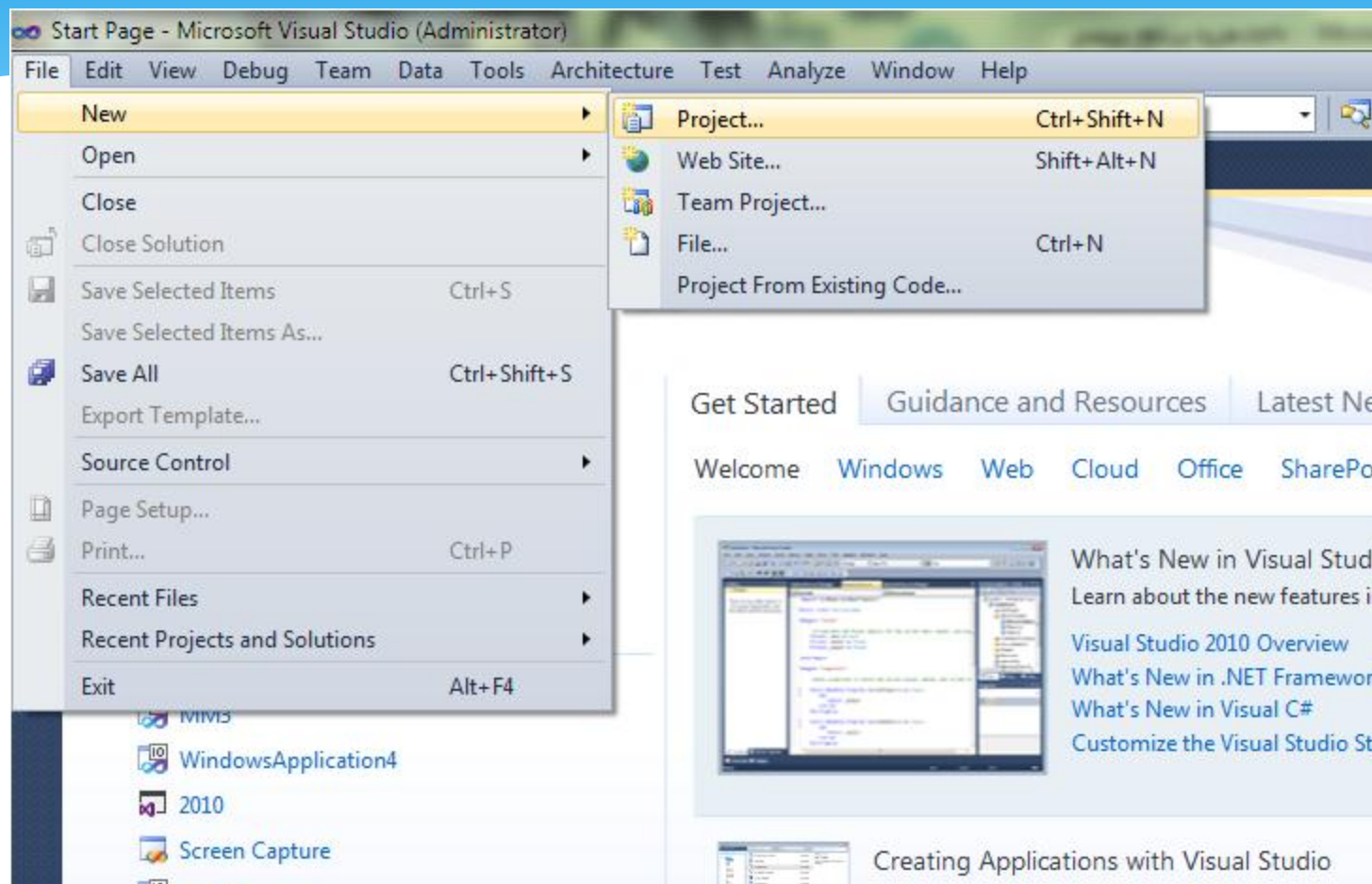
پنجمین و آخرین کاری که رایانه در زمان برنامه نویسی می تواند انجام دهد کنترل کنترل برنامه است. زمانی که یک برنامه ی نوشته شده اجرا می شود رایانه برنامه را به صورت خط به خط (دستور به دستور) از بالا به پایین اجرا می کند. منظور از کنترل برنامه محل اجرای برنامه در هر لحظه است. منظور از کنترل، کنترل برنامه این است که می توان کنترل برنامه را از یک خط به خط دیگر انتقال داد مثلاً برنامه تا خط ۵ اجرا شده حال می توان با دستورهایی کنترل برنامه را به خط ۳ یا خط ۱۲ و یا هر خط دیگری انتقال داد. این قسمت در بخش حلقه ها توضیح داده خواهد شد.

- نحوه ی نوشتن برنامه در محیط کنسول:
- (۱) برنامه ی visual studio 2005 یا ورژن بالاتر را نصب نمایید.
 - (۲) در منوی استارت ویندوز تایپ نمایید visual studio تا برنامه پیدا شود.
 - (۳) برنامه ی visual studio را اجرا نمایید و صبر کنید تا صفحه ی start آن باز شود.
- در شکل ۱ صفحه ی start ویژوال استودیو ۲۰۱۰ را مشاهده می کنید.



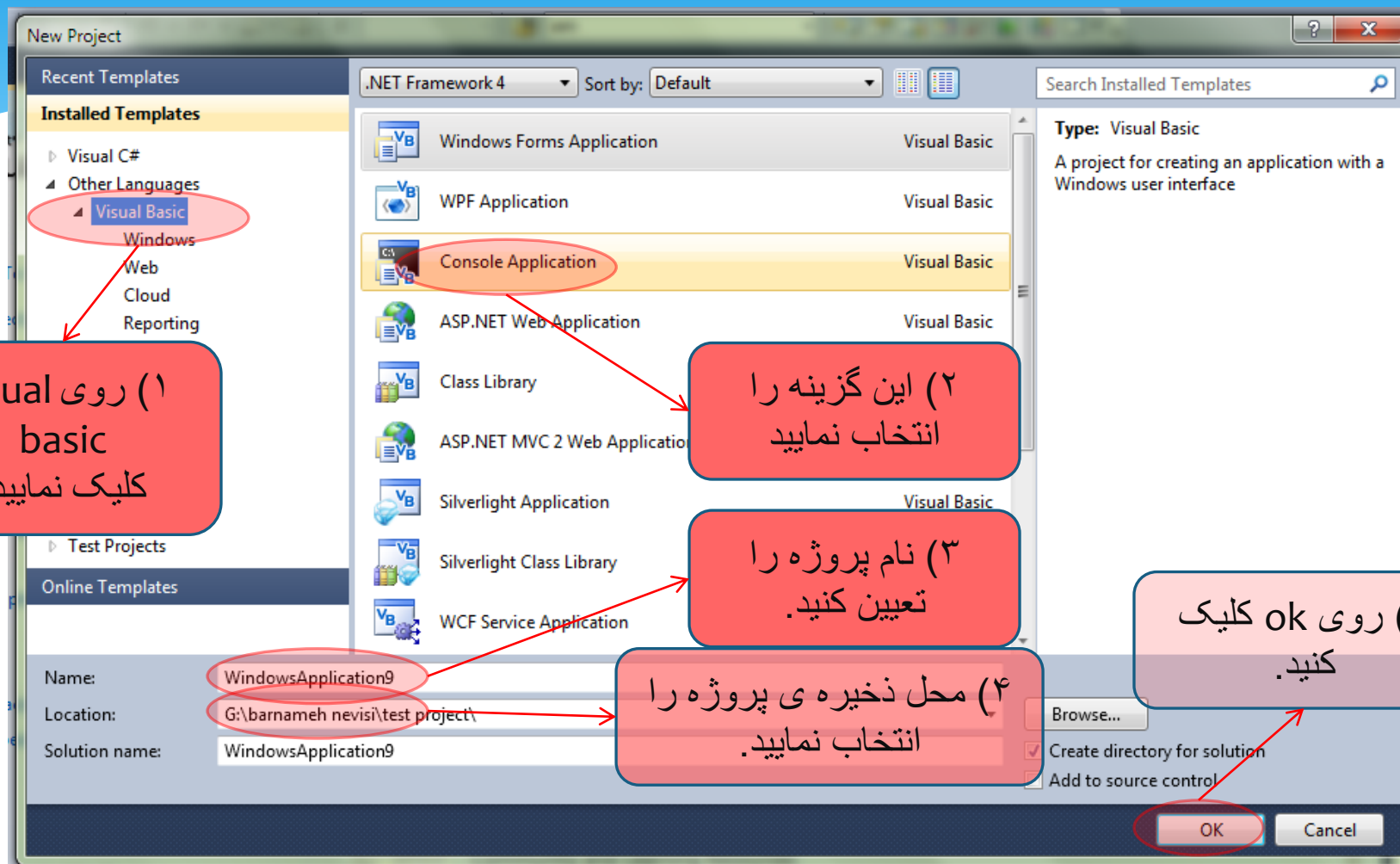
شکل ۱- صفحه ی start ویژوال استودیو ۲۰۱۰

نحوه ی نوشتن برنامه در محیط کنسول:
(۴) از منوی فایل گزینه ی New و بعد گزینه ی Project را انتخاب نمایید. شکل ۲



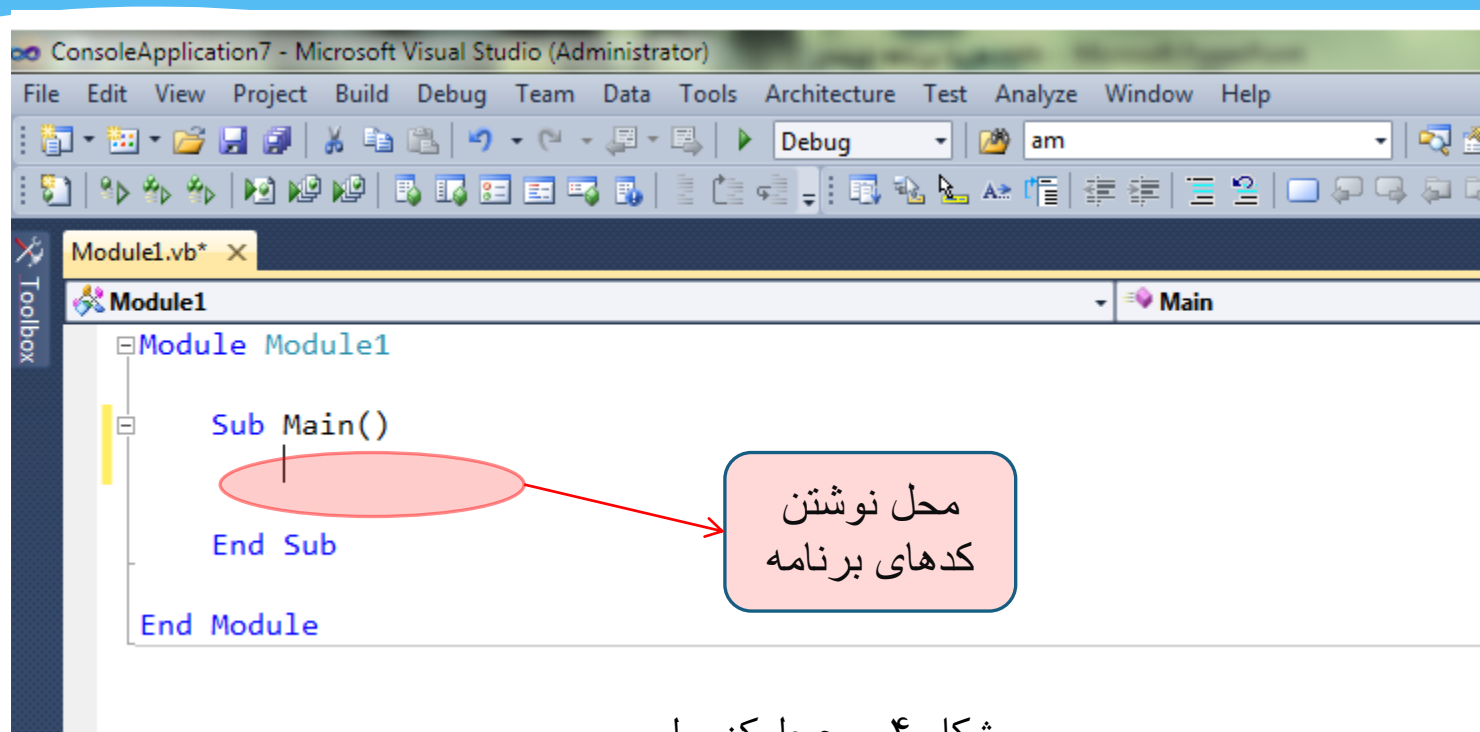
شکل ۲- انتخاب پروژه ی جدید

نحوه ی نوشتن برنامه در محیط کنسول:
(۵) حال پنجره ی new project باز میشود در این پنجره مطابق شکل ۳ عمل نمایید.



شکل ۳- انتخاب پروژه ی جدید

نحوه ی نوشتن برنامه در محیط کنسول:
(۶) حال محیط کنسول باز می شود و شما می توانید کدهای خود را در محل مشخص شده در شکل ۴ بین خطوط `sub main()` و `end sub` قرار دهید. برای این کار در انتهای خط `sub main()` کلیک کرده `enter` را بزنید حال برنامه ی خود را تایپ نمایید.



شکل ۴- محیط کنسول

(۷) قبل از اجرای برنامه اگر زیر کدی از برنامه خطی به رنگ آبی کشیده شده باشد یعنی اشکال تایپی و یا ساختاری در برنامه وجود دارد مثلاً اگر یک دستور `if` شامل عبارت `end if` نباشد برنامه زیر عبارت `if` خط آبی می کشد. تمامی اشکالات را برطرف کنید. برای تست و اجرای برنامه از کلید ترکیبی **Ctrl + F5** استفاده کنید.

مثال) برنامه ای در محیط کنسول بنویسید که ضرایب معادله درجه ۲ را دریافت کند و جواب معادله را حساب کرده در خروجی نمایش دهد.

ابتدا حالت کلی معادله درجه ۲ را در نظر می گیریم

$$ax^2 + bx + c = 0$$
$$\Delta = b^2 - 4ac$$
$$\Delta > 0 \left\{ \begin{array}{l} x_1 = (-b + \sqrt{\Delta}) / 2a \\ x_2 = (-b - \sqrt{\Delta}) / 2a \end{array} \right.$$
$$\Delta = 0 \quad x_1 = x_2 = -b / 2a$$
$$\Delta < 0 \rightarrow \text{معادله جواب ندارد}$$

```

Sub Main()
    Dim a As Double
    Dim b As Double
    Dim c As Double
    Dim d As Double
    Dim x1 As Double
    Dim x2 As Double
    a = Console.ReadLine()
    b = Console.ReadLine()
    c = Console.ReadLine()
    d = b ^ 2 - 4 * a * c
    If d > 0 Then
        x1 = (-b + Math.Sqrt(d)) / (2 * a)
        x2 = (-b - Math.Sqrt(d)) / (2 * a)
        Console.WriteLine(x1 & " " & x2)
    ElseIf d = 0 Then
        x1 = -b / (2 * a)
        x2 = x1
        Console.WriteLine(x1 & " " & x2)
    Else
        Console.WriteLine("no answer")
    End If
End Sub

```

در این دستور ما برای اینکه متغیر x1 و x2 را در یک خط در خروجی نمایش دهیم طوری که بین آنها مقداری فاصله باشد تا به هم نچسبند از عملگر اتصال رشته ی & استفاده کرده ایم و متغیر x1 را به رشته ی " " و آن را به متغیر x2 اتصال دادیم. (اگر متوجه نشدید نکات ۷ و ۸ تمرینات ص ۸۶ را بخوانید.)

تمرین

برای هر یک از خواسته‌های زیر ابتدا الگوریتمی بنویسید، سپس روندنمای متناظر با آنرا رسم کنید.

۱- منطق برنامه‌ای را طراحی نمایید که دو عدد را بخواند به گونه‌ای که یکی از آنها ۱ تا ۷ و معرف روز هفته باشد (۱ نشانگر شنبه، ۲ نشانگر یکشنبه، ... و ۷ نشانگر جمعه است) و دیگری از ۱ تا ۱۲ و معرف ماه باشد (۱ نشانگر فروردین، ۲ نشانگر اردیبهشت، ... و ۱۲ نشانگر اسفند است) و روز یا ماه متناظر با عدد را چاپ نماید.

۲- منطق برنامه‌ای را تعیین نمایید که سه عدد را در ورودی دریافت دارد و تعیین نماید کدامیک از سه عدد مضرب ۵ می‌باشند سپس عدد مضرب ۵ را به همراه توان ۳ آن چاپ نماید.

۳- منطق برنامه‌ای را طراحی نمایید که سه عدد را در ورودی دریافت دارد، سپس آنها را به صورت نزولی (از بزرگ به کوچک) نمایش دهد.

۴- منطق برنامه‌ای را طراحی نمایید که سه عدد را در ورودی دریافت دارد، سپس آنها را به صورت صعودی نمایش دهد.

۵- منطق برنامه‌ای را طراحی نمایید که مقدار x را در ورودی دریافت دارد، سپس با نگرش به وضعیت x مقدار y را به صورت زیر محاسبه و چاپ نماید.

$$y = \begin{cases} 2x^2 - 3x + 1 & x < 0 \\ 3 & x = 0 \\ (2x + 4)(2x - 5) & x > 0 \end{cases}$$

۶- منطق برنامه‌ای را طراحی نمایید که عددی را بخواند و اگر عدد مضرب ۱۲ باشد آنرا به نسبت‌های ۳ و ۴ و ۵ تقسیم کند و حاصل تقسیم به نسبتها را چاپ نماید.

۷- منطق برنامه‌ای را بنویسید که دو عدد M و N را در ورودی دریافت دارد و بررسی نماید اگر M کوچک‌تر از N است محتوای آنها را با هم عوض کند. سپس اگر M بر N بخش پذیر باشد خارج قسمت آنها را چاپ نماید.

۸- منطق برنامه‌ای را طراحی نمایید که ۳ جزء یک تناسب را بخواند و جزء چهارم تناسب را محاسبه نماید و اگر جزء چهارم تناسب مضرب ۳ است خارج قسمت تقسیم جزء چهارم بر ۳ را محاسبه نماید و به همراه جزء چهارم چاپ نماید، در غیر این صورت چهار جزء تناسب را چاپ نماید.

۹- منطق برنامه‌ای را طراحی نمایید که دو عدد را در ورودی دریافت دارد، سپس در صد عدد کوچک‌تر را نسبت به عدد بزرگ‌تر محاسبه و چاپ نماید.

۱۰- منطق برنامه‌ای را طراحی نمایید که دو عدد M و N را در ورودی دریافت دارد، سپس اگر $M > N$ باشد مقدار $2M + N$ را چاپ کند و اگر $M < N$ باشد مقدار $3N - M$ را چاپ کند و اگر $M = N$ باشد مربع اعداد را چاپ کند.

تمرین های صفحه ی ۸۶ کتاب:

نکات تمرینات صفحه ۸۶:

۱. عدد a مضرب عدد b است اگر باقی مانده تقسیم عدد a بر عدد b برابر صفر شود. در زبان VB باقی مانده تقسیم a/b را به صورت $a \bmod b$ نشان می دهند.

۲. برای تعویض محتوای دو متغیر a و b به صورت زیر عمل می کنیم:
(الف) با استفاده از متغیر سوم c

```
c = b  
b = a  
a = c
```

(ب) بدون استفاده از متغیر سوم

```
a = a - b  
b = a + b  
a = b - a
```

۳. نمایش خارج قسمت تقسیم a بر b در VB به صورت $\text{Int}(a/b)$ است.

۴. نماد مخالف در VB به صورت $<>$ است.

۵. متد FLOOR کلاس MATH برای نمایش جزء صحیح استفاده می شود همچنین می توان از عبارت INT هم برای اینکار بهره جست (مانند زیر).

```
Math.Floor(-2.3)  
Int(-2.3)
```

نکات تمرینات صفحه ۸۶:

۶. متدهای پرکاربرد کلاس Math:

متد	توضیح
Math.PI	عدد پی (π)
Math.E	عدد نپر (e)
Math.Exp(x)	e^x
Math.Abs(a)	$ a $
Math.Pow(a, b)	a^b
Math.Sqrt(a)	$\sqrt{a}$
Math.Log10(a)	$\log_{10} a$
Math.Log(a)	$\ln a$
Math.Round(a)	روند a
Math.Floor(a)	$[a]$

۷. برای نمایش حروف و کلمات (رشته) باید آنها را بین دابل کوتیشن قرار داد. مثلا
`Console.WriteLine(" Hello! ")`

۸. برای اتصال چند متغیر یا رشته به هم و نمایش آنها در یک سطر از عملوند اتصال رشته ی & استفاده می کنیم. مثلا عبارت `Console.WriteLine(a & "is" & b)` متغیر a را به رشته ی is و سپس به متغیر b اتصال داده و نمایش می دهد.

۹. برای نوشتن چند خط از برنامه در یک خط بین خطوط از علامت : استفاده می شود.
مثال:

```
n = c : c = a : a = n
```

۱۰. برای خلاصه نویسی در تعریف چند متغیر بین متغیرها از علامت , استفاده می شود.
مثال:

```
Dim a, b, c, n As Double
```

۱۱. برای تعریف متغیر از نوع رشته (حروف، اعداد و نمادها) از عبارت `dim a as string` استفاده می شود و برای تعریف متغیر از نوع صحیح از عبارت `dim a as integer`

۱۲. برای ریختن یک متغیر عددی مثل X به متغیر از نوع رشته مثل S از عبارت `s=str(x)` استفاده می شود و بلعکس برای ریختن یک متغیر رشته مثلا S که شامل تنها عدد باشد در یک متغیر عددی مثل X از عبارت `x=val(s)`.

۱۳. برای نوشتن عبارات توضیحی قبل از آن عبارت از علامت کوتیشن استفاده میشود عبارات توضیحی کمپایل نمیشوند و تاثیری در برنامه ندارند مثلا:

```
Dim n, k, s, rn As Integer
```

```
'input k
```

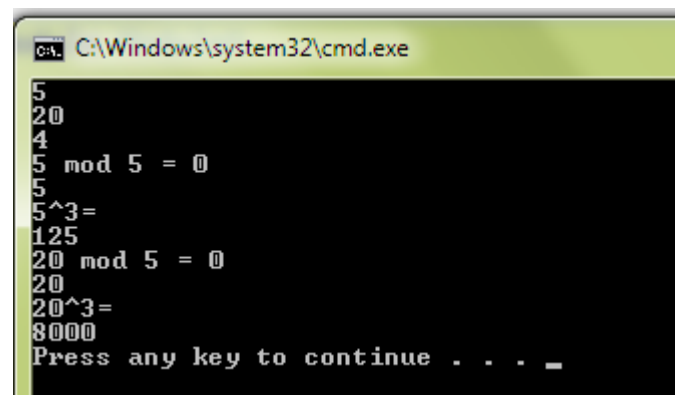
```
k = Console.ReadLine()
```

عبارت توضیحی

حل تمرین ۲ از تمرینات صفحه ۸۶:

۲- منطق برنامه‌ای را تعیین نمایید که سه عدد را در ورودی دریافت دارد و تعیین نماید کدامیک از سه عدد مضرب ۵ می‌باشند سپس عدد مضرب ۵ را به‌مراه توان ۳ آن چاپ نماید.

```
Sub Main()  
    Dim a, b, c As Double  
    a = Console.ReadLine()  
    b = Console.ReadLine()  
    c = Console.ReadLine()  
    If a Mod 5 = 0 Then  
        Console.WriteLine(a & " mod 5 = 0")  
        Console.WriteLine(a)  
        Console.WriteLine(a & "^3= ")  
        Console.WriteLine(Math.Pow(a, 3))  
    End If  
    If b Mod 5 = 0 Then  
        Console.WriteLine(b & " mod 5 = 0")  
        Console.WriteLine(b)  
        Console.WriteLine(b & "^3= ")  
        Console.WriteLine(Math.Pow(b, 3))  
    End If  
    If c Mod 5 = 0 Then  
        Console.WriteLine(c & " mod 5 = 0")  
        Console.WriteLine(c)  
        Console.WriteLine(c & "^3= ")  
        Console.WriteLine(Math.Pow(c, 3))  
    End If  
End Sub
```



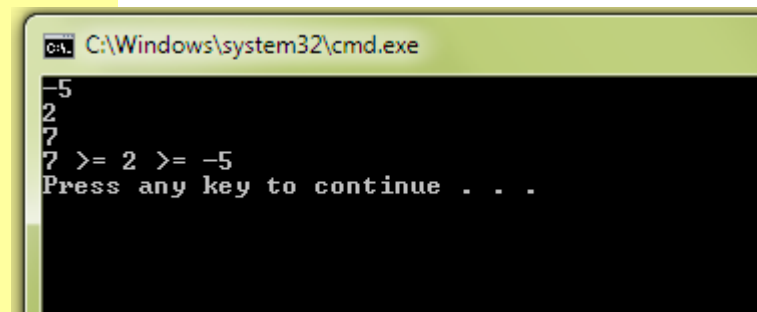
```
C:\Windows\system32\cmd.exe  
5  
20  
4  
5 mod 5 = 0  
5  
5^3=  
125  
20 mod 5 = 0  
20  
20^3=  
8000  
Press any key to continue . . . _
```

نمونه خروجی برنامه

حل تمرین ۳ از تمرینات صفحه ۸۶:

۳- منطق برنامه‌ای را طراحی نمایید که سه عدد را در ورودی دریافت دارد، سپس آنها را به صورت نزولی (از بزرگ به کوچک) نمایش دهد.

```
Sub Main()  
    Dim a, b, c, n As Double  
    a = Console.ReadLine()  
    b = Console.ReadLine()  
    c = Console.ReadLine()  
    'find min  
    If b < a Then  
        'swap a and b  
        n = b  
        b = a  
        a = n  
    End If  
    If c < a Then  
        'swap a and c  
        n = c : c = a : a = n  
    End If  
    'find max  
    If b > c Then  
        'swap b and c  
        n = b : b = c : c = n  
    End If  
    Console.WriteLine(c & " >= " & b & " >= " & a)  
End Sub
```



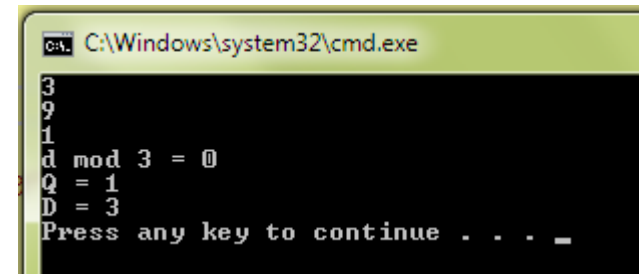
```
C:\Windows\system32\cmd.exe  
-5  
2  
7  
7 >= 2 >= -5  
Press any key to continue . . .
```

نمونه خروجی برنامه

حل تمرین ۸ از تمرینات صفحه ۸۶:

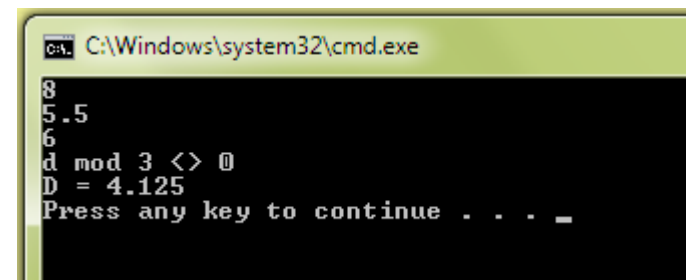
۸- منطق برنامه‌ای را طراحی نمایید که ۳ جزء یک تناسب را بخواند و جزء چهارم تناسب را محاسبه نماید و اگر جزء چهارم تناسب مضرب ۳ است خارج قسمت تقسیم جزء چهارم بر ۳ را محاسبه نماید و به همراه جزء چهارم چاپ نماید، در غیر این صورت چهار جزء تناسب را چاپ نماید.

```
Sub Main()  
    Dim a, b, c, d, q As Double  
    a = Console.ReadLine()  
    b = Console.ReadLine()  
    c = Console.ReadLine()  
    d = (b * c) / a  
    If d Mod 3 = 0 Then  
        Console.WriteLine("d mod 3 = 0")  
        q = Int(d / 3)  
        Console.WriteLine("Q = " & q)  
        Console.WriteLine("D = " & d)  
    Else  
        Console.WriteLine("d mod 3 <> 0")  
        Console.WriteLine("D = " & d)  
    End If  
End Sub
```



```
C:\Windows\system32\cmd.exe  
3  
9  
1  
d mod 3 = 0  
Q = 1  
D = 3  
Press any key to continue . . . _
```

نمونه ۱ از خروجی برنامه



```
C:\Windows\system32\cmd.exe  
8  
5.5  
6  
d mod 3 <> 0  
D = 4.125  
Press any key to continue . . . _
```

نمونه ۲ از خروجی برنامه

حلقه ها (کنترل کنترل برنامه):

(۱) حلقه ی for ... next

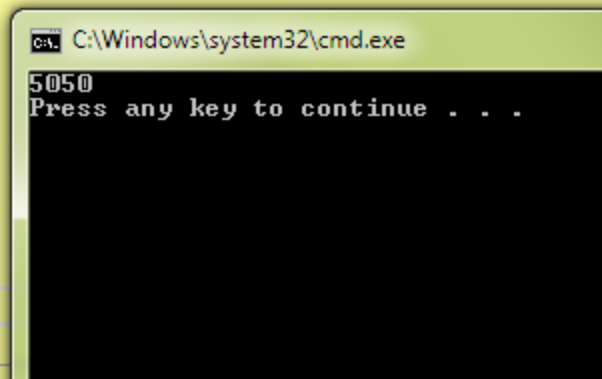
```
For counter = start To end [Step increment]
    ' statements
Next [counter]
```

عبارات داخل براکت
یعنی اختیاری

یکی از راه های ایجاد حلقه استفاده از عبارت for ... next است که در VB به صورت فرمت بالا نوشته می شود. زمانی که کنترل برنامه به خط عبارت for میرسد مقدار start را در متغیر counter قرار داده بعد بررسی میکند که آیا شرط $counter \leq end$ صحیح است یا نه در صورت صحیح بودن آن: خطوط بعد از خط for را اجرا میکند تا اینکه به خط next برسد حال به اندازه ی متغیر step به counter می افزاید و دوباره به خط for برمیگردد و حلقه ادامه می یابد. در صورت غلط بودن آن: کنترل برنامه به خط بعد از next میرود.

مثال: برنامه ای بنویسید که مجموع اعداد یک تا ۱۰۰ را نمایش دهد.

```
1 Module Module1
2
3 Sub Main()
4     Dim i As Integer
5     Dim s As Integer
6     For i = 1 To 100 Step 1
7         s = s + i
8     Next i
9     Console.WriteLine(s)
10 End Sub
11
12 End Module
13
```



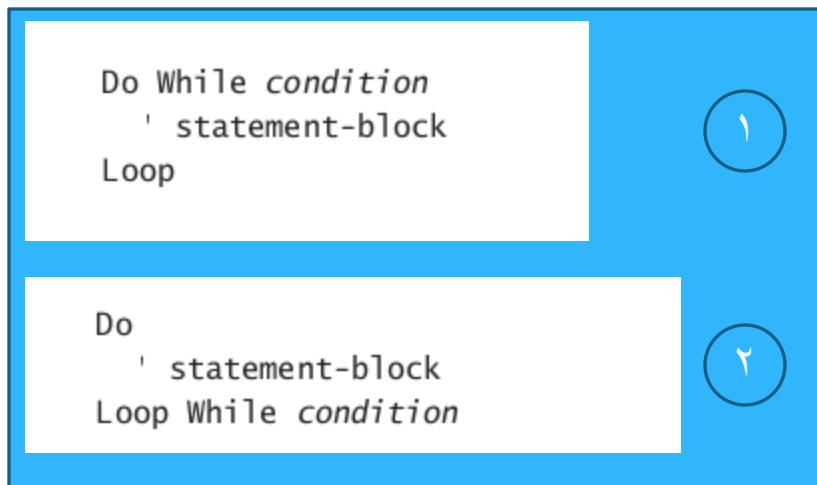
```
C:\Windows\system32\cmd.exe
5050
Press any key to continue . . .
```

حلقه ها (کنترل کنترل برنامه):

۲) حلقه ی do loop

حلقه ی for next زمانی استفاده می شود که تعداد تکرار حلقه مشخص باشد اما اگر در برنامه ای تعداد تکرار مشخص نیست و مثلاً ما میخواهیم تا زمانی که شرایطی (condition) برقرار است کارهای حلقه انجام شود از حلقه ی do loop استفاده می کنیم.

فرمت های حلقه ی do loop در VB:



While در اینجا یعنی تا وقتی که شرط برقرار است حلقه اجرا شود.

در فرمت ۱ زمانی که کنترل برنامه به عبارت do میرسد شرایط را بررسی می کند و در صورت برقرار بودن شرایط جملات درون حلقه اجرا میشود تا اینکه کنترل برنامه به عبارت loop برسد حال کنترل برنامه دوباره به خط do برگشته و شرایط را بررسی میکند و این روند ادامه می یابد تا اینکه شرایط برقرار نشود در این صورت کنترل برنامه خطوط بعد از خط loop را اجرا می کند.

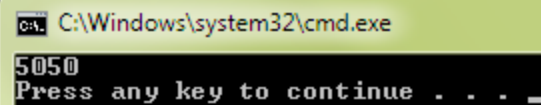
در فرمت ۲ زمانی که کنترل برنامه به عبارت do میرسد جملات درون حلقه اجرا میشود تا اینکه کنترل برنامه به عبارت loop برسد حال کنترل برنامه شرایط را بررسی می کند و در صورت برقرار بودن شرایط کنترل برنامه دوباره به خط do برگشته و جملات درون حلقه را اجرا می کند و این روند ادامه می یابد تا اینکه شرایط برقرار نشود در این صورت کنترل برنامه خطوط بعد از خط loop را اجرا می کند.

حلقه ها (کنترل کنترل برنامه):

مثال: با استفاده از حلقه do loop برنامه ای بنویسید که جمع اعداد یک تا ۱۰۰ را نشان دهد.

حل:

```
1 Module Module1
2
3 Sub Main()
4     Dim i, s As Integer
5     Do
6         i = i + 1
7         s = s + i
8     Loop While i < 100
9     Console.WriteLine(s)
10 End Sub
11
12 End Module
```



C:\Windows\system32\cmd.exe

5050
Press any key to continue . . . _

تمرینات صفحه ۱۷۶ کتاب :

تمرین

- برای هر یک از خواسته‌های زیر ابتدا الگوریتمی بنویسید، سپس روندنمای متناظر با آن را رسم کنید.
- ۱- منطق برنامه‌ای را طراحی نمایید که جمع اعداد چهار رقمی مضرب ۳ را محاسبه و چاپ نماید.
 - ۲- منطق برنامه‌ای را طراحی نمایید که جمع اعداد مضرب ۱۲ و ۲۴ و ۴۸ بزرگ‌تر از ۳۵ و کوچک‌تر از ۱۲۰۰۰ را محاسبه و چاپ نماید.
 - ۳- منطق برنامه‌ای را طراحی نمایید که جمع اعداد مضرب ۷ و ۱۳ کوچک‌تر از ۲۰۰۰ را محاسبه و چاپ نماید.
 - ۴- منطق برنامه‌ای را طراحی نمایید که جمع اعداد پنج رقمی فرد را محاسبه و چاپ نماید.
 - ۵- منطق برنامه‌ای را طراحی نمایید که عدد طبیعی N را بخواند و تعیین نماید که آیا این عدد بر مجموع ارقامش بخش پذیر است یا خیر؟
 - ۶- منطق برنامه‌ای را طراحی نمایید که کلیه اعداد سه رقمی که ارقام یکان و دهگان و صدگان آنها مساوی هستند و هریک از ارقام آنها فرد است را تعیین و چاپ نماید.
 - ۷- منطق برنامه‌ای را طراحی نمایید که ارقام یکان و صدگان اعداد چهار رقمی را با هم جابجا نماید و ارقام دهگان و هزارگان آنان را نیز با هم جابجا نماید و هر عدد را پس از جابجایی ارقامش چاپ نماید.
 - ۸- منطق برنامه‌ای را طراحی نمایید که مقلوب کلیه اعداد سه رقمی فرد را بسازد و چاپ نماید.
 - ۹- منطق برنامه‌ای را طراحی نمایید که کلیه حالت‌های ممکن جهت خرد کردن یک اسکناس هزار تومانی را با اسکناس‌های بالاتر از بیست تومان تعیین و چاپ نماید.
 - ۱۰- منطق برنامه‌ای را طراحی نمایید که با بهره‌گیری از نماد (*) یک مثلث متساوی‌الساقین رسم نماید به گونه‌ای که در رأس آن یک ستاره و در سطرها بعدی به ترتیب ۳، ۵، ۷، ۹ و ۱۱ ستاره رسم شود (یعنی مثلث در ۶ سطر رسم شود).
 - ۱۱- منطق برنامه‌ای را طراحی نمایید که در آمد هفتگی چهار ویزیتور از فروش کالایی را که هر فلم از آن ۸۰۰ ریال سود دارد و سود هر ویزیتور ۶ درصد سود می‌باشد با در نظر گرفتن میزان فروش هر ویزیتور به صورت زیر محاسبه و چاپ نماید:

الف- ویزیتور اول روز اول هفته ۴۸ عدد به فروش می‌رساند و روزهای بعد هر روز ۷ عدد بیشتر از روز قبل می‌فروشد.

- ب- ویزیتور دوم روز اول هفته ۵۵ عدد به فروش می‌رساند و روزهای بعد هر روز ۶ عدد بیشتر از روز قبل می‌فروشد.
- پ- ویزیتور سوم روز اول هفته ۴۲ عدد به فروش می‌رساند و روزهای بعد هر روز ۹ عدد بیشتر از روز قبل می‌فروشد.
- ت- ویزیتور چهارم روز اول هفته ۶۰ عدد به فروش می‌رساند و روزهای بعد هر روز ۱۰ عدد بیشتر از روز قبل می‌فروشد.
- ۱۲- منطق برنامه‌ای را طراحی کنید که کلیه اعداد زوج ۵ رقمی را که فاقد رقم صفر می‌باشند نمایش دهد و در ضمن جمع آنها را محاسبه و چاپ نماید.
- ۱۳- منطق برنامه‌ای را طراحی نمایید که نمرات ۶ درس ۳۰ نفر دانشجویان کلاسی را دریافت دارد و با فرض اینکه ۲ درس اول هر کدام ۳ واحد و سه درس بعدی هر کدام ۲ واحد و درس آخر ۴ واحد ارزش دارد معدل هریک از آنان را محاسبه و چاپ نماید. ضمناً معدل کل کلاس را نیز محاسبه و چاپ نماید.
- ۱۴- منطق برنامه خواسته ۱۳ را به گونه‌ای تغییر دهید که رتبه اول و رتبه آخر کلاس نیز مشخص شود و معدل آنها چاپ شود.
- ۱۵- منطق برنامه‌ای را طراحی نمایید که تعداد اعداد چهار رقمی فرد و زوج را که با مقفولیشان برابرند به تفکیک تعیین و چاپ نماید.
- ۱۶- منطق برنامه‌ای را طراحی نمایید که تعداد مقسوم‌علیه‌های اعداد زوج سه رقمی مضرب سه را به تفکیک تعیین و همراه هر عدد چاپ نماید.
- ۱۷- منطق برنامه‌ای را طراحی نمایید که عدد طبیعی N را در ورودی دریافت دارد سپس تعداد اعداد فرد، زوج، مضرب ۳، مضرب ۵، مضرب ۷ و مضرب ۹ از یک تا N را به تفکیک تعیین و چاپ نماید.
- ۱۸- منطق برنامه‌ای را طراحی نمایید که مجموع فاکتوریل ارقام اعداد زوج دو رقمی را به تفکیک محاسبه و چاپ نماید مثال:

$$۷۲۲ = ۷۲۰ + ۲ = ۶! + ۲! \rightarrow \text{مجموع فاکتوریل ارقام عدد ۲۶}$$

- ۱۹- منطق برنامه‌ای را طراحی نمایید که ۵۰ عدد طبیعی را در ورودی دریافت دارد سپس تعداد ارقام هر کدام از این اعداد را مشخص و چاپ نماید و تعداد اعداد اول در بین این ۵۰ عدد را نیز مشخص و چاپ نماید.
- ۲۰- منطق برنامه‌ای را طراحی نمایید که دو عدد X و N را در ورودی دریافت دارد سپس سری زیر را محاسبه نماید و مقدار آن را چاپ کند.

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots + \frac{x^n}{n!}$$

نکات مهمی که باید قبل از حل تمرینات صفحه ۱۷۶ بلد باشیم

(۱) در حالت کلی داریم:

$$S = \sum_{i=a}^b f(i)$$

معادل S در زبان برنامه نویسی VB

```
for i=a to b  
  s = s + f(i)  
Next i
```

وقتی کنترل برنامه به این محل برسد مقدار اس برابر مجموع است.

مثال: تمرین ۱ ص ۱۷۶

برنامه ای بنویسید که جمع اعداد ۴ رقمی مضرب ۳ را حساب کرده و چاپ نماید.

حل: در این گونه مسائل ابتدا فرمول $\sum_{i=a}^b f(i)$ را رابطه می سازیم

$f(i) = 3i$, $\sum_{i=a}^b 3i$

$3a = 1002 \Rightarrow a = \frac{1002}{3} = 334$

$3b = 9999 \Rightarrow b = \frac{9999}{3} = 3333$

$\Rightarrow$ فرمول = $\sum_{i=334}^{3333} 3i \Rightarrow$

حل مسئله می شود:

۱۰۰۲ اولین عدد ۴ رقمی مضرب ۳ و ۹۹۹۹ هم آخرین عدد ۴ رقمی مضرب بر ۳ است.

برنامه ای بنویسید که جمع اعداد ۴ رقمی مضرب ۳ را حساب کرده و چاپ نماید.

```

1 Module Module1
2
3 Sub Main()
4     Dim i, s As Double
5     For i = 334 To 3333
6         s = s + 3 * i
7     Next
8     Console.WriteLine(s)
9 End Sub
10
11 End Module
  
```

خروجی برنامه

C:\Windows\system32\cmd.exe

16501500

Press any key to continue . . .

(۲) در برخی مسائل نیاز است که تعداد ارقام یک متغیر را پیدا کنیم. تعداد ارقام یک متغیر برابر است با جزء صحیح لوگاریتم در مبنای ۱۰ آن متغیر بعلاوه یک

در نتیجه اگر Tr تعداد ارقام متغیر a باشد داریم:

$$tr = \text{Int}(\text{Math.Log10}(a)) + 1$$

(۳) در برخی مسائل نیاز است که رقم خاصی از یک متغیر را جدا کنیم و در متغیر دیگری بریزیم. در حالت کلی اگر Rn

رقم n اوم از متغیر a باشد داریم:

$$Rn = \text{Int}(a / (10 ^ (n - 1))) \text{ Mod } 10$$

از این فرمول نتیجه می شود:

$$\text{رقم اول} = R1 = a \text{ mod } 10$$

$$\text{رقم دوم} = R2 = \text{int}(a / 10) \text{ mod } 10$$

$$\text{رقم سوم} = R3 = \text{int}(a / 100) \text{ mod } 10$$

$$\text{رقم چهارم} = R4 = \text{int}(a / 1000) \text{ mod } 10$$

...

مثال: تمرین ۵ ص ۱۷۶

برنامه ای بنویسید که عدد طبیعی a را بخواند و تعیین کند که آیا این عدد بر مجموع ارقامش بخش پذیر است یا نه ؟

حل: توضیح: ابتدا باید بدانیم تعداد ارقام عدد a (tr) را تعیین کنیم. بعد در یک حلقه ی `for ... next` که به تعداد ارقام عدد a تکرار میشود باید ارقام عدد a (rn) را جدا کنیم و با استفاده از فرمول $S=S+rn$ مجموع ارقام را بدست آوریم حال خارج حلقه اگر عدد a بر S (مجموع ارقام) بخش پذیر باشد عبارت $a \bmod S = 0$ را چاپ میکنیم و در غیر این صورت عبارت $a \bmod S \neq 0$.

پس داریم:

```
1 Module Module1
2
3 Sub Main()
4     Dim a, rn, tr, s, n As Double
5     a = Console.ReadLine()
6     tr = Int(Math.Log10(a)) + 1
7     For n = 1 To tr
8         rn = Int(a / (10 ^ (n - 1))) Mod 10
9         s = s + rn
10    Next
11    Console.WriteLine("sum= " & s)
12    If a Mod s = 0 Then
13        Console.WriteLine(a & " mod " & s & " = 0")
14    Else
15        Console.WriteLine(a & " mod " & s & " <> 0")
16    End If
17 End Sub
18
19 End Module
```

```
C:\Windows\system32\cmd.exe
12340
sum= 10
12340 mod 10 = 0
Press any key to continue
```

نمونه خروجی وقتی که $a=12340$ وارد شود.

دقت کنیم که متغیر حلقه باید n باشد چرا که ما می خواهیم ارقام را در حلقه جدا کنیم وقتی حلقه اجرا میشود خط ۸ در هر بار اجرای حلقه یکی از ارقام a را جدا میکند ابتدا رقم اول بعد رقم دوم تا رقم آخر که $n=tr$ میشود خط ۹ هم در هر بار اجرای حلقه رقم جدا شده را با مقدار مجموع ارقام جدا شده ی قبل جمع میکند در حقیقت با توجه به نکته ی یک $f(i)$ در اینجا همان rn است در نتیجه خارج حلقه S برابر مجموع ارقام است.

(۴) حلقه های تو در تو:

این نکته را با حل تمرین ۹ توضیح می دهیم:

تمرین ۹ ص ۱۷۶: برنامه ای بنویسید که کلیه ی حالات ممکن برای خرد کردن یک اسکناس ۱۰۰۰ تومانی به اسکناس های ۵۰ تومانی ۱۰۰ تومانی ۲۰۰ تومانی و ۵۰۰ تومانی را چاپ نماید.

← در اینگونه مسائل و در حقیقت در هر مسئله ی برنامه نویسی ابتدا باید مسئله را به شیوه ی ریاضی حل کنیم سپس منطق حل مسئله را با عبارات برنامه نویسی وارد کامپیوتر نمایم. پس!

$$1000 = a \times 50 + b \times 100 + c \times 200 + d \times 500$$

این مسئله در حقیقت از ما می خواهد معادله ی بالا که یک معادله و مجهول دارد را حل کنیم. با تقسیم طرفین بر عدد ۵۰ داریم:

$$a + 2b + 4c + 10d = 20$$

می توانیم محدوده های متغیرهای a, b, c, d را به صورت زیر پیدا کنیم.

$$0 \leq a \leq 20$$

$$0 \leq b \leq 10$$

$$0 \leq c \leq 5$$

$$0 \leq d \leq 2$$

ما برنامه را طوری می نویسیم که کامپیوتر تمامی حالات ممکن برای a و b و c و d و ادر معادله قرار دهد و حالاتی که معادله برقرار باشد را به عنوان جواب معادله جدا کند و نمایش دهد. برای این کار باید از حلقه های تو در تو بهره جست پس داریم:

تمرین ۹ ص ۱۷۶: برنامه ای بنویسید که کلیه ی حالات ممکن برای خرد کردن یک اسکناس ۱۰۰۰ تومانی به اسکناس های ۵۰ تومانی ۱۰۰ تومانی ۲۰۰ تومانی و ۵۰۰ تومانی را چاپ نماید.

```
1 Module Module1
2
3 Sub Main()
4     Dim a, b, c, d As Integer
5     For a = 0 To 20
6         For b = 0 To 10
7             For c = 0 To 5
8                 For d = 0 To 2
9
10                    If a + 2 * b + 4 * c + 10 * d = 20 Then
11                        Console.WriteLine("1000= " & a & "x50 + " & b & "x100 + " &
12                            c & "x200 + " & d & "x500")
13                    End If
14                Next d
15            Next c
16        Next b
17    Next a
18 End Sub
19
20
21 End Module
22
```

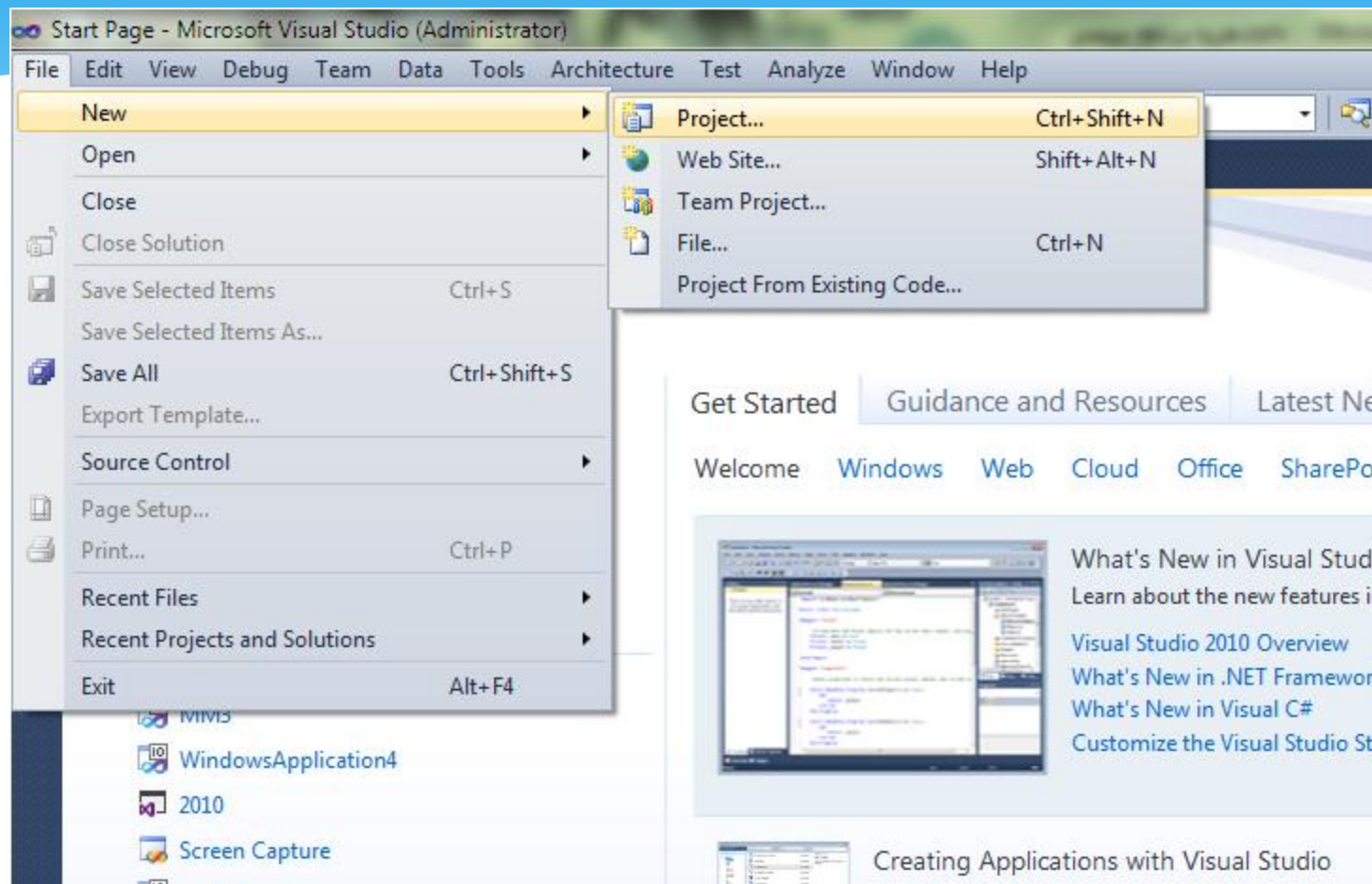
از آندرلاین برای
شکستن یک خط به
دو خط استفاده
میشود

دقت نمایید که در اینجا ۴ حلقه ی تو در تو ی
for ... next داریم و خطوط ۱۰ تا ۱۳ که
بین این حلقه ها قرار گرفته ۲۰*۵*۱۰*۲ مرتبه
اجرا میشود.

```
C:\Windows\system32\cmd.exe
1000= 0x50 + 0x100 + 0x200 + 2x500
1000= 0x50 + 0x100 + 5x200 + 0x500
1000= 0x50 + 1x100 + 2x200 + 1x500
1000= 0x50 + 2x100 + 4x200 + 0x500
1000= 0x50 + 3x100 + 1x200 + 1x500
1000= 0x50 + 4x100 + 3x200 + 0x500
1000= 0x50 + 5x100 + 0x200 + 1x500
1000= 0x50 + 6x100 + 2x200 + 0x500
1000= 0x50 + 8x100 + 1x200 + 0x500
1000= 0x50 + 10x100 + 0x200 + 0x500
1000= 2x50 + 0x100 + 2x200 + 1x500
1000= 2x50 + 1x100 + 4x200 + 0x500
1000= 2x50 + 2x100 + 1x200 + 1x500
1000= 2x50 + 3x100 + 3x200 + 0x500
1000= 2x50 + 4x100 + 0x200 + 1x500
1000= 2x50 + 5x100 + 2x200 + 0x500
1000= 2x50 + 7x100 + 1x200 + 0x500
1000= 2x50 + 9x100 + 0x200 + 0x500
```

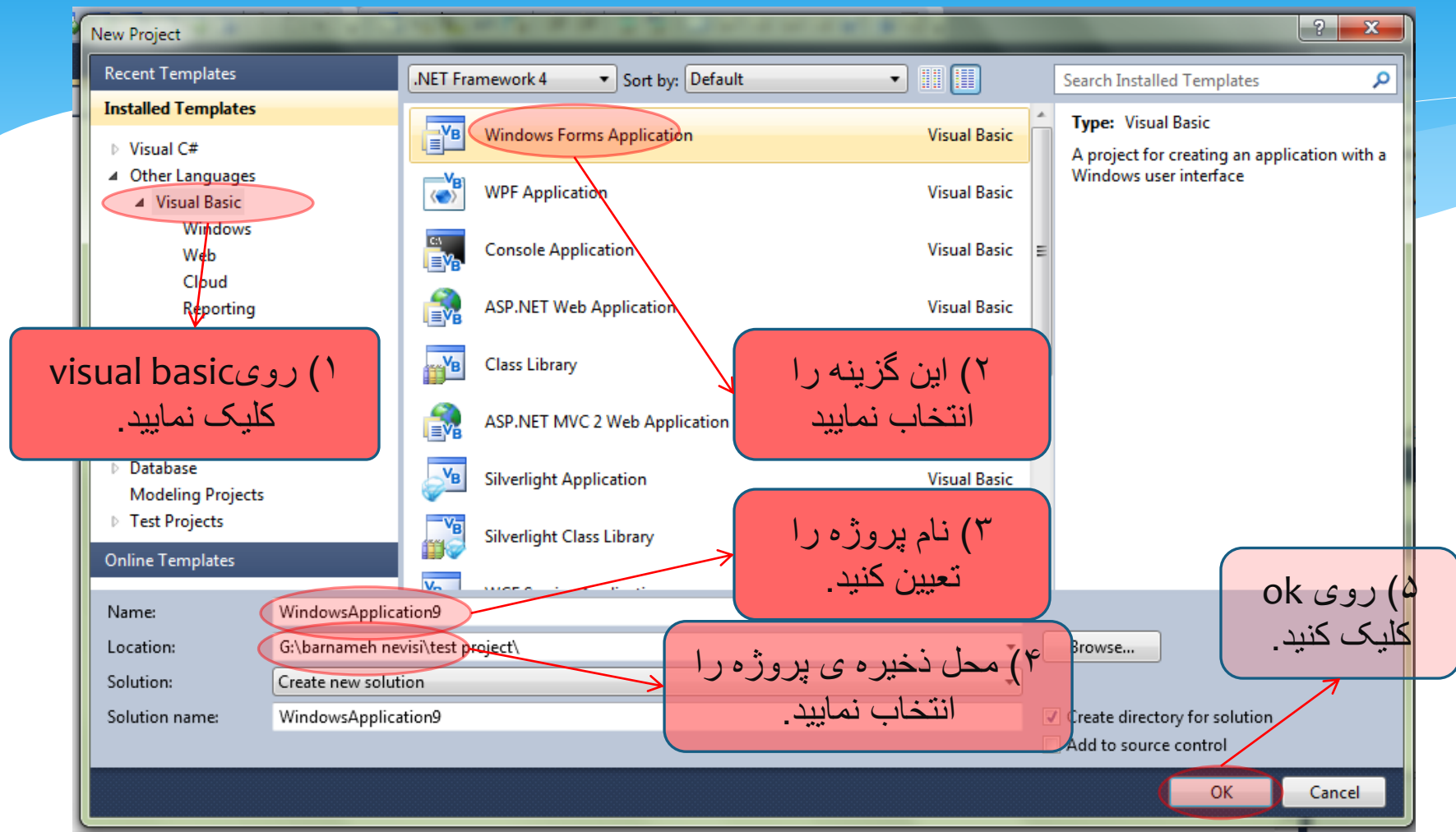
نحوه ی نوشتن برنامه در پلت فرم Windows Form Application:

(۱) بعد از اجرای visual studio از منوی فایل گزینه ی New و بعد گزینه ی Project را انتخاب نمایید. شکل ۱



شکل ۱- انتخاب پروژه ی جدید

نحوه ی نوشتن برنامه در پلت فرم Windows Form Application:
(۲) حال پنجره ی new project باز میشود در این پنجره مطابق شکل ۲ عمل نمایید.



شکل ۲- انتخاب پروژه ی جدید

نحوه ی نوشتن برنامه در پلت فرم Windows Form Application:
(۳) حال وارد محیط Windows Form شده ایم. در این محیط سه پنجره ی مهم داریم که در شکل ۳ مشخص شده است.

پنجره ی toolbox
از این پنجره برای آوردن
کنترل های مختلف در پنجره
ی فرم استفاده می شود.

پنجره ای که نمودار درختی
فایل های برنامه را نشان
می دهد

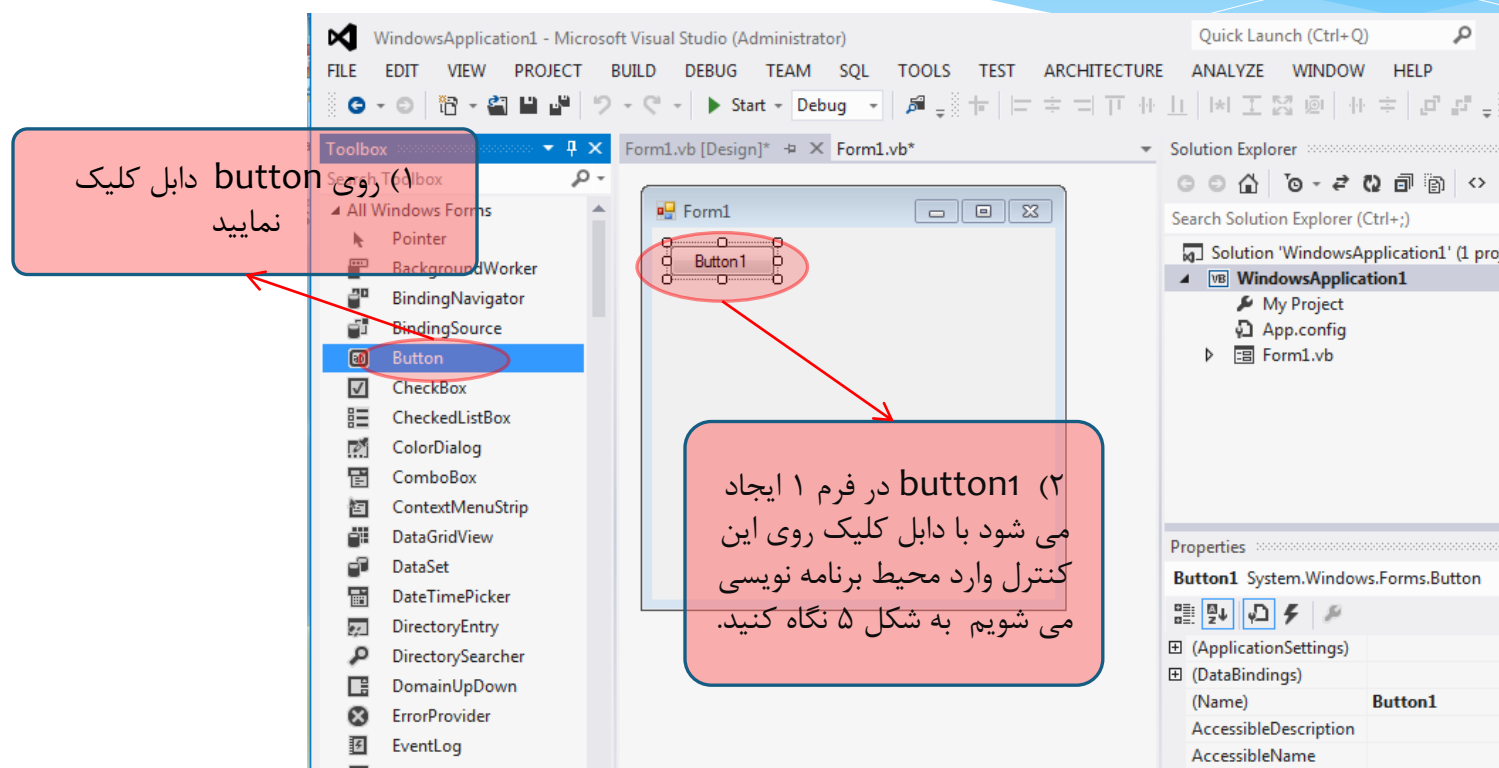
پنجره ی فرم
رابط گرافیکی
کاربر (GUI) یا
پنجره ای که برای
برنامه ی خود
طراحی می کنیم

پنجره ای که خصوصیات هر کنترلی که
در پنجره ی فرم در حالت انتخاب قرار
دارد را نشان می دهد. و می توان آنها
را تغییر داد.

Properties of Form1 (System.Windows.Forms.Form)	
ShowIcon	True
ShowInTaskbar	True
Size	414; 369
SizeGripStyle	Auto
StartPosition	WindowsDefaultLocation
Tag	
Text	Form1
TopMost	False

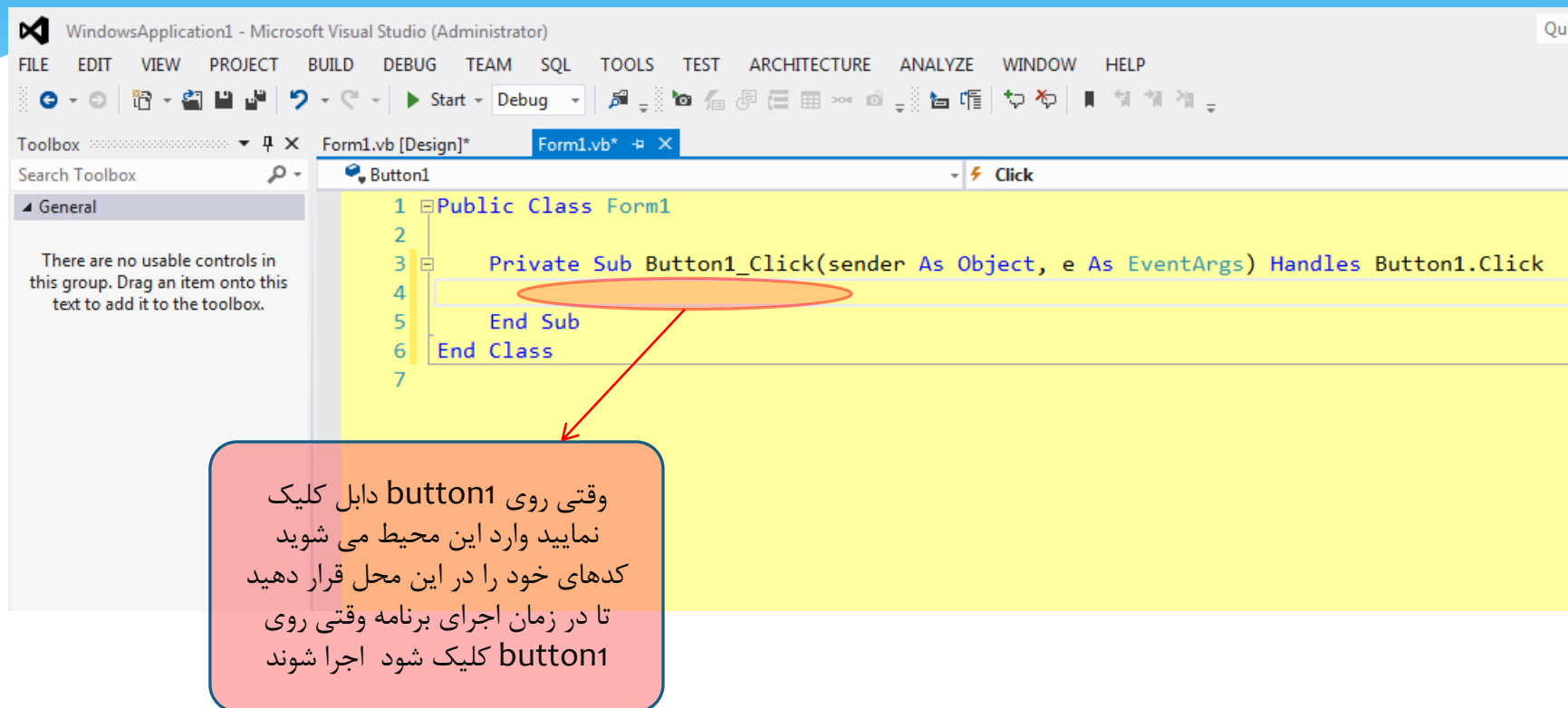
شکل ۳- محیط ویندوز فرم

سوال: در محیط ویندوز فرم کدهای برنامه ی خود را کجا قرار دهیم؟
جواب: چون محیط ویندوز فرم یک محیط رویدادگرا است لذا ساده ترین راه این است که برنامه ی خود را در رویداد کلیک یک دکمه (button) بنویسیم یعنی ساده ترین راه این است که برنامه ی ما با کلیک روی یک button اجرا شود برای این کار از پنجره ی toolbox روی button دابل کلیک نمایید با این کار یک دکمه به نام button1 در فرم ایجاد می شود اگر روی این دکمه (button1) دابل کلیک نمایید وارد محیط کد نویسی می شوید. شکل ۴



شکل ۴- ایجاد button

مطابق شکل ۵ کدهای خود را بین خطوط `private sub button1_click ...` و `end sub` بنویسید. برای اجرای برنامه کلید **f5** را فشار دهید.



شکل ۵- رویداد کلیک `button1`

گرفتن متغیر از ورودی و نمایش متغیر در خروجی در محیط Windows Form Application

(a) گرفتن متغیر از ورودی

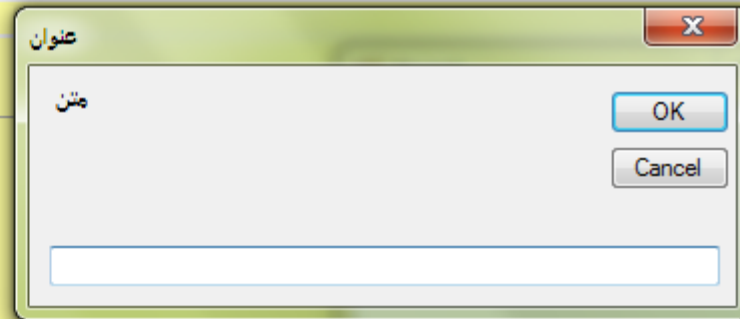
۱- استفاده از دستور `inputbox`:

این دستور را به این صورت می توان نوشت

نمونه ی استفاده از `inputbox` در شکل زیر می بینید.

("عنوان" , "متن") = `inputbox` نام متغیر

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4         Dim a As Double
5         a = InputBox("متن", "عنوان")
6     End Sub
7 End Class
8
```



(b) نمایش متغیر در خروجی

۱- استفاده از دستور `msgbox`:

این دستور را به این صورت می توان نوشت

(نام متغیر) `Msgbox`

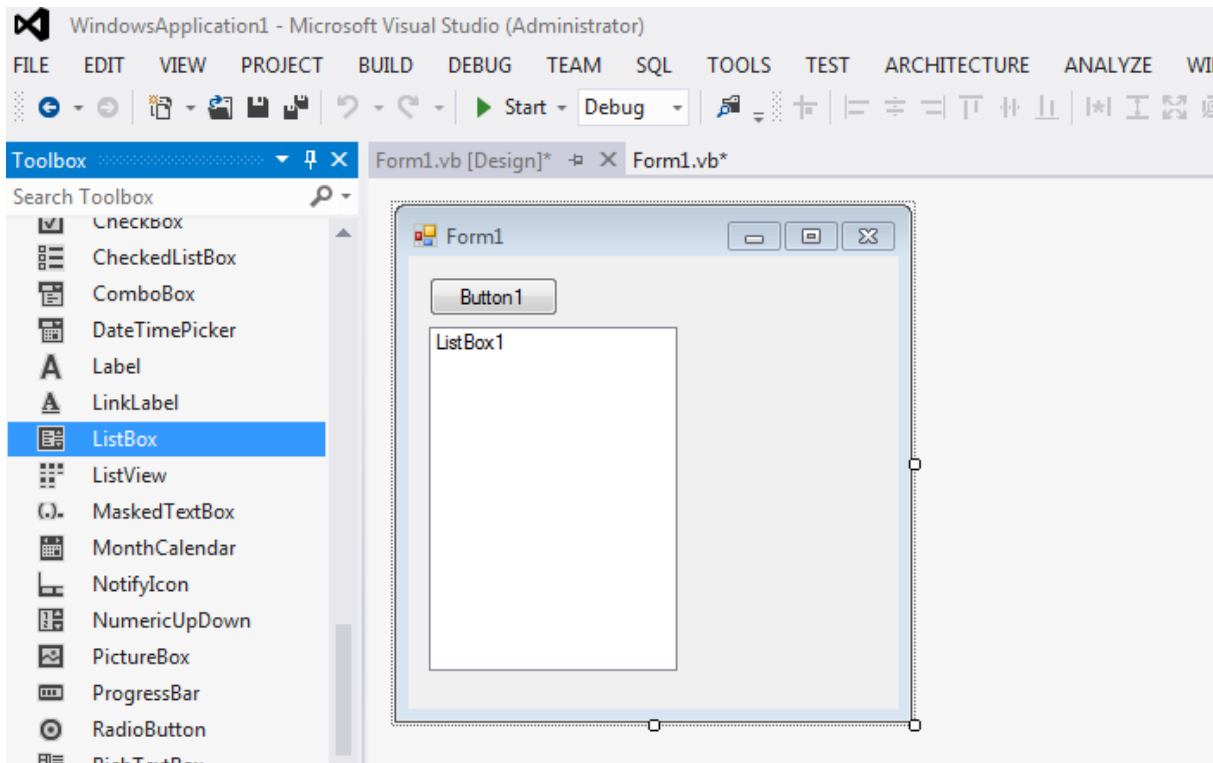
(b) نمایش متغیر در خروجی
۲- استفاده از کنترل `listbox`:

این دستور را به این صورت می توان نوشت

`Listbox1.items.add(نام متغیر)`

`Msgbox` زمانی مناسب است که تعداد خروجی ها کم باشد اگر تعداد خروجی ها زیاد بود از کنترل `listbox` برای نمایش خروجی استفاده می کنیم.

مثال: برنامه ای بنویسید که اعداد ۱ تا ۱۰۰۰ را در خروجی نمایش دهد.

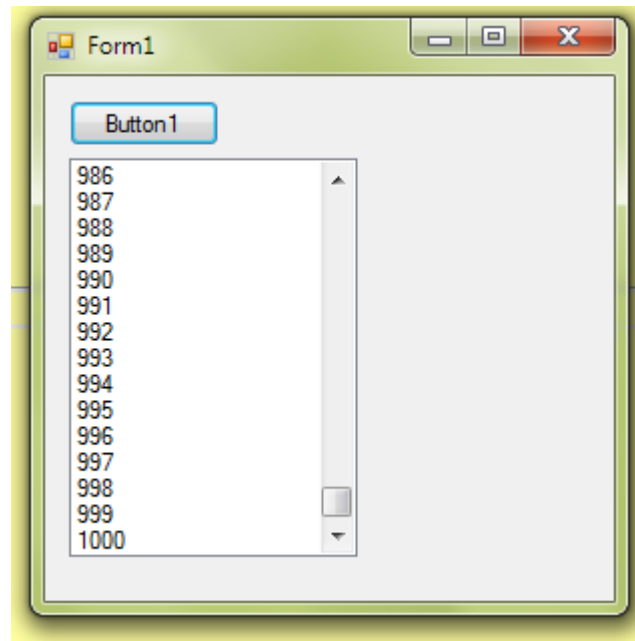


ابتدا یک `button` در فرم ایجاد میکنیم سپس یک `listbox` هم به آن می افزاییم و `button` `listbox` ای که ایجاد کرده ایم را در فرم مرتب میکنیم مانند تصویر حال روی `button1` دابل کلیک کرده و در محیط کدنویسی مانند شکل ۶ برنامه را می نویسیم.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4
5         Dim i As Double
6         For i = 1 To 1000
7             ListBox1.Items.Add(i)
8         Next
9
10    End Sub
11 End Class
12
```

شکل ۶- کدهای برنامه ی نمایش اعداد ۱ تا ۱۰۰۰

برای اجرا کلید f5 را می زنیم حال اگر روی button1 کلیک کنیم اعداد ۱ تا ۱۰۰۰ در listbox لیست خواهد شد. مانند شکل زیر.



نکته:

تمام تمریناتی که در محیط کنسول انجام دادیم را می توانیم به محیط ویندوز فرم هم انتقال دهیم برای این منظور تنها کافی است مانند مثال قبل یک `listbox` و یک `button` در فرم قرار دهیم و روی `button1` دابل کلیک کرده و در محل مربوطه کدهای برنامه ی کنسول مان را کپی کنیم و دو تغییر زیر را انجام دهیم.

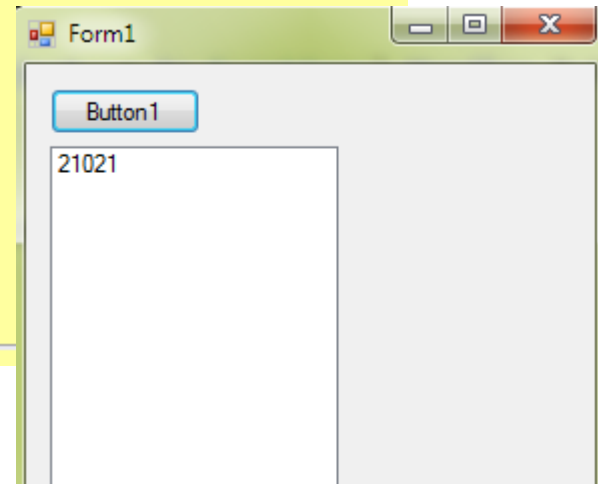
`a = Console.readline()` → تبدیل شود به → `a = inputbox("")`

`Console.WriteLine( a )` → تبدیل شود به → `listbox1.items.add( a)`

ادامه ی تمرینات صفحه ی ۱۷۶:
از این به بعد تمرینات را در محیط ویندوز فرم انجام می دهیم برای تمامی تمرینات یک button و یک listbox ایجاد می کنیم و کدهای خود را در رویداد button1 می نویسیم.

تمرین ۳) برنامه ای بنویسید که جمع اعداد مضرب ۷ و ۱۳ کوچکتر از ۲۰۰۰ را چاپ کند.
حل: یک راه ساده این است که در یک حلقه ی for ... next اعداد کوچکتر از ۲۰۰۰ را بسازیم سپس مضارب ۷ و ۱۳ را از دل آن بیرون بکشیم در این صورت می توانیم مجموع را با فرمول ساده ی $S = S + i$ پیدا کنیم.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.C
4
5         Dim i, s As Double
6         For i = 1 To 2000
7             If i Mod 13 = 0 And i Mod 7 = 0 Then
8                 s = s + i
9             End If
10        Next
11        ListBox1.Items.Add(s)
12
13    End Sub
14 End Class
15
```



در این مکان تنها i هایی می توانند حضور یابند که هم مضرب ۱۳ و ۷ باشند پس در پایان حلقه S برابر مجموع است.

نکته: چون اعداد بخش پذیر بر ۱۳ و ۷ را می خواهیم از کلمه ی and در مقایسه استفاده کرده ایم.

خروجی برنامه

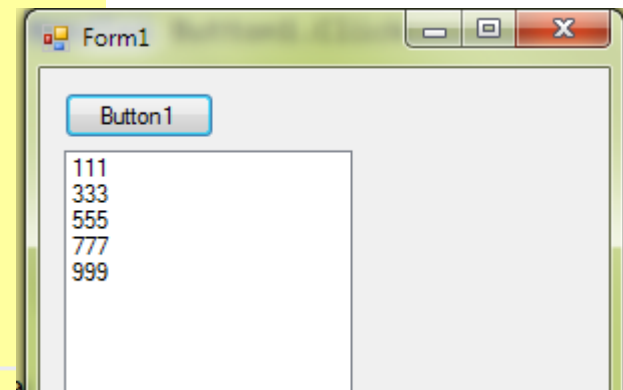
تمرین ۶) برنامه ای بنویسید که اعداد سه رقمی که ارقام یکان و دهگان و صدگان مساوی دارند و ارقامشان فرد است را چاپ کند.

حل: راه ساده این است که ابتدا کلیه ی اعداد سه رقمی را با استفاده از حلقه ی `for ... next` بسازیم بعد با فرمولی که برای جدا کردن ارقام ذکر شد ارقام را جدا کنیم سپس با استفاده از دستور `if` اعدادی که شرایط مسئله را دارند شکار کنیم و در لیست باکس خود لیست نماییم.

در این تمرین متغیر `a3` برای اعداد سه رقمی، `r1` برای رقم یکان، `r2` برای رقم دهگان و `r3` برای رقم صدگان تعریف شده اند.

```
1 Public Class Form1
2
3 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles
4
5     Dim a3, r1, r2, r3 As Double
6     For a3 = 100 To 999
7         ' جدا کردن ارقام
8         r1 = a3 Mod 10
9         r2 = Int(a3 / 10) Mod 10
10        r3 = Int(a3 / 100) Mod 10
11        ' شرط فرد بودن ارقام
12        If r1 Mod 2 <> 0 And r2 Mod 2 <> 0 And r3 Mod 2 <> 0 Then
13            ' شرط برابر بودن ارقام
14            If r1 = r2 And r2 = r3 Then
15                ' تنها ارقامی میتوانند به اینجا راه یابند
16                ' که شرایط مسئله را داشته باشند
17                ListBox1.Items.Add(a3)
18            End If
19        End If
20    Next
21
22
23 End Sub
24 End Class
```

کلمات بعد از کوتیشن به رنگ سبز در می آیند که به معنای عبارات توضیحی هستند و می توانند به هر زبانی باشند این کلمات کمپایل نمی شوند.



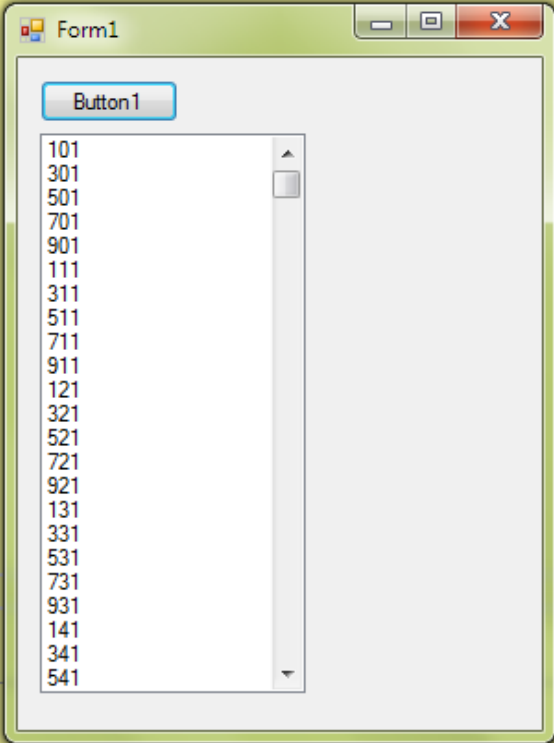
خروجی برنامه

تمرین ۸) برنامه ای بنویسید که مقلوب اعداد سه رقمی فرد را چاپ کند.

حل: توضیح: مقلوب یعنی جای ارقام عدد قرینه شود یعنی مثلاً مقلوب عدد ۱۲۳ می شود ۳۲۱ در این مسئله ما ابتدا کل اعداد سه رقمی را با حلقه ی for ... next می سازیم سپس اعداد فرد را جدا می کنیم و بعد ارقام را جدا می کنیم سپس با فرمول زیر مقلوب را می سازیم.

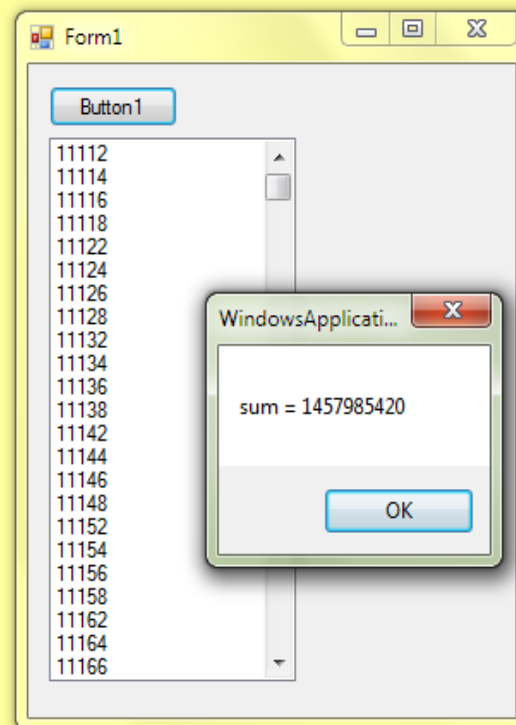
$$r1 * 100 + r2 * 10 + r3 = \text{مقلوب یک عدد سه رقمی} \rightarrow r3 * 100 + r2 * 10 + r1 = \text{یک عدد سه رقمی}$$

```
1 Public Class Form1
2
3 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4
5     Dim a3, r1, r2, r3, ma3 As Double
6     For a3 = 100 To 999
7         ' جدا کردن ارقام فرد
8         If a3 Mod 2 <> 0 Then
9             ' جدا کردن ارقام
10            r1 = a3 Mod 10
11            r2 = Int(a3 / 10) Mod 10
12            r3 = Int(a3 / 100) Mod 10
13            ' ایجاد مقلوب
14            ma3 = r3 + r2 * 10 + r1 * 100
15            ' نمایش
16            ListBox1.Items.Add(ma3)
17        End If
18    Next
19
20
21 End Sub
22 End Class
23
```



تمرین ۱۲) برنامه ای بنویسید که اعداد زوج ۵ رقمی فاقد رقم صفر را چاپ کند مجموع را هم چاپ کند.
حل: کل اعداد ۵ رقمی را با حلقه ی for می سازیم بعد ارقام آنها را جدا می کنیم بعد با دستور if ارقام فاقد رقم صفر را شکار می کنیم حال آنها را نمایش می دهیم مجموع را هم با دستور ساده ی $S = S + \dots$ حساب می کنیم و مسئله را ویلان می اندازیم.

```
Dim a5, r1, r2, r3, r4, r5, s As Double
For a5 = 10000 To 99999
    ' جدا کردن ارقام زوج
    If a5 Mod 2 = 0 Then
        ' جدا کردن ارقام
        r1 = a5 Mod 10
        r2 = Int(a5 / 10) Mod 10
        r3 = Int(a5 / 100) Mod 10
        r4 = Int(a5 / 1000) Mod 10
        r5 = Int(a5 / 10000) Mod 10
        ' شرط فاقد صفر
        If r1 <> 0 And r2 <> 0 And r3 <> 0 And r4 <> 0 And r5 <> 0 Then
            ' نمایش
            ListBox1.Items.Add(a5)
            s = s + a5
        End If
    End If
Next
' نمایش مجموع
MsgBox("sum = " & s)
```

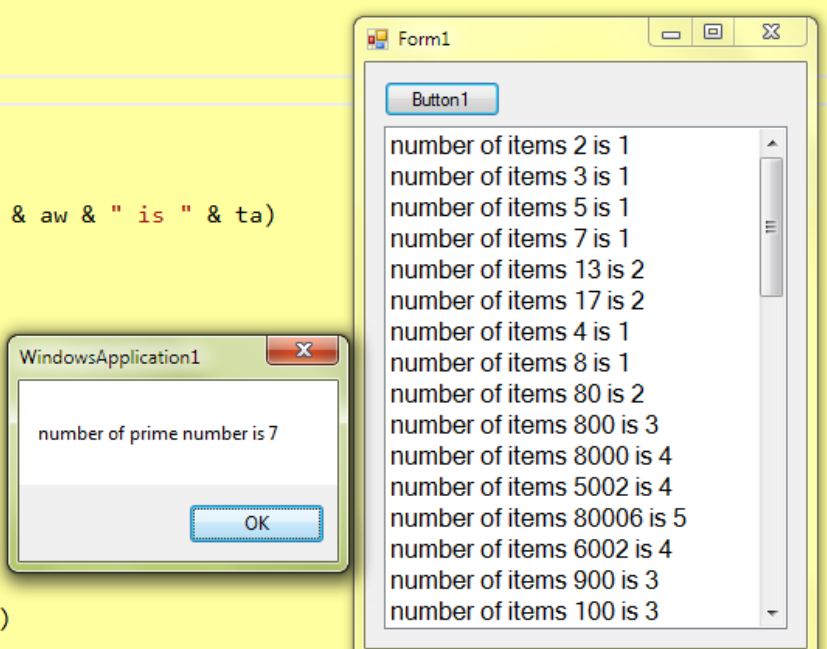


تمرین ۱۹) برنامه ای بنویسید که ۵۰ عدد طبیعی را بگیرد سپس تعداد ارقام هر یک را مشخص و چاپ کند و مشخص کند چه تعداد از این اعداد اول هستند.

حل: همانطور که قبلا گفتیم تعداد ارقام یک عدد برابر جزء صحیح لوگاریتم در مبنای آن عدد بعلاوه ی یک است. عددی اول است که هیچ مقسوم علیه ی بین عدد ۲ تا نصف آن عدد وجود نداشته باشد.
در این تمرین aw برای عدد ورودی، ta برای تعداد ارقام، taa برای تعداد عدد اول و tam هم برای تعداد عدد مرکب در نظر می گیریم.

نکته: برای خروج زود هنگام از حلقه ی for از عبارت exit for استفاده می کنیم. به علت نوشتن exit for در خط ۱۷ فکر کنید.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4
5         Dim aw, ta, taa, tam, i, j As Double
6         'گرفتن ۵۰ عدد'
7         For i = 1 To 50
8             aw = InputBox("input number")
9             'تعیین تعداد ارقام'
10            ta = Int(Math.Log10(aw)) + 1
11            ListBox1.Items.Add("number of items " & aw & " is " & ta)
12            'تعیین تعداد عدد مرکب'
13            If aw > 1 Then
14                For j = 2 To Int(aw / 2)
15                    If aw Mod j = 0 Then
16                        tam = tam + 1
17                        Exit For
18                    End If
19                Next
20            End If
21        Next
22        'تعیین تعداد عدد اول'
23        taa = 50 - tam
24        MsgBox("number of prime number is " & taa)
25    End Sub
```



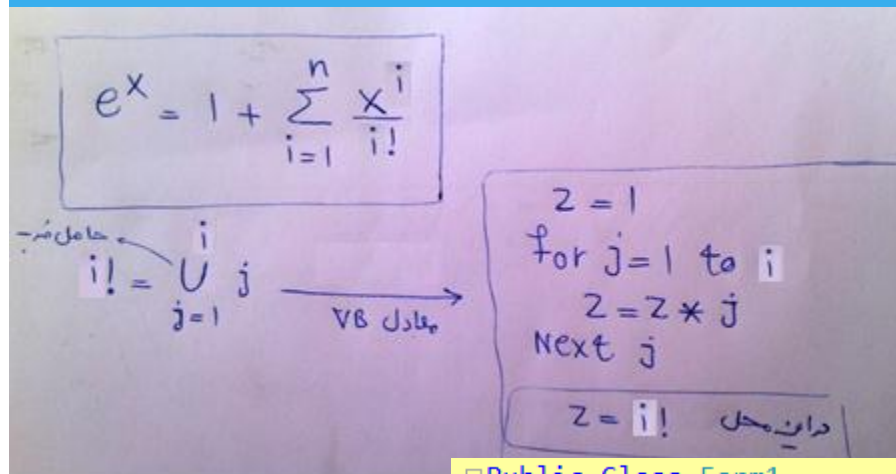
The screenshot shows a Windows application titled 'Form1' with a button labeled 'Button1'. Below the button is a list box containing the following text:

- number of items 2 is 1
- number of items 3 is 1
- number of items 5 is 1
- number of items 7 is 1
- number of items 13 is 2
- number of items 17 is 2
- number of items 4 is 1
- number of items 8 is 1
- number of items 80 is 2
- number of items 800 is 3
- number of items 8000 is 4
- number of items 5002 is 4
- number of items 80006 is 5
- number of items 6002 is 4
- number of items 900 is 3
- number of items 100 is 3

A message box titled 'WindowsApplication1' is displayed, showing the text 'number of prime number is 7' and an 'OK' button.

تمرین ۲۰) برنامه ای بنویسید دو عدد x و n را دریافت کند سپس سری زیر را محاسبه و چاپ نماید.

$$e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots + \frac{x^n}{n!}$$



حل: مطابق فرمول گفته شده برای مجموع عمل می کنیم.

```
Public Class Form1

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles B
        Dim x, n, i, j, s, z As Double
        x = InputBox("input x")
        n = InputBox("input n")
        For i = 1 To n
            ' محاسبه ی فاکتوریل
            z = 1
            For j = 1 To i
                z = z * j
            Next
            ' در اینجا z=i! می شود
            s = s + (x ^ i / z)
        Next
        ' چون فرمول ما بعلاوه یک داشت
        s = s + 1
        ListBox1.Items.Add(s)
    End Sub
End Class
```

The screenshot shows a Windows form titled "Form1" with a button labeled "Button1". Below the button, a text box displays the value "16.375". To the right of the text box, there is a box containing the text "خروجی به ازای x=3 n=4".

آرایه ها:

برای تعریف متغیر از نوع ماتریس از آرایه ها استفاده می شود.
در VB برای تعریف ماتریس یک بعدی می نویسیم:

نوع متغیر as (تعداد عنصر) نام متغیر Dim

مثلا برای تعریف یک ماتریس یک بعدی ۱۰ عضوی از نوع اعشاری داریم

```
Dim a(10) as double
```

حال فرض کنید می خواهیم عنصر ۴ روم از ماتریس a را برابر 3.14 قرار دهیم می نویسیم:

```
a(4)=3.14
```

نکته: در VB با تعریف مثلا a(10) یک ماتریس با ۱۱ عنصر شامل a(0),a(1),a(2) ... a(10) در رم ایجاد می شود
اما ما برای سادگی کار عنصر a(0) را نادیده می گیریم.

تعریف ماتریس دو بعدی، سه بعدی و چند بعدی:

نوع متغیر as (تعداد ستون,تعداد سطر) نام متغیر Dim

نوع متغیر as (تعداد ارتفاع ,تعداد ستون,تعداد سطر) نام متغیر Dim

نوع متغیر as (تعداد بعد آخر , ... ,تعداد بعد دوم,تعداد بعد اول) نام متغیر Dim

تمرین

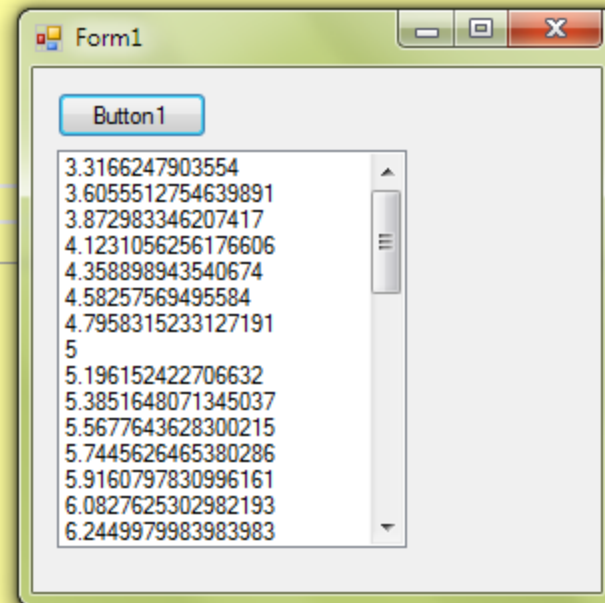
برای هر یک از خواسته‌های زیر ابتدا الگوریتمی بنویسید، سپس روند نمای متناظر آن را رسم کنید.

- ۱- منطق برنامه‌ای را طراحی نمایید که اسامی ۲۵ نفر دانش‌آموز را در یک آرایه ذخیره نماید و سپس آنها نمایش دهد.
- ۲- منطق برنامه‌ای را طراحی نمایید که اعداد فرد سه رقمی را در یک آرایه ذخیره نماید و سپس آنها را نمایش دهد.
- ۳- منطق برنامه‌ای را طراحی نمایید که جذر اعداد فرد دورقمی را در یک آرایه ذخیره نماید و سپس آنها را چاپ کند.
- ۴- منطق برنامه‌ای را طراحی نمایید که ۲۰ عدد را بخواند و در آرایه‌ای ذخیره نماید، سپس فاکتوریل هر یک از آنها را محاسبه نماید و در آرایه دیگری ذخیره نماید.
- ۵- منطق برنامه‌ای را طراحی نمایید که نمرات میان ترم و پایان ترم ۴۰ نفر دانشجویان کلاسی را در ورودی دریافت دارد و مجموع نمرات میان ترم و پایان ترم هر دانشجو را در آرایه‌ای ذخیره نماید، سپس میانگین مجموع نمرات را محاسبه و چاپ نماید.
- ۶- منطق برنامه‌ای را طراحی نمایید که اعداد اول کوچک‌تر از صد را مشخص و در خانه‌ای از آرایه که اندیس آن برابر عدد است ذخیره نماید، و در خانه‌ای از آرایه که عدد اول در آن ذخیره نمی‌شود صفر قرار گیرد. سپس محتویات آرایه را چاپ نماید.
- ۷- منطق برنامه‌ای را طراحی نمایید که اعداد مضرب ۵ کوچک‌تر از ۱۰۰۰ را در آرایه A و اعداد مضرب ۱۰ کوچک‌تر از ۱۰۰۰ را در آرایه B ذخیره نماید، سپس آرایه C حاوی تفاضل عناصر متناظر آرایه‌های A و B را ایجاد و چاپ نماید.
- ۸- منطق برنامه‌ای را طراحی نمایید که بیست عدد را بخواند در آرایه A ذخیره نماید، سپس جای عناصر خانه‌های ۱ تا ۱۰ را با عناصر خانه‌های ۱۱ تا ۲۰ عوض نماید به گونه‌ای که عنصر خانه اول به خانه ۱۱ و عنصر خانه ۱۱ به خانه ۱ منتقل شود و عنصر خانه ۱۰ به خانه ۲۰ و عنصر خانه ۲۰ به خانه ۱۰ منتقل شود.
- ۹- منطق برنامه‌ای را طراحی نمایید که ۲۸ جمله از دنباله فیبوناچی را تولید نماید و در آرایه F ذخیره نماید سپس مربع اعداد تولید شده را در آرایه M ذخیره نماید و نسبت هر عضو از آرایه F به عضو متناظر آن از آرایه M از عضو دوم به بعد را محاسبه و چاپ نماید.
- ۱۰- منطق برنامه‌ای را طراحی نمایید که ۳۲ عدد را بخواند و اگر عدد زوج باشد مربع آن را در آرایه EVEN قرار دهد و اگر عدد فرد باشد جذر آن را در آرایه ODD قرار دهد و میانگین مربعات و جذرها را به تفکیک محاسبه و چاپ نماید.

- ۱۱- منطق برنامه‌ای را طراحی نمایید که نام و نام خانوادگی و نام پدر و آدرس ۲۰۰ نفر اعضای یک شرکت تعاونی را بخواند و در آرایه MEMBER ذخیره نماید، سپس محتوای آرایه را از آخر به اول چاپ نماید.
- ۱۲- منطق برنامه‌ای را طراحی نمایید که ۵۶ عدد را بخواند و مربع آنها را محاسبه نماید و در یک ماتریس 7×8 به صورت ستونی ذخیره نماید.
- ۱۳- منطق برنامه‌ای را طراحی نمایید که ۷۲ عدد را بخواند و فاکتوریل هر عدد را محاسبه نماید و در یک ماتریس 9×8 به صورت سطری ذخیره نماید.
- ۱۴- منطق برنامه‌ای را طراحی نمایید که عدد طبیعی N را در ورودی دریافت دارد، سپس یک ماتریس مربع $N \times N$ به گونه‌ای بسازد که در آن درایه‌های قطری اصلی و فرعی ۱ و بقیه درایه‌ها صفر باشند.
- ۱۵- منطق برنامه‌ای را طراحی نمایید که نام و نام خانوادگی و شماره تلفن ۴۰ نفر کارکنان شرکتی را در ورودی دریافت دارد و در دو آرایه بهم مرتبط و مکمل هم ذخیره نماید، سپس بزرگ‌ترین شماره تلفن را مشخص نماید و به همراه نام و نام خانوادگی دارنده این شماره تلفن چاپ نماید.
- ۱۶- منطق برنامه‌ای را طراحی نمایید که کلیه اعداد زوج دو رقمی را بخواند و به صورت غیر تکراری در آرایه A ذخیره نماید، سپس جذر آنها را محاسبه نماید و در آرایه B ذخیره نماید و مربع هر یک از اعداد را نیز محاسبه نماید و در آرایه C ذخیره نماید. سرانجام مجموع جذرها و مجموع مربعات را محاسبه و چاپ نماید.
- ۱۷- منطق برنامه‌ای را طراحی نمایید که یک ماتریس 20×30 به گونه‌ای بسازد که در هر درایه از آن فاکتوریل عددی که از حاصل ضرب شماره سطر در شماره ستون مربوطه حاصل می‌شود ذخیره شود، سپس ماتریس را چاپ نماید.
- ۱۸- منطق برنامه‌ای را طراحی نمایید که به ازاء مقادیر x از ۸- تا ۶+ و گام $\frac{1}{2}$ (افزایش مرحله‌ای) مقادیر y از ۱۰ معادله $y = 3x^2 + 4x$ محاسبه نماید و x و y متناظر هم را در ستونهای ۱ و ۲ یک ماتریس ذخیره نماید و سپس چاپ کند.
- ۱۹- منطق برنامه‌ای را طراحی نمایید که ۲۴۰ عدد را بخواند و در یک ماتریس 20×12 ذخیره نماید، سپس درایه‌های ستونهای غیر متوالی را با هم جابجا نماید یعنی درایه‌های ستونهای ۱ را با ۳، ۵، ۷، ۹، ۱۱، ۱۳، ۱۵، ۱۷، ۱۹، ۲۱، ۲۳، ۲۵، ۲۷، ۲۹، ۳۱، ۳۳، ۳۵، ۳۷، ۳۹، ۴۱، ۴۳، ۴۵، ۴۷، ۴۹، ۵۱، ۵۳، ۵۵، ۵۷، ۵۹، ۶۱، ۶۳، ۶۵، ۶۷، ۶۹، ۷۱، ۷۳، ۷۵، ۷۷، ۷۹، ۸۱، ۸۳، ۸۵، ۸۷، ۸۹، ۹۱، ۹۳، ۹۵، ۹۷، ۹۹، ۱۰۱، ۱۰۳، ۱۰۵، ۱۰۷، ۱۰۹، ۱۱۱، ۱۱۳، ۱۱۵، ۱۱۷، ۱۱۹، ۱۲۱، ۱۲۳، ۱۲۵، ۱۲۷، ۱۲۹، ۱۳۱، ۱۳۳، ۱۳۵، ۱۳۷، ۱۳۹، ۱۴۱، ۱۴۳، ۱۴۵، ۱۴۷، ۱۴۹، ۱۵۱، ۱۵۳، ۱۵۵، ۱۵۷، ۱۵۹، ۱۶۱، ۱۶۳، ۱۶۵، ۱۶۷، ۱۶۹، ۱۷۱، ۱۷۳، ۱۷۵، ۱۷۷، ۱۷۹، ۱۸۱، ۱۸۳، ۱۸۵، ۱۸۷، ۱۸۹، ۱۹۱، ۱۹۳، ۱۹۵، ۱۹۷، ۱۹۹، ۲۰۱، ۲۰۳، ۲۰۵، ۲۰۷، ۲۰۹، ۲۱۱، ۲۱۳، ۲۱۵، ۲۱۷، ۲۱۹، ۲۲۱، ۲۲۳، ۲۲۵، ۲۲۷، ۲۲۹، ۲۳۱، ۲۳۳، ۲۳۵، ۲۳۷، ۲۳۹، ۲۴۱، ۲۴۳، ۲۴۵، ۲۴۷، ۲۴۹، ۲۵۱، ۲۵۳، ۲۵۵، ۲۵۷، ۲۵۹، ۲۶۱، ۲۶۳، ۲۶۵، ۲۶۷، ۲۶۹، ۲۷۱، ۲۷۳، ۲۷۵، ۲۷۷، ۲۷۹، ۲۸۱، ۲۸۳، ۲۸۵، ۲۸۷، ۲۸۹، ۲۹۱، ۲۹۳، ۲۹۵، ۲۹۷، ۲۹۹، ۳۰۱، ۳۰۳، ۳۰۵، ۳۰۷، ۳۰۹، ۳۱۱، ۳۱۳، ۳۱۵، ۳۱۷، ۳۱۹، ۳۲۱، ۳۲۳، ۳۲۵، ۳۲۷، ۳۲۹، ۳۳۱، ۳۳۳، ۳۳۵، ۳۳۷، ۳۳۹، ۳۴۱، ۳۴۳، ۳۴۵، ۳۴۷، ۳۴۹، ۳۵۱، ۳۵۳، ۳۵۵، ۳۵۷، ۳۵۹، ۳۶۱، ۳۶۳، ۳۶۵، ۳۶۷، ۳۶۹، ۳۷۱، ۳۷۳، ۳۷۵، ۳۷۷، ۳۷۹، ۳۸۱، ۳۸۳، ۳۸۵، ۳۸۷، ۳۸۹، ۳۹۱، ۳۹۳، ۳۹۵، ۳۹۷، ۳۹۹، ۴۰۱، ۴۰۳، ۴۰۵، ۴۰۷، ۴۰۹، ۴۱۱، ۴۱۳، ۴۱۵، ۴۱۷، ۴۱۹، ۴۲۱، ۴۲۳، ۴۲۵، ۴۲۷، ۴۲۹، ۴۳۱، ۴۳۳، ۴۳۵، ۴۳۷، ۴۳۹، ۴۴۱، ۴۴۳، ۴۴۵، ۴۴۷، ۴۴۹، ۴۵۱، ۴۵۳، ۴۵۵، ۴۵۷، ۴۵۹، ۴۶۱، ۴۶۳، ۴۶۵، ۴۶۷، ۴۶۹، ۴۷۱، ۴۷۳، ۴۷۵، ۴۷۷، ۴۷۹، ۴۸۱، ۴۸۳، ۴۸۵، ۴۸۷، ۴۸۹، ۴۹۱، ۴۹۳، ۴۹۵، ۴۹۷، ۴۹۹، ۵۰۱، ۵۰۳، ۵۰۵، ۵۰۷، ۵۰۹، ۵۱۱، ۵۱۳، ۵۱۵، ۵۱۷، ۵۱۹، ۵۲۱، ۵۲۳، ۵۲۵، ۵۲۷، ۵۲۹، ۵۳۱، ۵۳۳، ۵۳۵، ۵۳۷، ۵۳۹، ۵۴۱، ۵۴۳، ۵۴۵، ۵۴۷، ۵۴۹، ۵۵۱، ۵۵۳، ۵۵۵، ۵۵۷، ۵۵۹، ۵۶۱، ۵۶۳، ۵۶۵، ۵۶۷، ۵۶۹، ۵۷۱، ۵۷۳، ۵۷۵، ۵۷۷، ۵۷۹، ۵۸۱، ۵۸۳، ۵۸۵، ۵۸۷، ۵۸۹، ۵۹۱، ۵۹۳، ۵۹۵، ۵۹۷، ۵۹۹، ۶۰۱، ۶۰۳، ۶۰۵، ۶۰۷، ۶۰۹، ۶۱۱، ۶۱۳، ۶۱۵، ۶۱۷، ۶۱۹، ۶۲۱، ۶۲۳، ۶۲۵، ۶۲۷، ۶۲۹، ۶۳۱، ۶۳۳، ۶۳۵، ۶۳۷، ۶۳۹، ۶۴۱، ۶۴۳، ۶۴۵، ۶۴۷، ۶۴۹، ۶۵۱، ۶۵۳، ۶۵۵، ۶۵۷، ۶۵۹، ۶۶۱، ۶۶۳، ۶۶۵، ۶۶۷، ۶۶۹، ۶۷۱، ۶۷۳، ۶۷۵، ۶۷۷، ۶۷۹، ۶۸۱، ۶۸۳، ۶۸۵، ۶۸۷، ۶۸۹، ۶۹۱، ۶۹۳، ۶۹۵، ۶۹۷، ۶۹۹، ۷۰۱، ۷۰۳، ۷۰۵، ۷۰۷، ۷۰۹، ۷۱۱، ۷۱۳، ۷۱۵، ۷۱۷، ۷۱۹، ۷۲۱، ۷۲۳، ۷۲۵، ۷۲۷، ۷۲۹، ۷۳۱، ۷۳۳، ۷۳۵، ۷۳۷، ۷۳۹، ۷۴۱، ۷۴۳، ۷۴۵، ۷۴۷، ۷۴۹، ۷۵۱، ۷۵۳، ۷۵۵، ۷۵۷، ۷۵۹، ۷۶۱، ۷۶۳، ۷۶۵، ۷۶۷، ۷۶۹، ۷۷۱، ۷۷۳، ۷۷۵، ۷۷۷، ۷۷۹، ۷۸۱، ۷۸۳، ۷۸۵، ۷۸۷، ۷۸۹، ۷۹۱، ۷۹۳، ۷۹۵، ۷۹۷، ۷۹۹، ۸۰۱، ۸۰۳، ۸۰۵، ۸۰۷، ۸۰۹، ۸۱۱، ۸۱۳، ۸۱۵، ۸۱۷، ۸۱۹، ۸۲۱، ۸۲۳، ۸۲۵، ۸۲۷، ۸۲۹، ۸۳۱، ۸۳۳، ۸۳۵، ۸۳۷، ۸۳۹، ۸۴۱، ۸۴۳، ۸۴۵، ۸۴۷، ۸۴۹، ۸۵۱، ۸۵۳، ۸۵۵، ۸۵۷، ۸۵۹، ۸۶۱، ۸۶۳، ۸۶۵، ۸۶۷، ۸۶۹، ۸۷۱، ۸۷۳، ۸۷۵، ۸۷۷، ۸۷۹، ۸۸۱، ۸۸۳، ۸۸۵، ۸۸۷، ۸۸۹، ۸۹۱، ۸۹۳، ۸۹۵، ۸۹۷، ۸۹۹، ۹۰۱، ۹۰۳، ۹۰۵، ۹۰۷، ۹۰۹، ۹۱۱، ۹۱۳، ۹۱۵، ۹۱۷، ۹۱۹، ۹۲۱، ۹۲۳، ۹۲۵، ۹۲۷، ۹۲۹، ۹۳۱، ۹۳۳، ۹۳۵، ۹۳۷، ۹۳۹، ۹۴۱، ۹۴۳، ۹۴۵، ۹۴۷، ۹۴۹، ۹۵۱، ۹۵۳، ۹۵۵، ۹۵۷، ۹۵۹، ۹۶۱، ۹۶۳، ۹۶۵، ۹۶۷، ۹۶۹، ۹۷۱، ۹۷۳، ۹۷۵، ۹۷۷، ۹۷۹، ۹۸۱، ۹۸۳، ۹۸۵، ۹۸۷، ۹۸۹، ۹۹۱، ۹۹۳، ۹۹۵، ۹۹۷، ۹۹۹، ۱۰۰۱، ۱۰۰۳، ۱۰۰۵، ۱۰۰۷، ۱۰۰۹، ۱۰۱۱، ۱۰۱۳، ۱۰۱۵، ۱۰۱۷، ۱۰۱۹، ۱۰۲۱، ۱۰۲۳، ۱۰۲۵، ۱۰۲۷، ۱۰۲۹، ۱۰۳۱، ۱۰۳۳، ۱۰۳۵، ۱۰۳۷، ۱۰۳۹، ۱۰۴۱، ۱۰۴۳، ۱۰۴۵، ۱۰۴۷، ۱۰۴۹، ۱۰۵۱، ۱۰۵۳، ۱۰۵۵، ۱۰۵۷، ۱۰۵۹، ۱۰۶۱، ۱۰۶۳، ۱۰۶۵، ۱۰۶۷، ۱۰۶۹، ۱۰۷۱، ۱۰۷۳، ۱۰۷۵، ۱۰۷۷، ۱۰۷۹، ۱۰۸۱، ۱۰۸۳، ۱۰۸۵، ۱۰۸۷، ۱۰۸۹، ۱۰۹۱، ۱۰۹۳، ۱۰۹۵، ۱۰۹۷، ۱۰۹۹، ۱۱۰۱، ۱۱۰۳، ۱۱۰۵، ۱۱۰۷، ۱۱۰۹، ۱۱۱۱، ۱۱۱۳، ۱۱۱۵، ۱۱۱۷، ۱۱۱۹، ۱۱۲۱، ۱۱۲۳، ۱۱۲۵، ۱۱۲۷، ۱۱۲۹، ۱۱۳۱، ۱۱۳۳، ۱۱۳۵، ۱۱۳۷، ۱۱۳۹، ۱۱۴۱، ۱۱۴۳، ۱۱۴۵، ۱۱۴۷، ۱۱۴۹، ۱۱۵۱، ۱۱۵۳، ۱۱۵۵، ۱۱۵۷، ۱۱۵۹، ۱۱۶۱، ۱۱۶۳، ۱۱۶۵، ۱۱۶۷، ۱۱۶۹، ۱۱۷۱، ۱۱۷۳، ۱۱۷۵، ۱۱۷۷، ۱۱۷۹، ۱۱۸۱، ۱۱۸۳، ۱۱۸۵، ۱۱۸۷، ۱۱۸۹، ۱۱۹۱، ۱۱۹۳، ۱۱۹۵، ۱۱۹۷، ۱۱۹۹، ۱۲۰۱، ۱۲۰۳، ۱۲۰۵، ۱۲۰۷، ۱۲۰۹، ۱۲۱۱، ۱۲۱۳، ۱۲۱۵، ۱۲۱۷، ۱۲۱۹، ۱۲۲۱، ۱۲۲۳، ۱۲۲۵، ۱۲۲۷، ۱۲۲۹، ۱۲۳۱، ۱۲۳۳، ۱۲۳۵، ۱۲۳۷، ۱۲۳۹، ۱۲۴۱، ۱۲۴۳، ۱۲۴۵، ۱۲۴۷، ۱۲۴۹، ۱۲۵۱، ۱۲۵۳، ۱۲۵۵، ۱۲۵۷، ۱۲۵۹، ۱۲۶۱، ۱۲۶۳، ۱۲۶۵، ۱۲۶۷، ۱۲۶۹، ۱۲۷۱، ۱۲۷۳، ۱۲۷۵، ۱۲۷۷، ۱۲۷۹، ۱۲۸۱، ۱۲۸۳، ۱۲۸۵، ۱۲۸۷، ۱۲۸۹، ۱۲۹۱، ۱۲۹۳، ۱۲۹۵، ۱۲۹۷، ۱۲۹۹، ۱۳۰۱، ۱۳۰۳، ۱۳۰۵، ۱۳۰۷، ۱۳۰۹، ۱۳۱۱، ۱۳۱۳، ۱۳۱۵، ۱۳۱۷، ۱۳۱۹، ۱۳۲۱، ۱۳۲۳، ۱۳۲۵، ۱۳۲۷، ۱۳۲۹، ۱۳۳۱، ۱۳۳۳، ۱۳۳۵، ۱۳۳۷، ۱۳۳۹، ۱۳۴۱، ۱۳۴۳، ۱۳۴۵، ۱۳۴۷، ۱۳۴۹، ۱۳۵۱، ۱۳۵۳، ۱۳۵۵، ۱۳۵۷، ۱۳۵۹، ۱۳۶۱، ۱۳۶۳، ۱۳۶۵، ۱۳۶۷، ۱۳۶۹، ۱۳۷۱، ۱۳۷۳، ۱۳۷۵، ۱۳۷۷، ۱۳۷۹، ۱۳۸۱، ۱۳۸۳، ۱۳۸۵، ۱۳۸۷، ۱۳۸۹، ۱۳۹۱، ۱۳۹۳، ۱۳۹۵، ۱۳۹۷، ۱۳۹۹، ۱۴۰۱، ۱۴۰۳، ۱۴۰۵، ۱۴۰۷، ۱۴۰۹، ۱۴۱۱، ۱۴۱۳، ۱۴۱۵، ۱۴۱۷، ۱۴۱۹، ۱۴۲۱، ۱۴۲۳، ۱۴۲۵، ۱۴۲۷، ۱۴۲۹، ۱۴۳۱، ۱۴۳۳، ۱۴۳۵، ۱۴۳۷، ۱۴۳۹، ۱۴۴۱، ۱۴۴۳، ۱۴۴۵، ۱۴۴۷، ۱۴۴۹، ۱۴۵۱، ۱۴۵۳، ۱۴۵۵، ۱۴۵۷، ۱۴۵۹، ۱۴۶۱، ۱۴۶۳، ۱۴۶۵، ۱۴۶۷، ۱۴۶۹، ۱۴۷۱، ۱۴۷۳، ۱۴۷۵، ۱۴۷۷، ۱۴۷۹، ۱۴۸۱، ۱۴۸۳، ۱۴۸۵، ۱۴۸۷، ۱۴۸۹، ۱۴۹۱، ۱۴۹۳، ۱۴۹۵، ۱۴۹۷، ۱۴۹۹، ۱۵۰۱، ۱۵۰۳، ۱۵۰۵، ۱۵۰۷، ۱۵۰۹، ۱۵۱۱، ۱۵۱۳، ۱۵۱۵، ۱۵۱۷، ۱۵۱۹، ۱۵۲۱، ۱۵۲۳، ۱۵۲۵، ۱۵۲۷، ۱۵۲۹، ۱۵۳۱، ۱۵۳۳، ۱۵۳۵، ۱۵۳۷، ۱۵۳۹، ۱۵۴۱، ۱۵۴۳، ۱۵۴۵، ۱۵۴۷، ۱۵۴۹، ۱۵۵۱، ۱۵۵۳، ۱۵۵۵، ۱۵۵۷، ۱۵۵۹، ۱۵۶۱، ۱۵۶۳، ۱۵۶۵، ۱۵۶۷، ۱۵۶۹، ۱۵۷۱، ۱۵۷۳، ۱۵۷۵، ۱۵۷۷، ۱۵۷۹، ۱۵۸۱، ۱۵۸۳، ۱۵۸۵، ۱۵۸۷، ۱۵۸۹، ۱۵۹۱، ۱۵۹۳، ۱۵۹۵، ۱۵۹۷، ۱۵۹۹، ۱۶۰۱، ۱۶۰۳، ۱۶۰۵، ۱۶۰۷، ۱۶۰۹، ۱۶۱۱، ۱۶۱۳، ۱۶۱۵، ۱۶۱۷، ۱۶۱۹، ۱۶۲۱، ۱۶۲۳، ۱۶۲۵، ۱۶۲۷، ۱۶۲۹، ۱۶۳۱، ۱۶۳۳، ۱۶۳۵، ۱۶۳۷، ۱۶۳۹، ۱۶۴۱، ۱۶۴۳، ۱۶۴۵، ۱۶۴۷، ۱۶۴۹، ۱۶۵۱، ۱۶۵۳، ۱۶۵۵، ۱۶۵۷، ۱۶۵۹، ۱۶۶۱، ۱۶۶۳، ۱۶۶۵، ۱۶۶۷، ۱۶۶۹، ۱۶۷۱، ۱۶۷۳، ۱۶۷۵، ۱۶۷۷، ۱۶۷۹، ۱۶۸۱، ۱۶۸۳، ۱۶۸۵، ۱۶۸۷، ۱۶۸۹، ۱۶۹۱، ۱۶۹۳، ۱۶۹۵، ۱۶۹۷، ۱۶۹۹، ۱۷۰۱، ۱۷۰۳، ۱۷۰۵، ۱۷۰۷، ۱۷۰۹، ۱۷۱۱، ۱۷۱۳، ۱۷۱۵، ۱۷۱۷، ۱۷۱۹، ۱۷۲۱، ۱۷۲۳، ۱۷۲۵، ۱۷۲۷، ۱۷۲۹، ۱۷۳۱، ۱۷۳۳، ۱۷۳۵، ۱۷۳۷، ۱۷۳۹، ۱۷۴۱، ۱۷۴۳، ۱۷۴۵، ۱۷۴۷، ۱۷۴۹، ۱۷۵۱، ۱۷۵۳، ۱۷۵۵، ۱۷۵۷، ۱۷۵۹، ۱۷۶۱، ۱۷۶۳، ۱۷۶۵، ۱۷۶۷، ۱۷۶۹، ۱۷۷۱، ۱۷۷۳، ۱۷۷۵، ۱۷۷۷، ۱۷۷۹، ۱۷۸۱، ۱۷۸۳، ۱۷۸۵، ۱۷۸۷، ۱۷۸۹، ۱۷۹۱، ۱۷۹۳، ۱۷۹۵، ۱۷۹۷، ۱۷۹۹، ۱۸۰۱، ۱۸۰۳، ۱۸۰۵، ۱۸۰۷، ۱۸۰۹، ۱۸۱۱، ۱۸۱۳، ۱۸۱۵، ۱۸۱۷، ۱۸۱۹، ۱۸۲۱، ۱۸۲۳، ۱۸۲۵، ۱۸۲۷، ۱۸۲۹، ۱۸۳۱، ۱۸۳۳، ۱۸۳۵، ۱۸۳۷، ۱۸۳۹، ۱۸۴۱، ۱۸۴۳، ۱۸۴۵، ۱۸۴۷، ۱۸۴۹، ۱۸۵۱، ۱۸۵۳، ۱۸۵۵، ۱۸۵۷، ۱۸۵۹، ۱۸۶۱، ۱۸۶۳، ۱۸۶۵، ۱۸۶۷، ۱۸۶۹، ۱۸۷۱، ۱۸۷۳، ۱۸۷۵، ۱۸۷۷، ۱۸۷۹، ۱۸۸۱، ۱۸۸۳، ۱۸۸۵، ۱۸۸۷، ۱۸۸۹، ۱۸۹۱، ۱۸۹۳، ۱۸۹۵، ۱۸۹۷، ۱۸۹۹، ۱۹۰۱، ۱۹۰۳، ۱۹۰۵، ۱۹۰۷، ۱۹۰۹، ۱۹۱۱، ۱۹۱۳، ۱۹۱۵، ۱۹۱۷، ۱۹۱۹، ۱۹۲۱، ۱۹۲۳، ۱۹۲۵، ۱۹۲۷، ۱۹۲۹، ۱۹۳۱، ۱۹۳۳، ۱۹۳۵، ۱۹۳۷، ۱۹۳۹، ۱۹۴۱، ۱۹۴۳، ۱۹۴۵، ۱۹۴۷، ۱۹۴۹، ۱۹۵۱، ۱۹۵۳، ۱۹۵۵، ۱۹۵۷، ۱۹۵۹، ۱۹۶۱، ۱۹۶۳، ۱۹۶۵، ۱۹۶۷، ۱۹۶۹، ۱۹۷۱، ۱۹۷۳، ۱۹۷۵، ۱۹۷۷، ۱۹۷۹، ۱۹۸۱، ۱۹۸۳، ۱۹۸۵، ۱۹۸۷، ۱۹۸۹، ۱۹۹۱، ۱۹۹۳، ۱۹۹۵، ۱۹۹۷، ۱۹۹۹، ۲۰۰۱، ۲۰۰۳، ۲۰۰۵، ۲۰۰۷، ۲۰۰۹، ۲۰۱۱، ۲۰۱۳، ۲۰۱۵، ۲۰۱۷، ۲۰۱۹، ۲۰۲۱، ۲۰۲۳، ۲۰۲۵، ۲۰۲۷، ۲۰۲۹، ۲۰۳۱، ۲۰۳۳، ۲۰۳۵، ۲۰۳۷، ۲۰۳۹، ۲۰۴۱، ۲۰۴۳، ۲۰۴۵، ۲۰۴۷، ۲۰۴۹، ۲۰۵۱، ۲۰۵۳، ۲۰۵۵، ۲۰۵۷، ۲۰۵۹، ۲۰۶۱، ۲۰۶۳، ۲۰۶۵، ۲۰۶۷، ۲۰۶۹، ۲۰۷۱، ۲۰۷۳، ۲۰۷۵، ۲۰۷۷، ۲۰۷۹، ۲۰۸۱، ۲۰۸۳، ۲۰۸۵، ۲۰۸۷، ۲۰۸۹، ۲۰۹۱، ۲۰۹۳، ۲۰۹۵، ۲۰۹۷، ۲۰۹۹، ۲۱۰۱، ۲۱۰۳، ۲۱۰۵، ۲۱۰۷، ۲۱۰۹، ۲۱۱۱، ۲۱۱۳، ۲۱۱۵، ۲۱۱۷، ۲۱۱۹، ۲۱۲۱، ۲۱۲۳، ۲۱۲۵، ۲۱۲۷، ۲۱۲۹، ۲۱۳۱، ۲۱۳۳، ۲۱۳۵، ۲۱۳۷، ۲۱۳۹، ۲۱۴۱، ۲۱۴۳، ۲۱۴۵، ۲۱۴۷، ۲۱۴۹، ۲۱۵۱، ۲۱۵۳، ۲۱۵۵، ۲۱۵۷، ۲۱۵۹، ۲۱۶۱، ۲۱۶۳، ۲۱۶۵، ۲۱۶۷، ۲۱۶۹، ۲۱۷۱، ۲۱۷۳، ۲۱۷۵، ۲۱۷۷، ۲۱۷۹، ۲۱۸۱، ۲۱۸۳، ۲۱۸۵، ۲۱۸۷، ۲۱۸۹، ۲۱۹۱، ۲۱۹۳، ۲۱۹۵، ۲۱۹۷، ۲۱۹۹، ۲۲۰۱، ۲۲۰۳، ۲۲۰۵، ۲۲۰۷، ۲۲۰۹، ۲۲۱۱، ۲۲۱۳، ۲۲۱۵، ۲۲۱۷، ۲۲۱۹، ۲۲۲۱، ۲۲۲۳، ۲۲۲۵، ۲۲۲۷، ۲۲۲۹، ۲۲۳۱، ۲۲۳۳، ۲۲۳۵، ۲۲۳۷، ۲۲۳۹، ۲۲۴۱، ۲۲۴۳، ۲۲۴۵، ۲۲۴۷، ۲۲۴۹، ۲۲۵۱، ۲۲۵۳، ۲۲۵۵، ۲۲۵۷، ۲۲۵۹، ۲۲۶۱، ۲۲۶۳، ۲۲۶۵، ۲۲۶۷، ۲۲۶۹، ۲۲۷۱، ۲۲۷۳، ۲۲۷۵، ۲۲۷۷، ۲۲۷۹، ۲۲۸۱، ۲۲۸۳، ۲۲۸۵، ۲۲۸۷، ۲۲۸۹، ۲۲۹۱، ۲۲۹۳، ۲۲۹۵، ۲۲۹۷، ۲۲۹۹، ۲۳۰۱، ۲۳۰۳، ۲۳۰۵، ۲۳۰۷، ۲۳۰۹، ۲۳۱۱، ۲۳۱۳، ۲۳۱۵، ۲۳۱۷، ۲۳۱۹، ۲۳۲۱، ۲۳۲۳، ۲۳۲۵، ۲۳۲۷، ۲۳

تمرین ۳) برنامه ای که جذر اعداد دورقمی فرد را در یک آرایه ذخیره و نمایش دهد.
حل: در تعریف آرایه حتما باید تعداد عناصر آرایه مشخص باشد. در این مسئله تعداد عناصر آرایه برابر تعداد اعداد دو رقمی فرد که می شود $90/2 = 45$

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4         Dim a(45), i, j, jaf As Double 'jaf = جذر اعداد فرد
5         For i = 10 To 99
6             If i Mod 2 <> 0 Then
7                 'در اینجا متغیر آی برابر اعداد فرد است
8                 jaf = Math.Sqrt(i)
9                 'متغیر جی را برای دسترسی به عناصر آرایه تعریف کردیم
10                j = j + 1
11                a(j) = jaf
12                ListBox1.Items.Add(a(j))
13            End If
14        Next
15    End Sub
16 End Class
17
```



Form1

Button1

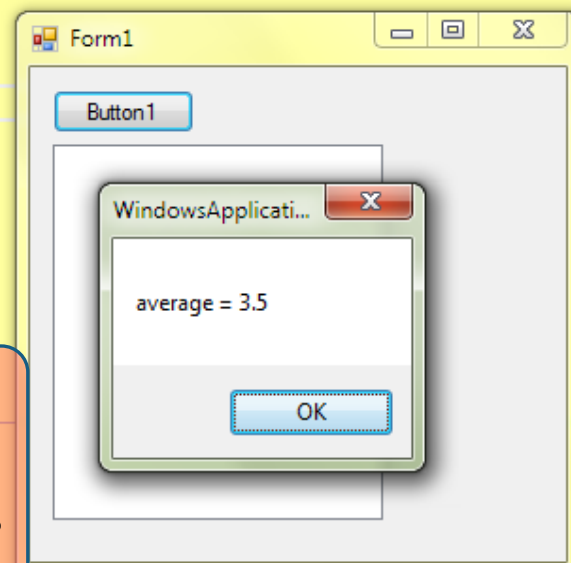
3.3166247903554
3.6055512754639891
3.872983346207417
4.1231056256176606
4.358898943540674
4.58257569495584
4.7958315233127191
5
5.196152422706632
5.3851648071345037
5.5677643628300215
5.7445626465380286
5.9160797830996161
6.0827625302982193
6.2449979983983983

تمرین ۵) برنامه ای که نمرات میان ترم و پایان ترم ۴۰ نفر دانشجویی کلاسی را دریافت کند سپس جمع این دو نمره را برای هر دانشجو در آرایه ای ذخیره کند و میانگین کلاس را نمایش دهد.

حل:

```
1 Public Class Form1
2
3 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4     Dim a(40) As Double
5     Dim nm, np, s, mn As Double 'nm = نمره میانترم, np= نمره یایانترم, mn= میانگین نمرات
6     Dim i As Integer 'برای استفاده در حلقه
7     For i = 1 To 40
8         nm = InputBox("enter nm", "Student " & i)
9         np = InputBox("enter np", "Student " & i)
10        a(i) = np + nm
11        'a(i) در این محل برابر مجموع نمرات دانشجویان است
12        s = s + a(i)
13    Next
14    mn = s / 40
15    MsgBox("average = " & mn)
16 End Sub
17 End Class
18
```

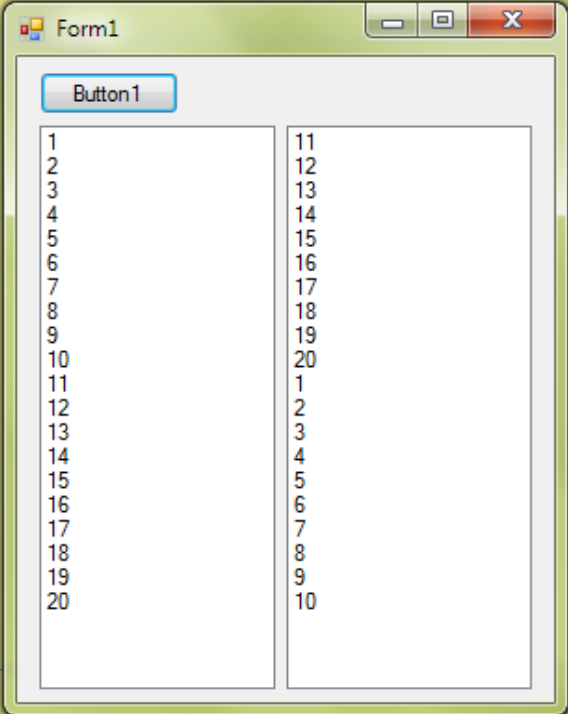
برای اینکه شماره ی دانشجو در عنوان پنجره ی inputbox نشان داده شود این طور نوشتیم می توان برای سادگی این عبارت را حذف کرد.



تمرین ۸) برنامه ای که ۲۰ عدد را دریافت کند در آرایه ای بریزد سپس جای عناصر ۱ تا ۱۰ را با ۱۱ تا ۲۰ عوض نماید به طوری که عنصر ۱ به خانه ی ۱۱ و عنصر ۱۱ به خانه ی ۱ برود تا ... عنصر ۱۰ به خانه ی ۲۰ و عنصر ۲۰ به خانه ۱۰ برود.

حل: برای تست اینکه برنامه درست نوشته شده یا نه دو عدد listbox ایجاد شده در یکی اعداد قبل از جابجایی و در یکی هم اعداد ارایه بعد از جابجایی نمایش داده شده اند. برای حل این گونه مسائل باید فکر کرد به خطوط ۱۳ تا ۱۷ فکر کنید. در خط ۱۳ تعداد تکرار حلقه ی for ۱۰ بار انتخاب شده چون تعداد جابجایی مسئله ۱۰ تا است. شما باید مهارت قرار دادن فرمول مناسب در اشاره ی صحیح به عناصر آرایه در یک حلقه (خطوط ۱۳ تا ۱۷ در این تمرین) را با حل اینگونه تمرین ها کسب کنید.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4         Dim a(20), n As Double 'n برای استفاده در جابجایی
5         Dim i As Integer
6         'گرفتن اعداد
7         For i = 1 To 20
8             a(i) = InputBox("")
9             'نمایش قبل از جابجایی
10            ListBox1.Items.Add(a(i))
11        Next
12        'جابه جایی
13        For i = 1 To 10
14            n = a(i + 10)
15            a(i + 10) = a(i)
16            a(i) = n
17        Next
18        'نمایش بعد از جابجایی
19        For i = 1 To 20
20            ListBox2.Items.Add(a(i))
21        Next
22    End Sub
23 End Class
24
```



ListBox1 (Before)	ListBox2 (After)
1	11
2	12
3	13
4	14
5	15
6	16
7	17
8	18
9	19
10	20
11	1
12	2
13	3
14	4
15	5
16	6
17	7
18	8
19	9
20	10

تمرین ۹) برنامه ای که ۲۸ جمله از دنباله فیبوناچی را بسازد و در آرایه F بریزد سپس مربع اعداد تولیدی را در آرایه M ریخته و نسبت هر عضو از آرایه F به عضو متناظر آن از آرایه M از عضو دوم به بعد را چاپ بنماید.

حل: توضیح: دنباله ی فیبوناچی دنباله ای است که جمله ی اول آن صفر است، جمله ی دوم آن یک و جملات بعدی از جمع دو جمله ی قبل بدست می آید پس:

دنباله فیبوناچی = 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, ...

```

3 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4     Dim F(28), M(28), n As Double, i As Integer
5     'ایجاد جملات فیبوناچی
6     F(1) = 0
7     F(2) = 1
8     For i = 3 To 28
9         F(i) = F(i - 1) + F(i - 2)
10    Next
11    'نمایش جملات
12    For i = 1 To 28
13        ListBox1.Items.Add(F(i))
14    Next
15    'ایجاد توان دوم جملات و نمایش آن
16    For i = 1 To 28
17        M(i) = F(i) ^ 2
18        ListBox2.Items.Add(M(i))
19    Next
20    'ایجاد نسبت و نمایش
21    For i = 2 To 28
22        n = F(i) / M(i)
23        ListBox3.Items.Add(n)
24    Next
25 End Sub
26 End Class
27

```

Form1

Button1

0	0	1
1	1	1
1	1	0.5
2	4	0.333333333333333
3	9	0.2
5	25	0.125
8	64	0.0769230769230769
13	169	0.0476190476190476
21	441	0.0294117647058823
34	1156	0.0181818181818181
55	3025	0.0112359550561797
89	7921	0.00694444444444444
144	20736	0.0042918454935622
233	54289	0.0026525198938992
377	142129	0.0016393442622950
610	372100	0.0010131712259371
987	974169	0.0006261740763932
1597	2550409	0.0003869969040247
2584	6677056	0.0002391772303276
4181	17480761	0.0001478196600147
6765	45765225	9.13575735428467E-5
10946	119814916	5.6462085709446108
17711	313679521	3.489548801339987E-5
28657	821223649	2.1566597653554176
46368	2149991424	1.3328890369876708
75025	5628750625	8.2377072813094656
121393	14736260449	5.0911830891262516
196418	38580030724	

تمرین ۱۳) برنامه ای که ۷۲ عدد را دریافت کند فاکتوریل هر کدام را محاسبه نماید و در یک ماتریس ۹X۸ به صورت سطری ذخیره نماید و نمایش دهد.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4         Dim a(9, 8), n, fac As Double
5         Dim i, j, k As Integer
6         'نمایش سطر آرایه'
7         For i = 1 To 9
8             For j = 1 To 8
9                 'دریافت عدد'
10                n = InputBox("")
11                'فاکتوریل عدد'
12                fac = 1
13                For k = 1 To n
14                    fac = fac * k
15                Next
16                'ذخیره فاکتوریل'
17                a(i, j) = fac
18                ListBox1.Items.Add _
19                ("a(" & i & " , " & j & ") = " & n & " != " & fac)
20            Next
21        Next
22    End Sub
23 End Class
```

برای اینکه فرمت خروجی برنامه
به صورت روبه رو باشد این گونه
نوشته شد.

Form1

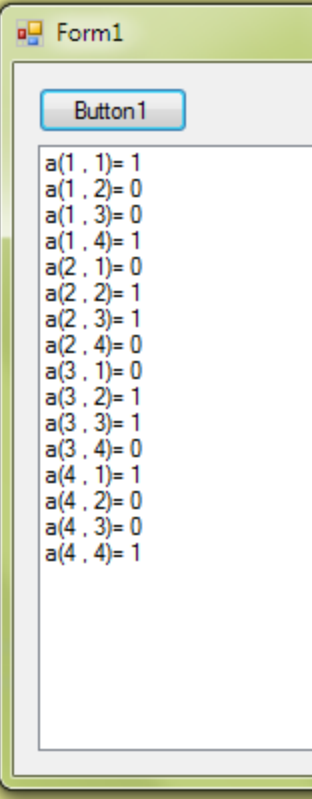
Button1

a(1, 1) = 1 != 1
a(1, 2) = 2 != 2
a(1, 3) = 3 != 6
a(1, 4) = 0 != 1
a(1, 5) = 4 != 24
a(1, 6) = 5 != 120
a(1, 7) = 6 != 720
a(1, 8) = 8 != 40320
a(2, 1) = 7 != 5040
a(2, 2) = 4 != 24
a(2, 3) = 1 != 1
a(2, 4) = 2 != 2
a(2, 5) = 3 != 6
a(2, 6) = 5 != 120
a(2, 7) = 8 != 40320
a(2, 8) = 9 != 362880
a(3, 1) = 10 != 3628800

تمرین ۱۴) برنامه ای که عدد طبیعی n را بگیرد سپس یک ماتریس مربعی $n \times n$ ایجاد کند که قطر اصلی و فرعی آن ۱ و بقیه ی درایه ها صفر باشد.

حل: در برنامه ی زیر خروجی به ازای $n=4$ است.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4         Dim n As Integer
5         'گرفتن n
6         n = InputBox("")
7         Dim a(n, n) As Byte
8         Dim i, j As Byte
9         For i = 1 To n
10             For j = 1 To n
11                 'در قطر فرعی داریم  $i+j=n+1$  , در قطر اصلی داریم  $i=j$ 
12                 If i = j Or (i + j) = (n + 1) Then
13                     a(i, j) = 1
14                 Else
15                     a(i, j) = 0
16                 End If
17                 'نمایش
18                 ListBox1.Items.Add _
19                     ("a(" & i & " , " & j & ")= " & a(i, j))
20             Next
21         Next
22
23     End Sub
24 End Class
```



The screenshot shows a Windows Form titled "Form1" with a button labeled "Button1". Below the button is a list box containing the following output:

```
a(1, 1)= 1
a(1, 2)= 0
a(1, 3)= 0
a(1, 4)= 1
a(2, 1)= 0
a(2, 2)= 1
a(2, 3)= 1
a(2, 4)= 0
a(3, 1)= 0
a(3, 2)= 1
a(3, 3)= 1
a(3, 4)= 0
a(4, 1)= 1
a(4, 2)= 0
a(4, 3)= 0
a(4, 4)= 1
```

۱۹- منطق برنامه‌ای را طراحی نمایید که ۲۴۰ عدد را بخواند و در یک ماتریس 20×12 ذخیره نماید، سپس درایه‌های ستونهای غیر متوالی را با هم جابجا نماید یعنی درایه‌های ستونهای ۱ را با ۳، ۵ را با ۷، ۹ را با ۱۱، ۲ را با ۴، ۶ را با ۸ و ۱۰ را با ۱۲ عوض کند. و سرانجام ماتریس را چاپ نماید.

```

1 Public Class Form1
2
3 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button
4     Dim a(20, 12), n, nm As Double 'nm = نگهداری محل
5     Dim i, j As Integer
6     'خواندن اعداد و ذخیره در ماتریس'
7     For i = 1 To 20
8         For j = 1 To 12
9             n = InputBox("")
10            a(i, j) = n
11        Next
12    Next
13    'جابجایی ستون‌های ۱ و ۵ و ۹'
14    For j = 1 To 12 Step 4
15        For i = 1 To 20
16            nm = a(i, j)
17            a(i, j) = a(i, j + 2)
18            a(i, j + 2) = nm
19        Next
20    Next

```

ادامه ی برنامه

```

21    'جابجایی ستونهای ۲ و ۶ و ۱۰'
22    For j = 2 To 12 Step 4
23        For i = 1 To 20
24            nm = a(i, j)
25            a(i, j) = a(i, j + 2)
26            a(i, j + 2) = nm
27        Next
28    Next
29    'نمایش'
30    For i = 1 To 20
31        For j = 1 To 12
32            ListBox1.Items.Add(a(i, j))
33        Next
34    Next
35 End Sub
36 End Class
37

```

۲۰- منطق برنامه‌ای را طراحی نمایید که یک ماتریس 3×4 ذخیره شده را در یک ماتریس 4×9 ذخیره شده ضرب نماید، سپس ۸ برابر عناصر ماتریس حاصل ضرب را در ماتریس دیگری ذخیره و چاپ نماید.

← برنامه‌ی ضرب ماتریس‌ها :

برای نوشتن برنامه‌ی ضرب ماتریس‌ها باید فرمول کلی آن را استخراج کنیم .

اگر C یک ماتریس به ابعاد $I \times Q$ باشد و a یک ماتریس به ابعاد $I \times J$ و b یک ماتریس به ابعاد $J \times Q$ باشد و داشته باشیم :

$$C = a \times b$$

برای یک درایه از C مثل $C(i, q)$ ماتریس شکل زیر خواهیم داشت :

$$\begin{matrix} a(I, J) & b(J, Q) & c(I, Q) \\ \left[\begin{array}{c} \boxed{\text{عناصر}} \\ \hline \end{array} \right]_{I \times J} & \times \left[\begin{array}{c} \boxed{\text{ستون}} \\ \boxed{J} \\ \hline \end{array} \right]_{J \times Q} & = \left[\begin{array}{c} \boxed{c(i, q)} \\ \hline \end{array} \right]_{I \times Q}
 \end{matrix}$$

$$c(i, q) = \sum_{j=1}^J a(i, j) \times b(j, q)$$

فرمول ضرب ماتریس

for i=1 to I

for q=1 to Q

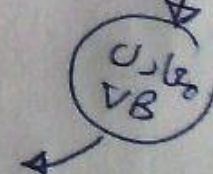
for j=1 to J

$c(i, q) = c(i, q) + a(i, j) \times b(j, q)$

next j

next q

next i



۲۰- منطق برنامه‌ای را طراحی نمایید که یک ماتریس 3×4 ذخیره شده را در یک ماتریس 4×9 ذخیره شده ضرب نماید، سپس ۸ برابر عناصر ماتریس حاصل ضرب را در ماتریس دیگری ذخیره و چاپ نماید.

```
4 Dim i, j, q As Integer
5 Dim a(3, 4), b(4, 9), c(3, 9), d(3, 9) As Double
6 'گرفتن ماتریس a
7 For i = 1 To 3
8     For j = 1 To 4
9         a(i, j) = InputBox("enter a(" & i & " , " & j & ")")
10    Next
11 Next
12 'گرفتن ماتریس b
13 For j = 1 To 4
14     For q = 1 To 9
15         b(j, q) = InputBox("enter b(" & j & " , " & q & ")")
16     Next
17 Next
18 'محاسبه حاصل ضرب
19 For i = 1 To 3
20     For q = 1 To 9
21         For j = 1 To 4
22             c(i, q) = c(i, q) + a(i, j) * b(j, q)
23         Next
24     Next
25 Next
```

```
26 'هشت برابر ماتریس حاصلضرب و نمایش
27 For i = 1 To 3
28     For q = 1 To 9
29         d(i, q) = 8 * c(i, q)
30         ListBox1.Items.Add _
31             ("d(" & i & " , " & q & ") = " & d(i, q))
32     Next
33 Next
```

(سوال ۲ از میانترم سال ۹۵ دکتر)

(۲) برنامه ای که تعداد اعداد زوج بین دو عدد ورودی از کاربر را در یک msgbox نمایش دهد.

```
5      Dim aw1, aw2, i, sh As Integer 'aw= عدد ورودی , sh= شمارنده
6      Dim mn As Integer 'تعریف شده برای جابجایی دو عدد ورودی
7      aw1 = InputBox("")
8      aw2 = InputBox("")
9      If aw2 < aw1 Then
10         'جابجایی دو عدد ورودی
11         mn = aw1
12         aw1 = aw2
13         aw2 = mn
14     End If
15     For i = aw1 To aw2
16         If i Mod 2 = 0 Then
17             sh = sh + 1
18         End If
19     Next
20     MsgBox(sh)
```

توضیح : در این برنامه باید با یک حلقه ی for که از عدد ورودی کوچکتر شروع می شود و در عدد ورودی دوم پایان می یابد تمام اعداد بین دو عدد ورودی را پیمایش کنیم و درون حلقه با شرط if اعداد زوج را جدا کرده و با یک متغیر شمارنده مثلا sh و نوشتن جمله ی $sh = sh + 1$ درون شرط اعداد زوج را بشماریم یادمان باشد که انتهای حلقه متغیر sh برابر تعداد اعداد زوج است. چون حلقه ی for ما باید از عدد کوچکتر به عدد بزرگتر بشمارد در صورتی که کاربر عدد دوم را کوچکتر از عدد اول وارد نماید باید اعداد جابجا شوند که خطوط ۹ تا ۱۴ این کار را انجام می دهند.

(سوال ۳ از میانترم سال ۹۵ دکتر)

۳) برنامه ی که ۱۰ عدد ذخیره شده در فایل به نام a.text در درایو D را در آرایه ی به نام N فراخوانی کرده و مجموع اعداد مضرب ۷ از این اعداد را در یک listbox نمایش دهد.

```
4      ' باز کردن فایل
5      FileOpen(1, "D:\a.txt", OpenMode.Input)
6      Dim n(10), i, s As Integer
7      ' ریختن اعداد در آرایه
8      For i = 1 To 10
9          Input(1, n(i))
10     Next
11     ' بستن فایل
12     FileClose(1)
13     ' محاسبه مجموع اعداد بخشپذیر بر هفت
14     For i = 1 To 10
15         If n(i) Mod 7 = 0 Then
16             s = s + n(i)
17         End If
18     Next
19     ' نمایش
20     ListBox1.Items.Add(s)
```

توضیح : دقت نمایید که فایل باید در مود input باز شود.

(سوال ۴ از میانترم سال ۹۵ دکتر)

۴) برنامه ای که عددی را دریافت کند سپس اولین عدد مربع کامل بعد از آن را یافته به توان عدد ورودی رسانده در msgbox نمایش دهد.

```
4 Dim n As Integer
5 Dim mk As Double 'مربع کامل=mk
6 'گرفتن عدد از ورودی
7 n = Int(InputBox(""))
8 'تعیین اولین عدد مربع کامل بعد عدد ورودی
9 mk = n
10 Do While Math.Sqrt(mk) <> Int(Math.Sqrt(mk))
11     mk = mk + 1
12 Loop
13 MsgBox("mk= " & mk)
14 'نمایش خواسته مسئله
15 MsgBox(mk ^ n)
```

توضیح : شرط مربع کامل این است که جذر عدد صحیح باشد در vb یعنی $\text{math.sqrt}(a)=\text{int}(\text{math.sqrt}(a))$
در این مسئله ما برای تعیین مربع کامل ابتدا عدد ورودی را در mk قرار دادیم سپس در یک حلقه ی do یکی یکی به mk افزودیم اما این افزایش باید تا وقتی که mk مربع کامل نیست ادامه یابد این شرط را باید به صورت خط ۱۰ بنویسیم.

(سوال ۵ از میانترم سال ۹۵ دکتر)

۵) برنامه ای که تعداد نامعینی عدد دریافت کند سپس آنها را در یک آرایه ریخته بزرگترین و کوچکترین آنها را نمایش دهد.

حل: برای اینکه تعداد نامعلومی داده دریافت کنیم نوع داده ی دریافتی را string انتخاب می کنیم و در یک حلقه ی do شرط اتمام حلقه را وارد کردن کلمه ی end توسط کاربر قرار می دهیم درون حلقه داده ها را دریافت می کنیم و یک متغیر شمارنده (در اینجا n) هم برای شمارش تعداد داده قرار می دهیم. چون می خواهیم داده ها را بعدا در یک آرایه بریزیم داده ها را درون یک listbox می ریزیم تا بعدا بتوانیم از انجا داده ها را فراخوانی کرده و درون آرایه بریزیم. برای خواندن یک داده از لیست باکس و قرار دادن آن در یک متغیر مانند a از عبارت (شماره جمله) `a=listbox1.items` استفاده میکنیم و چون شماره ی جمله از صفر شروع می شود در خط ۱۵ عبارت i-1 را در جای شماره جمله قرار دادیم.

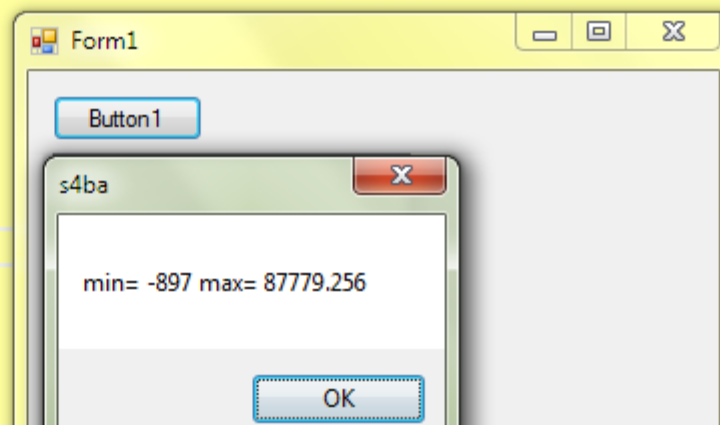
تابع val متغیر رشته ای را به متغیر عددی تبدیل می کند.

```
4 Dim n As Integer
5 Dim st As String
6 'دریافت تعداد نامعلومی داده
7 Do While st <> "end"
8     st = InputBox("")
9     ListBox1.Items.Add(st)
10    If st <> "end" Then n = n + 1
11 Loop
12 'قرار دادن داده ها در آرایه
13 Dim a(n) As Double, i As Integer
14 For i = 1 To n
15     a(i) = Val(ListBox1.Items(i - 1))
16 Next
17 'find min and max
18 Dim min, max As Double
19 min = 999999999
20 For i = 1 To n
21     If a(i) < min Then min = a(i)
22     If a(i) > max Then max = a(i)
23 Next
24 MsgBox("min= " & min & " max= " & max)
```

The screenshot shows a Windows application window titled 'Form1'. Inside the form, there is a button labeled 'Button1'. Below the button is a list box containing the following numbers: 78, 56, 32, 34, 5, 1, -87, 25, and end. A modal message box titled 't5mt' is displayed over the form. The message box contains the text 'min= -87 max= 78' and an 'OK' button.

(سوال ۵ از میانترم بدون ریختن داده ها در آرایه سال ۹۵ دکتر)
(۵) برنامه ای که تعداد نامعینی عدد دریافت کند سپس بزرگترین و کوچکترین آنها را نمایش دهد.
حل: برای اینکه تعداد نامعلومی داده دریافت کنیم نوع داده ی دریافتی را string انتخاب می کنیم و در یک حلقه ی do شرط اتمام حلقه را وارد کردن کلمه ی end توسط کاربر قرار می دهیم درون حلقه داده ها را دریافت می کنیم.

```
4 Dim st As String ' برای دریافت داده
5 Dim a As Double ' برای تبدیل داده ی دریافتی به نوع عددی
6 Dim min, max As Double
7 'مینموم را برابر یک عدد بزرگ قرار می دهیم چرا؟
8 min = 999999999
9 'دریافت تعداد نامعلومی داده
10 Do While st <> "end"
11     st = InputBox("")
12     If st <> "end" Then
13         a = Val(st)
14         'find min and max
15         If a < min Then min = a
16         If a > max Then max = a
17     End If
18 Loop
19 MsgBox("min= " & min & " max= " & max)
```



ب.م.م و ک.م.م

سوال: برنامه ای بنویسید که دو عدد را دریافت کرده و ب.م.م و ک.م.م آنها را چاپ نماید.

حل: برای یافتن ب.م.م دو عدد از روش نردبانی استفاده می کنیم در این روش ما یک دنباله می سازیم به این ترتیب که جمله ی اول برابر عدد بزرگتر جمله ی دوم برابر عدد کوچکتر است و جمله ی سوم به بعد برابر باقی مانده ی تقسیم دو جمله ی قبل است این دنباله در جمله ای به عدد صفر میرسد در این صورت یک جمله قبل از جمله ی صفر برابر ب.م.م است.

برای یافتن ک.م.م هم داریم:

ب.م.م / حاصلضرب دو عدد = ک.م.م

```
4 Dim a, b, c, r, z, bmm, kmm As Integer
5 a = InputBox("") : b = InputBox("")
6 z = a * b
7 'میخواهیم a < b باشد پس
8 If b < a Then 'جابجایی
9     c = b : b = a : a = c
10 End If
11 'تعیین ب.م.م
12 r = 1
13 Do While r <> 0
14     r = b Mod a
15     b = a : a = r
16 Loop
17 bmm = b
18 MsgBox("bmm= " & bmm)
19 'تعیین ک.م.م
20 kmm = z / bmm
21 MsgBox("kmm= " & kmm)
```

قطعه کد تعیین ب.م.م چون حلقه تا زمانی ادامه دارد که ۲ (باقی مانده) مخالف صفر است و چون در ابتدا ۲ صفر است قبل از ورود به حلقه ۲=1 قرار می دهیم. درون حلقه هم بعد از محاسبه ی باقی مانده باید جملات جدید را برای محاسبه ی باقی مانده ایجاد کنیم مطابق خط ۱۵ روی آن فکر کنید.

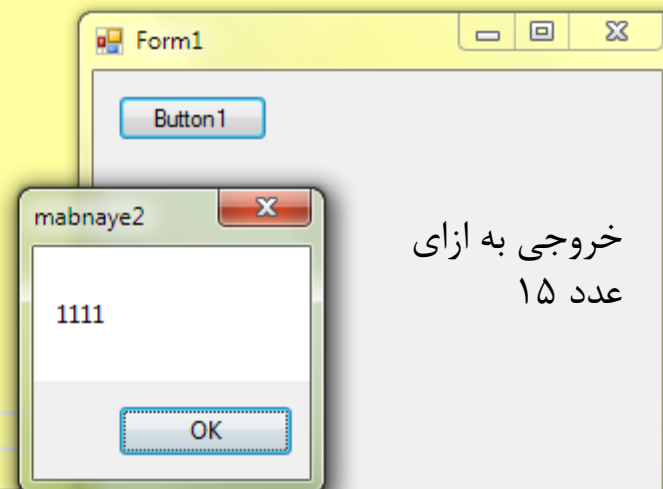
۵- برنامه‌ای بنویسید که یک عدد از کاربر دریافت کند و شکل مبنای ۲ آن را در یک msgbox نمایش دهد

برای بردن یک عدد به مبنای دیگر مثلاً مبنای ۲ به این صورت عمل می‌کنیم که رقم یکان عدد در مبنای ۲ می‌شود باقی مانده ی تقسیم عدد بر ۲ رقم ده گان می‌شود باقی مانده ی خارج قسمت تقسیم قبلی بر ۲ و برای ارقام بعدی همینطور تقسیمات را تا وقتی که خارج قسمت مخالف صفر است ادامه می‌دهیم به این ترتیب برنامه ی ما می‌شود:

```

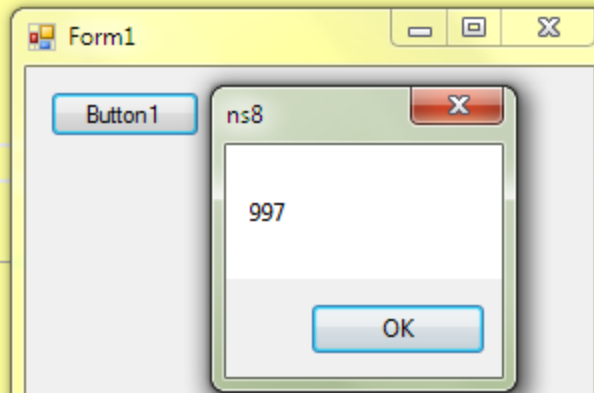
1 Public Class Form1
2
3 Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4     'am10=۱۰ عدد مبنای ۲ , am2= ۲ عدد مبنای ۱۰ , r= باقی مانده , q= خارج قسمت
5     'shr=شمارنده ی رقم
6     Dim am10, am2, r, q, shr As Integer
7     am10 = InputBox("")
8     q = am10
9     Do While q <> 0
10        r = q Mod 2
11        q = Int(q / 2)
12        shr = shr + 1
13        am2 = am2 + r * 10 ^ (shr - 1)
14    Loop
15    MsgBox(am2)
16 End Sub
17 End Class
18

```



۸- برنامه ای بنویسید که بزرگترین عدد اول کوچکتر از ۱۰۰۰ (هزار) را محاسبه و نمایش دهد.

```
1 Public Class Form1
2
3     Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
4         Dim n, i As Integer
5         n = 1000
6     100:
7         For i = 2 To Int(n / 2)
8             If n Mod i = 0 Then
9                 ' عدد مرکب است یکی از آن کاسته به لیبل ۱۰۰ میرویم
10                n = n - 1
11                GoTo 100
12            End If
13        Next
14        MsgBox(n)
15    End Sub
16 End Class
17
```



۹- برنامه ای بنویسد که از یک فایل با نام *data.txt* ذخیره شده در درایو *D:* تمامی اعداد را فراخوانی کند و سپس اعداد فرد را در ابتدا و اعداد زوج را بعد از اعداد فرد در همان فایل ذخیره کند. (محتویات قبلی فایل حذف گردد)

```

4      FileOpen(1, "D:\data.txt", OpenMode.Input)
5      Dim n, td As Integer 'td=تعداد داده
6      Do While EOF(1) = False
7          Input(1, n)
8          td = td + 1
9      Loop
10     Dim af(td), az(td), i, taz, taf As Integer
11     'af=تعداد عدد فرد, az=تعداد زوج, taz=تعداد زوج, taf=تعداد عدد فرد
12     FileClose(1)
13     FileOpen(1, "D:\data.txt", OpenMode.Input)
14     For i = 1 To td
15         Input(1, n)
16         If n Mod 2 = 0 Then
17             taz = taz + 1
18             az(taz) = n
19         Else
20             taf = taf + 1
21             af(taf) = n
22         End If
23     Next
24     FileClose(1)
25     FileOpen(1, "D:\data.txt", OpenMode.Output)
26     For i = 1 To taf
27         Print(1, af(i) & vbNewLine)
28     Next
29     For i = 1 To taz
30         Print(1, az(i) & vbNewLine)
31     Next
32     FileClose(1)

```

الگوریتم های مرتب سازی: دو الگوریتمی که دکتر درس داده اند یکی به نام الگوریتم انتخابی است و دیگری به نام الگوریتم حبابی تصویر توضیح این دو الگوریتم در کتاب به صورت زیر است.

آرایه پُر	آرایه جدید
۱ ۴۸۹۲۱۲	۱ ۱۵۲۱۲۹
۲ ۹۹۹۹۹۹	۲ ۲۳۴۰۱۱
۳ ۵۹۱۷۳۶	۳
۴ ۹۹۹۹۹۹	۴
۵ ۳۷۸۲۱۶	۵

در سومین گذر چون ۳۷۸۲۱۶ سومین عدد در مرتبه صعودی است به آرایه جدید منتقل می شود و مقدار آن در آرایه پر به ۹۹۹۹۹۹ تبدیل می گردد. وضعیت دو آرایه پر و جدید پس از گذر سوم به صورت زیر می باشد:

آرایه پُر	آرایه جدید
۱ ۴۸۹۲۱۲	۱ ۱۵۲۱۲۹
۲ ۹۹۹۹۹۹	۲ ۲۳۴۰۱۱
۳ ۵۹۱۷۳۶	۳ ۳۷۸۲۱۶
۴ ۹۹۹۹۹۹	۴
۵ ۹۹۹۹۹۹	۵

در چهارمین گذر چون ۴۸۹۲۱۲ چهارمین عدد در مرتبه صعودی است به آرایه جدید منتقل می شود و مقدار آن در آرایه پر به ۹۹۹۹۹۹ تبدیل می گردد. وضعیت دو آرایه پر و جدید پس از گذر چهارم به صورت زیر می باشد:

آرایه پُر	آرایه جدید
۱ ۹۹۹۹۹۹	۱ ۱۵۲۱۲۹
۲ ۹۹۹۹۹۹	۲ ۲۳۴۰۱۱
۳ ۵۹۱۷۳۶	۳ ۳۷۸۲۱۶
۴ ۹۹۹۹۹۹	۴ ۴۸۹۲۱۲
۵ ۹۹۹۹۹۹	۵

در پنجمین گذر عدد ۵۹۱۷۳۶ به آرایه جدید منتقل می شود و مقدار آن در آرایه پر به ۹۹۹۹۹۹ تبدیل می گردد. وضعیت دو آرایه پر و جدید پس از گذر پنجم به صورت زیر می باشد:

آرایه پُر	آرایه جدید
۱ ۹۹۹۹۹۹	۱ ۱۵۲۱۲۹
۲ ۹۹۹۹۹۹	۲ ۲۳۴۰۱۱
۳ ۹۹۹۹۹۹	۳ ۳۷۸۲۱۶
۴ ۹۹۹۹۹۹	۴ ۴۸۹۲۱۲
۵ ۹۹۹۹۹۹	۵ ۵۹۱۷۳۶

سرانجام پس از ۵ گذر آرایه به صورت صعودی مرتب شده است. چنانچه عملیات بالا به صورت انتخاب بزرگ ترین عضو آرایه پُر در هر گذر انجام گیرد آرایه جدید پس از تکمیل عضوهایش به ترتیب نزولی مرتب خواهند شد.

• مرتب سازی انتخابی (۲):

یکی از انواع روش های مرتب سازی روش مرتب سازی انتخابی می باشد که طی آن کراراً اقلام داده ها مورد بررسی قرار می گیرند و در هر گذر (۳) بسته به ترتیب صعودی یا نزولی یک عنصر انتخاب می شود و در لیست جدید قرار می گیرد و از لیست قبلی یا قدیمی حذف می گردد و این فرآیند آنقدر ادامه می یابد تا یک لیست جدید مرتب شده به دست آید. در مورد آرایه نیز عنصرهای آرایه یک به یک مورد بررسی قرار می گیرند و در هر گذر بسته به ترتیب صعودی یا نزولی یک عنصر انتخاب می شود و در یک آرایه خالی قرار می گیرد، تا سرانجام آرایه جدید به صورت صعودی یا نزولی مرتب شود. وقتی عنصری برای انتقال به آرایه خالی انتخاب می شود پس از انتقال به آرایه خالی مقدار آن در آرایه پُر به یک مقدار بالا یا عددی که کلیه ارقام آن ۹ باشد تبدیل می گردد اگر عنصر مورد انتقال رکورد یا سابقه (۴) باشد کلیه (۵) آن در آرایه پُر به یک مقدار بالا یا عددی که کلیه ارقام آن ۹ باشد تبدیل می گردد. برای آشنا شدن با طرز کار این نوع مرتب سازی فرض کنید یک آرایه ۵ عنصری نامرتب زیر وجود دارد و در نظر داریم عنصرهای آنرا به روش مرتب سازی انتخابی به صورت صعودی مرتب کنیم. ابتدا یک آرایه خالی هم اندازه آرایه پُر انتخاب می کنیم:

آرایه پُر	آرایه خالی (جدید)
۱ ۴۸۹۲۱۲	۱
۲ ۲۳۴۰۱۱	۲
۳ ۵۹۱۷۳۶	۳
۴ ۱۵۲۱۲۹	۴
۵ ۳۷۸۲۱۶	۵

در گذر اول چون ۱۵۲۱۲۹ از همه عناصر کوچک تر است انتخاب می شود و به آرایه خالی (جدید) منتقل می گردد و مقدار آن در آرایه پر به ۹۹۹۹۹۹ تبدیل می گردد. وضعیت دو آرایه پر و خالی پس از گذر اول به صورت زیر می باشد:

آرایه پُر	آرایه جدید
۱ ۴۸۹۲۱۲	۱ ۱۵۲۱۲۹
۲ ۲۳۴۰۱۱	۲
۳ ۵۹۱۷۳۶	۳
۴ ۹۹۹۹۹۹	۴
۵ ۳۷۸۲۱۶	۵

در دومین گذر چون ۲۳۴۰۱۱ دومین عدد در مرتبه صعودی است به آرایه جدید منتقل می شود و مقدار آن در آرایه پر به ۹۹۹۹۹۹ تبدیل می گردد. وضعیت دو آرایه پُر و جدید پس از گذر دوم به صورت زیر می باشد:

مثال: سوال ۴ از نمونه سوالات دکتر حل به روش انتخابی:

۴- برنامه‌ای بنویسید که n عدد از کاربر دریافت کرده و میانه آن‌ها را نمایش دهد (توضیح: میانه چند عدد، عددی است که پس از مرتب کردن بصورت صعودی یا نزولی در وسط قرار می‌گیرد. اگر تعداد ارقام فرد باشد عدد وسط و اگر تعداد اعداد زوج باشد میانگین دو عددی که در وسط قرار می‌گیرند میانه است).

```
4 Dim n, i, j As Integer
5 'دریافت تعداد داده
6 n = Val(InputBox("enter n"))
7 Dim a(n), b(n), mi, min As Double, mmin As Integer 'mmin=محل مینیموم
8 'دریافت داده ها
9 For i = 1 To n
10     a(i) = Val(InputBox("enter items"))
11 Next
12 'صعودی سازی با روش انتخابی
13 For i = 1 To n
14     'find min
15     min = 999999
16     For j = 1 To n
17         If a(j) < min Then
18             min = a(j)
19             mmin = j
20         End If
21     Next
22     a(mmin) = 999999
23     b(i) = min
24     'نمایش داده ها
25     ListBox1.Items.Add(b(i))
26 Next
27 'نمایش میانه
28 If n Mod 2 <> 0 Then
29     mi = b(Int(n / 2) + 1)
30 Else
31     mi = (b(n / 2) + b(n / 2 + 1)) / 2
32 End If
33 MsgBox("mi= " & mi)
```

Form1

Button1

-8
2
5.3
8
9
30
35
35.6
47
64

t4_re

mi= 19.5

OK

• مرتب سازی حبابی^(۱):

روش مرتب سازی مشهور دیگر مرتب سازی حبابی است. در این روش به جای یافتن کوچک ترین عضو آرایه و سپس انجام یک مبادله، پس از اینکه مشخص گردید دو عضو متوالی خارج از ترتیب هستند با هم جابجا می شوند.

اگر تعداد عضوهای آرایه n باشد با به کارگیری این رویکرد^(۲) $n-1$ گذر لازم است. در اولین گذر عضوهای اول و دوم با هم مقایسه می شوند و اگر خارج از ترتیب باشند با هم جابجا می شوند. این فرآیند جهت عضوهای دوم و سوم، سوم و چهارم و هر دو عضو پی در پی دیگر تکرار می گردد. اگر عمل جستجو برای پیدا کردن بزرگترین عضو از ابتدای آرایه آغاز شود، پس از اولین گذر آخرین عضو در محل درست خود قرار خواهد داشت. پس از دومین گذر عضو ماقبل آخر یعنی عضوی که مقدار آن از عضو آخر آرایه کوچکتر و از بقیه اعضا بزرگتر است در محل درست خواهد بود و همین طور تا آخر عضوهای مرتب شونده همانند حبابی از حبابهای آب به بالا می آیند و سرانجام آرایه مرتب شده است. می توان جستجو را برای پیدا کردن کوچکترین عضو از ابتدای آرایه نیز آغاز نمود. در این حالت پس از اولین گذر کوچکترین عضو در محل درست خود قرار خواهد داشت و در گذرهای بعدی عضوهای دیگر در جای خود قرار می گیرند.

بعد از هر گذر از آرایه، کنترل به عمل می آید تا تعیین شود که آیا هیچگونه جابجایی در طی آن گذر انجام یافته است یا خیر. اگر هیچگونه مبادله ای صورت نگرفته، آنگاه آرایه یا لیست به صورت مرتب شده می باشد و گذرهای دیگری لازم نیست.

در این روش مرتب سازی نیز به یک ناحیه کاری از حافظه یا محل موقتی برای حفظ و نگهداری موقتی مقدار عضو جابجا شونده طی جابجایی نیاز است. برای آشنا شدن با طرز کار این روش مرتب سازی فرض کنید یک آرایه ۴ عضوی نامرتب زیر وجود دارد و در نظر داریم عضوهای آنرا به روش مرتب سازی حبابی مرتب کنیم. ابتدا یک محل موقتی برای حفظ و نگهداری مقدار عضوهای جابجا شونده در طی جابجایی انتخاب می کنیم:

	آرایه پر	محل موقتی
۱	۶۵۴	
۲	۴۷۲	
۳	۹۲۳	
۴	۲۵۲	

در گام اول از گذر اول چون ۶۵۴ و ۴۷۲ خارج از ترتیب هستند با هم جابجا می شوند ولی در گام دوم از گذر اول چون ۶۵۴ و ۹۲۳ به ترتیب هستند جابجایی انجام نمی گیرد.

گذر اول - گام اول

	آرایه پر	محل موقتی
۱	۶۵۴	۴۷۲
۲	۴۷۲	
۳	۹۲۳	
۴	۲۵۲	

گذر اول - گام دوم

	آرایه پر
۱	۴۷۲
۲	۶۵۴
۳	۹۲۳
۴	۲۵۲

در گام سوم از گذر اول چون ۹۲۳ و ۲۵۲ خارج از ترتیب هستند با هم جابجا می شوند.

گذر اول - گام سوم

	آرایه پر	محل موقتی
۱	۴۷۲	۲۵۲
۲	۶۵۴	
۳	۹۲۳	
۴	۲۵۲	

وضعیت آرایه پر پس از گذر اول

	آرایه پر
۱	۴۷۲
۲	۶۵۴
۳	۲۵۲
۴	۹۲۳

پس از کامل شدن گذر اول بزرگترین عضو آرایه یعنی ۹۲۳ در جایگاه صحیح خود به عنوان آخرین عضو قرار خواهد داشت. دومین گذر نیز به روش مشابه انجام خواهد گرفت ولی در این گذر فقط عضوهای ۱ تا ۳ مقایسه می شوند.

در گام اول از گذر دوم چون ۴۷۲ و ۶۵۴ به ترتیب هستند جابجایی انجام نمی گیرد، ولی در گام دوم از گذر دوم چون ۶۵۴ و ۲۵۲ خارج از ترتیب هستند با هم جابجا می شوند.

پس از کامل شدن گذر دوم عضو ماقبل آخر یعنی عضوی که مقدار آن از عضو آخر آرایه کوچکتر و از بقیه اعضا بزرگتر است در محل درست قرار دارد.

گذر دوم - گام دوم

	آرایه پر	محل موقتی
۱	۴۷۲	۲۵۲
۲	۶۵۴	
۳	۲۵۲	
۴	۹۲۳	

وضعیت آرایه پر پس از گذر دوم

	آرایه پر
۱	۴۷۲
۲	۲۵۲
۳	۶۵۴
۴	۹۲۳

سومین گذر نیز به روش مشابه انجام خواهد گرفت ولی در این گذر فقط عضوهای ۱ و ۲ با هم مقایسه می شوند. در گام اول از گذر سوم چون ۴۷۲ و ۲۵۲ خارج از ترتیب هستند با هم جابجا می شوند.

مثال: سوال ۴ از نمونه سوالات دکتر حل به روش حبابی:

۴- برنامه‌ای بنویسید که n عدد از کاربر دریافت کرده و میانه آن‌ها را نمایش دهد (توضیح: میانه چند عدد، عددی است که پس از مرتب کردن بصورت صعودی یا نزولی در وسط قرار می‌گیرد. اگر تعداد ارقام فرد باشد عدد وسط و اگر تعداد اعداد زوج باشد میانگین دو عددی که در وسط قرار می‌گیرند میانه است).

```
4 Dim n, i, j As Integer
5 'دریافت تعداد داده
6 n = InputBox("enter n")
7 Dim a(n), m As Double
8 'دریافت داده ها
9 For i = 1 To n
10     a(i) = InputBox("enter items")
11 Next
12 'صعودی سازی با روش حبابی
13 For i = 1 To n
14     For j = 2 To n
15         If a(j) < a(j - 1) Then
16             'جابجایی بین a(j-1) و a(j)
17             m = a(j - 1)
18             a(j - 1) = a(j)
19             a(j) = m
20         End If
21     Next
22 Next
23 'نمایش داده ها برای اطمینان از صحت صعودی سازی
24 For i = 1 To n
25     ListBox1.Items.Add(a(i))
26 Next
27 'محاسبه ی میانه
28 Dim mi As Double 'mi=میانه
29 If n Mod 2 <> 0 Then
30     mi = a(Int(n / 2) + 1)
31 Else
32     mi = (a(n / 2) + a(n / 2 + 1)) / 2
33 End If
34 MsgBox("mi= " & mi)
```

قطعه کد صعودی سازی به روش حبابی برای n داده روی آن فکر کرده و آن را به خاطر بسپارید.

Form1

Button1

-98
15
24
32
57
59
68
95
325.2
470356

t4

mi= 58

OK