

آشنايے با برنامہ نویسی

# برنامه نویسی

راهی برای **دستور دادن** به کامپیوتر/ماشین برای:

انجام وظایف مختلف

حل مساله ها با کامپیوتر

خودکار کردن کارهای ساده و تکراری

و ...



# برنامہ نویسی

دو تا عدد 3 و 4 رو باہم جمع کن و نمایش بدہ!



زبان برنامہ نویسی

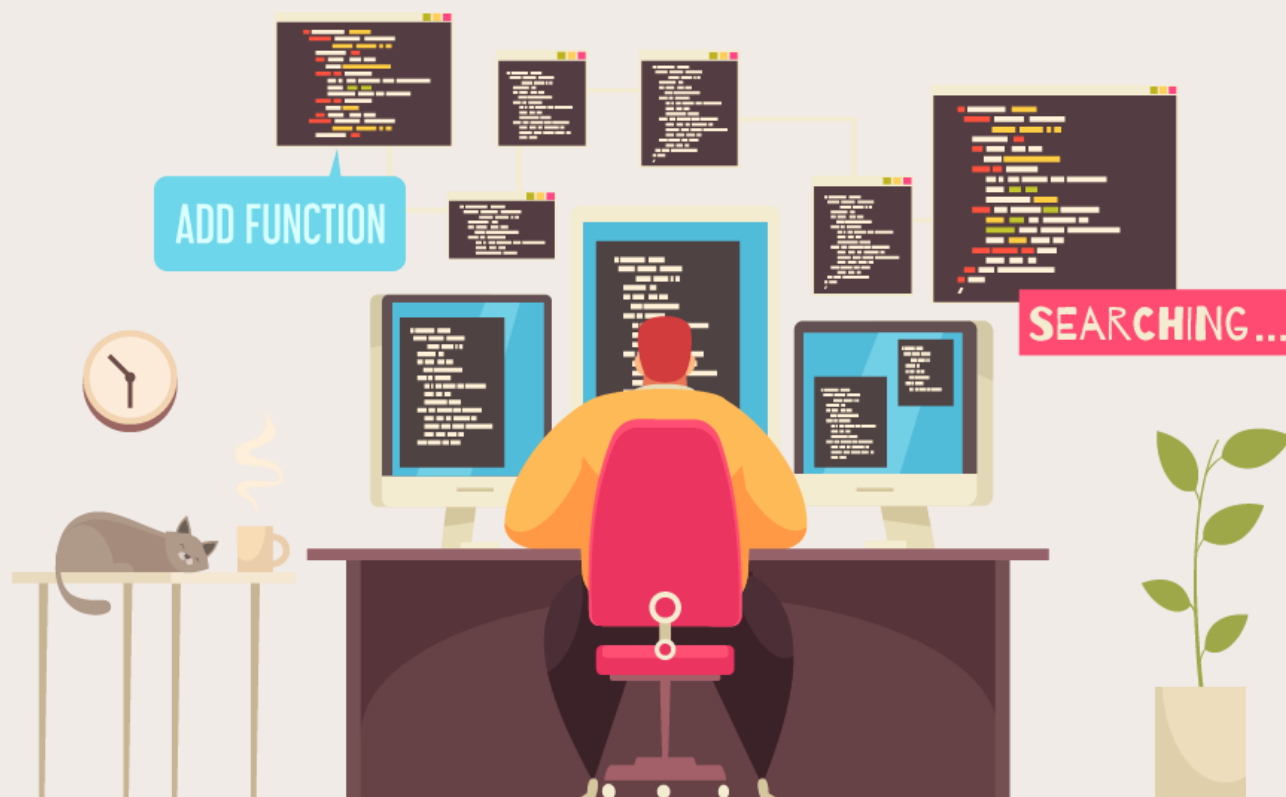
# زبان برنامہ نویسی

زبان راہ برقراری ارتباط میان انسان ها



# زبان برنامه نویسی

زبان برنامه نویسی راه برقراری ارتباط میان انسان و رایانه



A word cloud featuring various computer science and programming terms. The words are arranged in a dense, overlapping manner, with some appearing in larger, bolder fonts than others. The colors are primarily white and light blue, set against a dark background. The words include: Java, Computer Science, python, swift, Android, C++, C, JavaScript, Technology, ANDROID, SWIFT, PYTHON, JAVASCRIPT, and others. The overall composition is a collage of digital-related vocabulary.

# زبان برنامه نویسی





# زبان برنامه نویسی

دو تا عدد 3 و 4 رو باهم جمع کن و نمایش بده!

Command Window

```
>> a = 3;
```

```
>> b = 4;
```

```
>> c = a + b;
```

```
>> disp(c)
```

```
c =
```

```
7
```



تعام!

فرآیند برنامہ نویسی

# فرآیند برنامه نویسی

برنامه‌ای بنویسید که جواب‌های حقیقی معادله  $ax^2 + bx + c = 0$  را پیدا و چاپ کند.

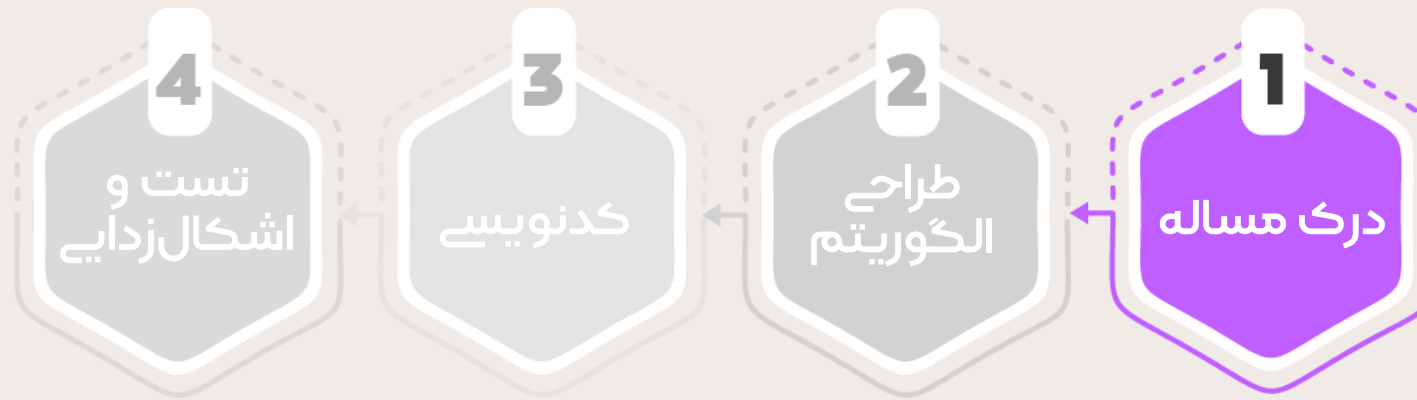


# فرآیند برنامه نویسی

برای حل یک مساله با برنامه نویسی چه مراحل باید طی شود؟



# فرآیند برنامه نویسی -> درک مساله



# فرآیند برنامه نویسی -> درک مساله

چند سوال مهم برای درک هر مساله ای:

هدف از حل این مساله چیست؟

ورودی ها چیست؟

خروجی چیست؟

می توانید روی کاغذ مساله را حل کنید؟

و بسیاری از سوالات بسیار بسیار مهم دیگر!



# فرآیند برنامه نویسی -> درک مساله

مثلا درمورد مساله  $ax^2 + bx + c = 0$ :

هدف از حل این مساله چیست؟

یافتن ریشه های معادله درجه دوم

ورودی ها چیست؟

ضرایب  $a$ ،  $b$  و  $c$

خروجی چیست؟

ریشه های معادله

می توانید روی کاغذ مساله را حل کنید؟

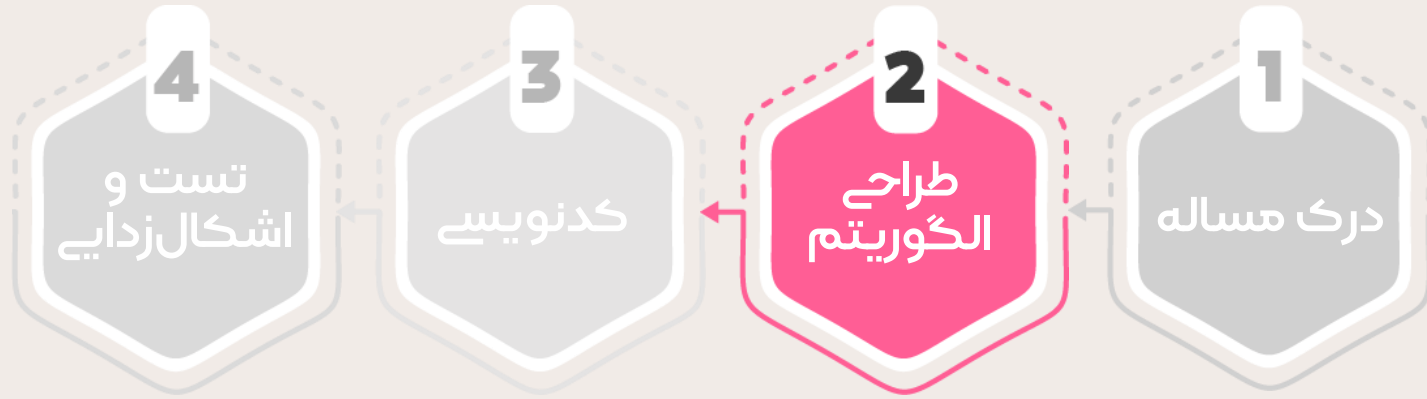
بله ☺

$$ax^2 + bx + c = 0 \rightarrow \Delta = b^2 - 4ac \rightarrow \begin{cases} \Delta = 0 \rightarrow x = -\frac{b}{2a} \\ \Delta < 0 \rightarrow ?! \\ \Delta > 0 \rightarrow x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \end{cases}$$





# فرآیند برنامه نویسی - > طراحی الگوریتم



# فرآیند برنامه نویسی -> طراحی الگوریتم

الگوریتم یعنی چه؟

پروسه انجام یک کار همراه با جزئیات هر مرحله



# فرآیند برنامه نویسی -> طراحی الگوریتم

مرتب کردن براساس قد



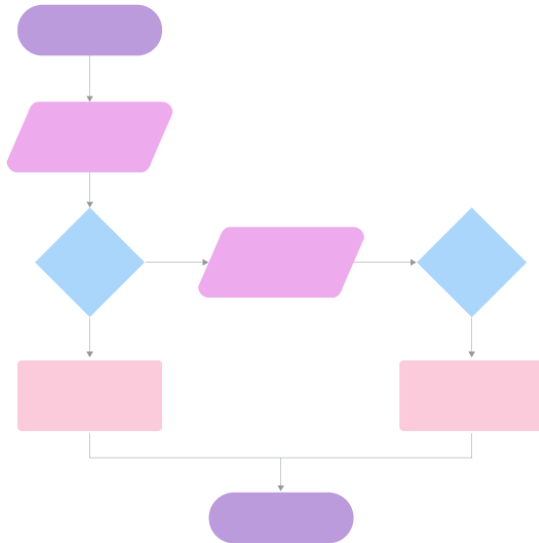
# فرآیند برنامه نویسی -> طراحی الگوریتم

انواع روش بیان الگوریتم:

با متن  
(فارسی/انگلیسی)

1. Choose the number of clusters( $K$ ) and obtain the data points
2. Place the centroids  $c_1, c_2, \dots, c_k$  randomly
3. Repeat steps 4 and 5 until convergence or until the end of a fixed number of iterations
4. for each data point  $x_i$ :
  - find the nearest centroid( $c_1, c_2 \dots c_k$ )
  - assign the point to that cluster
5. for each cluster  $j = 1..k$ 
  - new centroid = mean of all points assigned to that cluster
6. End

فلوچارت  
(Flowchart)



# فرآیند برنامه نویسی -> طراحی الگوریتم

برگردیم به معادله درجه دوم و طراحی الگوریتم آن:

شروع

ضرایب  $a$ ،  $b$  و  $c$  را از ورودی دریافت کن.

مقدار  $\Delta$  را حساب کن. اگر:

$\Delta > 0$  آنگاه از رابطه روبرو برای یافتن ریشه‌ها استفاده کن.  $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$

$\Delta = 0$  آنگاه از رابطه روبرو برای یافتن ریشه‌ها استفاده کن.  $x = -\frac{b}{2a}$

$\Delta < 0$  آنگاه ریشه حقیقی ندارد!

چاپ نتیجه

پایان



# فرآیند برنامه نویسی - > کدنویسی



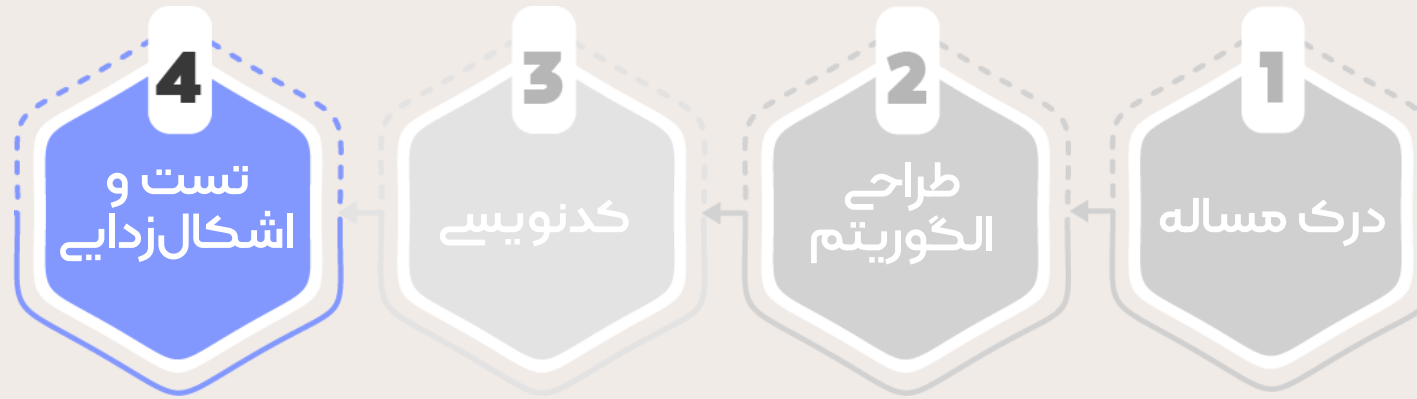
# فرآیند برنامه نویسی -> کدنویسی

و حالا کدنویسی معادله درجه دوم:

```
1 a = 4;
2 b = 1;
3 c = -2;
4
5 delta = b^2 - 4*a*c;
6
7 if delta >= 0
8     x1 = (-b + sqrt(delta))/(2*a);
9     x2 = (-b - sqrt(delta))/(2*a);
10    disp([x1, x2])
11 elseif delta == 0
12     x = b/(2*a);
13     disp(x)
14 else
15     disp('No real roots!')
16 end
```



# فرآیند برنامه نویسی -> تست و اشکال زدایی





# فرآیند برنامه نویسی -> تست و اشکال زدایی

■ **تست و اشکال زدایی** دو مرحله جدا از هم

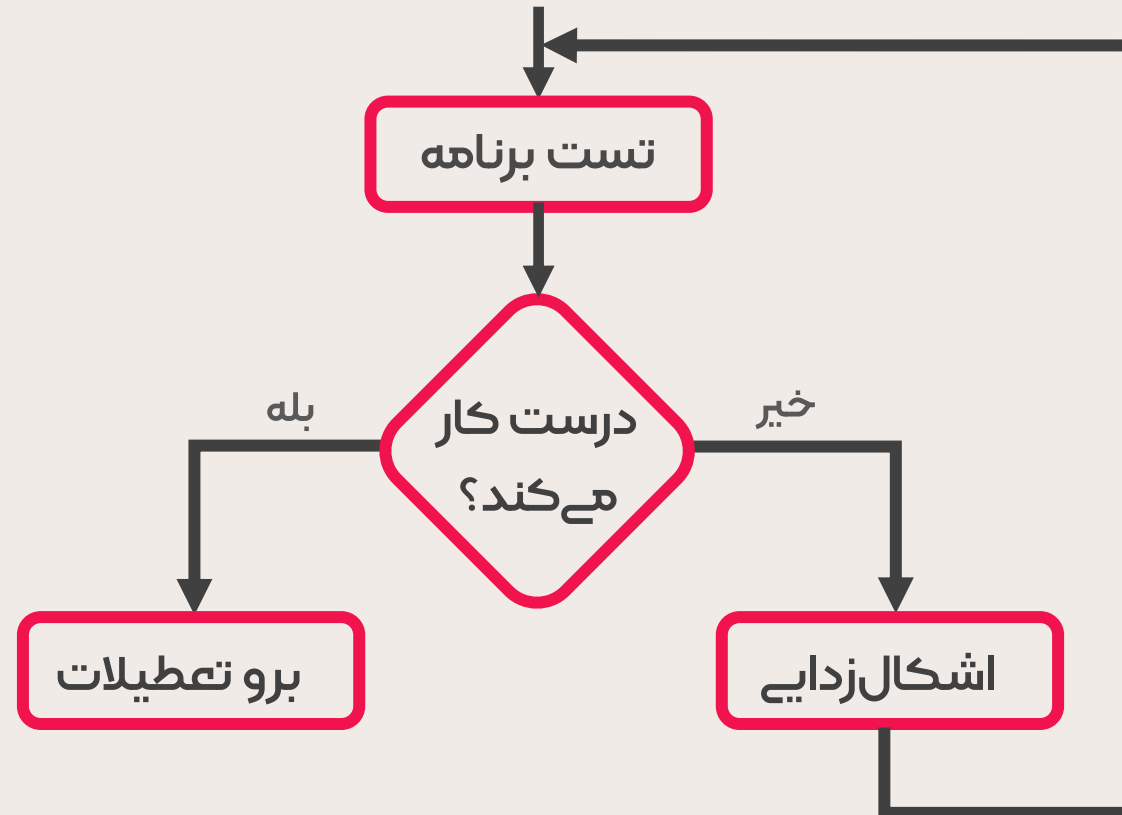
■ **تست:** برنامه را تست می کنیم تا از عملکرد آن آگاه شویم!

■ **اشکال زدایی یا دیباگ:** یافتن اشکالات کد/برنامه و رفع آنها



# فرآیند برنامه نویسی -> تست و اشکال زدایی

فرآیند تست و دیباگ:



# فرآیند برنامه نویسی -> تست و دیباگ

بازهم مساله معادله درجه دوم:

ابتدا تست

متاسفانه باگ دارد!

پس دیباگ

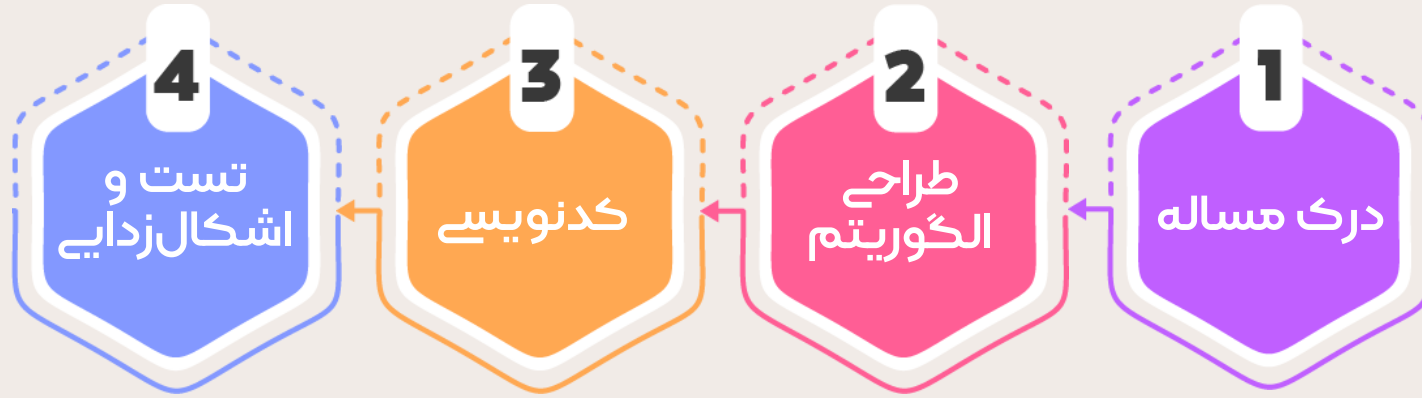
```
1 a = 4;
2 b = 1;
3 c = -2;
4
5 delta = b^2 - 4*a*c;
6
7 if delta >= 0
8     x1 = (-b + sqrt(delta))/(2*a);
9     x2 = (-b - sqrt(delta))/(2*a);
10    disp([x1, x2])
11 elseif delta == 0
12     x = -b/(2*a);
13     disp(x)
14 else
15     disp('No real roots!')
16 end
```



تعام!

نکته مهم!

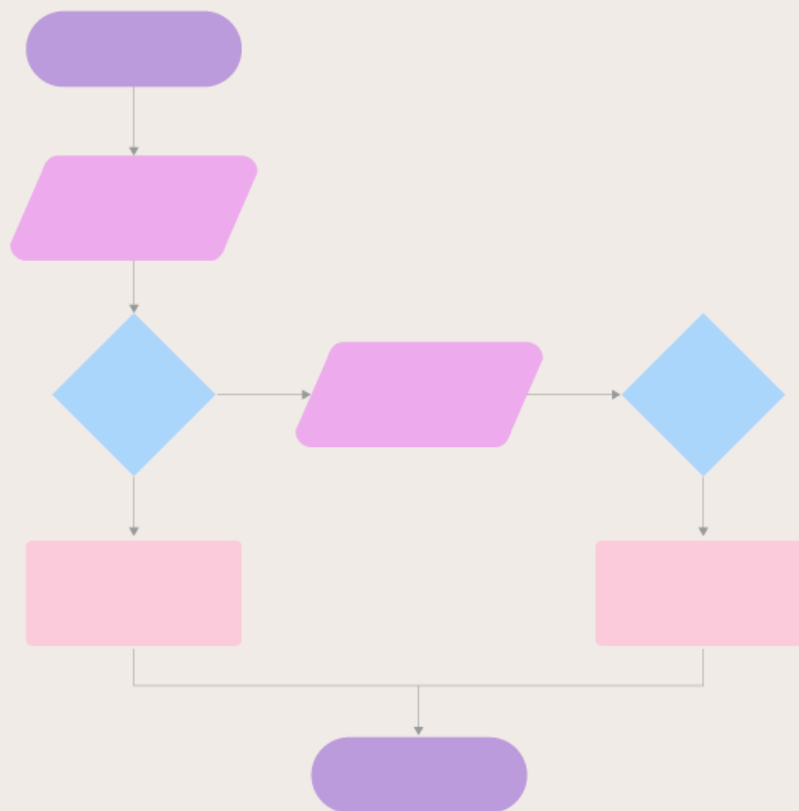
# فرآیند برنامه نویسی



فلوچارت

# فلوچارت

■ دیاگرام یک پروسه یا فرآیند





# فلوچارت



نمادهای مهم در فلوچارت:

علامت شروع و پایان

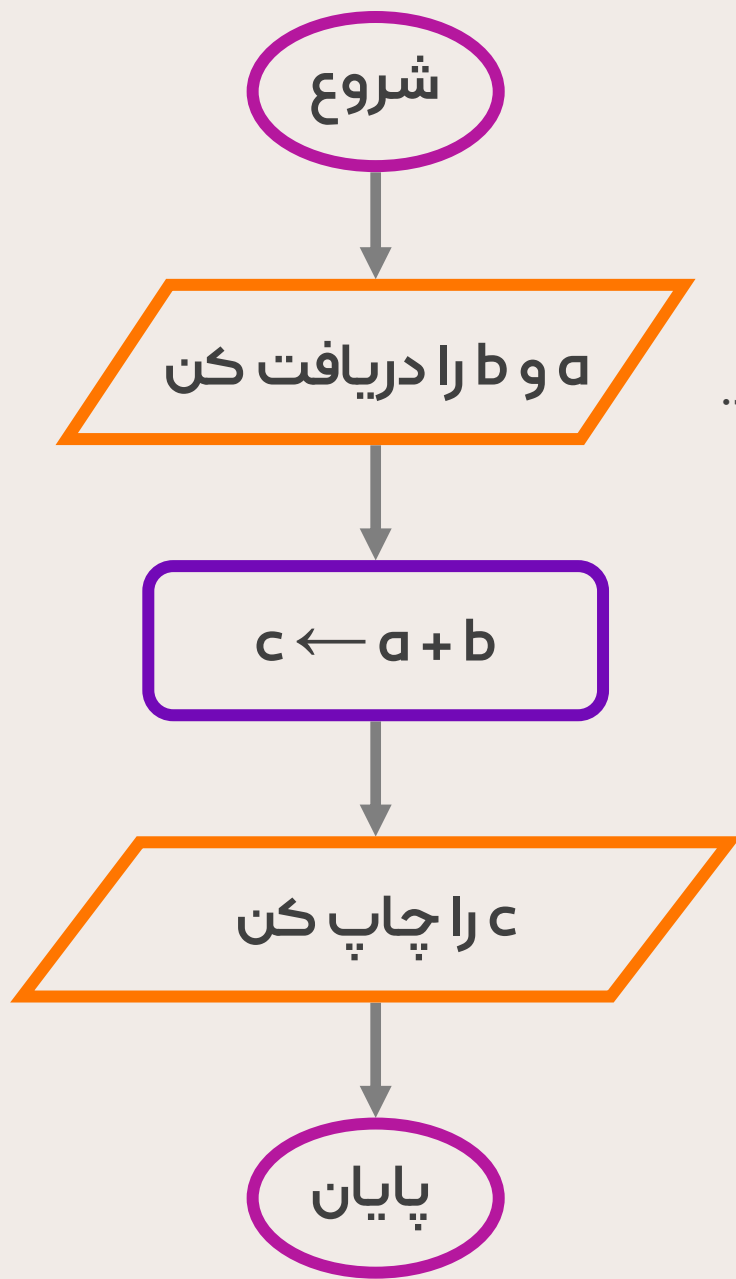
دریافت ورودی یا چاپ خروجی

محاسبات (پروسه)

...



# فلوچارت



مثال 1: الگوریتمی بنویسید که جمع دو عدد را محاسبه کند.

شروع

اعداد a و b را از ورودی دریافت کن.

اعداد a و b را باهم جمع کن و در c ذخیره کن.

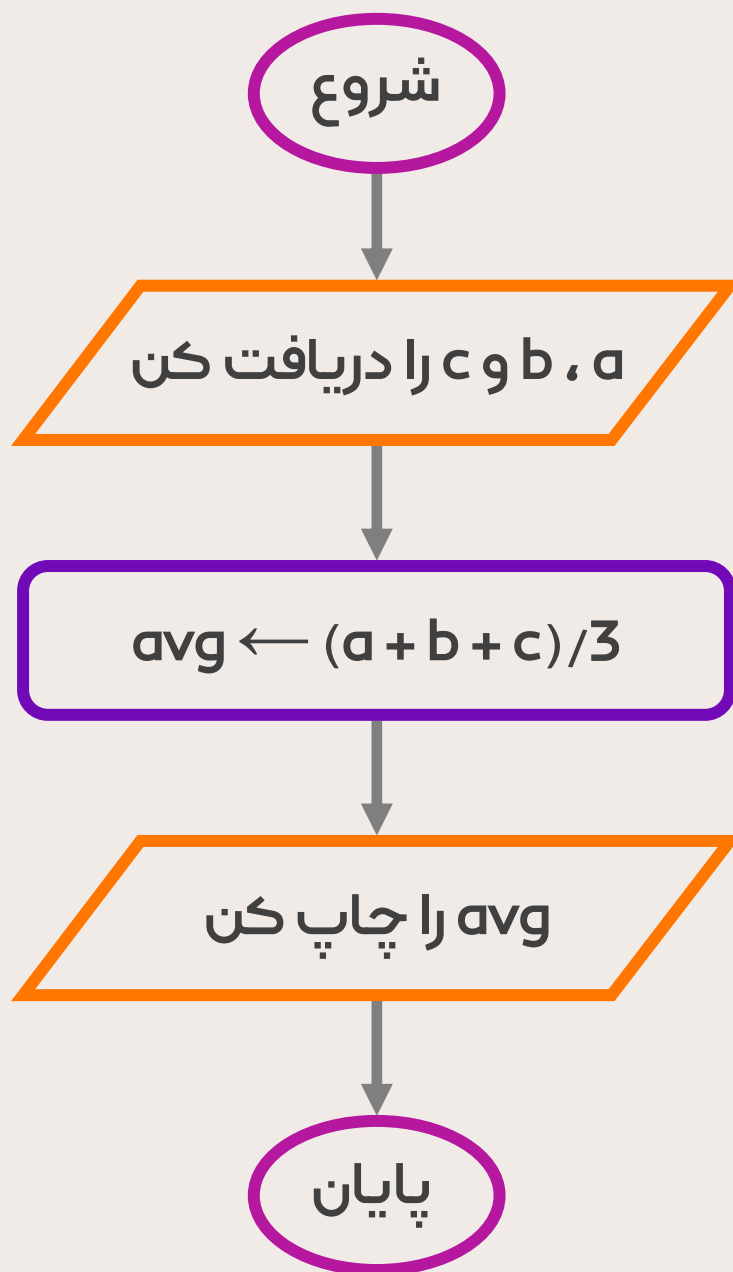
c را به عنوان خروجی نمایش بده.

پایان

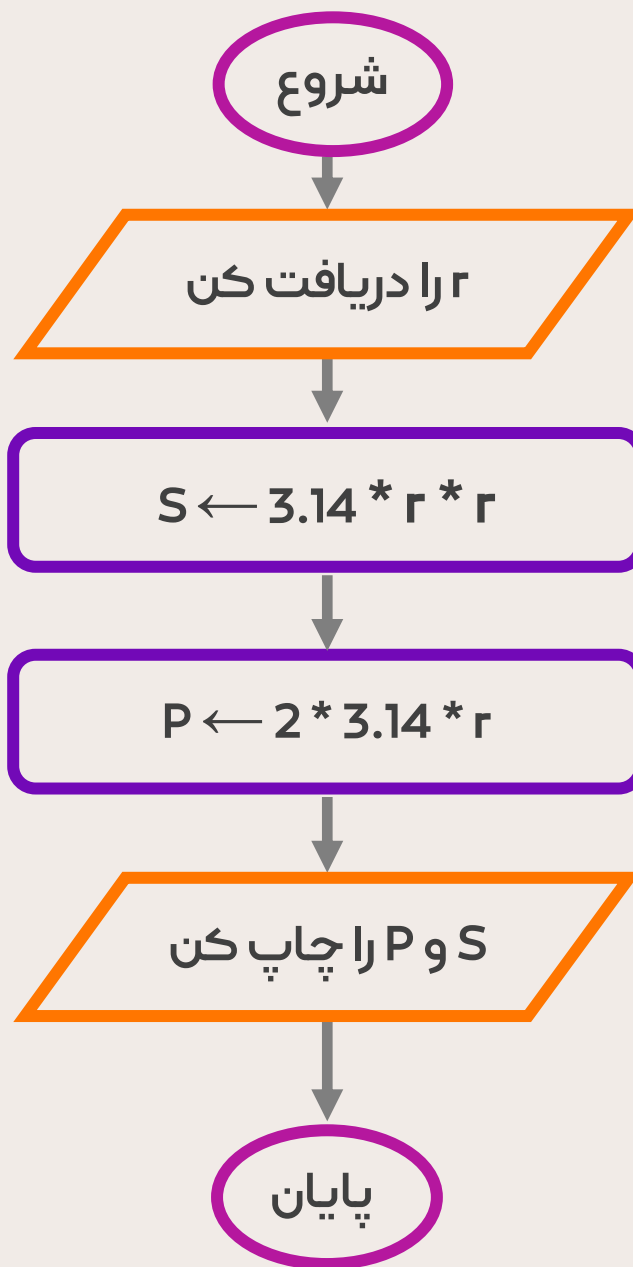


# فلوچارت

مثال 2: الگوریتمی بنویسید که سه عدد را دریافت و میانگین آنها را محاسبه کند.



# فلوچارت

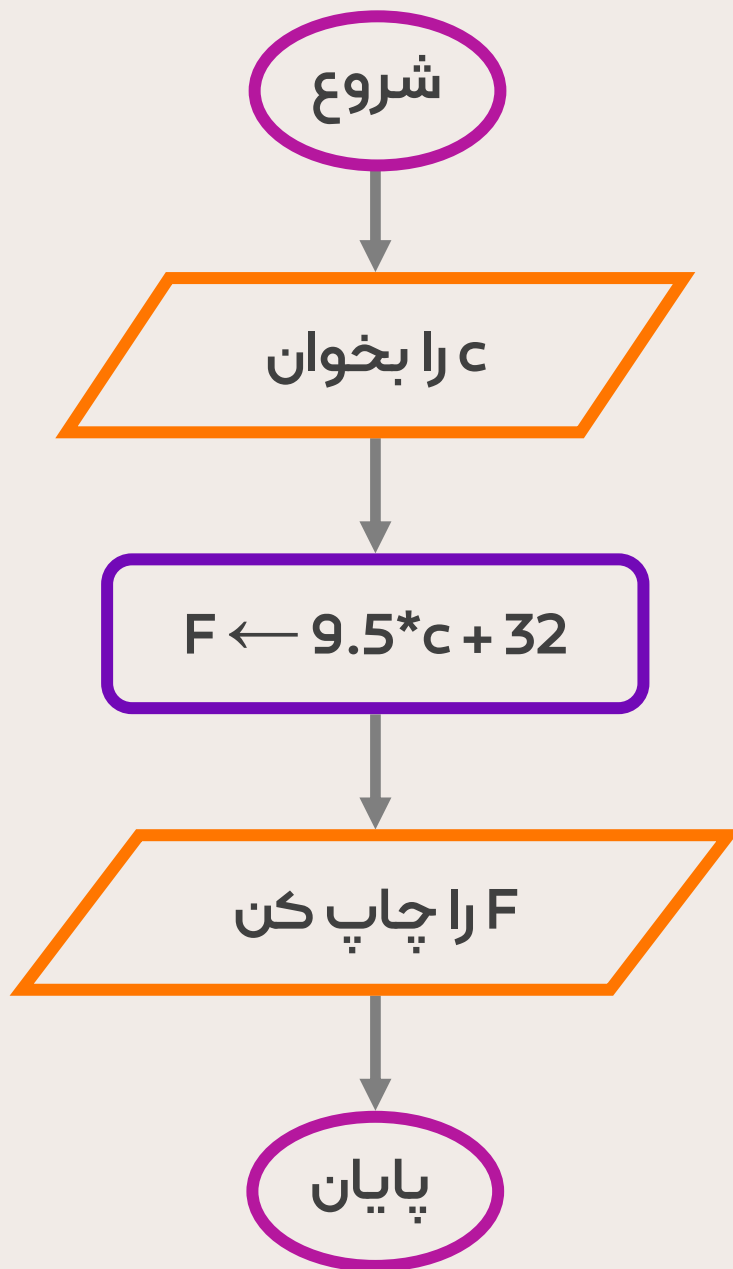


مثال 3: الگوریتمی بنویسید شعاع دایره را دریافت و مساحت و محیط را محاسبه کند.



# فلوچارت

مثال 4: الگوریتمی بنویسید که درجه حرارت سانتی گراد را به فارنهایت تبدیل کند.



تعام!

**شرط در فلوچارت**

# فلوچارت -> شرط

مثال 5: الگوریتمی بنویسید که دو ورودی دریافت کند و عدد بزرگتر را در خروجی چاپ کند.

شروع

ورودی‌های  $x$  و  $y$  را دریافت کن.

اگر  $x$  بزرگتر است،  $x$  را در  $\max$  ذخیره کن.

درغیراین‌صورت،  $y$  را در  $\max$  ذخیره کن.

پایان





# فلوچارت -> شرط

مثال 5: الگوریتمی بنویسید که دو ورودی دریافت کند و عدد بزرگتر را در خروجی چاپ کند.

شروع

ورودی‌های  $x$  و  $y$  را دریافت کن.

اگر  $x$  بزرگتر است،  $x$  را در  $\max$  ذخیره کن.

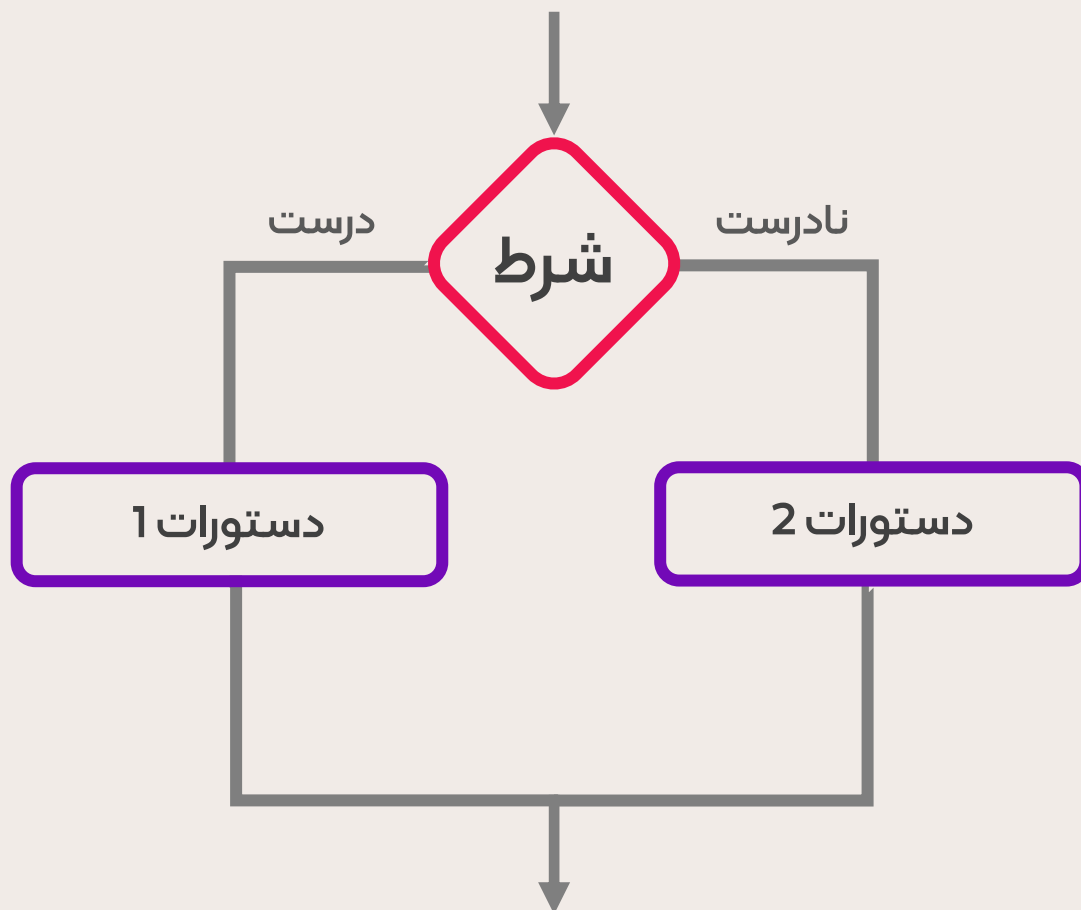
در غیر این صورت،  $y$  را در  $\max$  ذخیره کن.

پایان

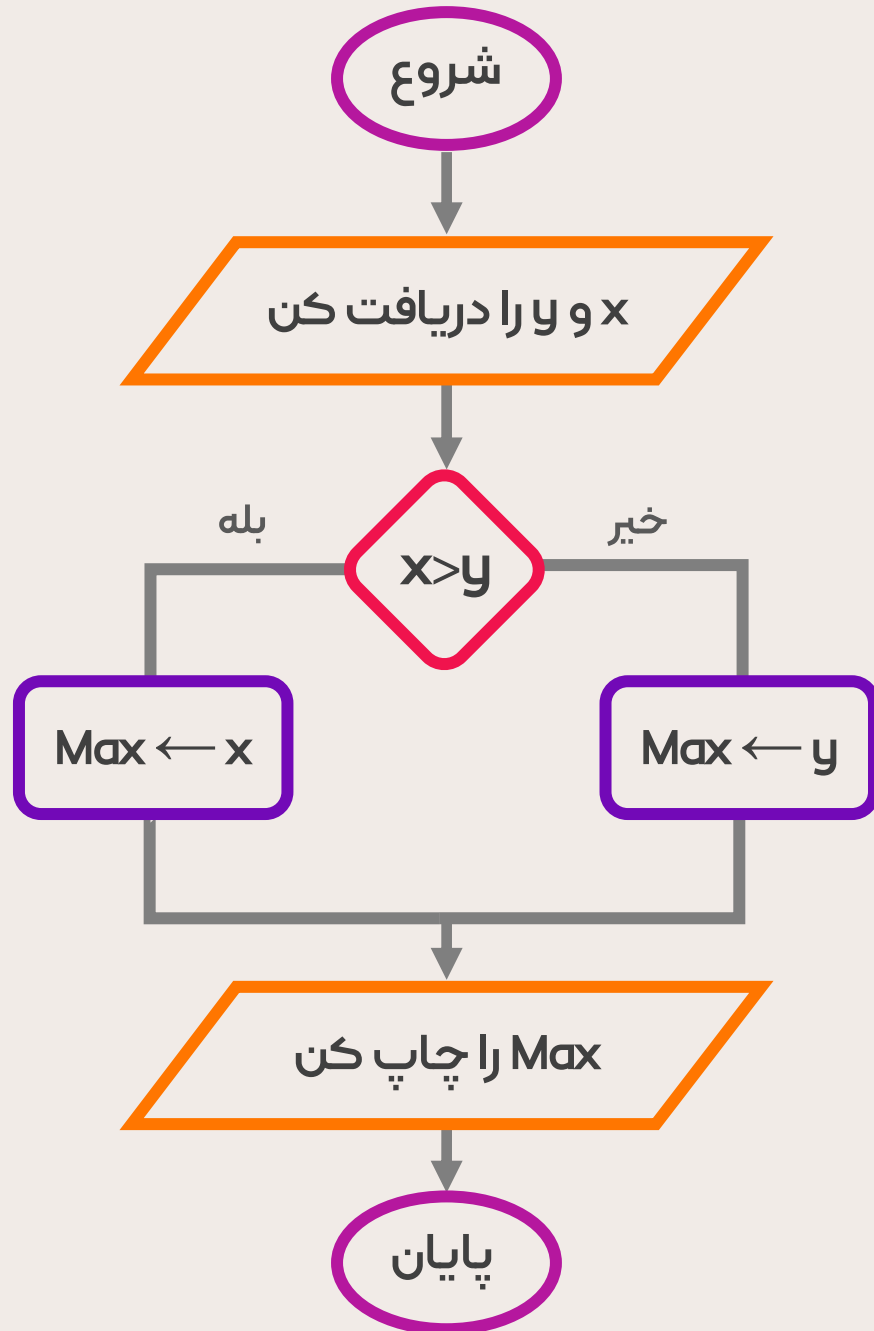


# فلوچارت -> شرط

■ علامت شرط در فلوچارت:



# فلوچارت -> شرط



مثال 5: الگوریتمی بنویسید که دو ورودی دریافت کند

و عدد بزرگتر را در خروجی چاپ کند.

شروع

ورودی‌های x و y را دریافت کن.

اگر x بزرگتر است، x را در max ذخیره کن.

درغیراین‌صورت، y را در max ذخیره کن.

max را چاپ کن.

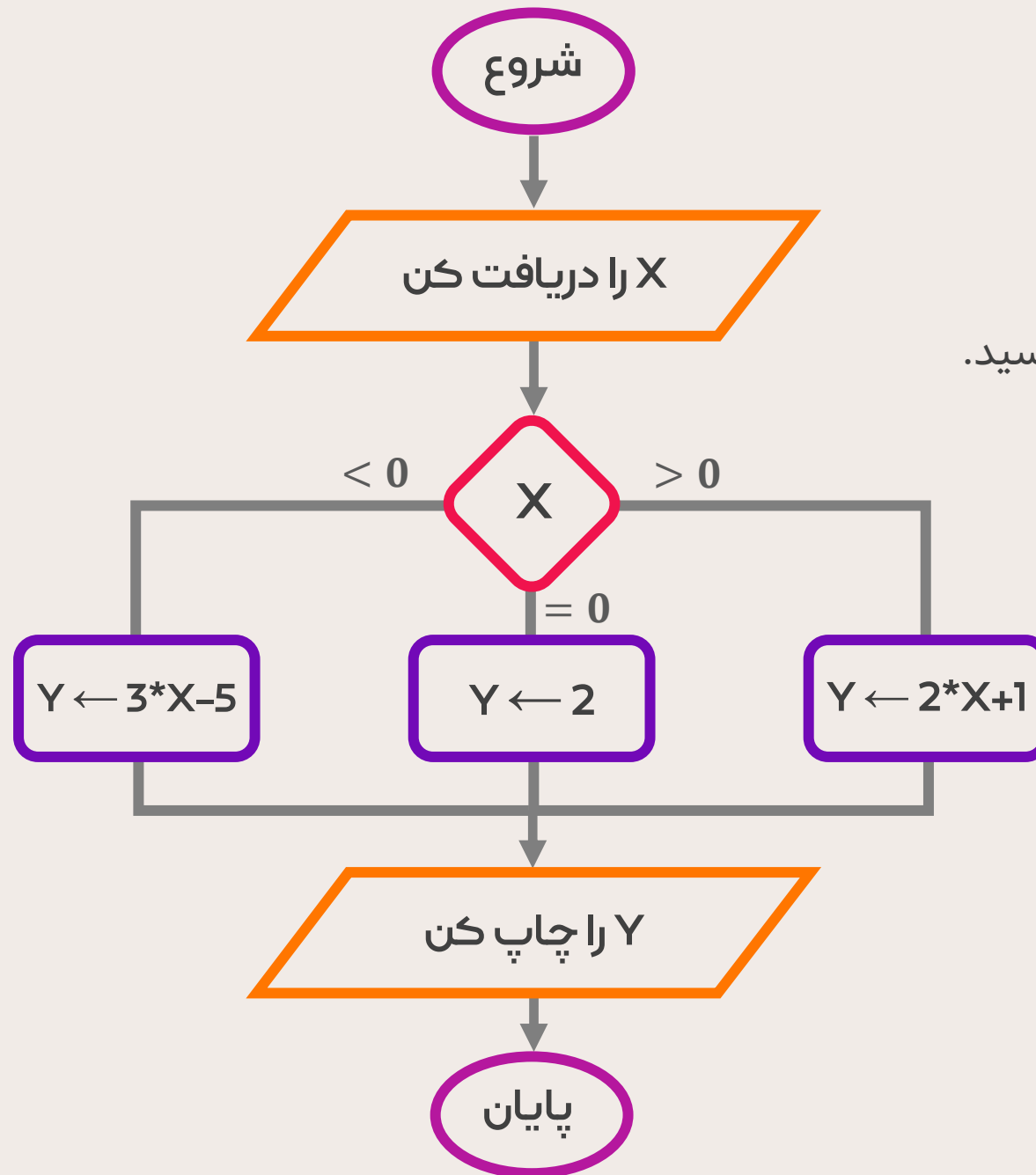
پایان



# فلوچارت -> شرط

مثال 6: الگوریتمی برای رابطه زیر بنویسید.

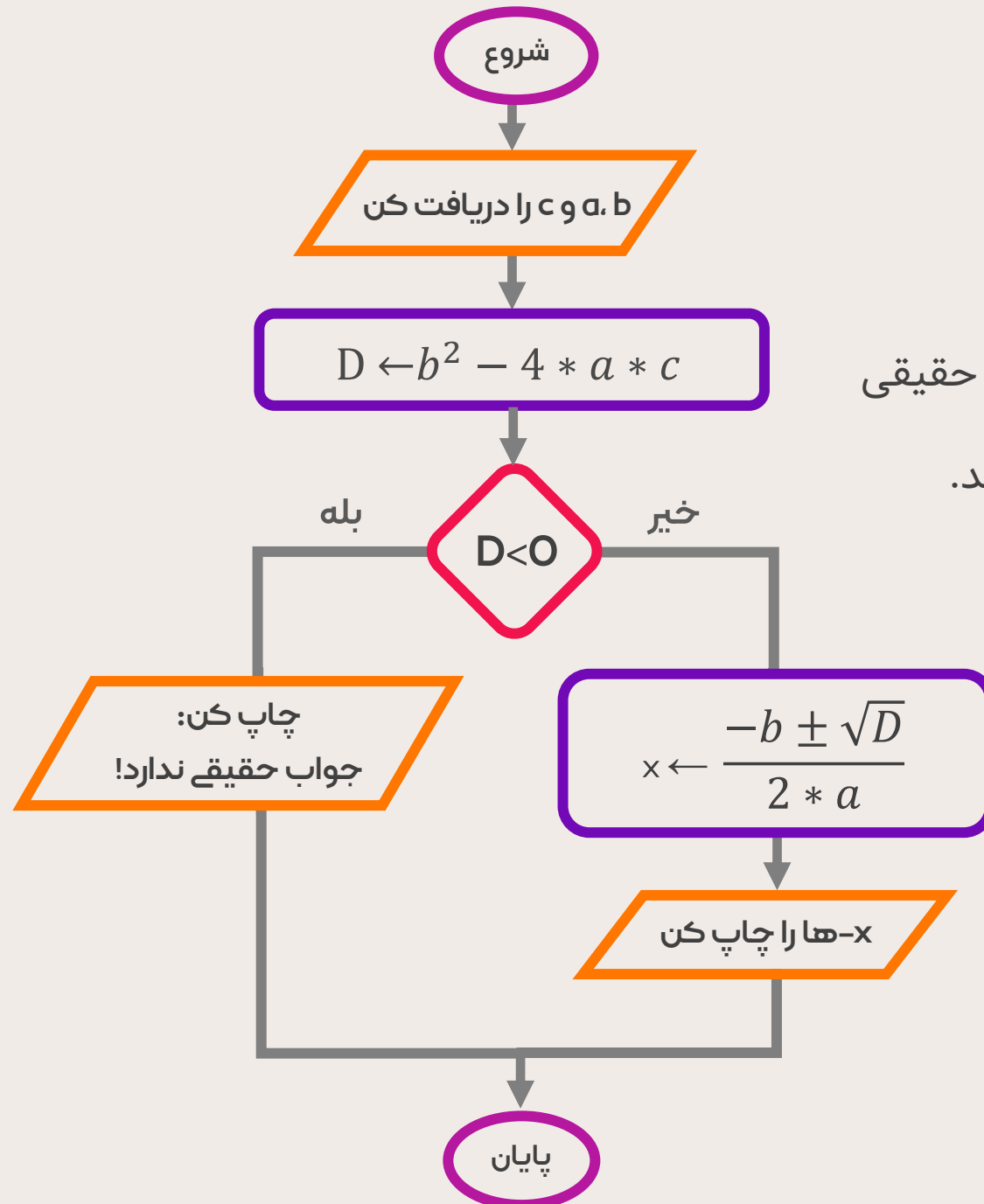
$$Y = \begin{cases} 3X - 5 & X < 0 \\ 2 & X = 0 \\ 2X + 1 & X > 0 \end{cases}$$



# فلوچارت - شرط

مثال 7: الگوریتمی بنویسید که جواب‌های حقیقی

معادله  $ax^2 + bx + c = 0$  را پیدا و چاپ کند.



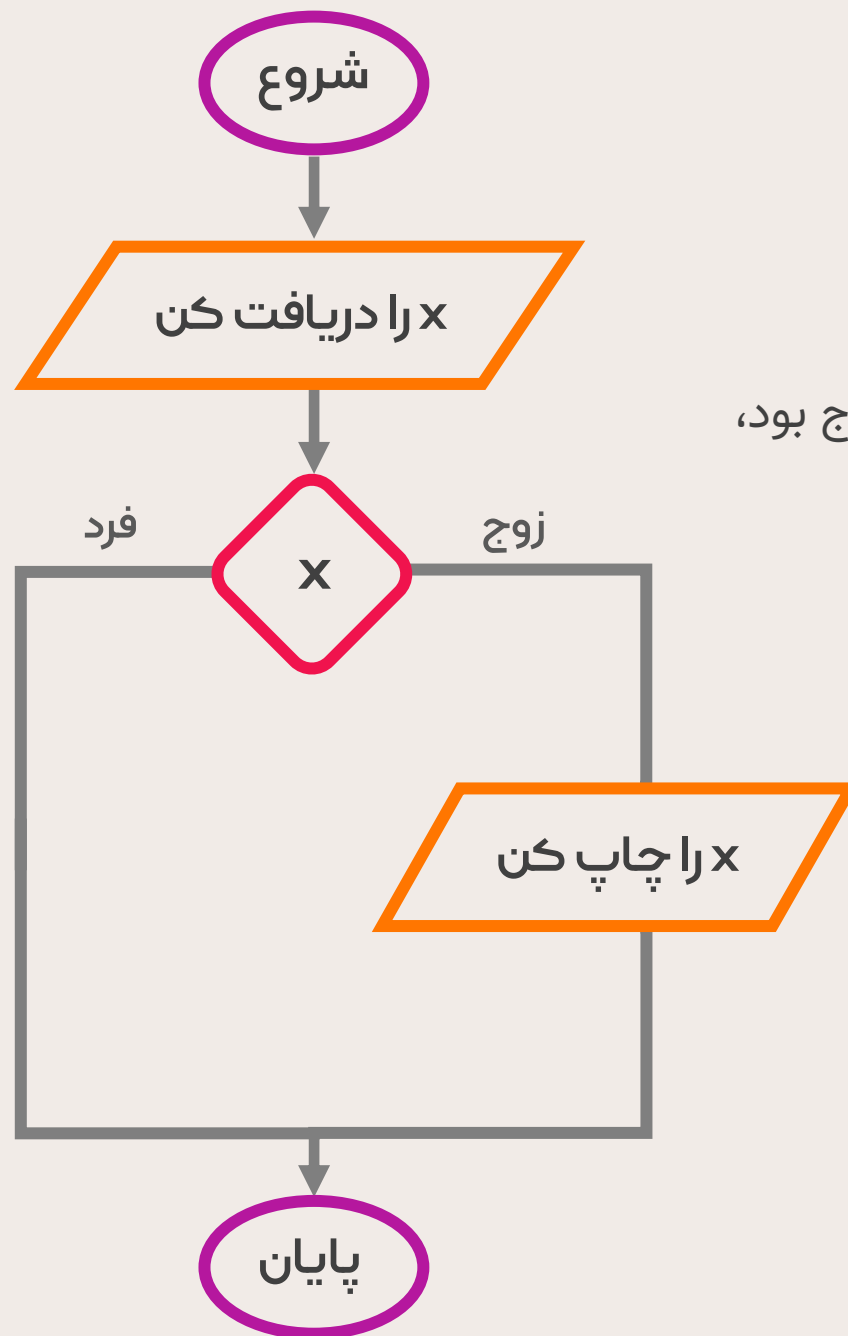
تعام!

# حلقه در فلوچارت

## فلوچارت - حلقه

مثال 8: الگوریتمی بنویسید که اگر ورودی زوج بود،

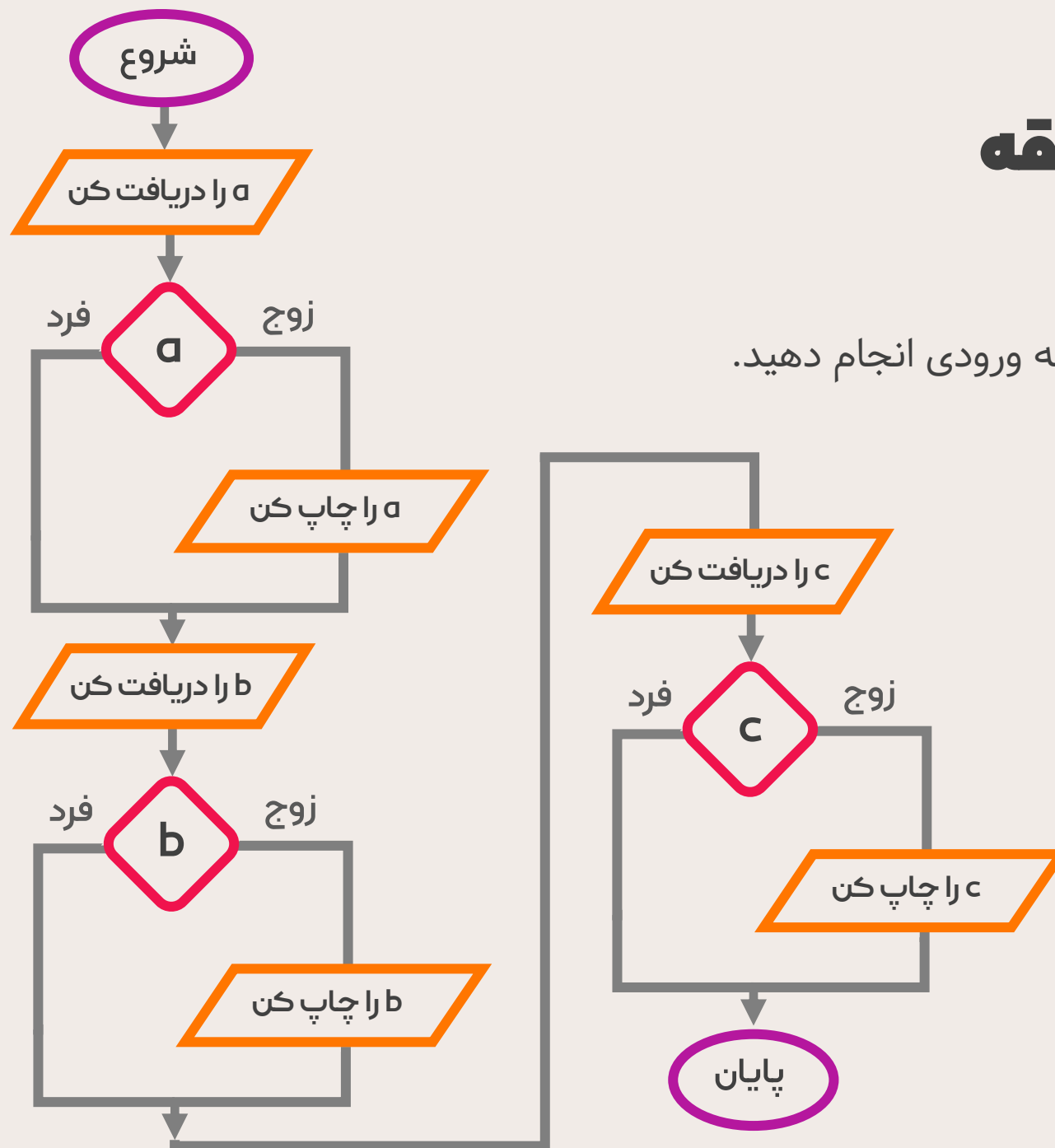
در خروجی چاپ شود.





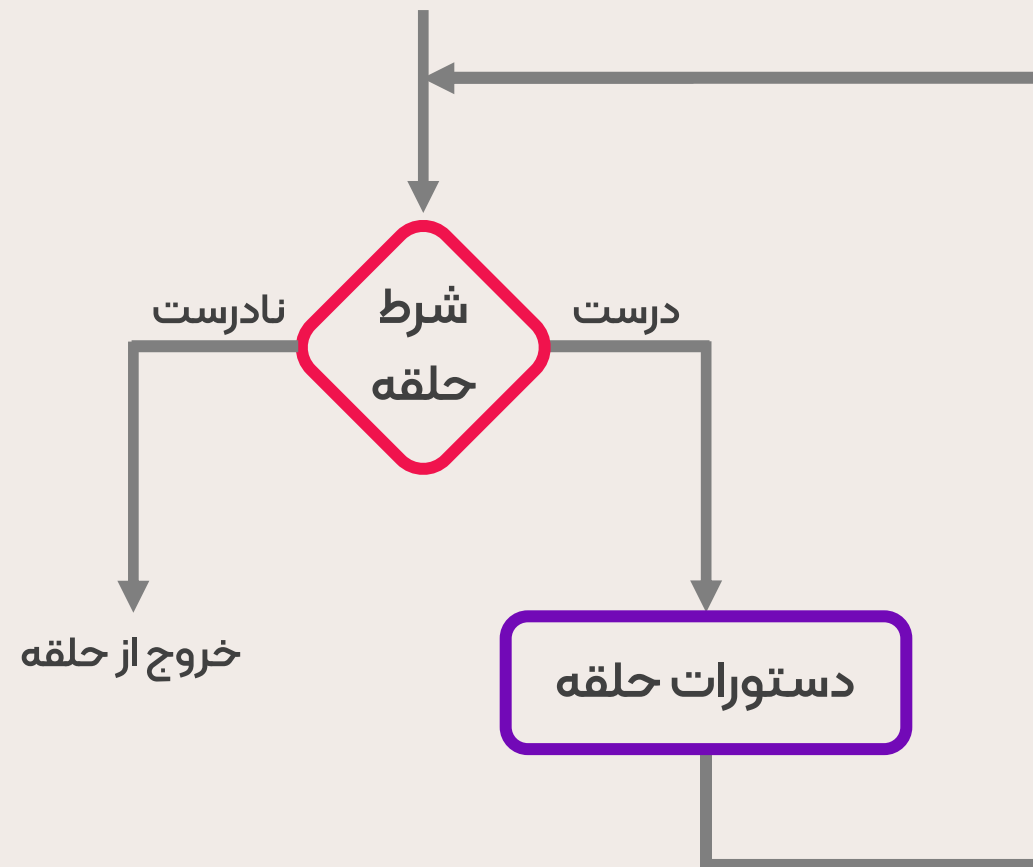
# فلوچارت - حلقه

مثال 9: مثال 8 را برای سه ورودی انجام دهید.



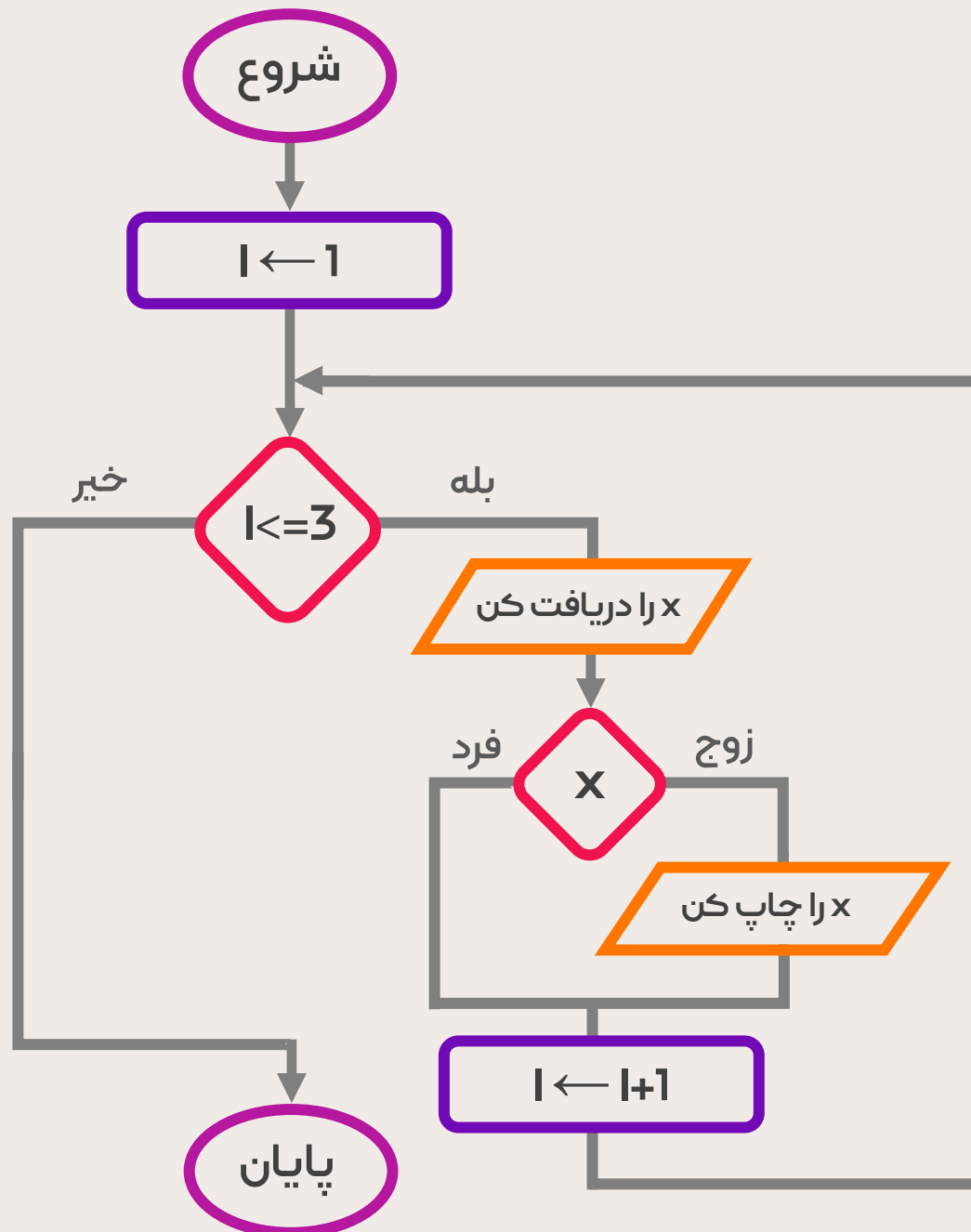
# فلوچارت - < حلقه

■ علامت حلقه در فلوچارت:



# فلوچارت - حلقه

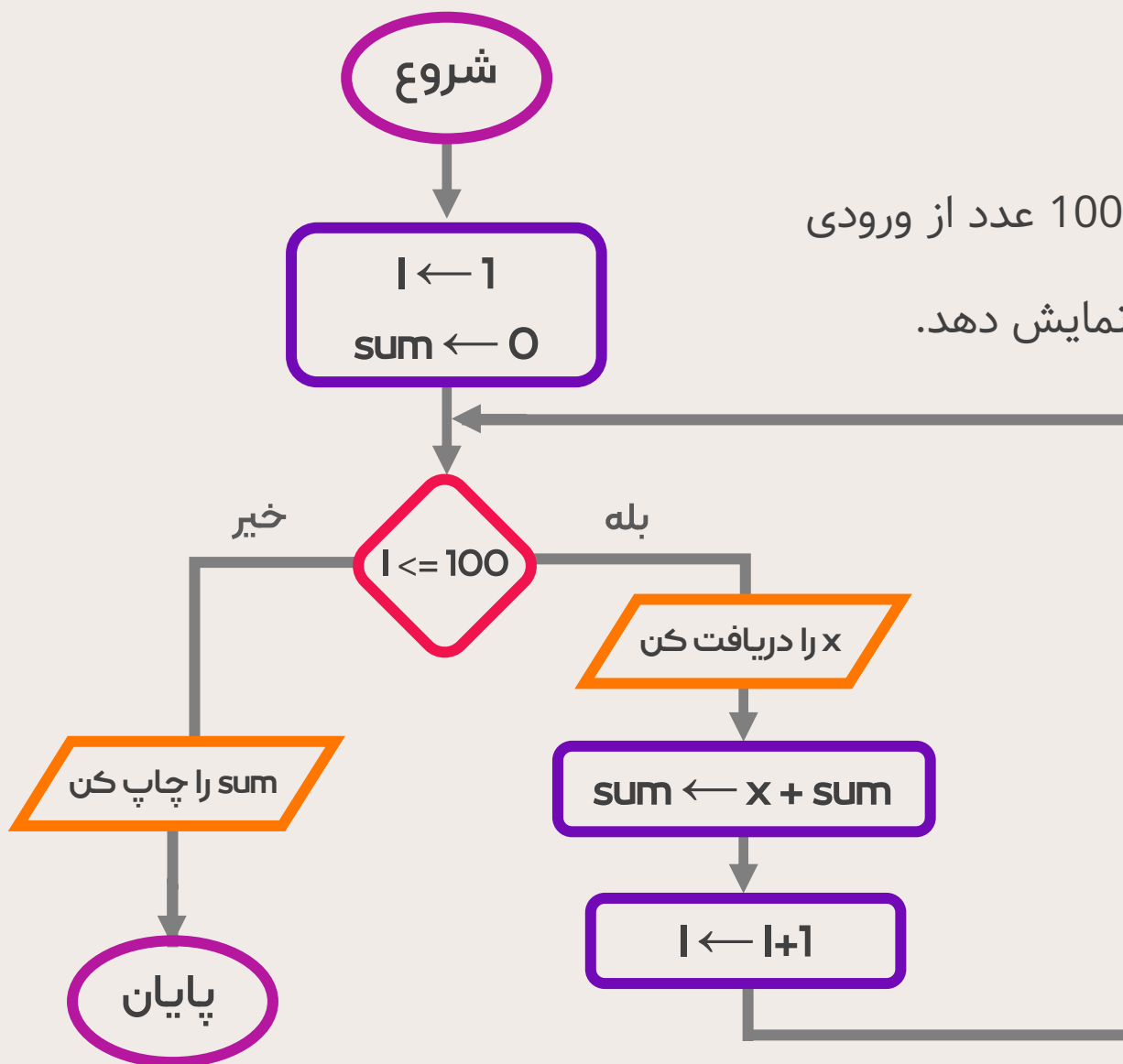
مثال 9: مثال 8 برای سه ورودی



# فلوچارت - حلقه

مثال 10: الگوریتمی بنویسید که 100 عدد از ورودی

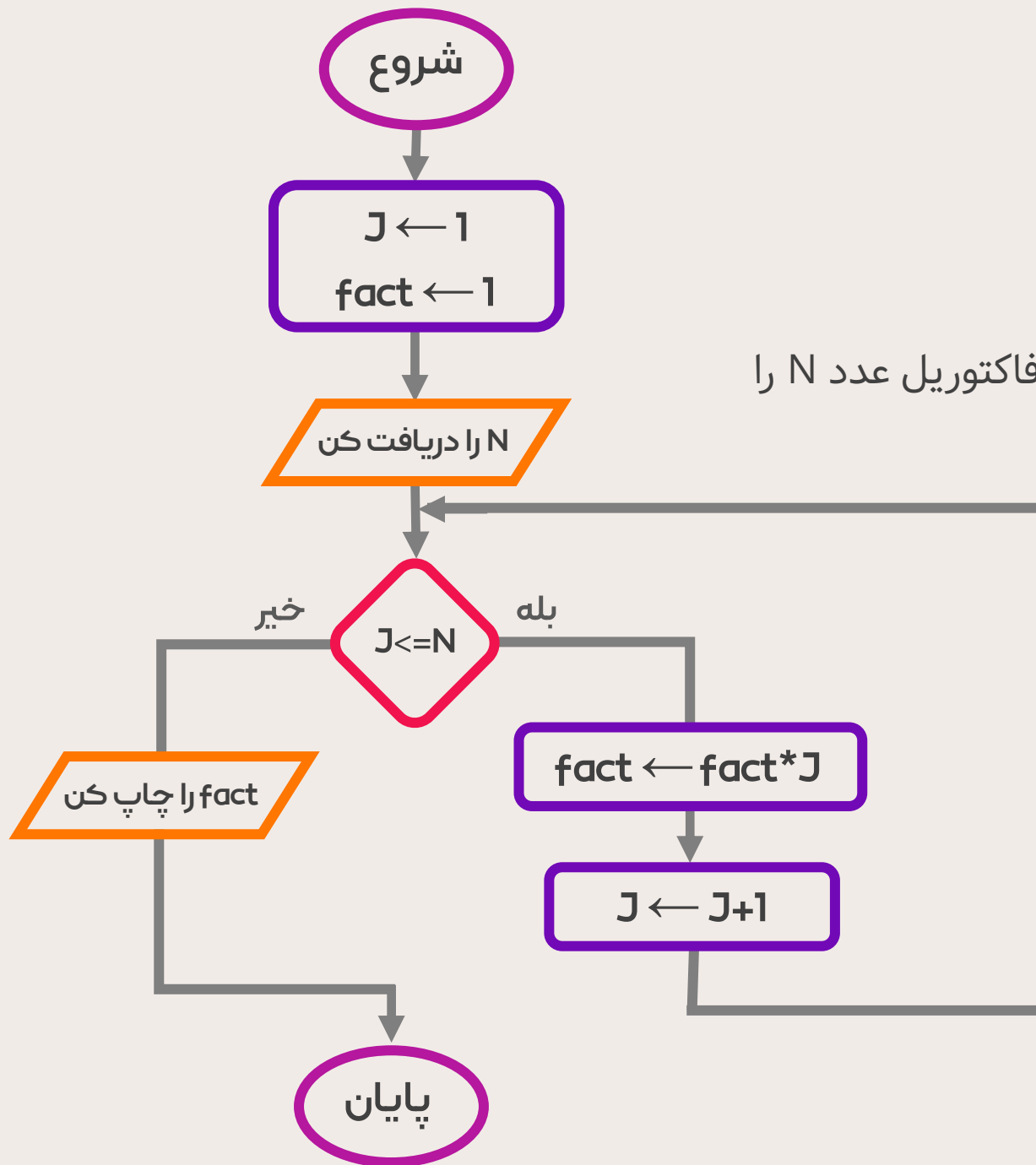
دریافت و مجموع آن را در خروجی نمایش دهد.



## فلوچارت - حلقه

مثال 11: الگوریتمی بنویسید که فاکتوریل عدد  $N$  را

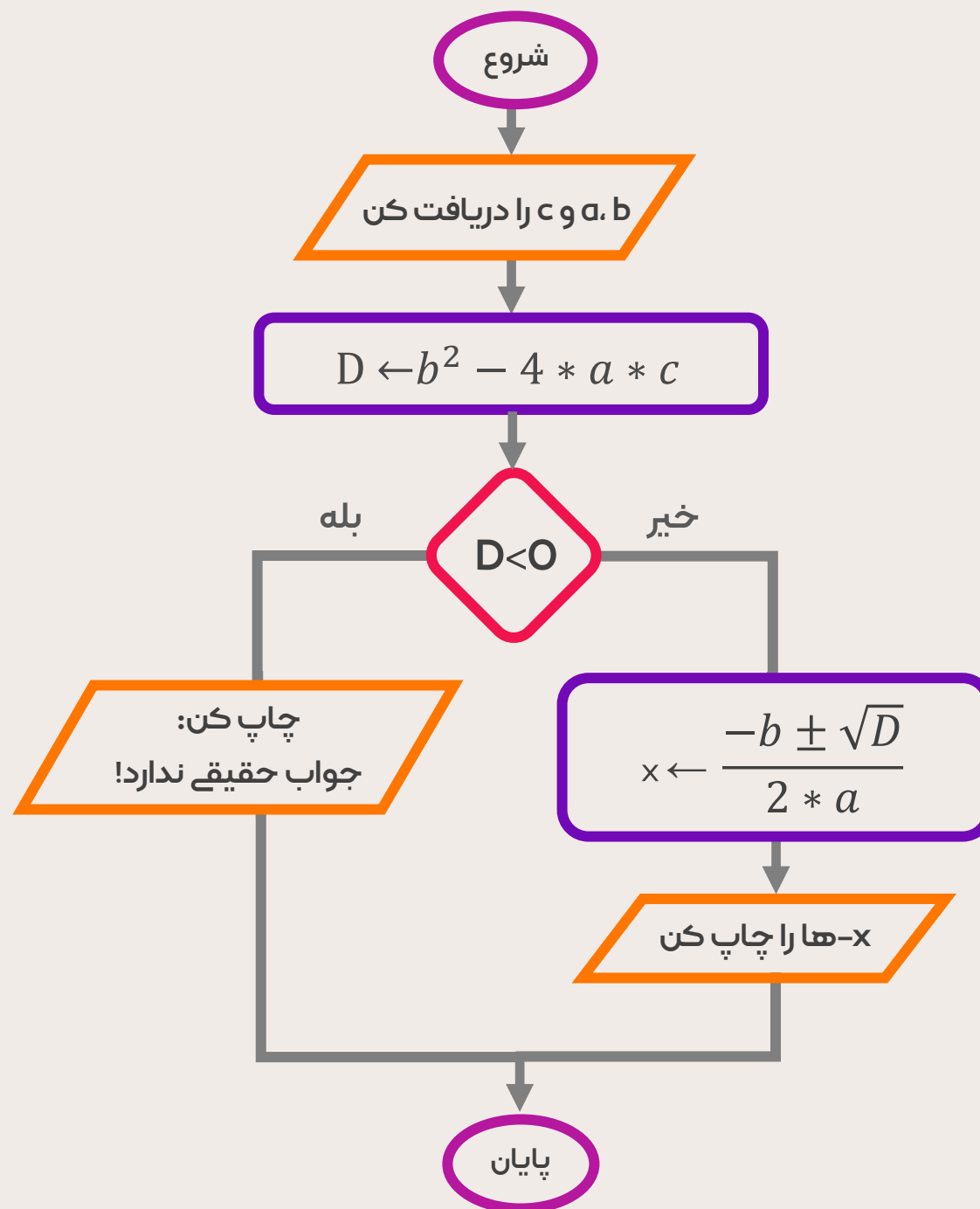
محاسبه کند.



**تعام!**

**دَقَّتْ كُنْ!**

# دقت کن!





**تعام!**

**حالا نوبت شماست...**