

آموزش ویژوال بیسیک 6

مهدی امینی دانشجوی ترم اول برنامه نویسی کامپیوتر

ایمیل: aminimahdi14@yahoo.com

وبلاگ: www.m-a-h-d-i.loxblog.com

Learning Visual Basic 6

2011

استاد محترم: خانم عبدالهی زاد

دانشگاه آزاد اسلامی سردرود

منبع: www.m-a-h-d-i.loxblog.com ✓

صفحه 1 تا 20
❖ 1- الگوریتم (ویژگی ها) و فلوچارت (اشکال و قوانین) (همراه مثال های کلاسی)
صفحه 21
❖ 2- مزایای ویژوال بیسیک
صفحه 22
❖ 3- جدول نوع داده
صفحه 23 تا 25
❖ 4- اولویت عملگرها (عملگرهای محاسباتی ، عملگرهای رابطه ای و عملگرهای رابطه ای).
صفحه 25 تا 26
❖ 5- ساختارهای تصمیم (دستورات شرطی) (همراه مثال های کلاسی)
صفحه 26 تا 28
❖ تابع IIF (همراه مثال های کلاسی)
صفحه 28 تا 32
❖ تابع Select Case (همراه مثال های کلاسی)
صفحه 32 تا 34
❖ حلقه تکرار (For) (همراه مثال های کلاسی)
صفحه 34 تا 35
❖ حلقه For تو در تو (همراه مثال های کلاسی)
صفحه 35 تا 39
❖ حلقه های تکرار While , Wend , Do...While , Do Loop (همراه مثال های کلاسی)
صفحه 39 تا 43
❖ آرایه ها (همراه مثال های کلاسی)
صفحه 43 تا 45
❖ جستجو ها (دودویی و حبابی) (همراه مثال های کلاسی)
صفحه 45 تا 48
❖ برنامه ها و توابع (همراه مثال های کلاسی)
صفحه 48 تا 49
❖ آموزش افزودن ماژول به برنامه (همراه مثال های کلاسی)
صفحه 50
❖ سوالات میان ترم

الگوریتم:

الگوریتم : دستورالعمل هایی است برای حل مسئله

- 1- به زبان دقیق گفته شود
- 2- ترتیب مراحل آن مشخص باشد.
- 3- جزئیات کامل مسئله را داشته باشد.
- 4- خاتمه عملیات مشخص باشد.

بیان روش حل مسئله به زبانی ساده و قابل فهم باشد.

ویژگی های الگوریتم

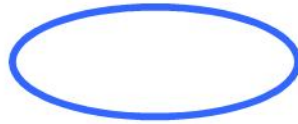
- 1- الگوریتم عبارت است از حل مسئله بصورت مرحله به مرحله بصورتی که قابل فهم باشد.
- 2- الگوریتم بایستی دارای یک نقطه شروع و یک نقطه پایان باشد.
- 3- به جمله آغازین الگوریتم که شروع نامیده می شود عدد 0 و به جمله پایانی الگوریتم که عبارت پایان می نامند آخرین عدد را انتصاب می کنیم.
- 4- الگوریتم باید پایان پذیر باشد یعنی در حلقه بی نهایت به دام نیفتد.
- 5- مراحل الگوریتم توسط جملات فارسی و یا ریاضی نوشته می شود.
- 6- هر مرحله از الگوریتم بایستی بصورت ساده و غیر مبهم نوشته شود.

یک الگوریتم مراحل زیر را دارد:

- 1- خواندن داده
 - 2- انجام محاسبات
 - 3- چاپ خروجی
- برای حل یک مسئله و طراحی الگوریتم موارد زیر باید مشخص باشد:
- 1- ورودی مسئله
 - 2- خروجی مسئله
 - 3- رابطه بین ورودی و خروجی
 - 4- بررسی راه حل های مختلف تهیه الگوریتم مناسب
 - 5- اشکال زدائی الگوریتم

بیان الگوریتم توسط شکل‌های هندسی:

شروع و پایان



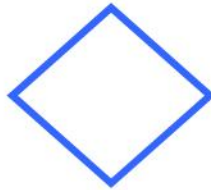
دستورات ورودی و خروجی



دستورات محاسباتی

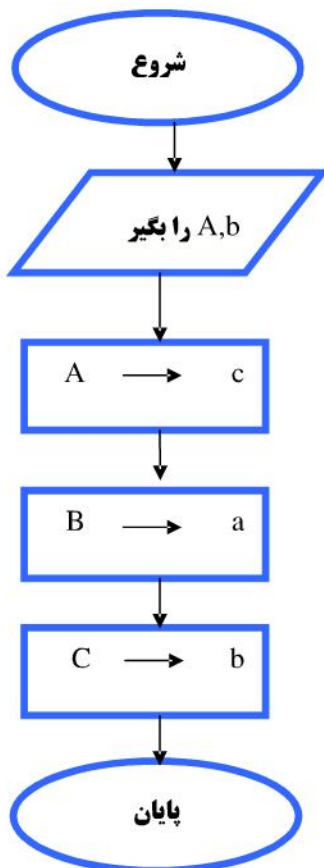


دستورات شرطی



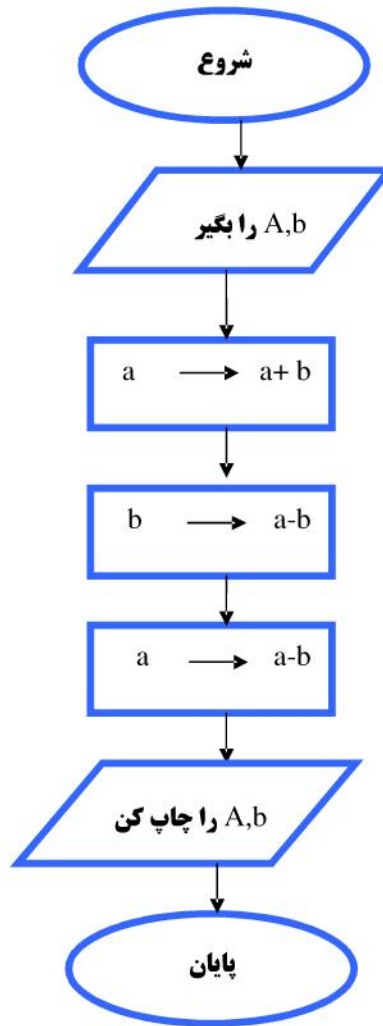
مثال:

الگوریتمی بنویسید که دو مقدار را از ورودی خوانده و در متغیرهای a, b قرار دهد و سپس محتویات این دو را عوض کند:



تمرین در منزل :

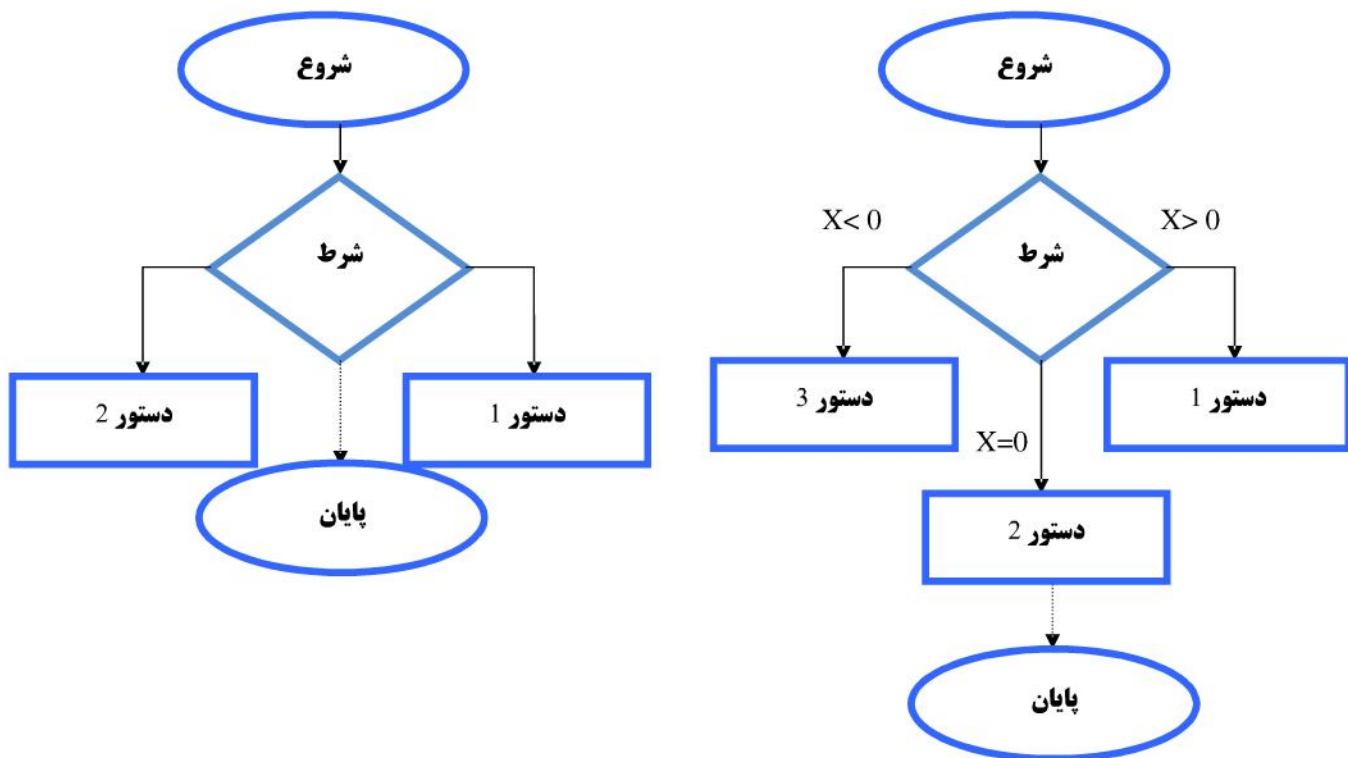
جای متغیر a,b را بدون استفاده از متغیر کمکی عوض کنید:



دستورالعمل های شرطی:

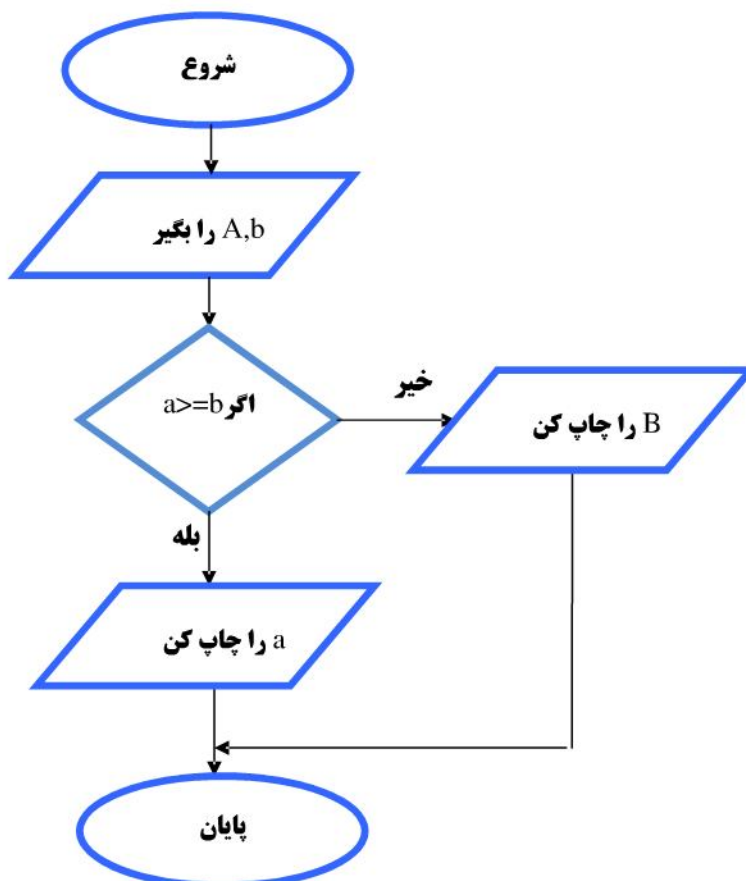
الگوریتمی وجود دارد که در آنها براساس شرایطی که اتفاق می افتد دستورات خاصی اجراء می شود:

دستور 2	وگرنه	دستور 1	آنگاه	شرط	اگر
دستور 2	وگرنه	دستور 1	→	شرط	if

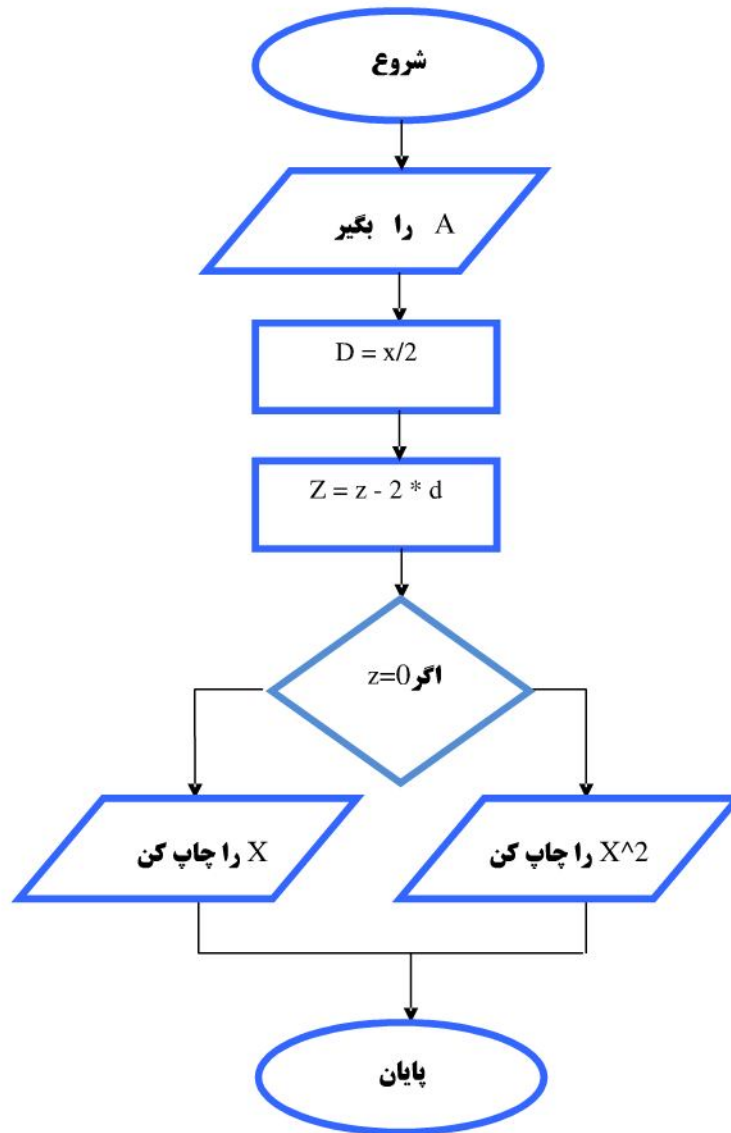


مثال 1:

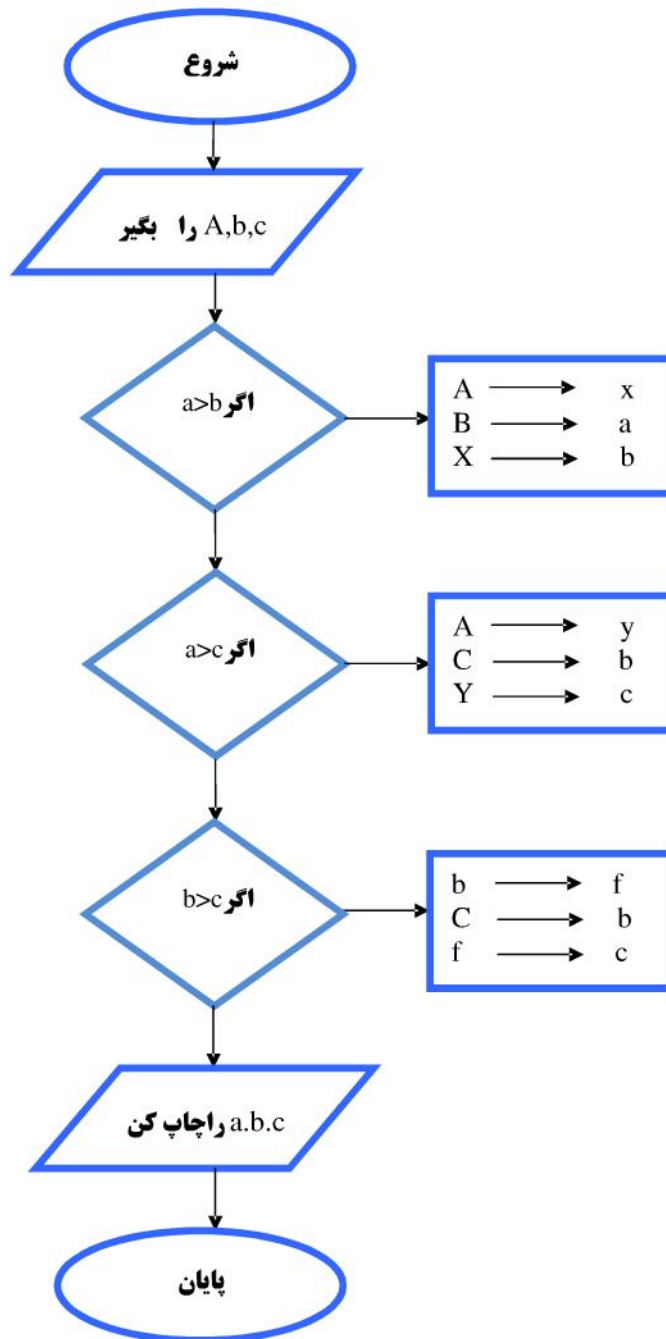
الگوریتمی بنویسید که 2 مقدار را از ورودی خوانده و مقدار بزرگتر را در متغیری مانند \max قرار دهد و خروجی را چاپ کند:



مثال 2: الگوریتمی بنویسید که a را از ورودی خوانده اگر عدد زوج باشد عدد را بتوان 2 برساند و اگر فرد باشد خود عدد را در خروجی چاپ کند.



مثال 3: الگوریتمی بنویسید که 3 مقدار را از ورودی خوانده و آنها را بترتیب در خروجی چاپ کند:



حلقه تکرار:

فرض کنید می خواهیم 100 عدد را از ورودی خوانده و جمع آنها را حساب کند:

روش اول: صد تا دستور می نویسیم یعنی یکی یکی اعداد را با هم جمع می کنیم.

روش دوم: یک دستور یا دو دستور می نویسیم و صد بار تکرار می کنیم.

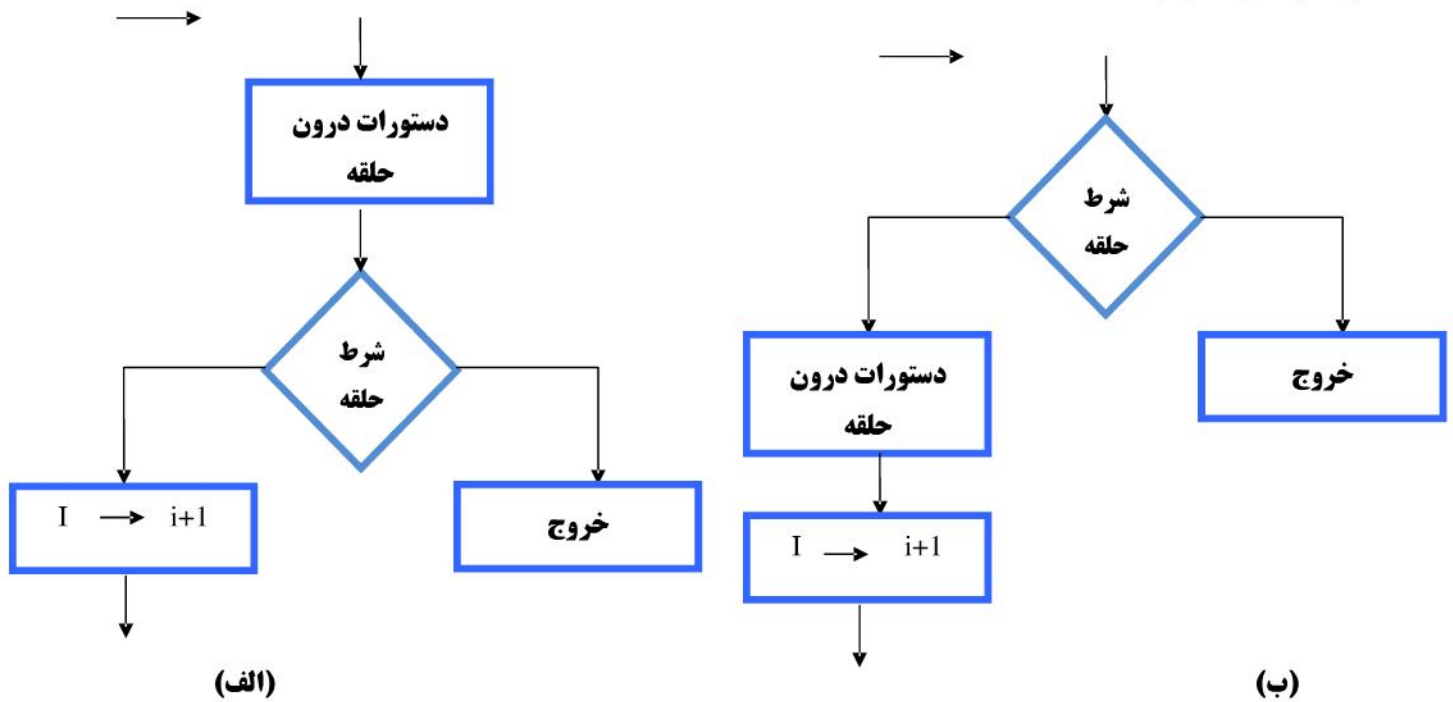
شرط حلقه تکرار:

1) مشخص کنید حلقه تکرار چه وقت خاتمه پیدا می کند.

2) شمارنده حلقه تکرار مشخص می کند که تاکنون حلقه چندبار اجرا شده است.

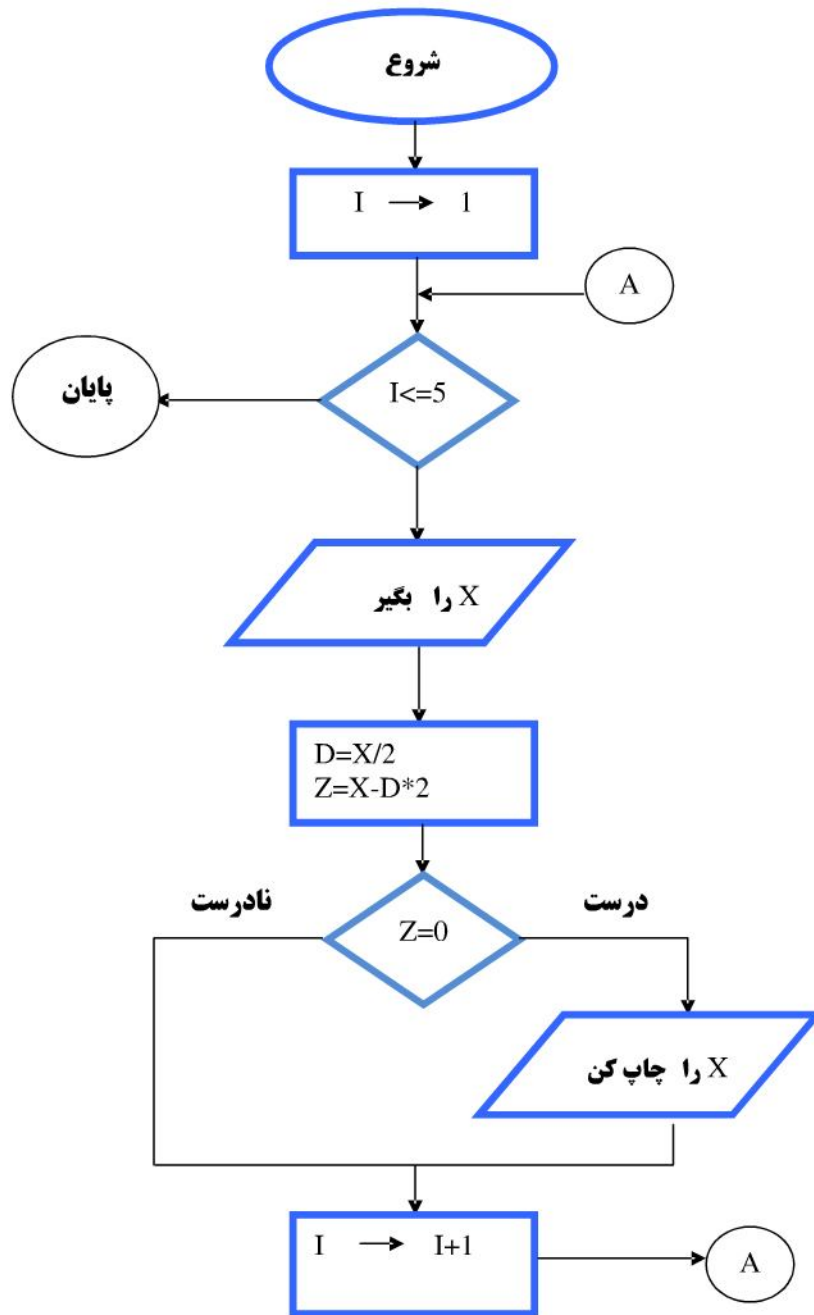
3) به ازای هر بار اجرای حلقه گام افزایش چه مقداری است و چه مقدار به شمارنده حلقه اضافه می شود.

تفاوت در شکل الف و ب

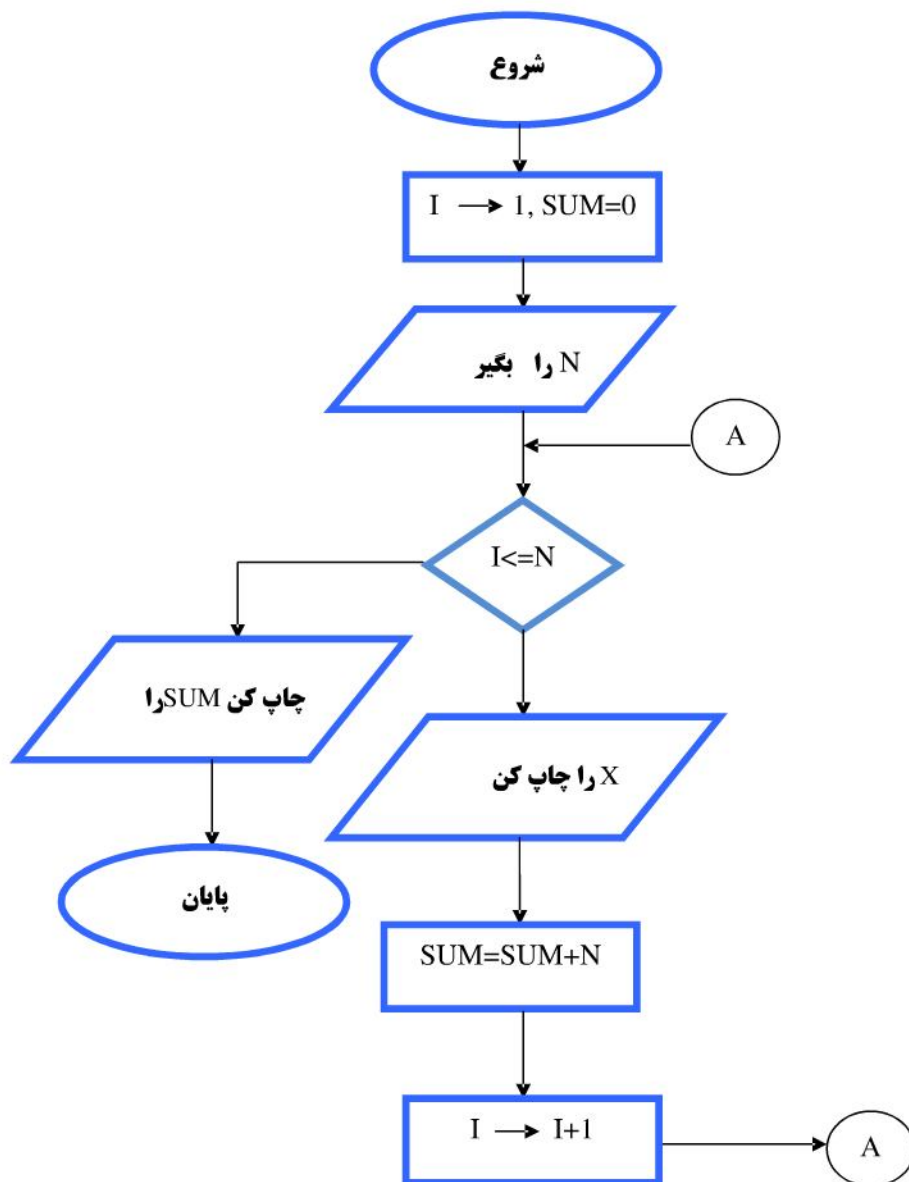


در شکل (الف) ابتدا دستورات حلقه اجرا شده و سپس شرط حلقه چک می شود. یعنی در صورت نادرست بودن شرط حلقه حداقل یک بار اجرا می شود ولی در شکل (ب) برعکس.

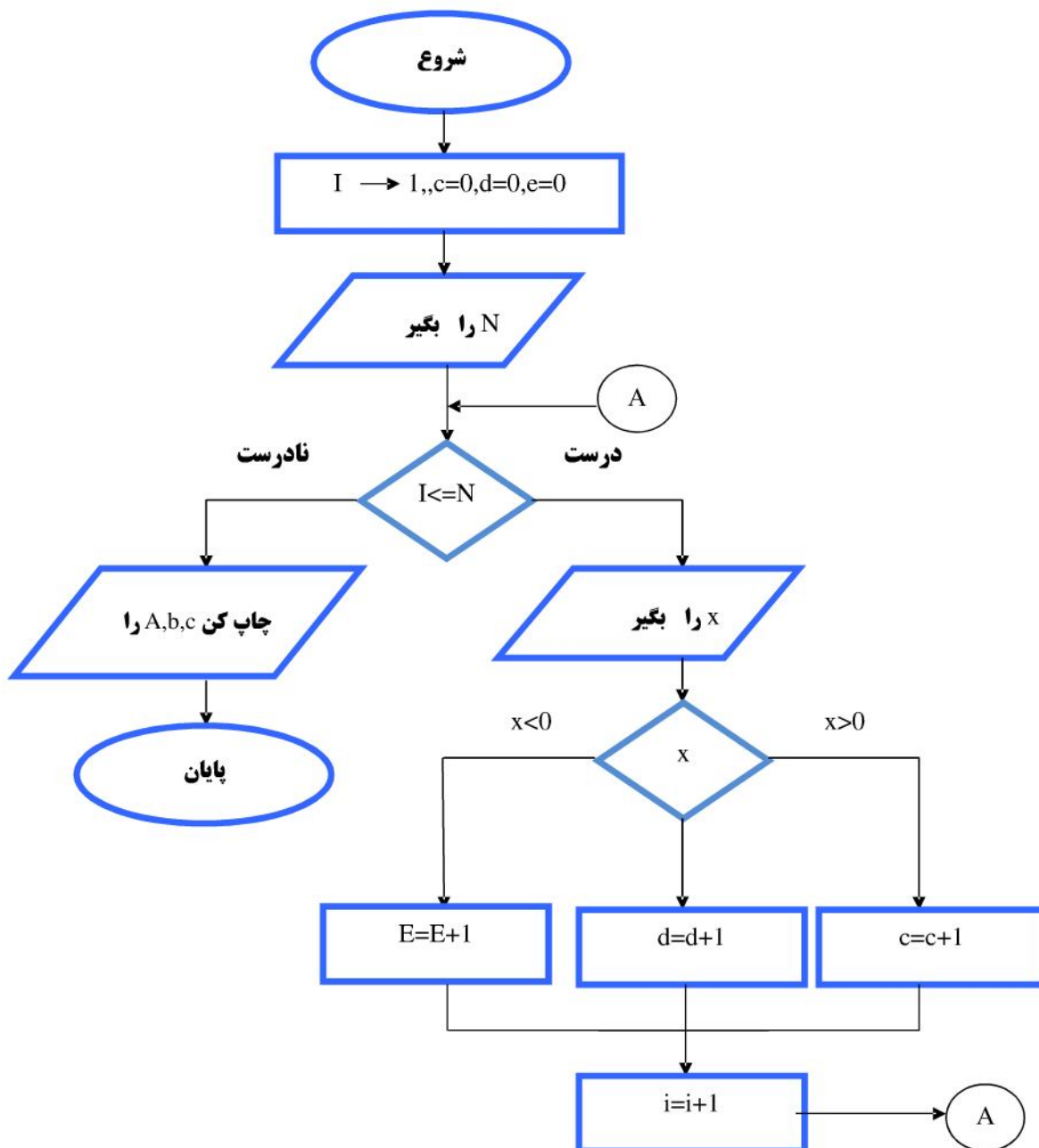
الگوریتمی بنویسید که تعداد ۵ عدد را از ورودی خوانده و اعداد زوج را در خروجی چاپ کند:



الگوریتمی بنویسید که تعداد N عدد را از ورودی خوانده و مجموع آنها را در خروجی چاپ کند:

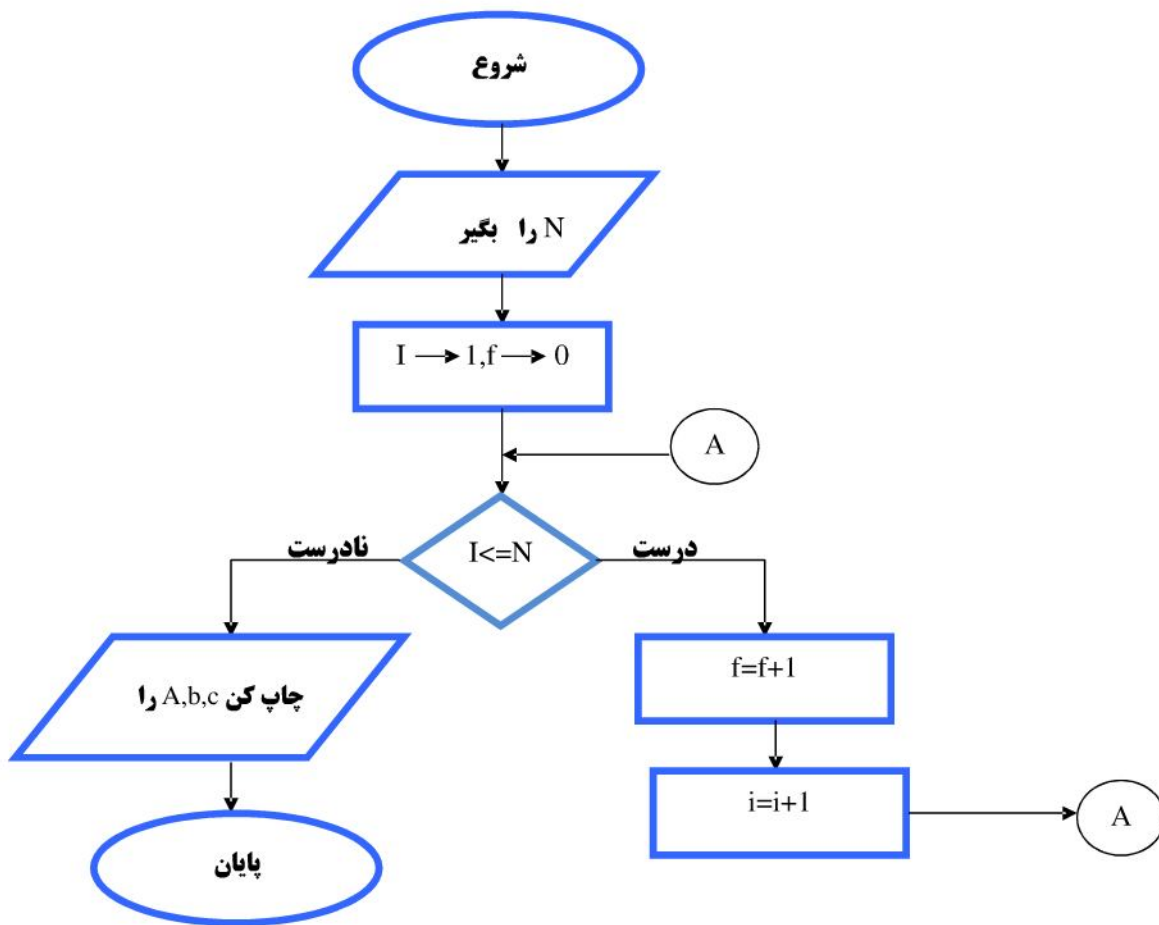


الگوریتمی بنویسید که تعداد N عدد را از ورودی خوانده و تعداد اعداد منفی و صفر و مثبت را شمارش کند:

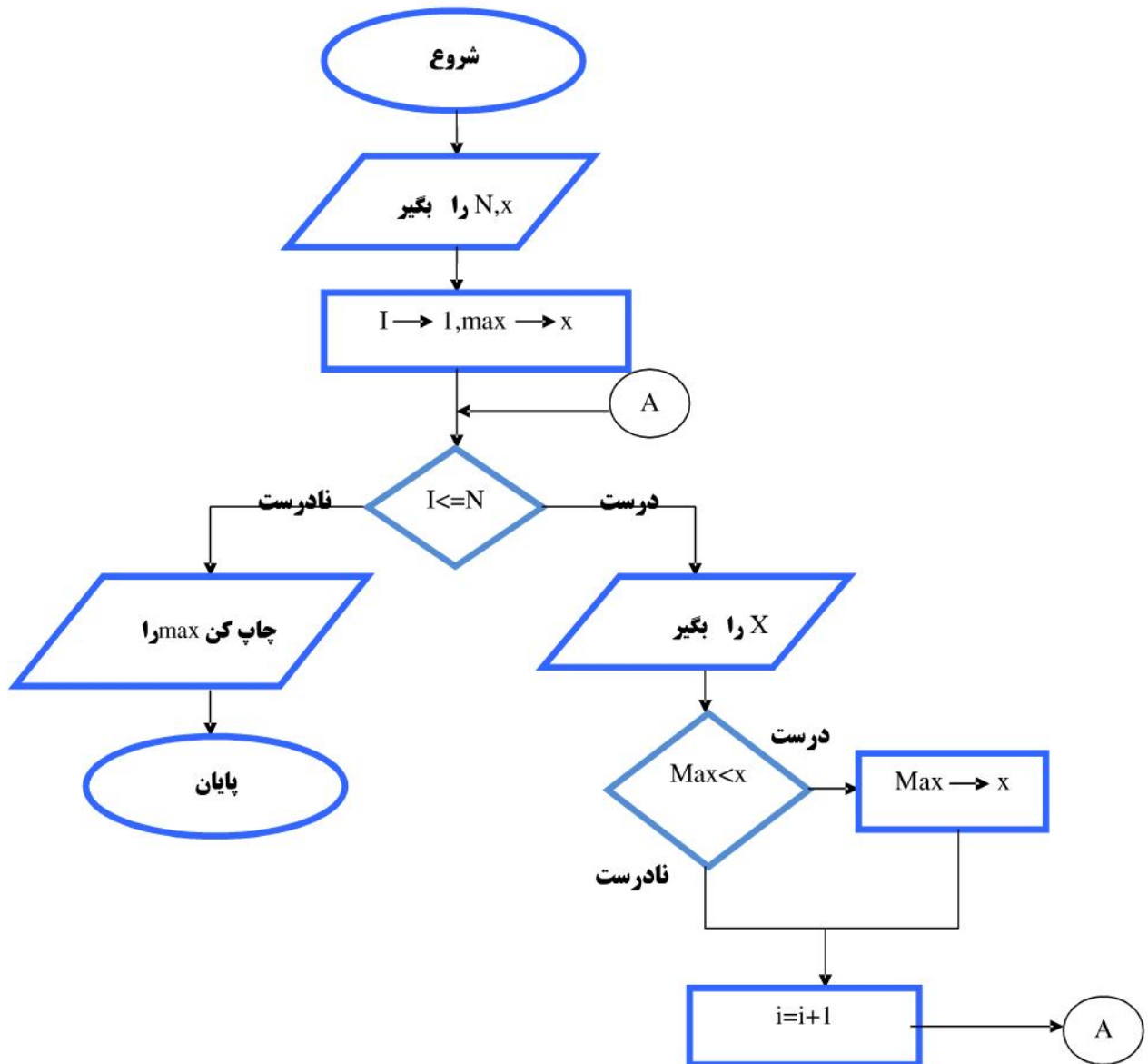


الگوریتمی بنویسید که عدد صحیح و مثبت n را از ورودی گرفته و فاکتوریل آن را حساب کن:

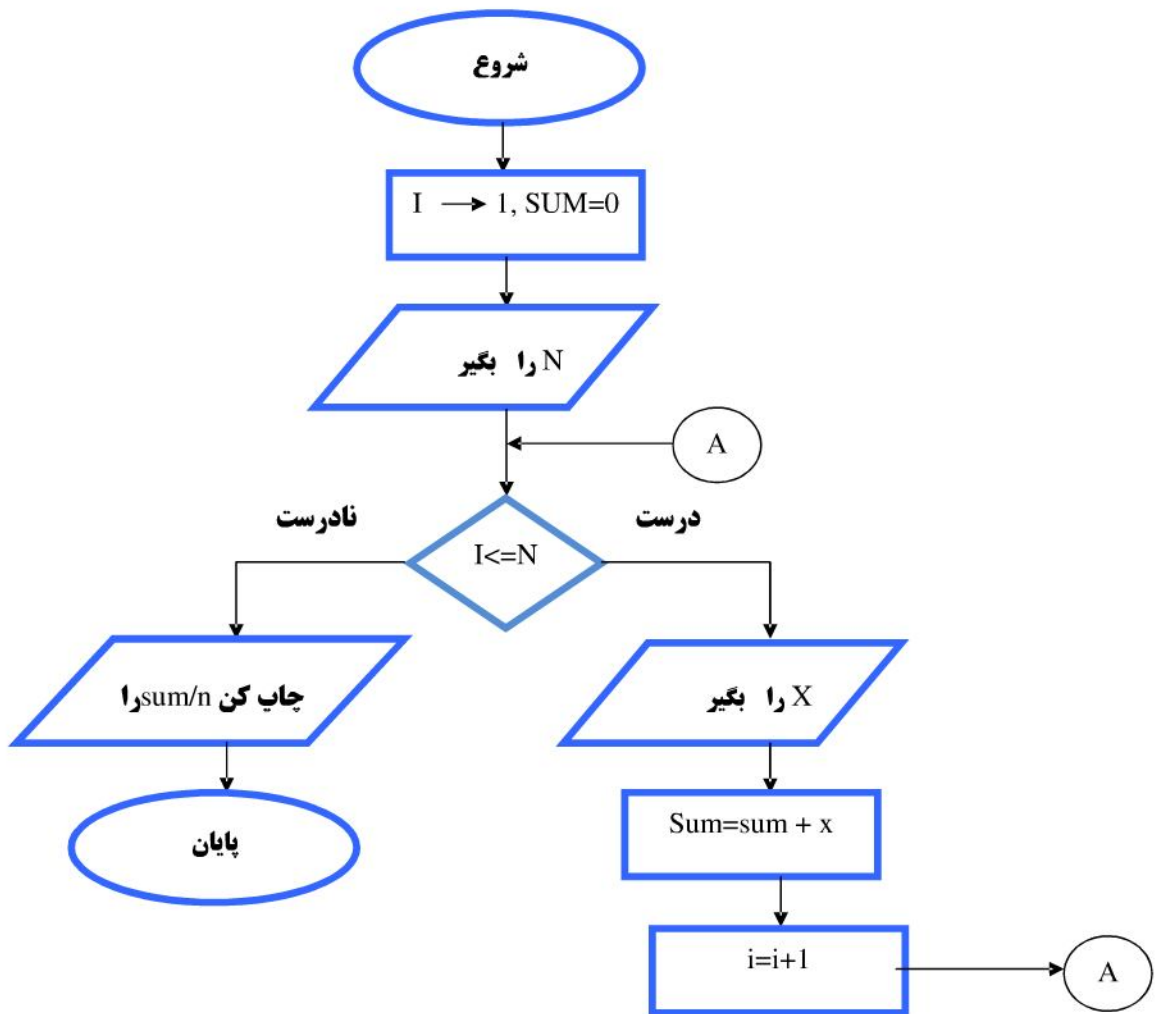
$$S=1!+2!+ 3!+4! +5!+\dots+n!$$



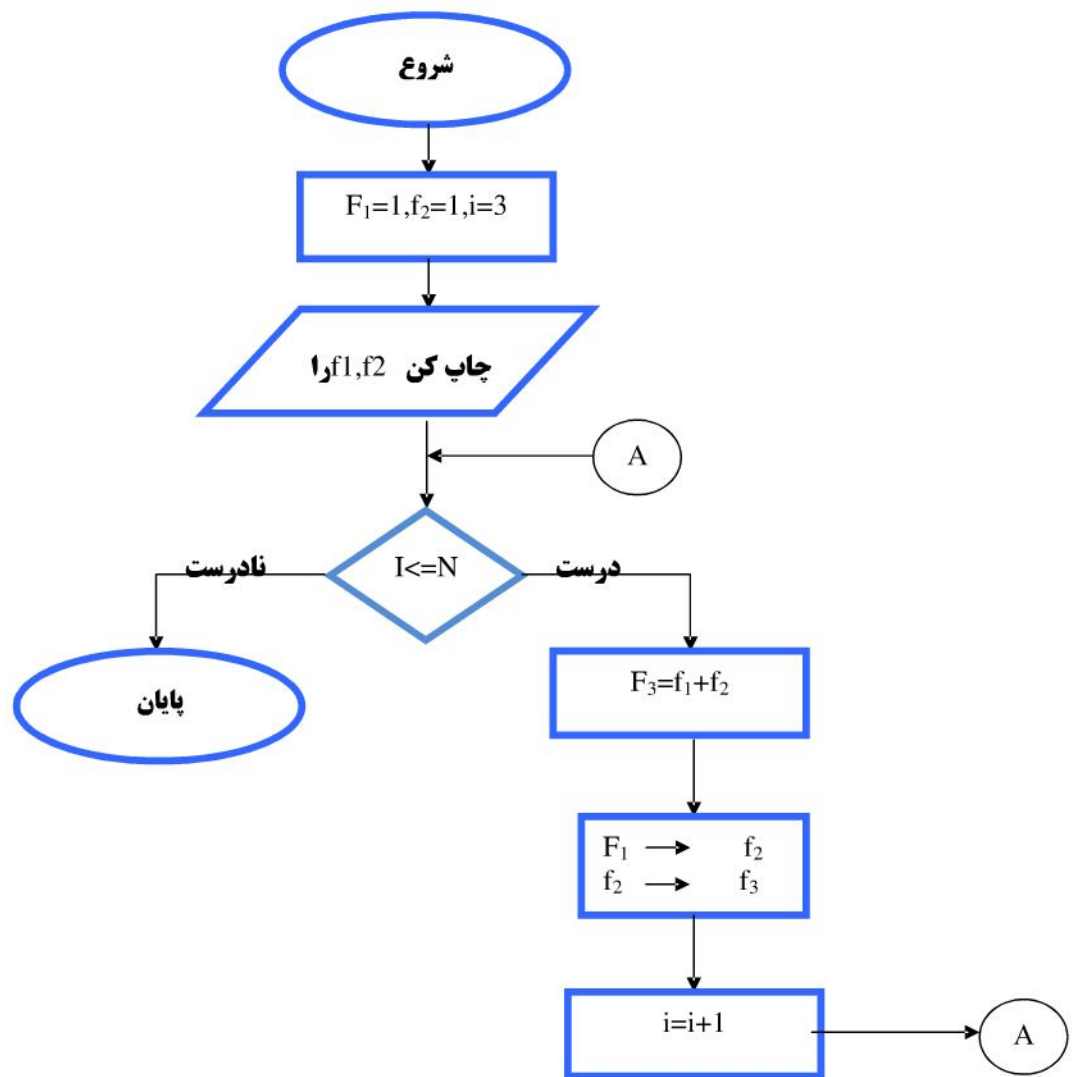
الگوریتمی بنویسید که تعداد n عدد را خوانده و بزرگترین عدد را از بین این اعداد پیدا کرده و چاپ کند:



الگوریتمی بنویسید که میانگین معدل n دانشجو را حساب کند:

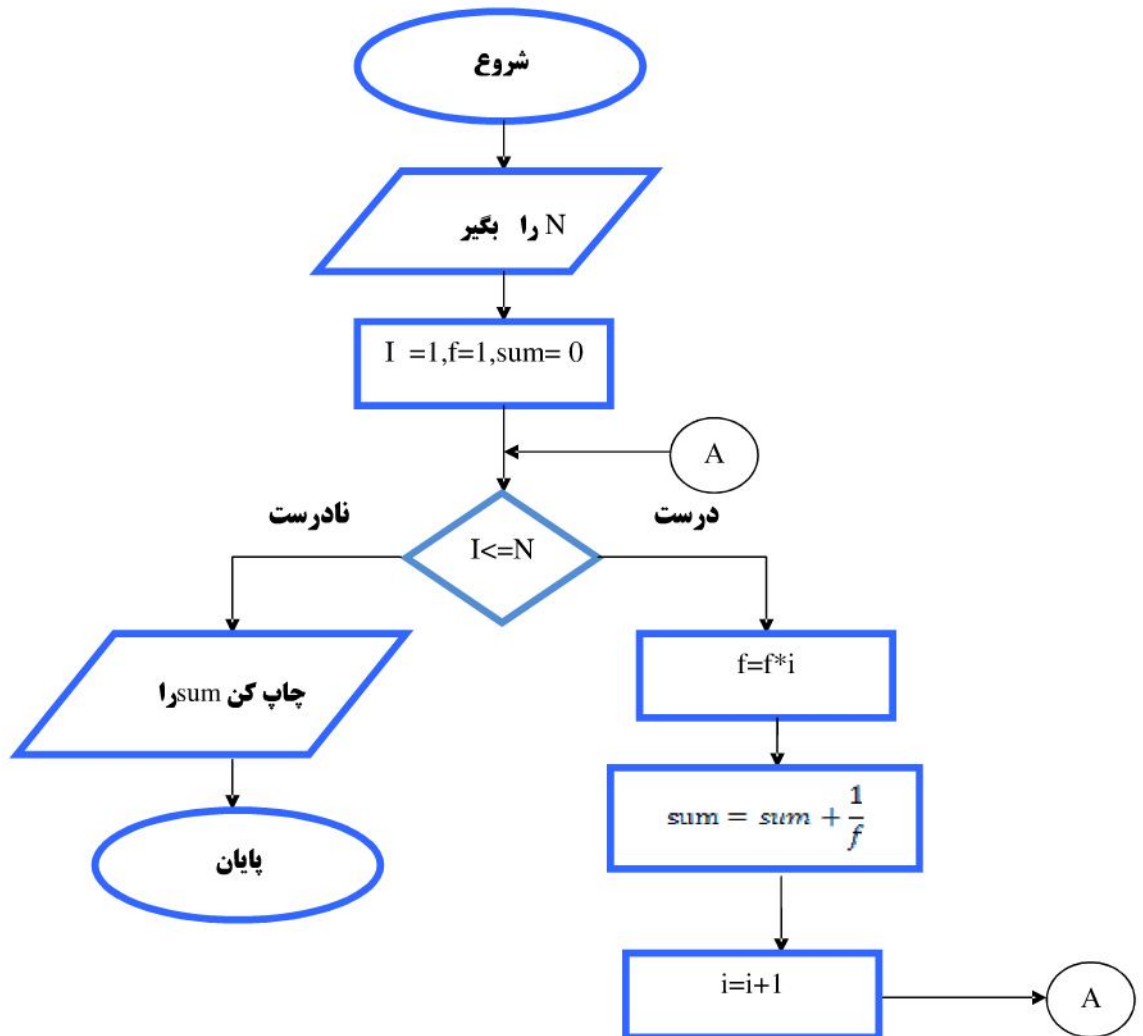


الگوریتمی بنویسید که تعداد n جمله از سری فیبوناچی را چاپ کند:



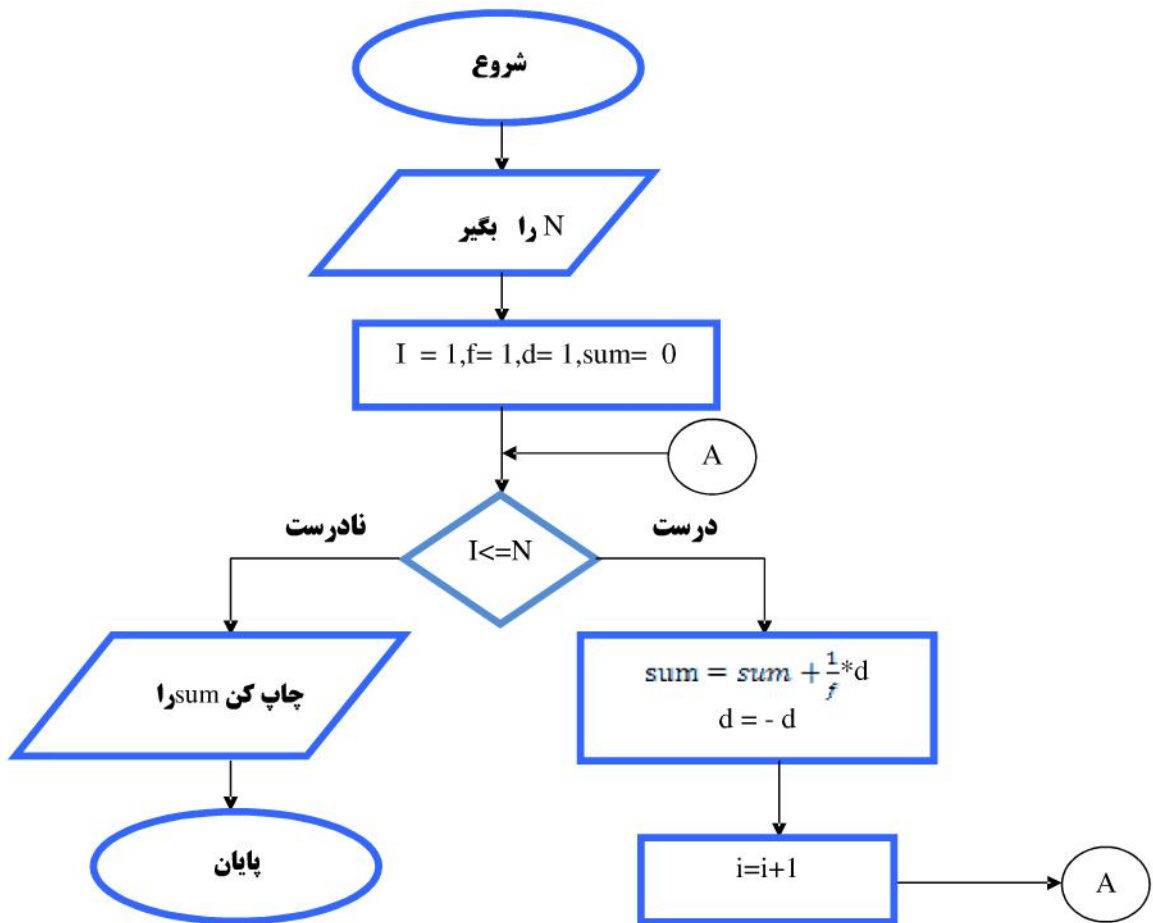
الگوریتمی بنویسید که تعداد n جمله از دنباله زیر را گرفته و چاپ کند:

$$s = \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{n!}$$

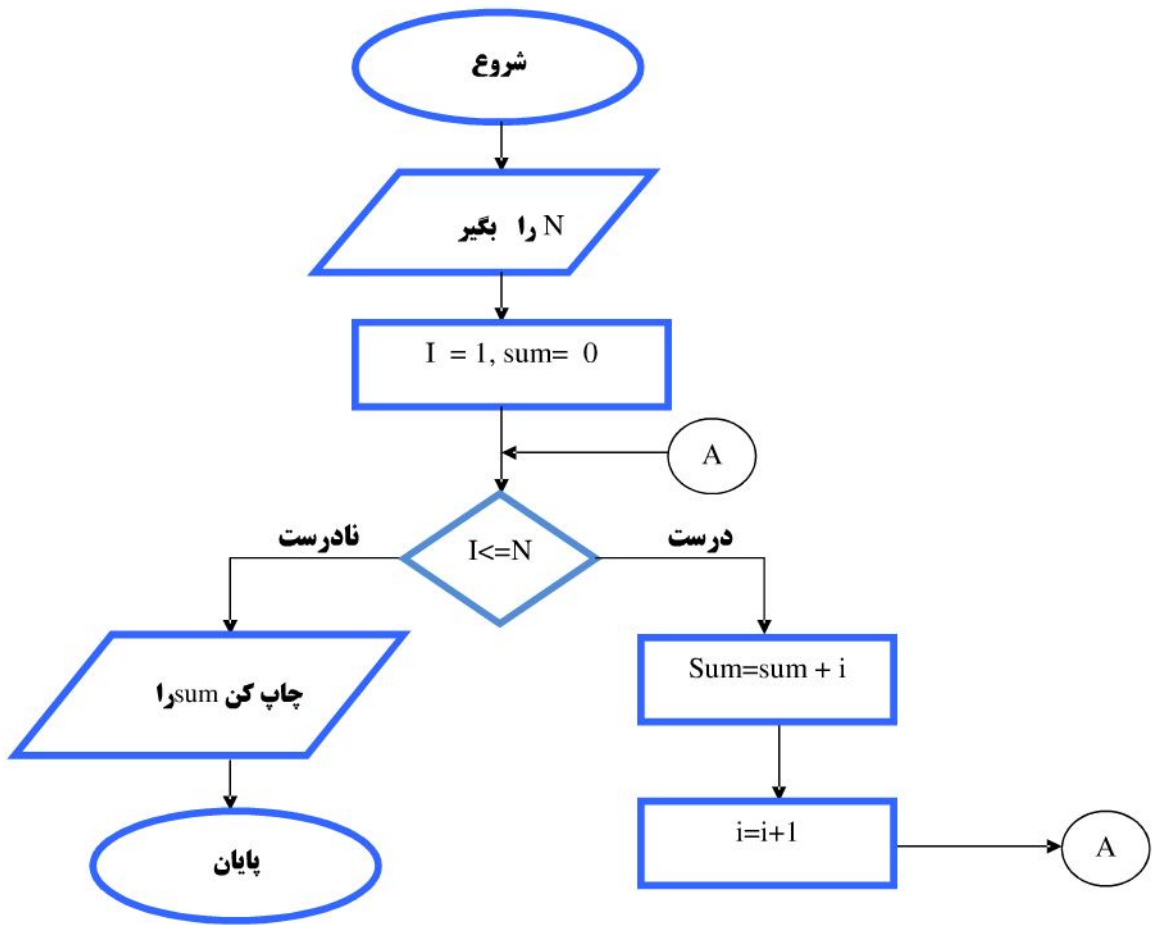


الگوریتمی بنویسید که تعداد n جمله از دنباله زیر را گرفته و چاپ کند:

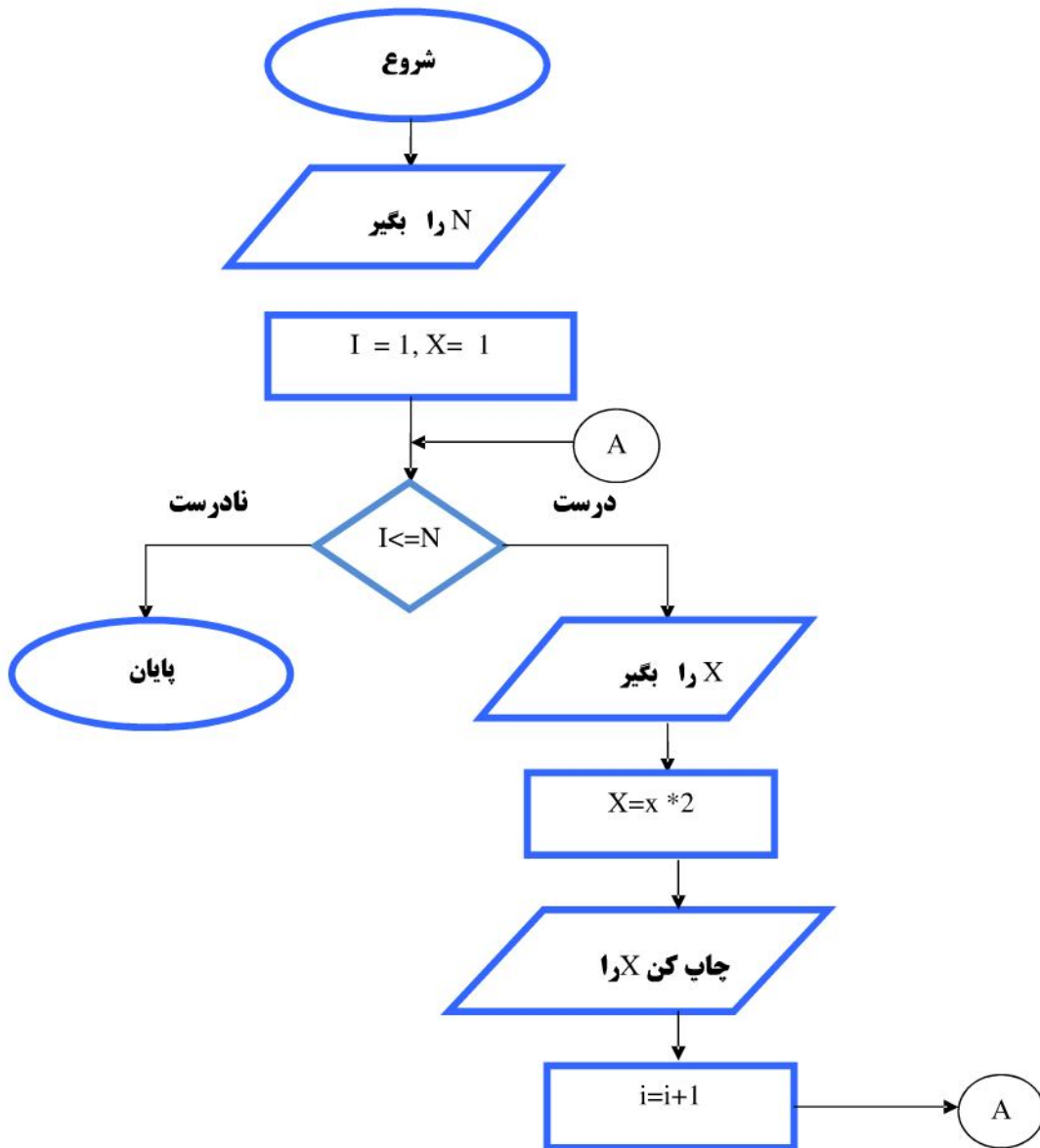
$$s = \frac{1}{1!} - \frac{1}{2!} + \frac{1}{3!} - \dots - \frac{1}{n!}$$



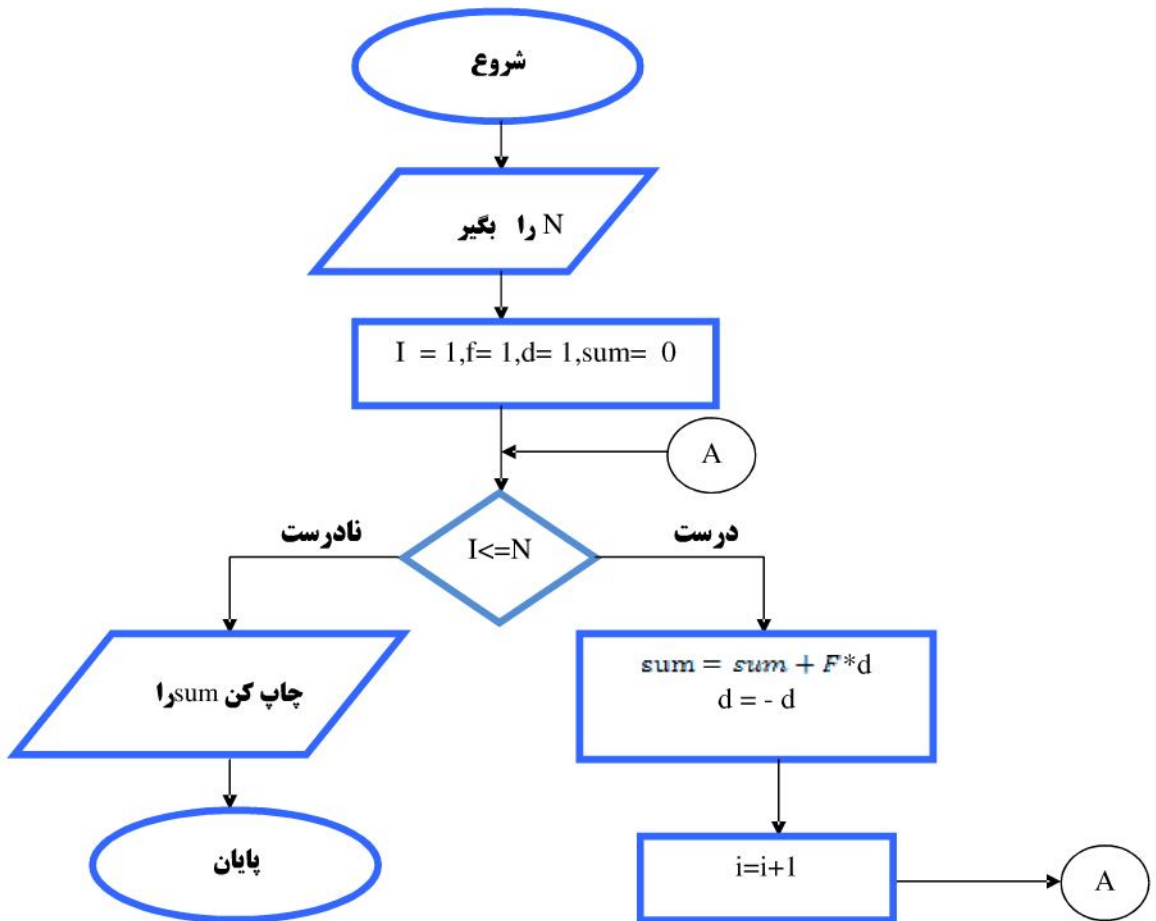
$$S=1+2+3+4+5+\dots+n$$



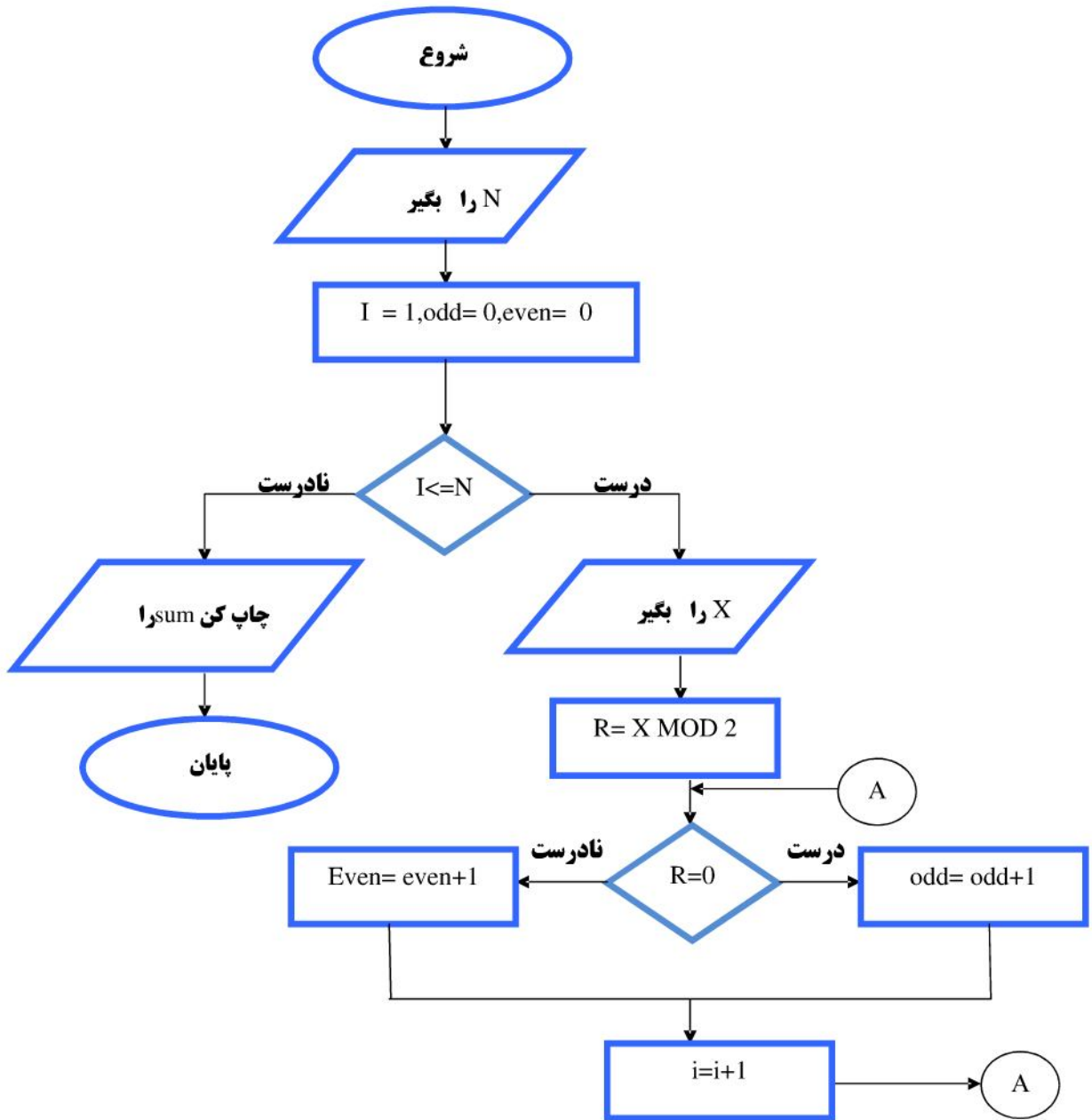
$S=1, 2, 4, 8, 16, 32, \dots, n$



$$S=1!+2!+3!+4!+5!+\dots,N!$$



الگوریتمی بنویسید که تعداد N عدد را گرفته تعداد اعداد زوج و فرد را چاپ کند:



مزایای ویژوال بیسیک:

خاصیت GUT: به وسیله آن می توان به آسانی برنامه های تحت سیستم عامل را ایجاد کرد و قبل از اجراء اشکال ظاهری آن را مشاهده کرد.

ویژوال بیسیک به صورت گرافیکی با کاربر و برنامه نویسی ارتباط برقرار می کند.

2- خاصیت RAD: طراحی و تولید برنامه در VB به دلیل وجود ابزار های مناسب به سرعت انجام می شود.

3- خاصیت IDC: به ما امکان می دهد که برنامه های تحت ویندوز خود را بدون نیاز به استفاده از برنامه های کاربردی دیگر اجراء و خطایابی کنیم.

4- خاصیت شی گراء: برنامه نویسی شی گراء شامل تعدادی (شی) با خواص و رفتارهای متفاوت است که رابطه خاصی بین آنها برقرار است بعد از انتقال یک شی بر روی فرم می توان با برنامه نویسی برای آن رفتار مشخصی تعریف کرد و رابطه آن را با اشیاء دیگر مشخص کرد، تعدادی از اشیاء را می توان بصورت یک مجموعه کلاس تعریف کرد و مورد استفاده قرار داد. خاصیت تابع گرائی: در این روش یک تابع اصلی یا بدنه اصلی وجود دارد که تابع از درون آن صدا زده می شود و تابع از بالا به پایین و خط به خط اجراء می شود.

5- خاصیت ساخته یافته: برنامه به بخش های کوچکتر و مجزا از هم تقسیم میشوند در نتیجه خوانایی برنامه بیشتر می شود، خطایابی و خطازدایی برنامه راحت تر می شود در نتیجه هزینه و سرعت تولید نرم افزار بهتر می شود. داده: برای انجام محاسبات نیاز به ورود داده و نگهداری آنها در حافظه داریم برای اینکار از متغیر ها و ثابتها استفاده می کنیم. متغیر: مکان هایی از حافظه هستند که برای نگهداری داده ها استفاده می شوند. متغیر ها به وسیله نام و نوع داده معرفی می شوند، نوع داده تعیین می کند که چگونه داده در فیلد های حافظه کامپیوتر ذخیره می شوند به وسیله نوع داده VB متوجه می شویم باید چه نوع و با چه نوع داده ای از اطلاعات سروکار داشته باشیم.

نوع	میزان حافظه	بازه قابل قبول
integer	2 بایت	32768- تا 32767
double	8 بایت	منفی از $-1/79769313486232+e308$ تا $-4/94065645841247e-324$ مثبت از $4/94065645841247e-324$ تا $-1/79769313486232+e308$
currency	8 بایت (برای محاسبه پولی)	از $5807/922337203685477$ تا $-922337203685477/5808$
byte	1 بایت	0 تا 255
long	4 بایت	2147483648- تا 214748364
single	4 بایت (اعشاری)	منفی از $-1/401298e-45$ تا $-3/402823e+38$ مثبت از $3/402823e+38$ تا $1/401298e-45$
String (نوع ثابت)	طول رشته ای	از یک تا 65400 کاراکتر
String (نوع متغیر)	طول رشته +10 بایت	از 0 تا 2 میلیارد کاراکتر
Boolean	2 بایت	True یا false
date	تاریخ	-----
variant (عددی)	16 بایت	به اندازه نوع عددی double
variant (رشته ای)	طول +22 بایت	مانند رشته طول متغیر

Dim	نام متغیر	As	نوع متغیر
Dim	X	As	Integer
Dim	نام متغیر	As	String
Dim	نام متغیر رشته ای (طول ثابت)	As	نوع متغیر n
Dim	chr	As	String*3

عملگرها: محاسباتی - رابطه ای - منطقی

1- محاسباتی

نتیجه	مثال	نام	عملگر
9	$3 \wedge 2$	توان	\wedge
20	$4 * 5$	ضرب	$*$
8	$16 / 2$	تقسیم	$/$
7	$2 + 5$	جمع	$+$
5	$9 - 4$	تفریق	$-$
0	$15 \text{ MOD } 3$	باقیمانده	MOD
4	$14 \setminus 3$	تقسیم صحیح	\setminus
"Ali"	"Al" + "i"	الحاق رشته ها	& , +

2- رابطه ای

نتیجه	نام	عملگر
$X < y$	کوچکتر	$<$
$X \leq y$	کوچکتر مساوی	\leq
$x > y$	بزرگتر	$>$
$X \geq y$	بزرگتر مساوی	\geq
$X \diamond y$	نامساوی	\diamond

3- منطقی

نتیجه	نام	عملگر
NOT X	نقض	NOT
X AND Y	و	AND
X OR Y	یا	OR
X XOR Y	بای انحصاری	XOR
X EQV Y	هم ارزی	EQV
X IMP Y	مشمول	IMP

X IMP Y	X EQV Y	X XOR Y	X OR Y	X AND Y	NOT X	Y	X
T	T	F	T	T	F	T	T
F	F	T	T	F	F	F	T
T	F	T	T	F	T	T	F
F	T	F	F	F	T	F	F

اولویت عملگرها:

اگر در یک عبارت ، عملگرهای محاسباتی ، رابطه ای و منطقی با هم وجود داشته باشند، تقدم آنها به شرح زیر است:

1- عملگرهای محاسباتی (عبارتهای ریاضی)

2- عملگرهای رابطه ای

3- عملگرهای منطقی

تمرین 1: برنامه ای بنویسید که دو عدد را از ورودی خوانده و جمع و تفریق کند؟

مراحل طراحی و اجرای برنامه

یک عنصر label و دو عنصر Text Box و سه عنصر Command Button به فرم اضافه کنید.

تغییرات : خاصیت Caption و Name سه دکمه را به ترتیب Run1 و Run2 و Exit را تغییر دهید. (مانند شکل بالا)

دکمه Run1 را دوبار کلیک کرده و دستورات زیر را تایپ می کنیم:

```
Private Sub Run1_Click ()
Dim Jam as long
Sum=val(text1.text)+val(text2.text)
Label1.caption= jam
End Sub
```

دکمه Run2 را دوبار کلیک کرده و دستورات زیر را تایپ می کنیم:

```
Private Sub Run2_Click ()
Dim Tafrig as long
Sum=Val (text1.text) +Val (text2.text)
Label1.caption = Tafrig
End Sub
```

دکمه Exit را دو بار کلیک کرده و دستور زیر را تایپ می کنیم:

```
Private Sub Run1_Click ()
END
End Sub
```

تمرین 2: برنامه ای بنویسید که سن شما را بر حسب روز از ورودی گرفته و تعیین کند چند سال و چند ماه و چند روز سن شماست؟

توضیح: برای انجام این مثال از تقسیم صحیح و باقی مانده تقسیم استفاده می شود. برای بدست آوردن سال ، باید تعداد روزها را بر

365 روز سال به دست آورد. این روند برای محاسبه تعداد ماه ، هفته و روز نیز به کار می رود.

در این

برنامه DAYS سن شما به روز، Y تعداد سال، M تعداد ماه و W تعداد هفته است. سرانجام روزهای باقی مانده نیز در DAYS قرار می گیرند.

مراحل طراحی و اجرای برنامه

دو عنصر Text Box و دو عنصر Command Button به فرم اضافه کنید.

تغییرات: خاصیت caption و Name دو دکمه را به ترتیب exit و display را تغییر دهید. (مانند شکل بالا)

دکمه display را دوبار کلیک کرده و دستورات زیر را تایپ می کنیم:

```
Private Sub Run1_Click ()
Dim days as double
Dim y as integer
Dim m as integer
Dim w as integer
Dim d as integer
Days=val(text1.text)
Y=days \ 365
Days=days mod 365
M=days \ 30
Days=days mod 30
W=days \ 7
Days=days mod 7
Text2.text="year="+str(y)+"month="+str(m)+"week="+str(w)+"days="+str(days)
End Sub
```

دکمه exit را دوبار کلیک کرده و دستورات زیر را تایپ می کنیم:

```
Private Sub exit_Click ()
End
End Sub
```

تمرین در منزل 1: برنامه ای بنویسید که دو عدد صحیح را از ورودی خوانده و با کلیک یک دکمه مجموع مربعات و با کلیک دکمه دیگر حاصل ضرب دو عدد را نمایش دهد؟

مراحل طراحی و اجرای برنامه

سه عنصر Text Box و دو عنصر Command Button به فرم اضافه کنید.

تغییرات: خاصیت caption و Name دو دکمه را به ترتیب Multiplay و Sumsqr را تغییر دهید. (مانند شکل بالا)

دکمه Sumsqr را دوبار کلیک کرده و دستورات زیر را تایپ می کنیم:

```
Private Sub Sumsqr_Click ()
Num1=val(text1.text)
Num2=val(text2.text)
Num3=num1^2+num2^2
Text3.text="Sumsqr;" +str(num3)
End Sub
```

دکمه Multiply را دوبار کلیک کرده و دستورات زیر را تایپ می کنیم:

```
Private Sub Multiply_Click ()  
Num1=Val (text1.text)  
Num2=Val (text2.text)  
Num3=num1*num2  
Text3.text="multiply;"+str (num3)  
End Sub
```

ساختار های شرطی:

اگر	شرط	آنگاه	دستور
اگر	$x > 0$	آنگاه	$x + 1$
If	شرط	then	
دستورات			
Else			
دستورات			
End if			

مثال 1: برنامه ای بنویسید که دو مقدار را از ورودی خوانده و کوچکترین مقدار را در خروجی چاپ کند؟

مراحل طراحی و اجرای برنامه

یک عنصر label و یک عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به min و exit تغییر دهید.

دکمه min را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub min_Click ()  
Dim a as integer  
Dim b as integer  
Dim min as integer  
A=text1.text  
B=text2.text  
If a>b then  
Min=b  
Else  
Min=a  
End if  
Label1.caption=min  
End Sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
end  
End Sub
```

تمرین در منزل 2: برنامه ای بنویسید که عددی را از ورودی گرفته تعداد رقم های آن را چاپ کند؟

مراحل طراحی و اجرای برنامه

دو عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
Dim num as integer
Dim digit as integer
Num=Val (text1.text)
If num<10 then
Digit=1
Else If num<100 then
Digit=2
Else If num<1000 then
Digit=3
Else If num<10000 then
Digit=4
Else
Digit="bozorg tar az 4 ragam"
Text2.text=digit
End Sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
end
End Sub
```

تابع IIF

این تابع همانند دستور IF عمل می کند با این تفاوت که می توان از این تابع همواره در دستور IF با بدنه یک سطر استفاده کرد. تابع IIF به صورت زیر است

(دستور 2 و دستور 1 و شرط)

در این تابع چنانچه شرط درست باشد ، دستور 1 و گرنه دستور 2 اجرا می گردد. به عنوان مثال، دستور زیر را در نظر بگیرید:

MIN=IIF (A<B,A,B)

در مثال بالا چنانچه a کوچکتر از B باشد در MIN قرار می گیرد و گرنه B در MIN قرار می گیرد. این دستور معادل دستور IF زیر می باشد.

```
IF A<B THEN
MIN=A
ELSE
MIN=B
END IF
```

IF های تو در تو

if شرط Then

دستورات 1

Else

دستورات 2

Else

دستورات 3

Else

...

End if

مثال 1: برنامه ای بنویسید که عددی از ورودی خوانده و تعیین علامت کند اگر:

positive $\longrightarrow x > 0$

negative $\longrightarrow x < 0$

zero $\longrightarrow x = 0$

مراحل طراحی و اجرای برنامه

دو عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
Dim a as integer  
A=text1.text  
If a>0 then  
Text2.text="positive"  
Else if a<0 then  
Text2.text="negative"  
Else  
Text2.text="zero"  
End if  
End Sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
end  
End Sub
```


مثال 2: برنامه ای بنویسید که عدد را از ورودی خوانده اگر عدد از بزرگتر از 0 باشد به توان 2 برساند و اگر بزرگتر از 0 نباشد عبارت مناسبی را چاپ کند؟

مراحل طراحی و اجرای برنامه

یک label و دو عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
Text2.text=""  
A=text1.text  
If a>0 then  
Text2.text=a^2  
Else  
Label1.caption="not true number"  
End if  
End Sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

دستور select case :

این دستور یک ساختار تصمیم گیری چندگانه است . این ساختار بسته به اینکه مقدار عبارت مورد آزمایش با عبارت کدام case برابر است یکی از چندمین گروه از دستورات را اجرا می کند.

```
select case <عبارت>  
Case  
دستورات 1  
Case  
دستورات 2  
Case  
دستورات 3  
Case  
دستورات 4  
...  
Case else  
مجموعه دستورات n  
End select
```

مثال 1: برنامه ای بنویسید که عدد را از ورودی گرفته اگر عدد 1 باشد از 0 باشد عبارت own و اگر بین دو تا ده باشد عبارت 10 < number is old اگر بین عدد 10 تا 20 باشد عبارت 20 < number is small اگر عدد 22 و 24 و 26 باشد چاپ کند number is old در غیر این صورت درون یک msgbox چاپ کند number is not true

مراحل طراحی و اجرای برنامه

یک label و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
Dim a as integer
A=inputbox("add number")
Select case a
Case 1
Label1.caption="own"
Case 2 to 10
Label1.caption="number is small<10"
Case 11 to 20
Label1.caption=" number is small<20"
Case 22,24,26
Label1.caption=" number is old"
Case else
Msgbox ("number is not true ")
End select
End Sub
```

دکمه end را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

مثال 2: برنامه ای بنویسید که عدد را از ورودی گرفته و تعیین علامت کند:

مراحل طراحی و اجرای برنامه

یک label و یک عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید. راهنمایی (از select case استفاده کنید)

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
X=text1.text
Select case x
Case is >0
Label1.caption=" +"
Case is <0
Label1.caption=" -"
Case is =0
Label1.caption=" 0"
```

```
End select
```

```
End sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
```

```
End
```

```
End Sub
```

مثال 3: برنامه ای بنویسید که یک عدد مثبت را از ورودی گرفته و تعداد ارقامش را مشخص کند:

مراحل طراحی و اجرای برنامه

یک label و یک عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
```

```
Dim a as integer
```

```
Select case x
```

```
Case is <10
```

```
Label1.caption=" 1"
```

```
Case is <100
```

```
Label1.caption=" 2"
```

```
Case is <1000
```

```
Label1.caption="3"
```

```
Case is <10000
```

```
Label1.caption=" 4"
```

```
Case is <100000
```

```
Label1.caption=" 5"
```

```
Case is <1000000
```

```
Label1.caption=" 6"
```

```
Case is <10000000
```

```
Label1.caption=" bozorgtar az 6 ragam"
```

```
End select
```

```
End Sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
```

```
End
```

```
End Sub
```

مثال 4: برنامه ای بنویسید که روزهای هفته را با استفاده از دکمه های option نمایش دهد ، با انتخاب یک دکمه از option

که نام روز هفته می باشد ، شماره روزهای هفته نمایش داده شود و با ورود یک عدد (شماره روز هفته) textbox نام روز

هفته را نمایش دهد و دکمه option مربوط به آن را فعال کند:

راهنمایی (از select case استفاده کنید)

یک label و یک عنصر textbox و سه عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به number و select و exit تغییر دهید.

دکمه number را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub number_Click ()
X=text1.text
Select case x
Case 1
Label1.caption=" shanbe"
Option1.value=true
Case 2
Label1.caption="1 shanbe"
Option2.value=true
Case 3
Label1.caption="2 shanbe"
Option3.value=true
Case 4
Label1.caption="3 shanbe"
Option4.value=true
Case 5
Label1.caption="4shanbe"
Option5.value=true
Case 6
Label1.caption="5 shanbe"
Option6.value=true
Case 7
Label1.caption="jomeh"
Option7.value=true
End select
End Sub
```

دکمه select را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub select_Click ()
Dim x as integer
If option 1.value=true then text1.text=1
If option 2.value=true then text1.text=2
If option 3.value=true then text1.text=3
If option 4.value=true then text1.text=4
If option 5.value=true then text1.text=5
If option 6.value=true then text1.text=6
If option 7.value=true then text1.text=7
else if text1.text==" not true number"
End Sub
```

دکمه exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

حلقه تکرار :

بسیاری از کارها تکراری هستند مثل خواندن اسامی 20 نفر از دانشجو یا خواندن 20 عدد صحیح برای انجام این کارها از حلقه های تکرار استفاده می کنیم:

```
For مقدار اولیه=شمارنده To مقدار نهایی Step (گام افزایشی)  
  
مجموعه دستورات  
  
NEXT
```

مثال 1: برنامه ای بنویسید که اعداد زوج بین 1 تا 100 را چاپ کند:

راهنمایی (از حلقه For استفاده کنید)

مراحل طراحی و اجرای برنامه

دو عنصر List و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
For I=1 to 100  
List1.additem str (i)  
Next i  
For j=2 to 100  
List2.additem str (j)  
Next j  
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

مثال 2: برنامه ای بنویسید که مضرب های 3 بین 1 تا 100 و مجموع آنها را نیز چاپ کند:

راهنمایی (از حلقه For استفاده کنید)

مراحل طراحی و اجرای برنامه

دو عنصر List و یک عنصر textbox دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
Dim sum as integer  
For I=1 to 100  
Sum=sum +i  
List1.additem str (i)  
Next i  
Text1.text=str(sum)  
For j=3 to 100 step 3  
List2.additem str (j)  
Next j  
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

مثال 3: برنامه ای بنویسید که مجموع مربعات اعداد 1 تا 10 را نمایش دهد:
راهنمایی (از حلقه For استفاده کنید)

مراحل طراحی و اجرای برنامه

دو عنصر List و یک عنصر label و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
Sum=0  
For i=1 to 10  
A=inputbox= ("Enter number")  
Sum=sum+a ^ 2  
List 1.additem (i)  
Next i  
Label1.caption= ("sum="+str(sum))  
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

مثال 3: برنامه ای بنویسید که با استفاده از جمع، دو عدد را در هم ضرب کند:

راهنمایی (از حلقه For استفاده کنید)

مراحل طراحی و اجرای برنامه

یک عنصر label و دو عنصر textbox و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
Dim x as integer
Dim y as integer
Dim m as integer
Dim sum as integer
X=Val (text1.text)
Y=Val (text2.text)
M=y
If y<0 then m=-y
Sum=0
For i= 1 to m
Sum=sum+x
Next i
If y<0 then sum=-sum
Label1.caption=str(sum)
End if
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

حلقه for تو در تو

به ازای هر یک بار اجرای حلقه for بیرونی حلقه for داخلی به طور کامل اجرا می شود:

مثال:

```
FOR I=1 TO 4
FOR J=1 TO 3
X=X+1
NEXT J
NEXT I
```

مثال 1: برنامه ای بنویسید که اشکال ستاره ای شکل را به صورت نزولی چاپ کند:

راهنمایی (از حلقه For های تو در تو استفاده کنید)

مراحل طراحی و اجرای برنامه

دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
Dim I as integer  
Dim J as integer  
For I=1 TO 5  
For J=1 TO 5  
Print="*"  
Next j  
Print  
Next i  
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

مثال 2: برنامه ای جدول ضرب 10*10 را چاپ کند:

راهنمایی (از حلقه For های تو در تو استفاده کنید)

مراحل طراحی و اجرای برنامه

یک عنصر MsFlexGrid و دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
For i=1 to 10  
For j=1 to 10  
MsFlexGrid1.Row=i-1  
MsFlexGrid1.col=j-1  
MsFlexGrid1.text=i*j  
Next j  
Next i  
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

حلقه های تکرار While, Wend, Do... While, Do Loop

در حلقه while...wend اگر شرط درست باشد دستورات درون شرط اجرا می شود.

While شرط

دستورات باید اجرا شوند

Wend

در این روش ، شرط موجود در حلقه تست می شود، چنانچه شرط درست باشد کنترل برنامه وارد حلقه تکرار می شود و دستورات داخل حلقه تکرار اجرا می شوند و اگر شرط برقرار نباشد ، دستورات حلقه اجرا نمی شوند، شرط حلقه تکرار باید داخل حلقه نقض شود تا شرط نادرستی پیدا کند و کنترل برنامه از حلقه خارج شود. به عنوان مثال دستورات زیر را در نظر بگیرید.

```
N=Val (text1.text)
List1.Clear
I=5
While I <= n
List1.AddItem Str(I)
I=I+5
Wend
```

این دستورات ، یک عدد را از ورودی گرفته اعداد مضرب 5 از 1 تا عدد خوانده شده را ، به گزینه های List1 اضافه می کند:

در دستورات Do...Loop حتی اگر شرط نادرست باشد دستورات حداقل یک بار اجرا می شوند.

Do < شرط >

دستورات

Loop

ساختار 1

```
I=0
Count=List1.Count
Do
List1.Removeitem 0
I=I+1
LoopWhileI<Count
```

Do

دستورات

While Loop < شرط >

ساختار 2

ساختار 1 دقیقاً شبیه While...Wend است ولی تفاوت ساختار 2 با ساختار While...Wend این است که دستورات حلقه تکرار ساختار 2 ، حداقل 1 بار اجرا می شوند سپس شرط تست می شود. به عنوان مثال دستورات زیر را در نظر بگیرید:

Do

مجموعه دستورات

Loop Until<شرط>

For (خروج از حلقه) Exit for
 Loop (خروج از حلقه) Exit for
 For I= 1 to 10
 If i=3 Then Exit For

وقتی کنترل اجراء برنامه به این ساختار رسید، دستورات داخل حلقه اجرا می شوند. پس از اجرای دستورات ، شرط حلقه تست می شود ، چنانچه شرط نادرست باشد اجرای دستورات ادامه می یابد و گرنه کنترل از حلقه تکرار خارج می شود. در مورد این حلقه دقت داشته باشید که چنانچه شرط حلقه نادرست باشد دستورات حلقه یک بار اجرا می شوند.

✓ تمرین 1: برنامه ای بنویسید که یک عدد چهار رقمی را از ورودی گرفته و به صورت (?) کدگذاری کند:

الف: باقی مانده هر رقم را بر چهار بدست آورد؟

ب: جای عدد اول و سوم را عوض کند؟

تمرین 2: برنامه ای بنویسید که شماره کارمندی و حقوق یک کارمند را دریافت کرده و براساس تعریف زیر حقوق آن را حساب کرده و در خروجی چاپ کند :

معاف از مالیات ≤ 4600000 حقوق

7.10 مالیات نسبت به مازاد ≤ 5600000 ≤ 4600001 = حقوق

7.13 مالیات نسبت به مازاد ≤ 7600000 ≤ 5600001 = حقوق

7.25 مالیات نسبت به مازاد ≤ 7600001 = حقوق

راهنمایی (از select case استفاده کنید)

مراحل طراحی و اجرای برنامه

یک عنصر label دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run و exit تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()
Dim a as long
a=Inputbox ("Enter Number")
Select case a
Case is <=4600000
Label1.caption=a
Case is 4600001 to 5600000
z=a-4600000
h=z*1.1
f=h-z
Label1.caption=f
Case 5600001 to 7600000
x=a-4600000
y=x-1000000
q=w-y+100000
```

```

Label1.caption=q
Case is >=7600001
r=a-7600000
u=r*1.25
p=u-r+360000
label1.caption=p
End select
End sub

```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```

Private Sub exit_Click ()
End
End Sub

```

مثال 1: برنامه ای بنویسید که دو عدد را به روش تفریق بر هم تقسیم کند:

مراحل طراحی و اجرای برنامه

یک عنصر label یک عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به run تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```

Private Sub run_Click ()
Dim a as integer
Dim b as integer
A=Val (text1.text)
B=Val (text2.text)
I=0
While (a>=b)
A=a-b
I=i+1
Wend
Print i
Print a
End sub

```

مثال 2: برنامه ای بنویسید که دو عدد را به روش تفریق بر هم تقسیم کند:

مراحل طراحی و اجرای برنامه

یک عنصر label دو عنصر command button به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به Count و exit تغییر دهید.

دکمه Count را دو بار کلیک کنید و دستورات زیر را تایپ کنید:

```

Private Sub Cout_Click ()
Dim number as integer
Dim a as integer
Number=0
A=Val (text1.text)
While a>0
If a Mod 10 <> 0 then number = number+1

```



```
A=a\10
Text2.text=str (number)
Wend
End sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

مثال 3: برنامه ای بنویسید که یک عدد را خوانده مجموع اعداد زوج تا آن عدد را یا مجموع اعداد فرد تا آن عدد را حساب کند یک عنصر Check Box تعیین کند چه عملی انجام شود. اگر Checkbox انتخاب شود مجموع اعداد زوج و گرنه مجموع اعداد فرد تا آن عدد را محاسبه کند.

مراحل طراحی و اجرای برنامه

دو عنصر label دو عنصر command button و یک عنصر Check Box و یک عنصر Textbox به فرم اضافه کرده ، خاصیت name و caption دو دکمه را به ترتیب به Display و exit تغییر دهید.

دکمه Display را دو بار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub Display_Click ()
:10
A=Val (text1.text)
If a < 1 then Goto 10
If Check1.Value=1 Then
I=2
Else
I=1
End if
Sum=0
Do
Sum=sum+ i
I=i+2
Loop Until I >a
Label1.caption=str (sum)
End sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

آرایه ها:

مجموعه ای از متغیر ها می باشد که دارای یک نوع مثلا عددی یا رشته ای بوده و دارای نام یکسانی هستند برای دستیابی به هر یک از آنها از اندیس استفاده می شود

توجه: دستور Option Base باید قبل از دستور معرفی آرایه قرار گیرد. به عنوان مثال، دستورات زیر را در نظر بگیرید

```
Option Base 1  
Dim a (5) As Integer
```

A (0)	A (1)	A (2)	A (3)	A (4)
-------	-------	-------	-------	-------

Dim a (1 to 6) as integer

چنانچه اندیس شروع آرایه ذکر نشود از اندیس ابتدایی صفر خواهد بود A (0)

مثال 1: برنامه ای بنویسید که دو عدد را به روش تفریق بر هم تقسیم کند:

مراحل طراحی و اجرای برنامه

دو عنصر List و دو عنصر Command Button به فرم اضافه کرده، خاصیت name و caption دکمه را به ترتیب به exit و run تغییر دهید.

دکمه run را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub run_Click ()  
Dim a (1 to 25) As Integer  
Dim sum As Long  
Dim Msum As Long  
For i=1 to 25  
A (i) =Val (inputbox ("Enter Number"+str (i), ""))  
List1.AddItem Str (a(i))  
Next i  
For i=1 to 25  
Sum =sum+a (i)  
List2.AddItem Str (a (i))  
Next i  
Label1.Caption =Val (sum)  
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()  
End  
End Sub
```

مثال 2: برنامه ای بنویسید که 10 عدد را گرفته و در آرایه ای قرار دهد و سپس کوچکترین عدد و مکان آن را نمایش دهد:

مراحل طراحی و اجرای برنامه

یک عنصر List و دو عنصر Textbox و دو عنصر Command Button به فرم اضافه کرده، خاصیت name و caption دکمه را به ترتیب به Min و exit تغییر دهید.

دکمه Min را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub Min_Click ()
Dim I, min, lmin as integer
Dim a (1 to 10) As Integer
For i=1 to 10
A (i) =Val (inputbox ("Enter Number"+str (i), ""))
Next i
Min=a(1)
Lmin=1
For i=2 to 10
If a(i) < min then min=a(i) : lmin=i
Next i
For i=1 to 10
List1.AddItem Str (a (i))
Next i
Text1.text= Str (Min)
Text2.text= Str (Lmin)
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

مثال 3: برنامه ای بنویسید که 10 عدد را گرفته و در آرایه ای قرار دهد و سپس مشخص کند چه عددی بیشتر از همه تکرار شده است (چنانچه عددی تکراری وجود نداشت پیامی صادر شود):

مراحل طراحی و اجرای برنامه

یک عنصر List و دو عنصر Textbox و دو عنصر Command Button به فرم اضافه کرده ، خاصیت name و caption دکمه را به ترتیب به Start و exit تغییر دهید.

```
Private Sub Start_Click ()
Dim a (1 to 10) as integer
For i=1 to 10
Prompt="Enter Number "+Str (i)
A (i) =Val (inputbox (prompt, "Enter Date", ""))
List1.AddItem Str (a (i))
Next i
Max count=0
For i=1 to 10
Count1=0
For j=i to 10
If a (i) =a (j) Then Count1=count+1
Next j
If Count1>maxcount then
Max count = count1
Mean=a (i)
```

www-m-a-h-d-i.loxblog.com

```

End if
Next i
If maxcount > 1 then
Text1.text="number=" + Str (mean)
Text2.text="Max Count=" + Str (Maxcount)
Else
Text1.text= ""
Text2.text=""
Msgbox ("not true number")
End if
End Sub

```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```

Private Sub exit_Click ()
End
End Sub

```

نکته: پس از وارد کردن 10 عدد، عددی که بیشترین تکرار را دارد در text1 ذخیره شده و تعداد تکرار آن در text2 ذخیره می شود.

مثال 4: برنامه ای بنویسید که 10 عدد را گرفته و میانگین عدد های وارده را حساب کند و اعدادی که بیشترین میانگین هستند همراه با میانگین نشان دهد، همچنین انحراف معیار هر کدام را چاپ کند:

مراحل طراحی و اجرای برنامه

سه عنصر List و دو عنصر Textbox و دو عنصر Command Button به فرم اضافه کرده، خاصیت name و caption دکمه را به ترتیب به Start و exit تغییر دهید.

```

Private Sub Start_Click ()
Dim a(1 to 10) as integer
Dim sum, avg as integer
Sum=0
For i=1 to 10
A (i) =Val (inputbox (prompt, "Enter Date", ""))
Sum=sum + a (i)
List1.AddItem Str (a (i))
Next i
Avg=sum/10
For i=1 to 10
List3.AddItem Val (a (i))-Val (avg)
If a (i) >=avg then List2.add Item Str (a (i))
Next i
Text1.text=str (avg)
End Sub

```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```

Private Sub exit_Click ()
End
End Sub

```

نکته: پس از وارد کردن 10 عدد، همه اعداد در list1 قرار گرفته، اعدادی که بیشتر از میانگین هستند به همراه میانگین در list2 قرار گرفته، در list3 انحراف معیار، میانگین اعداد در عنصر text1 چاپ می شوند.

www-m-a-h-d-i.loxblog.com

تمرین : برنامه ای بنویسید که معدل 20 دانشجو را گرفته و دومین معدل را از نظر بزرگی پیدا کند :
 دکمه Max را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub Max_Click ()
Dim I, max as integer
Dim a (1 to 20) as Integer
For i=1 to 20
A (i) =Val (inputbox (prompt, "Enter Date", ""))
List1.AddItem Str (a (i))
Next i
For i=1 to 20
For j=I to 20
If a (i) > a (j)
A (j) = Temp
Next j
Next i
For i= 1 to 20
List2.AddItem Str (a (i))
Next i
Max= a (1)
For i=2 to 20
If a (i) > max then max=a (i)
Next i
For i=1 to 20
If a(i)=max then
Max= Str (a (i-1))
End if
Next i
Text1.text=max
End Sub
```

مرتب سازی به روش حبابی:

در این روش ، ابتدا اولین عنصر لیست با بقیه عناصر لیست مقایسه می شود ، اگر عنصری از اولین عنصر لیست کوچکتر باشد ، جای آن با اولین عنصر لیست عوض می شود . بدین ترتیب کوچکترین عنصر لیست به ابتدای لیست منتقل می شود و در مرحله دوم ، دومین عنصر با بقیه عناصر ها مقایسه می شود...

50	30	20	60	10
60	30	20	50	10
60	50	20	30	10
60	50	30	20	10
60	50	30	20	10

مرتب سازی به روش جستجوی دو دوئی:

جستجوی دو دوئی: عنصر وسط لیست پیدا می شود عدد مورد جستجو یا عنصر وسط لیست مقایسه می شود اگر مساوی بود عدد پیدا شده و اگر مساوی نبود عمل جستجو را یا در نیمه پایانی لیست یا در نیمه بالائی لیست ادامه می دهیم:

Low 60 50 40 30 20 10 1 High
Mid

Low=1 High=Mid+1
Low=Mid-1 High = 7

$$\text{Mid} = (\text{Low} + \text{High}) / 2$$

مثال 1: برنامه ای بنویسید که 10 عدد را گرفته و به روش حبابی مرتب کند:
دکمه Start را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub Start_Click ()
Dim a (1 to 10) as integer
Dim I as integer
Dim j, min as integer
For i=1 to 10
For j=I to 10
If a (i) < a (j)
Temp = a (i)
A (j)=temp
End if
Next j
Next i
For i=1 to 10
List2.AddItem Str (a (i))
Next i
End Sub
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

مثال 2: برنامه ای بنویسید که 10 عدد را گرفته و ب صورت صعودی یا نزولی به ترتیب در آرایه ای که خوانده و نیز یک عدد از ورودی گرفته و در این آرایه جستجو کند اگر پیدا کرد پیغام "number is found" اگر پیدا نشد پیغام "number is not found" را چاپ کند

دکمه Start را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub Start_Click ()
Dim a (1 to 10) as integer
Dim I,j as integer
For i=1 to 10
A (i) = Val (inputbox (Enter number, +Str (i), ""))
List1.AddItem Str (a (i))
Next i
```

www-m-a-h-d-i.loxblog.com

```

For i=1 to 10
For j=1 to 10
If a (i) < a (j) Then
Temp = a (i)
A (i) = A (j)
A (j) = temp
End if
Next j
Next i
For i=1 to 10
List2.AddItem Str (a (i))
Next i
Num=Val (text1.text)
Low=1
High=10
Find=True
While (Low + High) and Find=false
Mid1= (low + High) \2
If Num <a (Mid1) Then
Low=Mid1+1
Else
High=Mid1-1
End if
Wend
If Find=true then
R=msgbox ("The Num"+Str (Num)+ "Number Is a Find",vbOkOnly+vbQuestion, "message box")
Else
R=msgbox ("The Num"+Str (Num)+ "Number Is a not Find",vbOkOnly+vbQuestion, "message box")
End if
End Sub

```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```

Private Sub exit_Click ()
End
End Sub

```

زیر برنامه ها و توابع:

زیر برنامه ها به دو دسته تقسیم می شوند:

1) رویه زیر روالی

2) رویه تابعی

در روی زیر روالی چندین مقدار به برنامه فراخوان تحویل می شود ولی در رویه تابعی یک مقدار به برنامه فراخوان تحویل می شود.

Public Private Sub نام زیر روال (آرگومانها)

بدنه زیر روال

Col نام زیر روال

End sub

www-m-a-h-d-i.loxblog.com

در حالت Public زیر روال را در همه برنامه ها می توان اجرا نمود ولی در حالت Private فقط در برنامه ای که تعریف شده استفاده می شود . اما اگر بخواهیم از یک زیر روال در چندین ماژول استفاده کنیم از واژه Public استفاده می کنیم . در هنگام فراخوانی زیر برنامه ، اطلاعاتی را به آن تحویل می دهیم که به آن پارامتر می گویند . هنگام تعریف زیر برنامه باید پارامتر های آن را مشخص کرد و در هنگام فراخوانی زیر برنامه باید پارامتر ها را به زیر برنامه منتقل کرد.

انواع پارامتر های زیر روال :

- 1- پارامتر های مقدار با نام Value می باشد ✓
 2- پارامتر های ارجاع که همان Reference می باشد. ✓

پارامتر های مقدار (Value) : هر تغییری در پارامتر های مجازی تاثیری در پارامتر های واقعی ندارد پارامتر های مقدار با کلمه By Val و پارامتری که فاقد این کلمه می باشد از نوع ارجاع است و ممکن است با By Ref نیز نوشته شود به عنوان مثال عبارت زیر را در نظر بگیرید:

```
A=5
B=10
Coll Proc1 (a, b)
Text1.text=A
Text2.text=B
Private Sub Proc1 (By Val X as integer, By Val X as integer)
X=X*5
Y=Y-z
```

این مجموعه دستورات ، یک زیر روال به نام Proc1 را تعریف می کند که دو پارامتر از نوع Integer دارد.

انتقال آرایه به زیر روال:

آرایه را می توان مثل متغیر های معمولی به زیر برنامه انتقال داد.

ساختار برنامه اصلی:

```
Private sub proc ()
Dim a (10) As Integer
Coll Sum (a ())
End Sub
```

ساختار تابع:

```
Private sub (a ()As Integer)
.
.
End Sub
```

وقتی آرایه ای به صورت ارجاع به زیر برنامه ارسال شود آدرس آرایه به زیر برنامه فرستاده می شود و از طریق آدرس آرایه به تمام عناصر آن در برنامه دسترسی پیدا می کنیم . برای انتقال آرایه به زیر برنامه ، نام آرایه همراه با پارانتز باز و بسته به عنوان پارامتر واقعی ذکر می شود و پارامتر مجازی متناظر آن باید متغیری از نوع آرایه باشد به عنوان مثال ، دستورات زیر را در نظر بگیرید:

```
Private Sub Call_Click ()
Dim a (10) As Integer
.
.
```

```

Call Sum (a ())
End Sub
Private Sub Sum (B () as Integer)
.
.
End Sub

```

این دستورات ، یک آرایه به نام a را به زیر روال Sum انتقال می دهد . در هنگام فراخوانی زیر روال Sum، آرایه به عنوان پارامتر واقعی به صورت A() ذکر شده است . در هنگام تعریف ، پارامتر مجازی به صورت آرایه تعریف شد.(B () as Integer)

مثال 1: برنامه ای بنویسید که دو عدد صحیح را از ورودی گرفته و توسط زیر روال بزرگترین عدد را پیدا کرده و در خروجی چاپ کند.

```

Dim X, Y as Integer
X=Val (Text1.text)
Y=Val (Text2.text)
Coll Proc 1 (X, Y, Max)
End Sub

```

```

-----
Private Sub Proc1 (By Val as Integer, By Val B as Integer, By Val C as Integer)
If A>B then
CV = A
Else
C=B
End if
End sub

```

مثال 2: برنامه ای بنویسید که دو عدد صحیح را از ورودی گرفته و به روش حبابی و به صورت نزولی مرتب کرده و چاپ کند سپس اعداد را به صورت صعودی چاپ کند.

```

Private Sub Command1_click ()
Dim I, j, Temp, a (1 to 10) As Integer
Call Read (a ())
Call Hobabi (a ())
Call Show1 (a ())
End Sub
Private Sub Read (b () As Integer)
For i=1 to 10
B (i) = inputbox ("Vared", "Adad ra ")
List1.AddItem Str (b (i))
Next i
End sub
Private Sub Hobabi (b () As Integer)
For i = 1 to 10
For j= I +1 to 10
If B (i) >= B (j) Then
Temp=B (i)
B (i) = B (j)
B (j) = Temp
End If
Next j
Next i
End Sub

```

```
Private Sub Show1 (b () As Integer)
For I=10 to 1 Step -1
List2.AddItem Str (b (i))
Next i
End Sub
```

افزودن ماژول به برنامه :

مراحل افزودن ماژول به برنامه به صورت زیر است :

- ✗ 1- گزینه Project /Add Module را اجرا می کنیم، با اجرای این گزینه پنجره Add Module ظاهر می گردد. این پنجره دارای 2 دکمه است که عبارت است از:
 - ✗ دکمه New: برای افزودن ماژول جدید به کار می رود.
 - ✗ دکمه Existing: برای افزودن ماژول موجود به کار می رود. با انتخاب این دکمه می توان نام ماژول را انتخاب کرد.
- ✗ 2- دکمه New را انتخاب می کنیم تا صفحه ورودی ماژول ظاهر شود در این صفحه توابع و زیر روال های ماژول را تایپ می کنیم.
- ✗ 3- ماژول را ذخیره می کنیم.

در عبارت Function همیشه فقط یک عنصر تعریف می شود و ساختار آن به صورت ذیل می باشد:

Private Public Function Name (لیست آرگومان) as نتیجه

دستورات

```
Name=      عبارت
End Function
```

مثال 1: برنامه ای بنویسید که یک عدد از ورودی گرفته و فاکتوریل آن را چاپ کند:

توجه (این دستورات با استفاده از ماژول نوشته شده است)

```
Private Sub Fact_click ()
Dim a As Integer
A=Inputbox ("Enter Number")
Label1.Caption=Fact (a)
End Sub
```

بعد از انتخاب صفحه ماژول به ترتیبی که در بالا گفته شد دستورات زیر را در آن می نویسیم:

```
Public Function Fact (a As Integer) As Double
Dim I as Integer
Fact=Fact*i
Next i
End Function
```

دکمه Exit را دوبار کلیک کنید و دستورات زیر را تایپ کنید:

```
Private Sub exit_Click ()
End
End Sub
```

مثال 2: برنامه ای بنویسید که یک عدد از ورودی گرفته و فاکتوریل آن را چاپ کند:

$$S=1! +2! +3! +\dots+n!$$

```
Public Function S (a As Integer) As Double
Dim I as Integer
Fact=1
S=0
For I=1 To a
Fact=Fact * i
S=Fact +S
Next i
End Function
```

```
-----
Private Sub Command1_Click ()
Dim a As Integer
a=Inputbox ("Enter Number")
Label1.Caption=S (a)
End Sub
```

```
-----
Private Sub Command2_Click ()
End
End Sub
```

مثال 3: برنامه ای بنویسید که یک عدد از ورودی گرفته و فاکتوریل آن را چاپ کند:

$$S=1! -2! +3! -4! +5!\dots+n!$$

```
Public Function S (a As Integer) As Double
Dim I as Integer
Fact=1
S=0
H=1
For I=1 To a
Fact=Fact * i
S=(Fact *h)+3
H= -1 * h
Next i
End Function
```

```
-----
Private Sub Command1_Click ()
Dim a As Integer
a=Inputbox ("Enter Number")
Label1.Caption=S (a)
End Sub
```

```
-----
Private Sub Command2_Click ()
End
End Sub
```

سوالات میان فرم

1: برنامه ای بنویسید که یک عدد را از ورودی گرفته و توان 2 آن را به روش زیر محاسبه و چاپ کند:

N عدد فرد از 1 تا n مجموعه

$$6^2 = 1+3+5+7+9+11=36$$

```
Private Sub Command1_Click ()
A=Val (Text1.text)
Sum = 1
For I=1 to Val (a * 2) Step 2
Sum=Sum + I
Next I
Text2.text=Sum-1
Label1.Caption=sum-1
End Sub
```

```
-----
Private Sub Exit_Click ()
End
End Sub
```

2: برنامه ای بنویسید که با استفاده از حلقه تو در تو و عنصر MsFlexGrid خروجی زیر را چاپ کند:

```
1
2 2
3 3 3
4 4 4 4
.
.
.
10 10 10 10 10 10 10 10 10 10
```

```
Private Sub Command1_Click ()
Dim I as Integer
Dim j as Integer
For i=1 To 10
For j= i to 10
MsFlexGrid1.col=i-1
MsFlexGrid1.Text=j
Next j
Next i
End Sub
```

```
-----
Private Sub Exit_Click ()
End
End Sub
```