

دسته عملگرهای رابطه‌ای:

دسته این عملگرها به بررسی روابط بین دو عبارت می‌پردازد.

در: $==$ (مساوی): تساوی دو عبارت را بررسی می‌کند.

می: $\sim$ (مخالف): مخالف بودن دو عبارت را بررسی می‌کند.

$<$ (کوچک‌تر): کوچک‌تر بودن دو عبارت را بررسی می‌کند.

$>$ (بزرگ‌تر): بزرگ‌تر بودن دو عبارت را بررسی می‌کند.

نکته $<=$ (کوچک‌تر مساوی): کوچک‌تر یا مساوی بودن دو عبارت را بررسی می‌کند.

$>=$ (بزرگ‌تر مساوی): بزرگ‌تر یا مساوی بودن دو عبارت را بررسی می‌کند.

خروجی عبارت‌های رابطه‌ای و منطقی، یک عبارت منطقی می‌باشد. که رابطه یا درست است که به آن True می‌گوییم

و مقدار یک دارد؛ و یا رابطه درست نیست که به آن False می‌گوییم و مقدار صفر دارد.

مثال: عملگرهای رابطه‌ای را بر روی دو عدد a, b بررسی کنید؟

$a=5$	$b=-1$
$a==b$	$\rightarrow ans=0$
$a\sim b$	$\rightarrow ans=1$
$a<b$	$\rightarrow ans=0$
$a<=b$	$\rightarrow ans=0$
$a>b$	$\rightarrow ans=1$
$a>=b$	$\rightarrow ans=1$

دستورات شرطی

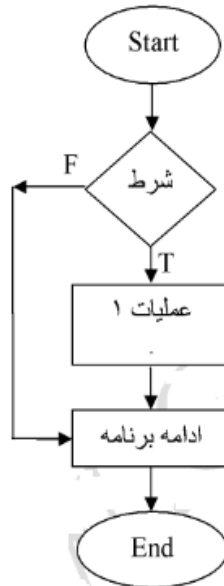
دستور if

در برخی از برنامه‌ها ما نیاز به تصمیم‌گیری در سر دو راهی‌ها یا سه راهی‌ها و یا ... را داریم. جهت انجام تصمیم‌گیری می‌توان از دستور if استفاده نمود که دارای ساختارهای متفاوتی است که بر حسب نیاز از آن‌ها استفاده می‌شود.

(A) ساختار تک انتخابی

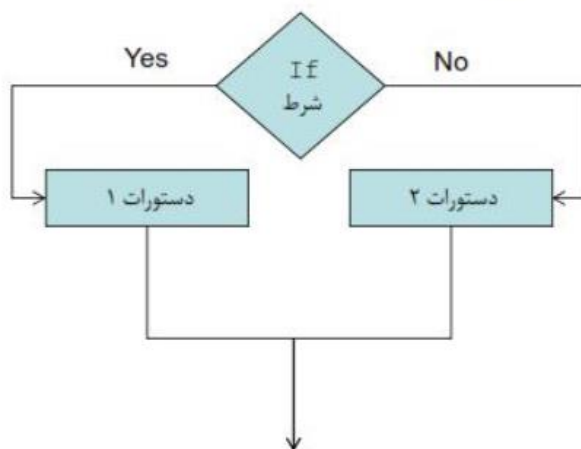
نکته: هر دستور if باید با یک دستور end بسته شود.

if جمله شرطی
 دستورات
 .
 .
end



۱-۲-۲-۱ دستورالعمل شرط

در برخی از برنامه‌ها لازم است بر اساس شرایطی که اتفاق می‌افتد، دستورالعمل خاصی اجرا شود. برای پیاده‌سازی این‌گونه مسائل دستورالعمل‌های شرطی بکار می‌روند. برای نمایش دستورات شرطی می‌توان از سمبل لوزی بصورت ذیل استفاده نمود:



فقط باید توجه کرد که در این‌گونه مسائل نحوه ارتباط مراحل مختلف در صورت برقرار نبودن شرط بدرستی مشخص شده باشد.

if... (else...) end – دستور if برنامه را قادر می سازد که تصمیم بگیرد که چه دستورهایی باید اجرا گردند. مثال:

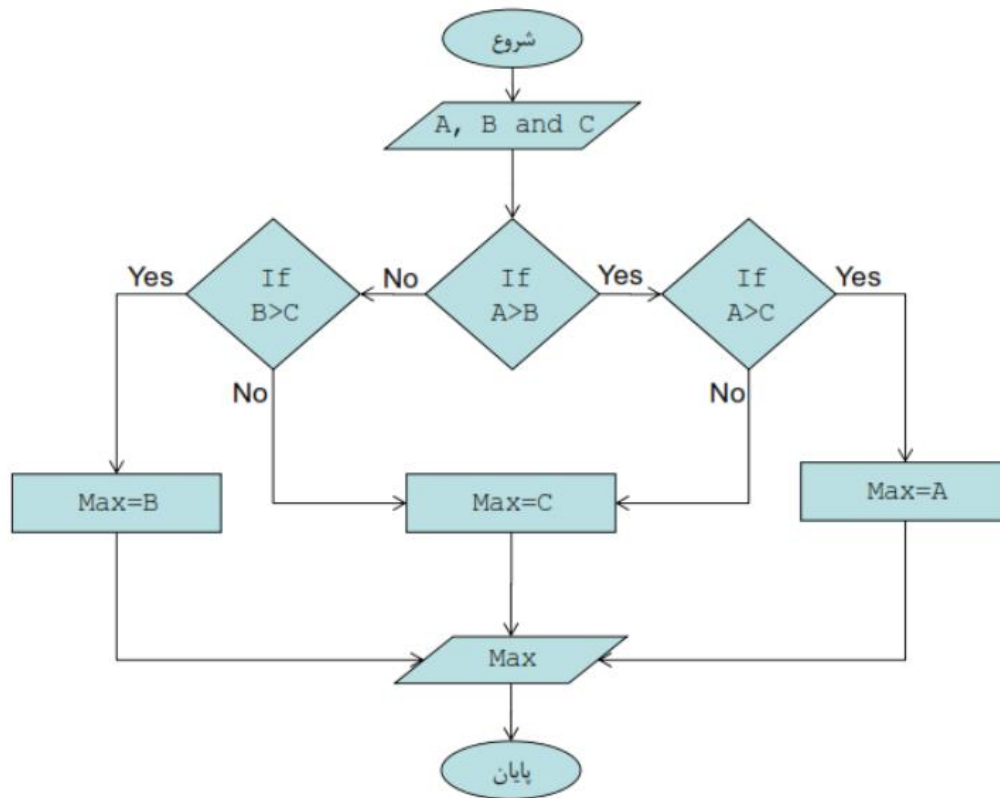
```
x = input(' x = ');  
if x >= 0  
    y=x^2  
end
```

عبارتی که به دنبال کلمه if می آید باید یک عبارت منطقی باشد. در صورت درست بودن این عبارت منطقی، دستورهایی که در سطرهای بین if و end قرار دارند بترتیب اجرا می گردند و در صورت نادرست بودن این عبارت منطقی، دستورهای گفته شده نادیده گرفته می شوند.

شما همچنین می توانید از دستور else استفاده کنید. مثال:

```
x = input(' x = ');  
if x >= 0  
    y=x^2  
else  
    y=-x^2  
end
```

مثال ۴: فلوچارت برنامه‌ای که سه عدد A ، B و C را از ورودی گرفته و بزرگترین آن را نمایش دهد، بصورت زیر ترسیم می‌گردد:



مثال: برنامه‌ای بنویسید که یک عدد را گرفته و با یک پیام مشخص کند که عدد زوج است یا فرد؟

```

clc
clear
format short g
a=input('a=');
if mod(a,2)==0
disp('zaj')
end
if mod(a,2)~=0
disp('fard')
end
  
```

مثال: برنامه‌ای بنویسید که ریشه‌های یک معادله درجه دو را بدست آورد و در صورتی که دلتا (b^2-4ac) منفی شود پیغامی مبنی بر عدم وجود ریشه چاپ کند؟

```
clc
clear
format short g
a=input('a=');
b=input('b=');
c=input('c=');
d=b^2-4*a*c;
if d<0
disp('risheh nadarad')
end
if d>0
D=d^0.5;
x1=(-1*b+D)/(2*a);
x2=(-1*b-D)/(2*a);
x=[x1 x2];
disp(' **** x1 ***** x2 ****')
disp(x)
end
```

مثال: برنامه‌ای بنویسید که با گرفتن سه زاویه با یک پیغام مشخص کند که آیا می‌توان با این سه زاویه یک مثلث تشکیل داد یا خیر؟

```
clc
clear
format short g
a=input('a=');
b=input('b=');
c=input('c=');
s=a+b+c;
if s==180
disp('mosalas')
end
if s~=180
disp('no mosalas')
end
```

مثال: برنامه‌ای بنویسید که یک عدد را گرفته و با یک پیام مثبت یا منفی بودن آن عدد را چاپ کند؟

```
clc
clear
format short g
a=input('a=');
if a>0
disp('+')
end
if a<0
disp('-')
end
```

مثال: برنامه‌ای بنویسید که برای ورود به برنامه یک کد عبور از کاربر درخواست کند در صورت درست بودن کد وارد برنامه شده و سه زاویه از یک مثلث را گرفته و در صورتی که این سه زاویه بتواند یا نتواند تشکیل یک مثلث دهد با یک پیام نمایش دهد.

```
clc
clear
format short g
pass=input('pass key=');
if pass==Rastgou
a=input('a=');
b=input('b=');
c=input('c=');
s=a+b+c;
if s==180
disp('mosalas')
end
if s~=180
disp('no mosalas')
end
end
```