



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

برنامه نویسی کامپیوتر

آموزش نرم افزار MATLAB

فصل ششم: حلقه ها و شرط ها



هدف فصل :

در ابتدای فصل به اهمیت دستورات شرطی و نیز حلقه ها در ایجاد برنامه ها اشاره گردید. در این فصل به بیان این مفهوم مهم در محیط نرم افزار متلب خواهیم پرداخت .

برای نوشتن یک برنامه خوب ابتدا باید خوب الگوریتم کشید!



عملگرهای رابطه ای و منطقی:

هنگامی که در مورد انواع داده در متلب صحبت می کردیم، داده های منطقی و به اصطلاح (logical) معرفی گردید.

بله (True)



در متلب یعنی (1)

خیر (False)



در متلب یعنی (0)



عملگرهای رابطه ای:

عملگرهای رابطه ای در متلب شامل عملگرهای مقایسه ای مرسوم می باشند. از این عملگرها می توان برای مقایسه دو آرایه با یک اندازه یا مقایسه یک آرایه با یک مقدار عددی استفاده کرد.

معنی	عملگر رابطه ای
علامت کوچکتر	$<$
علامت کوچکتر و مساوی	$<=$
علامت بزرگتر	$>$
علامت بزرگتر و مساوی	$>=$
علامت مساوی	$==$
علامت نامساوی	\neq



عملگر منطقی :

این عملگرها برای ترکیب عبارت ها به کار می روند.

معنی	عملگر منطقی
AND	&
OR	
NOT	~



مثال:

```
>> 3>5
ans =
     0
>> 3<7
ans =
     1
>> a=[1 2;3 4]
a =
     1     2
     3     4
>> b=[1 3;5 2]
b =
     1     3
     5     2
>> a==b
ans =
     1     0
     0     0
```

```
>> a|b
ans =
     1     1
     1     1
>> a~=b
ans =
     0     1
     1     1
>> (a>2) | (b<3)
ans =
     1     0
     1     1
>> (a>2) & (b<3)
ans =
     0     0
     0     1
```

```
>> a
a =
     1     2
     3     4
>> b
b =
     1     3
     5     2
>> a(a>0)
ans =
     1
     3
     2
     4
>> b(b>2)
ans =
     5
     3
```



تقدم و تأخر :

در انجام عملیات محاسباتی و منطقی باید توجه کرد که این عملیات سلسله وار انجام می گیرد و در هنگام نوشتن یک دستور باید به این نکته توجه نمود. مثلا عملیات توان نسبت به عملیات جمع زودتر انجام می گیرد.

```
>> (2+3)^2-1*2  
ans =  
    23
```

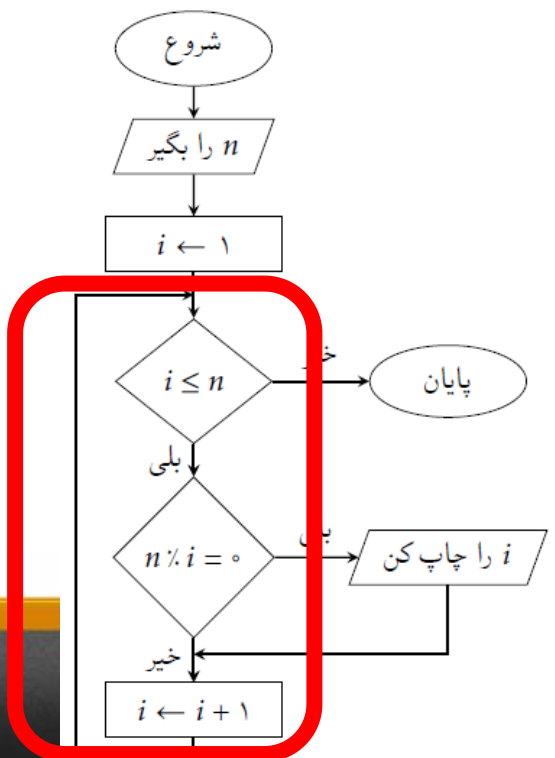
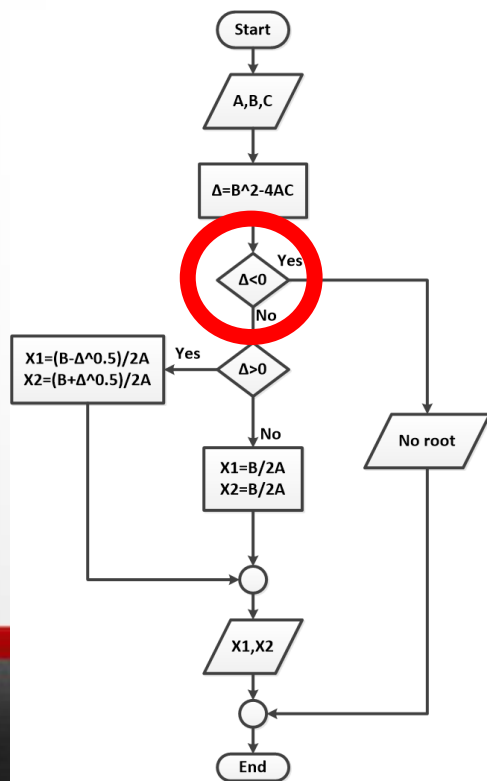
```
>> 1&0  
ans =  
     0  
  
>> 1|0  
ans =  
     1  
  
>> 1|0&0  
ans =  
     1
```

```
>> (1|0)&0  
ans =  
     0  
  
>> 1|(0&0)  
ans =  
     1
```



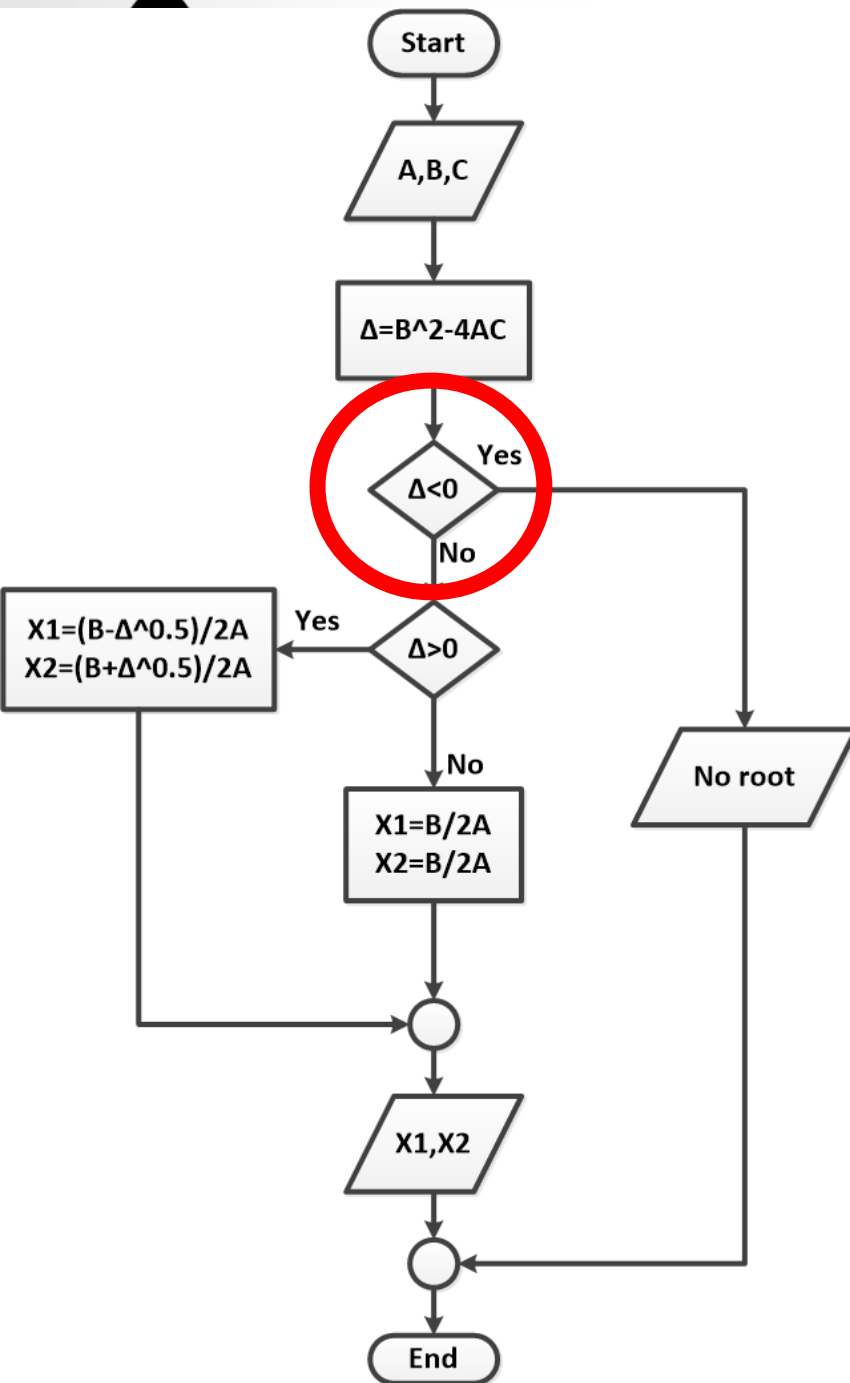
کنترل برنامه :

همان گونه که در مبحث الگوریتم بدان اشاره گردید، حلقه ها و دستورات شرطی کلیدی ترین مفهوم در امجام برنامه ها و توابع به شمار می روند. شاید بتوان گفت این فصل مهم ترین فصل در آموزش مقدمات نرم افزار متلب به شمار می رود.



شرط if-else-end :

این دستور برای بررسی یک شرط مورد استفاده قرار می گیرد.





شرط if-else-end :

این دستور برای بررسی یک شرط مورد استفاده قرار می گیرد.

If (شرط ۱)
دستورات ۱

Elseif (شرط ۲)
دستورات ۲

Elseif (شرط ۳)
دستورات ۳

Else
دستورات 4

End

Syntax

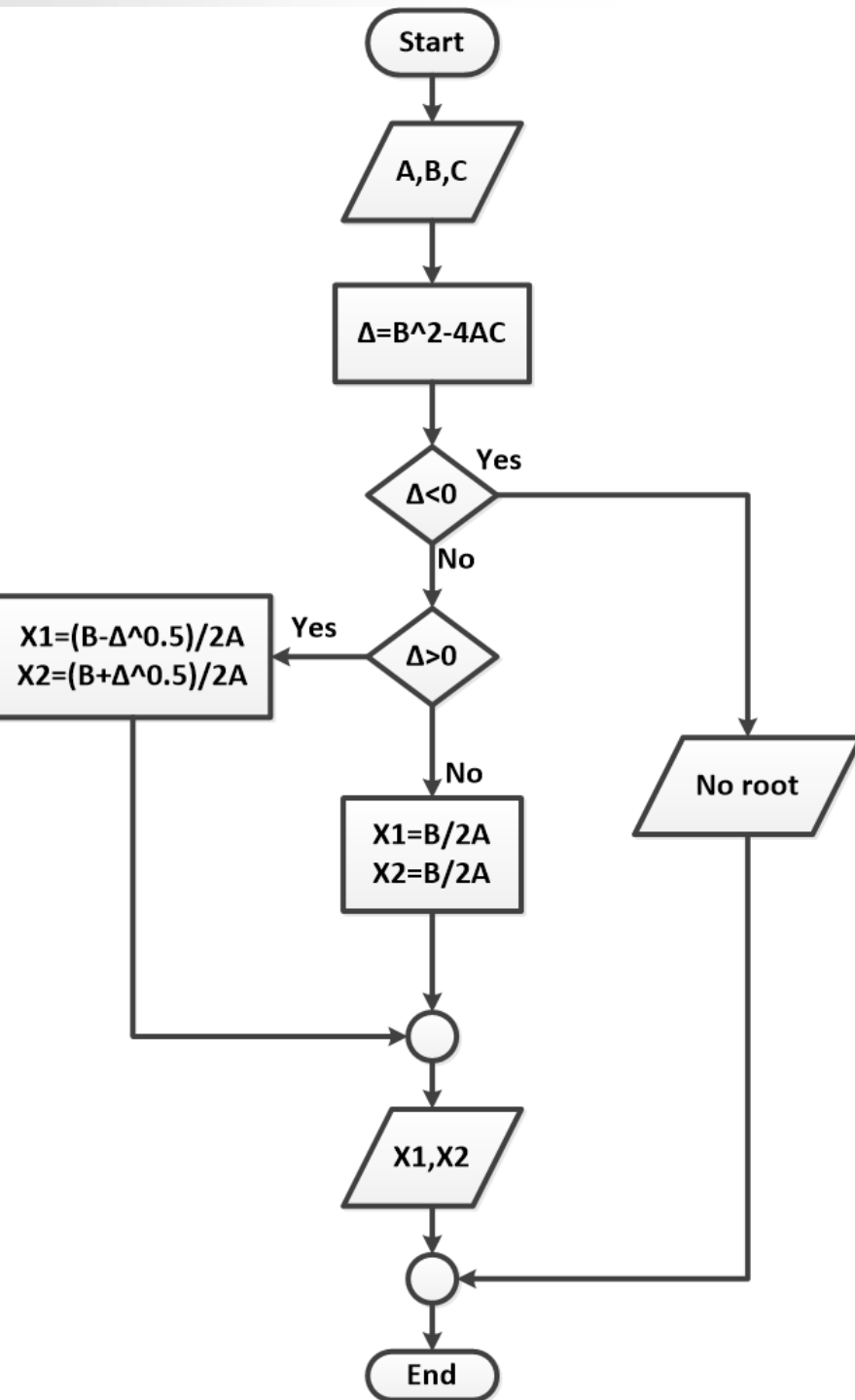
```
if expression
  statements
elseif expression
  statements
else
  statements
end
```

مثال ۱:

میخواهیم تابعی بنویسیم که :
ریشه های معادله درجه دوم

$$Ax^2 + Bx + C = 0$$

را تعیین نماید. (یعنی ضرایب A,B,C را
گرفته و ریشه ها را نمایش دهد).





مثال ١ :

```
1 function delta( A,B,C )
2     D=B^2-4*A*C;
3     if (D<0)
4         disp ('No Root')
5     elseif (D==0)
6         x1=-B/(2*A)
7         x2=-B/(2*A)
8     else
9         x1=(-B-sqrt(D))/(2*A)
10        x2=(-B+sqrt(D))/(2*A)
11    end
12 end
```

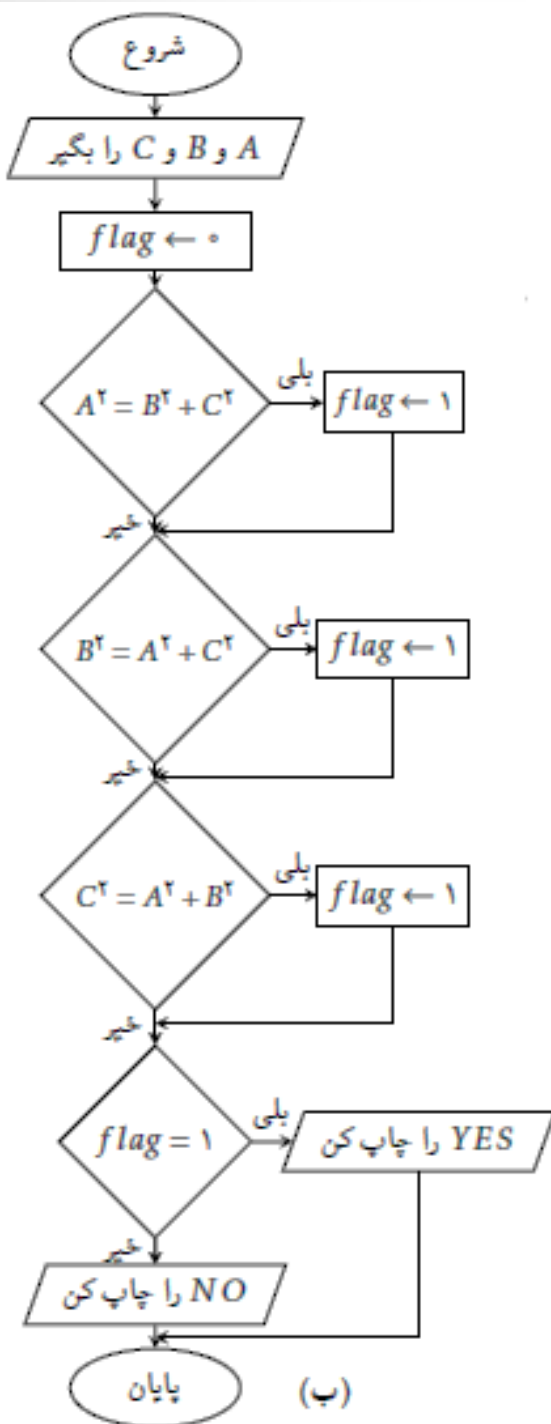
1

Command Window

```
>> delta(1,2,1)
x1 =
    -1
x2 =
    -1
>> delta(1,12,-23)
x1 =
   -13.6811
x2 =
    1.6811
>> delta(1,2,3)
No Root
```

2

مثال ۲ :



میخواهیم تابعی بنویسیم که :
سه ضلع یک مثلث را گرفته و تعیین نماید که
مثلث قائم الزاویه است یا خیر؟



مثال ٢ :

```
1 function ghaem(A,B,C)
2     flag=0;
3     if (A^2==B^2+C^2)
4         flag=1;
5     elseif (B^2==A^2+C^2)
6         flag=1;
7     elseif (C^2==A^2+B^2)
8         flag=1;
9     end
10    if (flag==1)
11        disp ('Yes')
12    else
13        disp ('NO')
14    end
15 end
```

1

Command Window

```
>> ghaem(3,4,5)
```

```
Yes
```

```
>> ghaem(1,2,4)
```

```
NO
```

```
>> ghaem(23,43,34)
```

```
NO
```

2



مثال ۳- بازی حدس عدد :

ابتدا کامپیوتر یک عدد سه رقمی را در نظر میگیرد. سپس از کاربر میخواهد که این عدد را حدس بزند و با راهنمایی (بالا تر و پایین تر) کاربر هدایت کند تا عدد را حدس بزند.

Command Window

```
>> hads_adad_1(500)
Go down
>> hads_adad_1(300)
Go up
>> hads_adad_1(440)
Go down
>> hads_adad_1(430)
Go up
>> hads_adad_1(435)
Go down
>> hads_adad_1(434)
OK
```



تمرین بازی حدس عدد:

```
1  function hads_adad_1(a)
2  -   adad=434;
3  -   if (a>adad)
4  -       disp('Go down')
5  -   elseif(a<adad)
6  -       disp('Go up')
7  -   elseif(a==adad)
8  -       disp('OK')
9  -   else
10 -       disp('your inputs was incorrect')
11 -   end
```




شرط switch-case :

اگر خواسته باشیم برای حالت‌های مختلف تصمیم مختلفی بگیریم مورد استفاده قرار می‌گیرد.

```
Switch a
Case (1)      دستورات ۱
Case (2)      دستورات ۲
...
otherwise     دستورات
End
```

Syntax

```
switch switch_expression
  case case_expression
    statements
  case case_expression
    statements
  ...
  otherwise
    statements
end
```

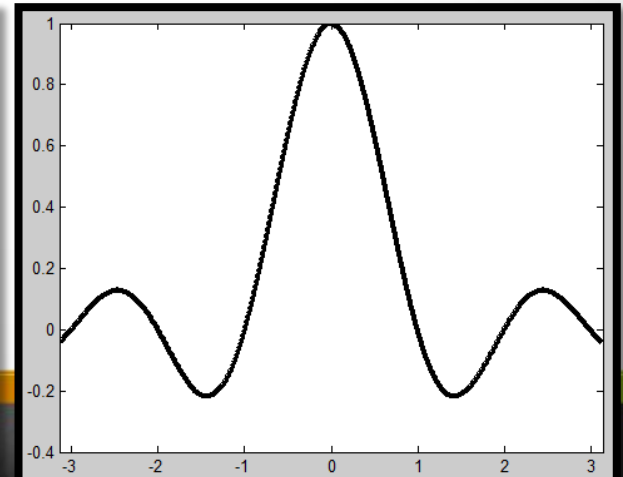
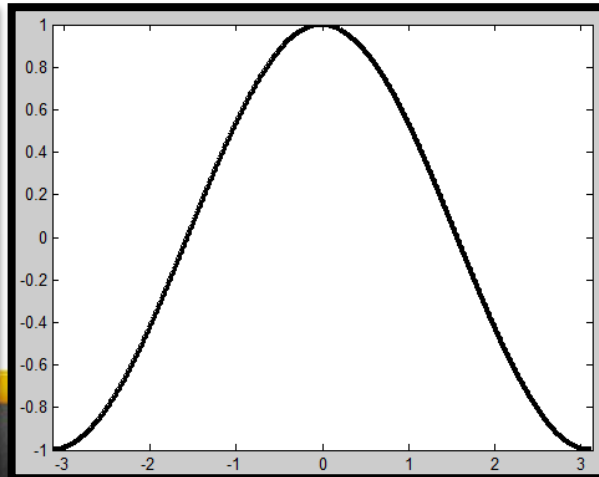
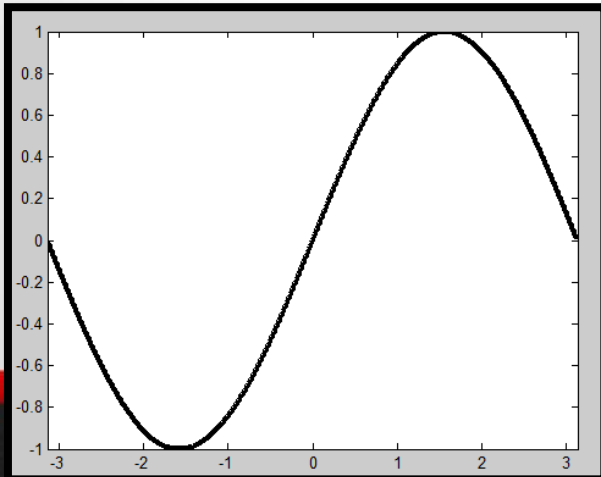


مثال:

برنامه ای بنویسید که به صورت انتخابی نمودارهای \sin , \cos , sinc را رسم کند.

Command Window

```
>> plot_switch(1)
>> plot_switch(2)
>> plot_switch(3)
>> plot_switch(6)
enter 1,2,3
```





مثال :

```
1 function plot_switch( n )
2 close all
3 x=-pi:0.001:pi;
4 switch n
5     case 1
6         y=sin(x);
7     case 2
8         y=cos(x);
9     case 3
10        y=sinc(x);
11    otherwise
12        disp('enter 1,2,3')
13 end
14 plot(x,y,'k','LineWidth',3);
15 xlim([-pi pi])
16 end
```

مثال :

برنامه ای بنویسید که شعاع دایره را گرفته و هم چنین متناسب با خواسته کاربر محیط یا مساحت را محاسبه نماید و در صورتی که اطلاعات وارده غلط باشد پیغامی صادر نماید.

Command Window

```
>> help prj_switch
    s ==> masahat
    m ==> mohit
    for example : prj_switch('m',2)
                  prj_switch('s',2)

>> prj_switch('m',1)
m =
    6.2832
>> prj_switch('s',1)
s =
    3.1416
```



مثال ۱ :

```
1 function prj_switch(a,r)
2 % s ==> masahat
3 % m ==> mohit
4 % for example : prj_switch('m',2)
5 % prj_switch('s',2)
6 switch a
7     case 'm'
8         m=2*pi*r
9     case 's'
10        s=pi*r^2
11    otherwise
12        disp('...error...')
13 end
14 end
```



حلقه for :

اگر خواسته باشیم به صورت معلوم یک عمل را انجام دهیم، از دستور for استفاده می کنیم.

```
For variable=a
```

دستوارت

```
End
```

Syntax

```
for index = values  
    program statements  
    :  
end
```

مثال :

برنامه ای بنویسید که تمام اعداد دو رقمی مضرب عدد دلخواه را نمایش دهد.

Command Window

```
>> help mazrab  
for example: mazrab(8)
```

```
>> mazrab(11)
```

```
i =  
    11  
i =  
    22  
i =  
    33  
i =  
    44  
i =  
    55
```

```
i =  
    66  
i =  
    77  
i =  
    88  
i =  
    99
```



مثال :

```
1  - function mazrab( a )
2  -   % for example: mazrab(8)
3  -  - for i=a:99
4  -      k=mod(i,a);
5  -      if k==0
6  -          i
7  -      end
8  -  - end
9  -  - end
```


مثال :

برنامه ای بنویسید که جدول ضرب پنج در پنج ایجاد نماید.

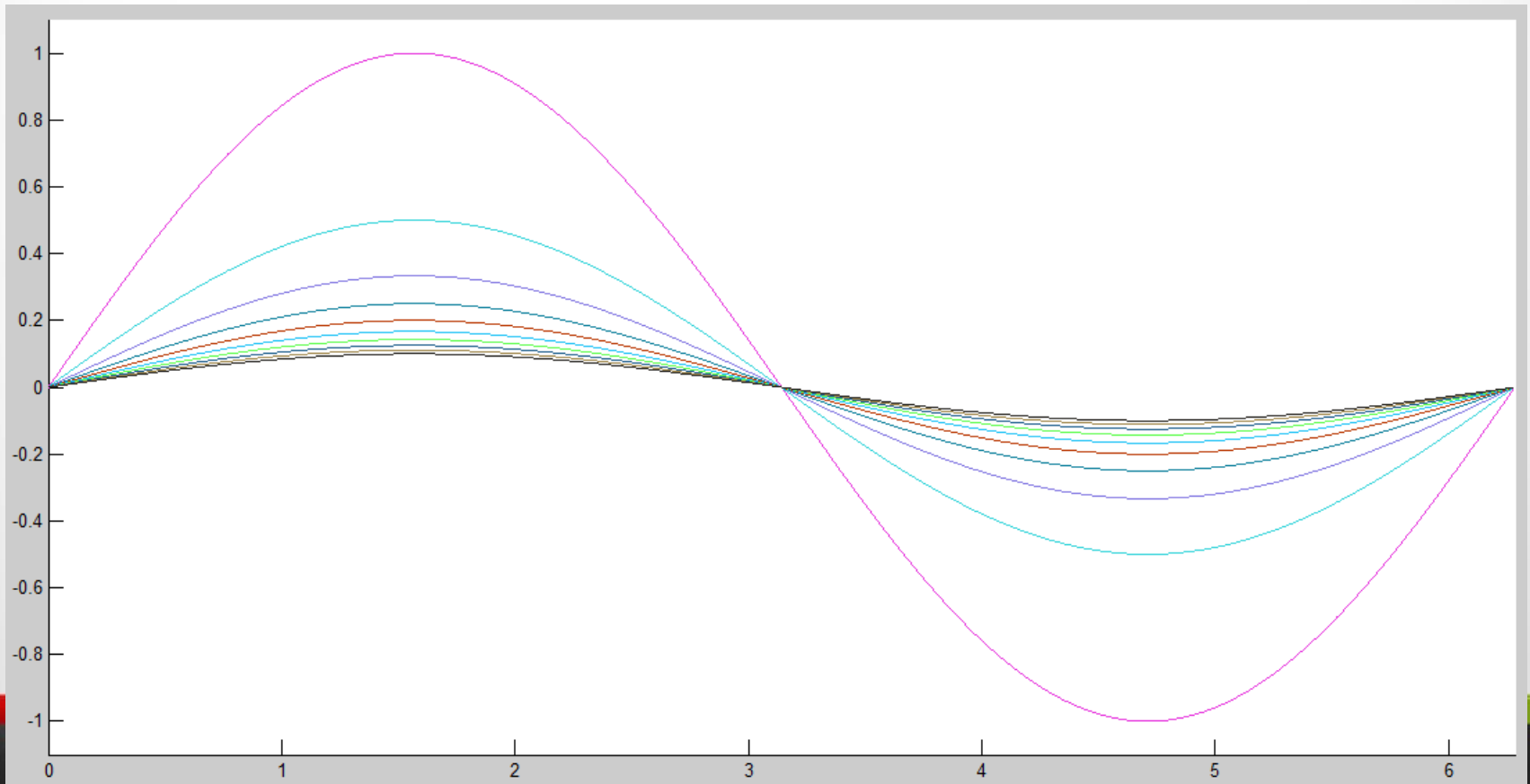
```
1 function zarb_1( )
2     Z=zeros(5);
3     for i=1:5
4         for j=1:5
5             Z(i,j)=i*j;
6         end
7     end
8     disp(Z)
9 end
```

```
>> zarb_1
     1     2     3     4     5
     2     4     6     8    10
     3     6     9    12    15
     4     8    12    16    20
     5    10    15    20    25
```



مثال :

نمودارهای $\sin(x)/1, \sin(x)/2, \sin(x)/3, \dots, \sin(x)/10$ را در یک شکل رسم نمایید.



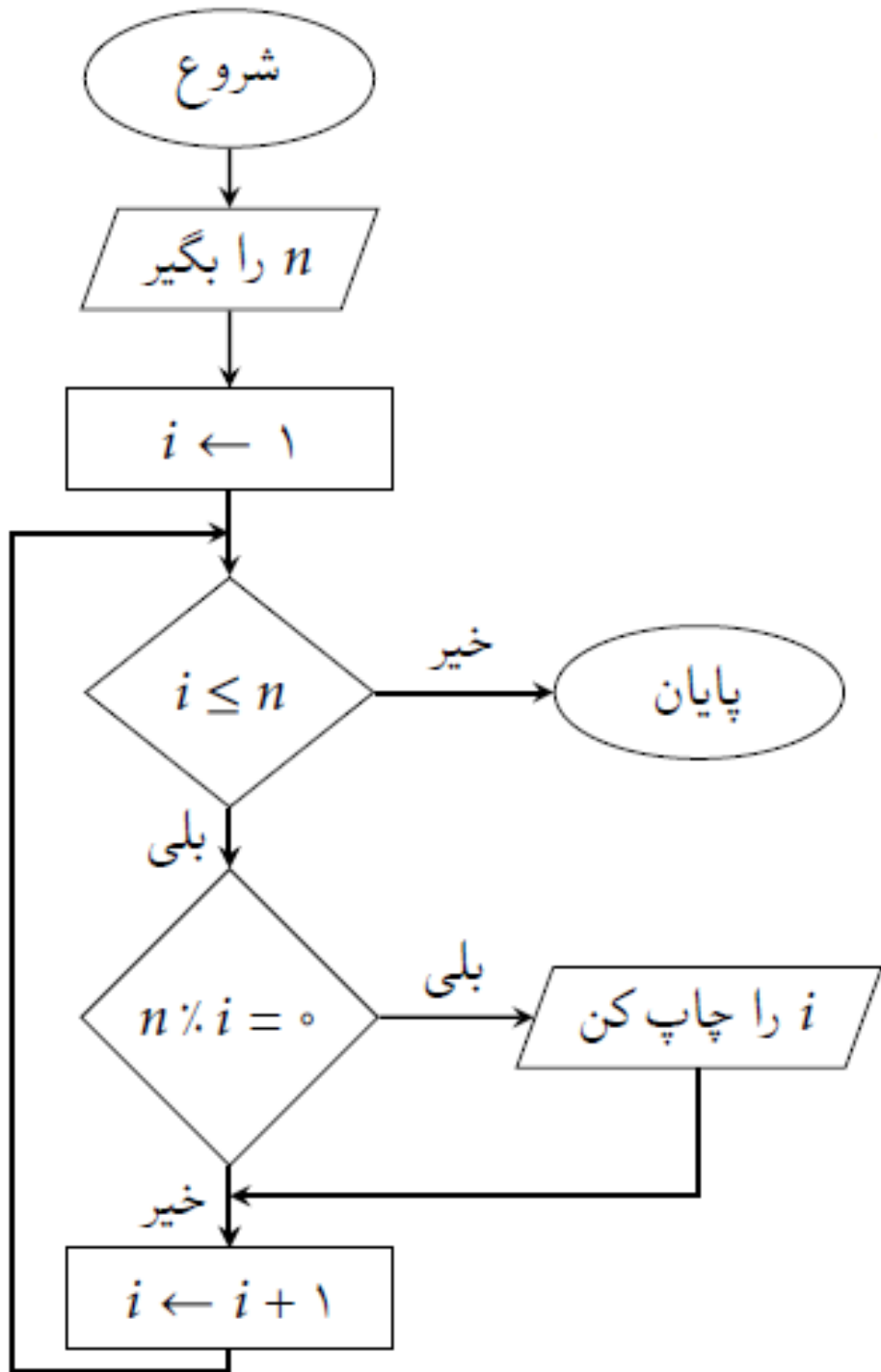


مثال :

```
1  - function plotting1( )
2  -     x=0:0.001:2*pi;
3  -     figure
4  -     xlim([0 2*pi]);
5  -     ylim([-1.1 1.1]);
6  -     hold on
7  -     for i=1:10
8  -         y=(sin(x))/i;
9  -         plot(x,y,'color',rand(1,3));
10 -     end
```

مثال :

برنامه ای بنویسید که تمام مقسوم علیه یک عدد دلخواه را نمایش دهد.



Command Window

```
>> maghsoom_adad(24)
```

```
1
```

```
2
```

```
3
```

```
4
```

```
6
```

```
8
```

```
12
```

```
24
```

```
>> maghsoom_adad(10)
```

```
1
```

```
2
```

```
5
```

```
10
```



مثال:

```
1  - function maghsoom_adad(a)
2  - for i=1:a
3  -     m=mod(a,i);
4  -     if (m==0)
5  -         disp(i);
6  -     end
7  - end
8  - end
```



مثال :

برنامه ای بنویسید که تمام اعداد اول بین هر دو عدد دلخواه را محاسبه نماید.

Command Window

```
>> help aval_1
aval_1(min,max)
for example : aval_1(12,20)
ans =
    13    17    19

>> aval_1(10,30)
    11    13    17    19    23    29

>> aval_1(980,1000)
    983    991    997
```

```
1 function aval_1(min,max)
2 % aval_1(min,max)
3 % for example : aval_1(12,20)
4 % ans =
5 %     13    17    19
6 add_aval=[];
7 for a=min:max
8     b=0;
9     for i=1:a
10        m=mod(a,i);
11        if (m==0)
12            b=b+1;
13        end
14    end
15    if (b==2)
16        add_aval(end+1)=a;
17    end
18 end
19 disp(add_aval)
20 end
```



حلقه while :

اگر خواهان استفاده از حلقه ای با تکرار نامعلوم و به نوعی تکرار تا وقتی که یک شرط خاص برقرار باشد، از این دستورالعمل استفاده می کنیم.

While (شرط)

دستوارت

End

Syntax

```
while expression
  statements
end
```




مثال:

برنامه ای بنویسید که عددی دلخواه را گرفته و از صفر تا آن عدد اعداد زوج را نمایش دهد.

```
1 function prj_while( a )
2     i=0;
3     while (i<=a)
4         disp(i);
5         i=i+2;
6     end
7 end
```

Command Window

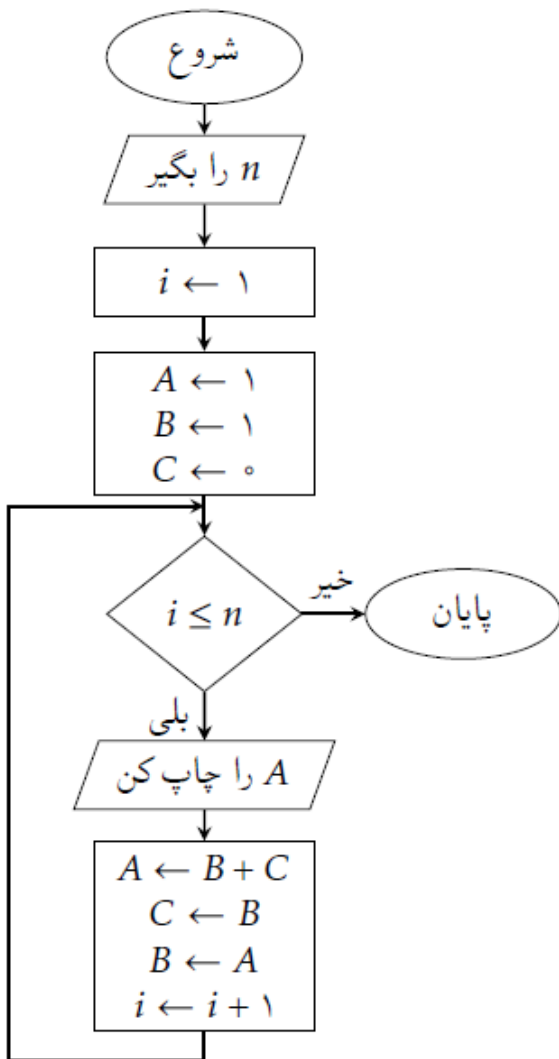
```
>> prj_while(10)
0
2
4
6
8
10
```

1

2

مثال :

برنامه ای بنویسید که تا جمله دلخواه از سری فیبوناچی را نمایش دهد.



Command Window

```
>> fibo(3)
    1
    1
    2

>> fibo(10)
    1
    1
    2
    3
    5
    8
   13
   21
   34
   55
```



مثال :

```
1  - function fibo( n )
2  -     A=1;
3  -     B=1;
4  -     C=0;
5  -     i=1;
6  -     while (i<=n)
7  -         disp (A)
8  -         A=B+C;
9  -         C=B;
10 -         B=A;
11 -         i=i+1;
12 -     end
13 - end
```

مثال:

برنامه ی بازی حدس عدد را با استفاده از دستور `while` بنویسید. (در این حالت حدس اولیه توسط کامپیوتر صورت گیرد.)

Command Window

```
>> hads_adad_2  
please enter number between [0-1000]
```

```
=====
```

```
number = 500  
Go down  
number = 250  
Go up  
number = 375  
Go down  
number = 335  
Go down  
number = 300  
Go down
```

```
number = 275  
Go up  
number = 285  
Go down  
number = 280  
Go down  
number = 278  
Go up  
number = 279  
OK
```



```
1  function hads_adad_2()  
2  -   disp('please enter number between [0-1000]')  
3  -   disp('=====')  
4  -   flag=1;  
5  -   adad=randi([0 1000],1,1);  
6  -   while (flag)  
7  -       a=input(' number = ');  
8  -       if (a>adad)  
9  -           disp('Go down')  
10 -       elseif (a<adad)  
11 -           disp('Go up')  
12 -       elseif (a==adad)  
13 -           disp('OK')  
14 -           flag=0;  
15 -       end  
16 -   end  
17 - end
```



تمرین :

بازی حدس عدد را به گونه ای بنویسید که این بار کاربر یک عدد دلخواه را در نظر گرفته و کامپیوتر این عدد را حدس بزند.
در این برنامه مشابه مثال ها کاربر راهنمایی های لازم برای حدس عدد به کامپیوتر ارائه می دهد.



تمرین :

برنامه ای بنویسید که جدول سودوکو را حل نماید.
در این برنامه لازم است تا بتوان در ابتدا چند عدد را به صورت پیش فرض در
نظر گرفت و سپس برنامه جدول را تکمیل نماید.

	2			3			4	
6								3
		4				5		
			8		6			
8				1				6
			7		5			
		7				6		
4								8
	3			4			2	

9	2	5	6	3	1	8	4	7
6	1	8	5	7	4	2	9	3
3	7	4	9	8	2	5	6	1
7	4	9	8	2	6	1	3	5
8	5	2	4	1	3	9	7	6
1	6	3	7	9	5	4	8	2
2	8	7	3	5	9	6	1	4
4	9	1	2	6	7	3	5	8
5	3	6	1	4	8	7	2	9



تمرین :

برنامه ای بنویسید که عدد دلخواه را از کاربر گرفته و شکل زیر را نمایش دهد.

