

# الگوریتم و فلوچارت

- در زندگی روزمره، انسان با مسائل مختلفی روبروست و برای هر کدام از این مسائل (حل مشکلات) راه حلی و روشی را بر می‌گزیند. مسائلی از قبیل راه رفتن، غذا خوردن، خوابیدن و غیره که بشر تقریباً هر روز آنها را پیش روی خود دارد.
- همه این مسائل نیاز به روشی برای حل کردن دارند مثلاً راه رفتن باید با ترتیب خاصی و مراحل معینی انجام شود تا مسئله راه رفتن برای بشر حل شود. اصطلاحاً به این مراحل قدم به قدم برای حل مسئله را الگوریتم آن مسئله می‌نامند
- کلمه الگوریتم از نام دانشمند پراوازه ایرانی به نام الخوارزمی گرفته شده است.

# الگوریتم و فلوچارت

- شناخت مسئله

– بررسی داده ها و یا معلومات (ورودیها)، مجهولات (خروجیها) و یافتن ارتباط منطقی بین داده ها و مجهولات (تحلیل راه حل)

– مثال : یافتن مساحت یک مثلث با داشتن اندازه قاعده و ارتفاع آن

- داده ها (ورودی)

- اندازه ارتفاع و قاعده مثلث

- مجهولات (خروجی)

- مساحت مثلث

- رابطه منطقی :

- روش محاسبه مساحت مثلث (ارتفاع  $\times$  قاعده  $\div 2$ )

– اغلب، مسائل دارای راه حلهای گوناگونی می باشند، یافتن بهترین راه حل به ابتکار، تمرین و از همه مهمتر تجربه بستگی دارد.

# تعریف الگوریتم

تعدادی دستورالعمل پشت سرهم که مراحل مختلف کاری را  
با زبانی دقیق و با جزئیات کافی بیان نماید بطوریکه ترتیب  
مراحل و خاتمه پذیر بودن عملیات در آن کاملاً مشخص  
شده باشد را الگوریتم گویند.

# مراحل الگوریتم

برای حل یک مسئله باید الگوریتم آن مسئله را مشخص کنیم (یا بیابیم). که اصطلاحاً طراحی الگوریتم برای آن مسئله نامیده می‌شود. در طراحی الگوریتم معمولاً سه مرحله زیر را از هم جدا می‌کنند:

- خواندن داده‌ها
- انجام محاسبات
- خروجی‌ها

# مجری الگوریتم

- الگوریتم ها می توانند به دو صورت انجام شوند
  - توسط ماشین (کامپیوتر)

- بعد از تبدیل شدن به زبان مناسب برای ماشین بطور اتوماتیک توسط ماشین قابل اجرا هستند.

– توسط انسان

- برای حصول اطمینان از عملکرد صحیح الگوریتم، گاهی خودمان آن را بصورت دستی دنبال نموده و در واقع خود مجری الگوریتم می شویم.

# کاربرد الگوریتم

- همه ما در طی روز برای انجام کارهای روزمره از روش الگوریتمی (و یا منطقی) استفاده می کنیم.
  - مانند مطالعه کتاب
  - تعویض چرخ پنجر شده
  - پختن غذا
  - پختن کیک و ....
- در واقع برای انجام هر یک از این کارها، لازم است تعدادی دستورالعملهای ساده تر را به ترتیب مناسب اجراء کرده تا به نتیجه مطلوب برسیم.

# اجزای اصلی الگوریتم

- هر مساله راه حل و الگوریتم خاص خود را دارد.
- می توان برای حل یک مساله روشهای گوناگونی را ارائه داد.
- اما تمام الگوریتم ها دارای این اجزاء هستند:
  1. **نقطه شروع:** حل مساله از کجا آغاز می گردد
    - فقط یک نقطه شروع در الگوریتم وجود دارد
  2. **نقطه پایان:** جایی که مراحل حل مساله پایان می پذیرد.
    - به هر حال الگوریتم بایستی در یک نقطه خاتمه یابد.
    - می توان چندین نقطه پایان برای الگوریتم داشت.
  3. **دستورالعملها و یا جملات اجرایی**

# روشهای بیان الگوریتم

- الگوریتم را می توان به صورتهای مختلفی بیان کرد:

1. بیان الگوریتم با جملات فارسی

2. بیان ریاضی الگوریتم

3. بیان الگوریتم توسط شکلها (فلوچارت)



# متغیر (Variable)

- به خانه ای از حافظه که داده ها و اطلاعات ورودی یا خروجی و یا اطلاعات موقت را در خود نگه می دارد متغیر گفته می شود.
- مقدار متغیر می تواند در طول اجرای الگوریتم تغییر داشته باشد.

0							
1	1	1	1	0	0	0	1
2							
i	1	1	0	1	0	0	1
N							

← A

# مثالی از یک الگوریتم

- الگوریتم محاسبه و چاپ مجموع دو عدد ۱۰ و ۲۰

1. شروع
2. عدد ۱۰ را در خانه (متغیر) A قرار بده
3. عدد ۲۰ را در خانه B قرار بده
4. محتویات خانه های A , B را با هم جمع کن و در خانه C قرار بده
5. مقدار خانه C را بعنوان نتیجه چاپ کن
6. پایان

# استفاده از بیان ریاضی

- بیان الگوریتم در قالب جملات نوشتاری طولانی و فهم الگوریتم را دشوار می سازد.
- الگوریتم محاسبه و چاپ مجموع دو عدد ۱۰ و ۲۰
  1. شروع
  2.  $A \leftarrow 10$
  3.  $B \leftarrow 20$
  4.  $C \leftarrow A+B$
  5. چاپ مقدار C
  6. پایان

# انواع جملات مورد استفاده در الگوریتم ها

- جملات شرطی
- جملات عملیاتی ( یا محاسباتی )
- جملات ورودی / خروجی
- جملات توضیحی

# مثال

- الگوریتمی بنویسید که اعداد زوج دو رقمی را چاپ کند.

(می دانیم که کوچکترین عدد زوج دو رقمی ۱۰ و اعداد زوج به اندازه ۲ واحد از هم فاصله دارند)

1. شروع
2.  $J \leftarrow 10$
3. J را چاپ کن
4.  $J \leftarrow J + 2$
5. اگر  $J \leq 98$  است آنگاه به مرحله ۳ برو
6. پایان

مثال : الگوریتمی بنویسید که دو عدد از ورودی دریافت کرده مجموع دو عدد را محاسبه و چاپ نماید.

ورودیها

a , b

انجام محاسبات

جمع دو عدد

خروجیها

مجموع دو عدد

- ۱- شروع
- ۲- a , b را بخوان.
- ۳- مجموع a , b را محاسبه و در sum قرار بده.
- ۴- sum را در خروجی چاپ کن
- ۵- پایان

مثال: الگوریتمی بنویسید که سه عدد از ورودی دریافت کرده مجموع و میانگین سه عدد را محاسبه و چاپ کند.

ورودیها

a

b , c

انجام محاسبات

محاسبه مجموع

محاسبه میانگین

خروجیها

چاپ مجموع

چاپ میانگین

۱- شروع

۲- سه عدد از ورودی بخوان

۳- مجموع سه عدد را محاسبه و در sum قرار بده.

۴- sum را بر سه تقسیم کرده، در ave قرار بده.

۵- sum , ave را در خروجی چاپ کن.

۶- پایان.

# مثال

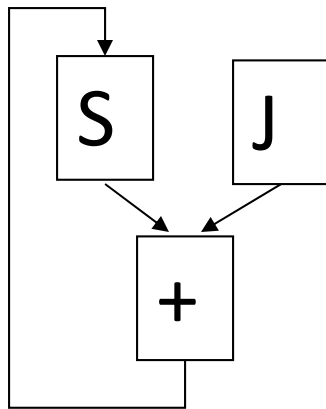
- الگوریتمی بنویسید که اعداد زوج از ۱۰۰۰ تا ۲۰۰۰ را تولید کرده و مجموع آنها را هم محاسبه کند.
- 1. شروع
- 2.  $S \leftarrow 0$  ،  $J \leftarrow 1000$
- 3.  $J$  را چاپ کن و  $S \leftarrow S + J$
- 4.  $J \leftarrow J + 2$
- 5. اگر  $J \leq 2000$  است آنگاه به مرحله ۳ برو در غیر اینصورت مقدار  $S$  را چاپ کن
- 6. پایان



# نکته

- اگر بخواهیم مقدار حاصل جمعی را محاسبه کنیم (مانند مثال قبل که مجموع اعداد زوج از ۱۰۰۰ تا ۲۰۰۰)

- ابتدا متغیری (مانند S) در نظر می گیریم و مقدار اولیه آن را صفر می گذاریم (یعنی اینکه هنوز هیچ مجموعی را حساب نکرده ایم)،  $S \leftarrow 0$
- سپس **تک تک** جملاتی را که قرار است با هم جمع کنیم را تولید کرده (با کمک یک متغیر دیگر، مثلاً J در مثال قبلی) و با متغیر S جمع نموده و حاصل را در S قرار می دهیم (در واقع مقدار جدید را به حاصل قبلی می افزاییم - مانند انباره)،  $S \leftarrow S + J$
- در نهایت حاصل مجموع در این متغیر S قرار می گیرد.



# ویژگیهای یک الگوریتم خوب

اگر چه یک مساله راه حل‌های مختلفی دارد، مهم یافتن بهترین راه حل است

- **سادگی**

- حتی الامکان ساده و عاری از ابهام و پیچیدگی باشد.

- **در نظر گرفتن تمام حالات خاص**

- الگوریتم بتواند در برابر حالات و شرایط مختلف پاسخ و جواب مناسبی ارائه دهد.

- بطور مثال هنگام حل معادله درجه دوم حالت‌های منفی زیر رادیکال را در نظر بگیرد.

- **روان بودن متن الگوریتم**

- دستورالعمل‌ها گویا بوده و منظور آنها بسادگی درک شود.

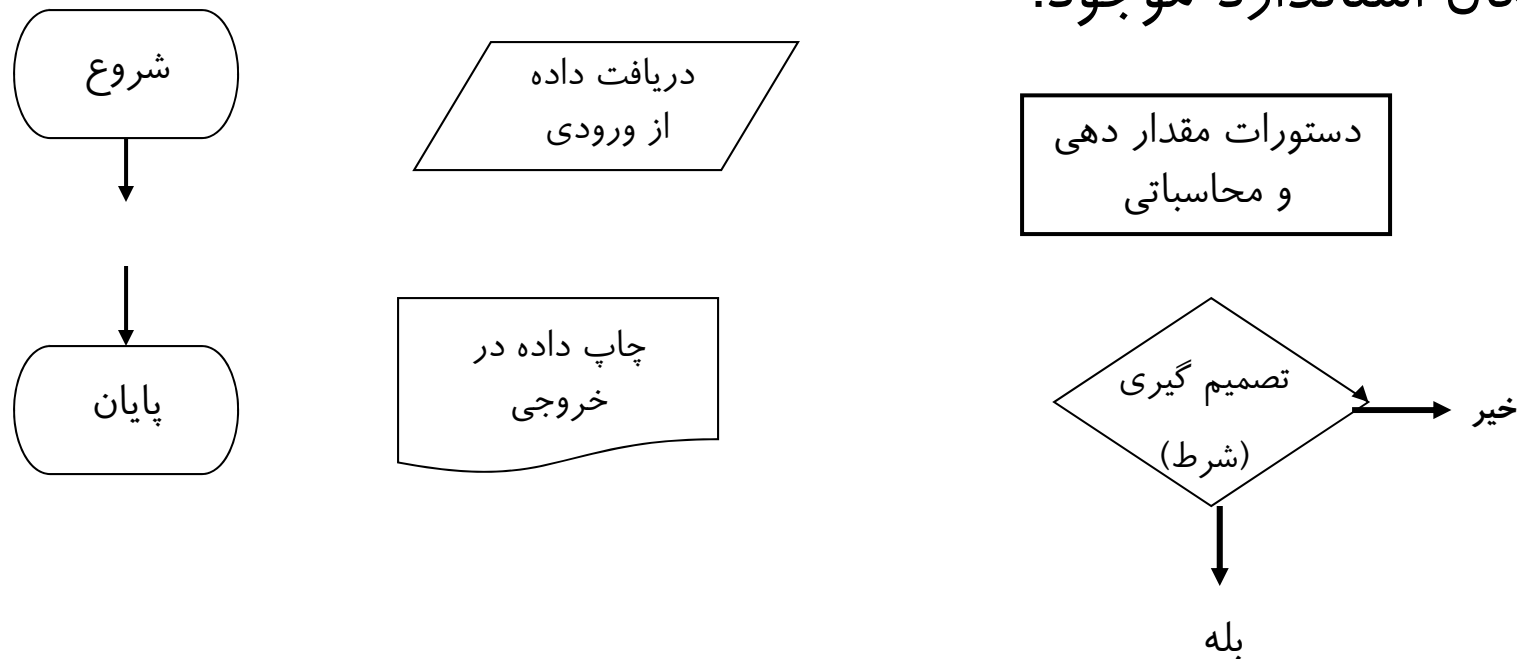
- **حداقل بودن تعداد دستورات و جملات**

# ایجاد حلقه های تکرار (Loops)

- گاهی اوقات برای حل مساله باید یک یا چند مرحله از دستورات را تکرار نمود.
- به مرحله‌ای از الگوریتم که اجرای آنها چندین بار تکرار می شود حلقه (Loop) و یا حلقه تکرار گفته می شود
- بطور کلی حلقه های تکرار از اجزای زیر تشکیل شده است:
  - شمارنده حلقه (Counter)
    - یک متغیر کمکی که پیش از شروع حلقه به آن مقدار اولیه داده می شود
    - از طریق آن می توان تعداد دفعات تکرار حلقه را نشان داد.
  - گام افزایش (Step)
    - مقداری که پس از هر بار مراحل حلقه به شمارنده اضافه می شود.
  - شرط پایانی
    - مقدار و یا متغیری است که پس از اجرای دستورات حلقه با شمارنده حلقه مقایسه می گردد و زمان پایان اجرای دستورات حلقه را مشخص می سازد.
  - بدنه حلقه
    - دستورالعملها و جملاتی که عملیات اصلی حلقه را تشکیل می دهند.

# فلوچارت (Flowchart) یا نمودار گردش

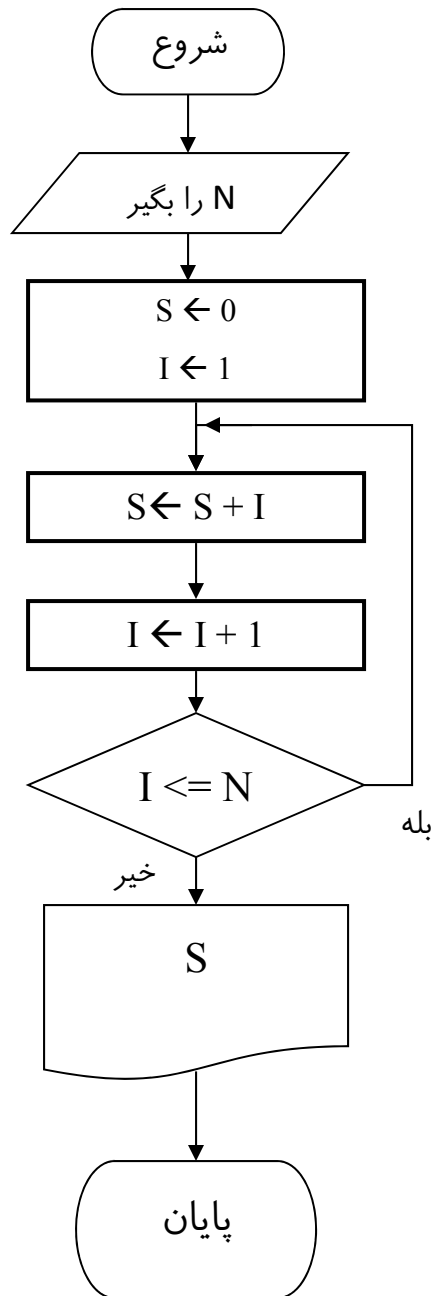
- فلوچارت، بیان تصویری الگوریتم با کمک مجموعه ای استاندارد از اشکال ساده می باشد
- فلوچارت یکی از روشهای برقراری ارتباط منطقی بین مراحل مختلف حل مساله است.
- اشکال استاندارد موجود:



## فلوچارت

مثال: با دریافت N از ورودی،  
مجموع زیر را حساب  
کند.

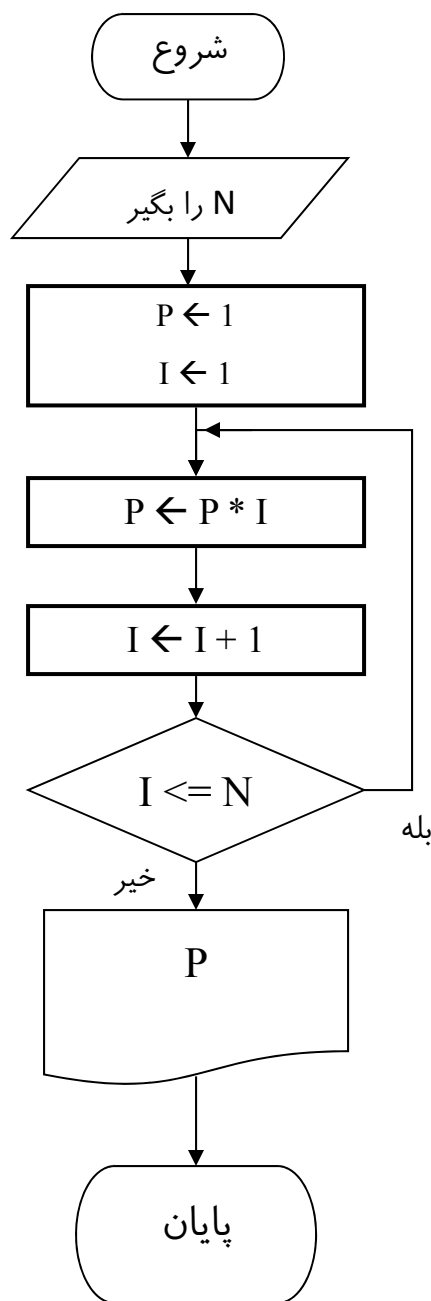
$$S = 1+2+3+ \dots +N$$



## مثال

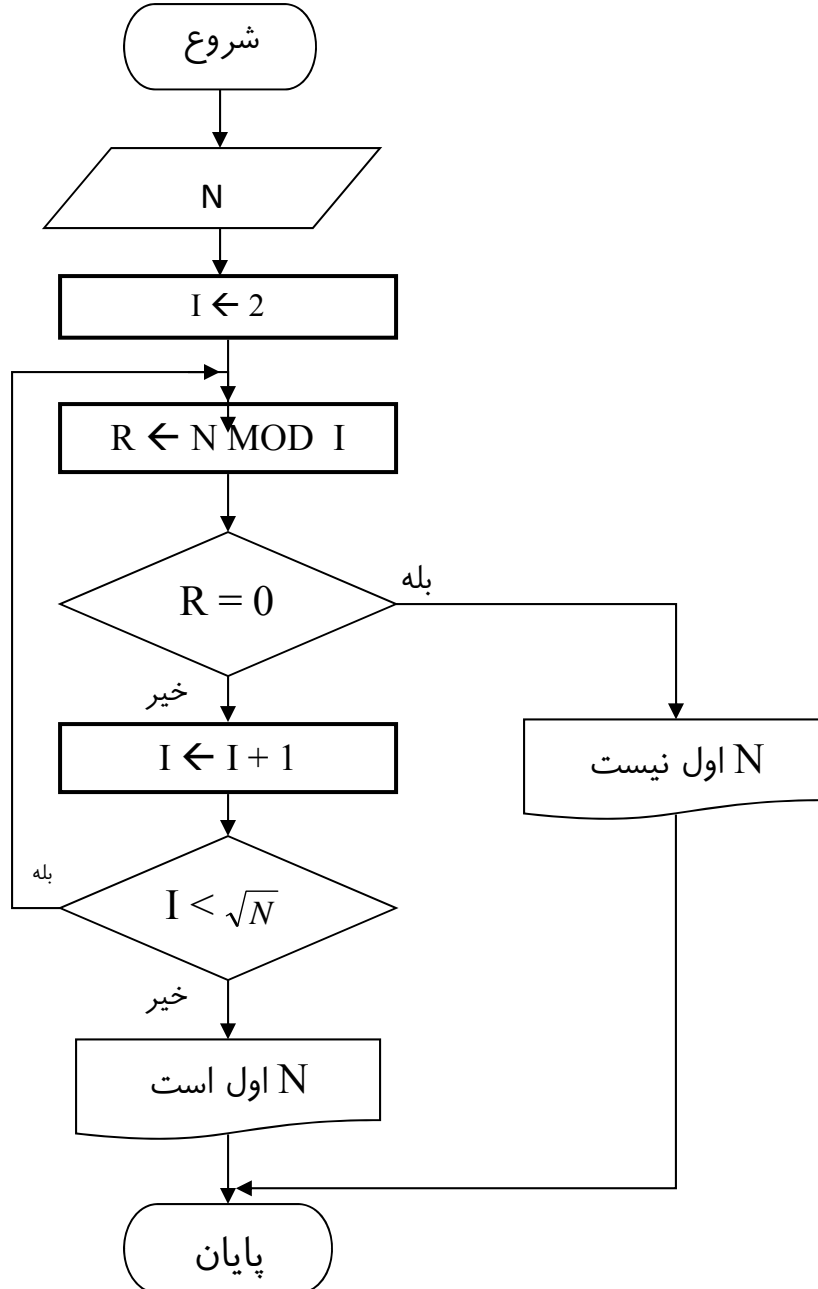
مثال: با دریافت N از ورودی ،  
فاکتوریل آنرا محاسبه و  
چاپ کند.

$$N! = 1 * 2 * 3 * \dots * (N-1) * N$$



## مثال

- عدد طبیعی  $N$  را دریافت کند و معین کند که  $N$  عددی اول است یا خیر؟



## مثال

• عدد طبیعی  $N$  را دریافت کند و معین کند که  $N$  عددی کامل است یا خیر؟

• عدد کامل است که خودش برابر با مجموع مقسوم علیه های کوچکتر از خودش باشد.

$$6 \rightarrow 1, 2, 3$$

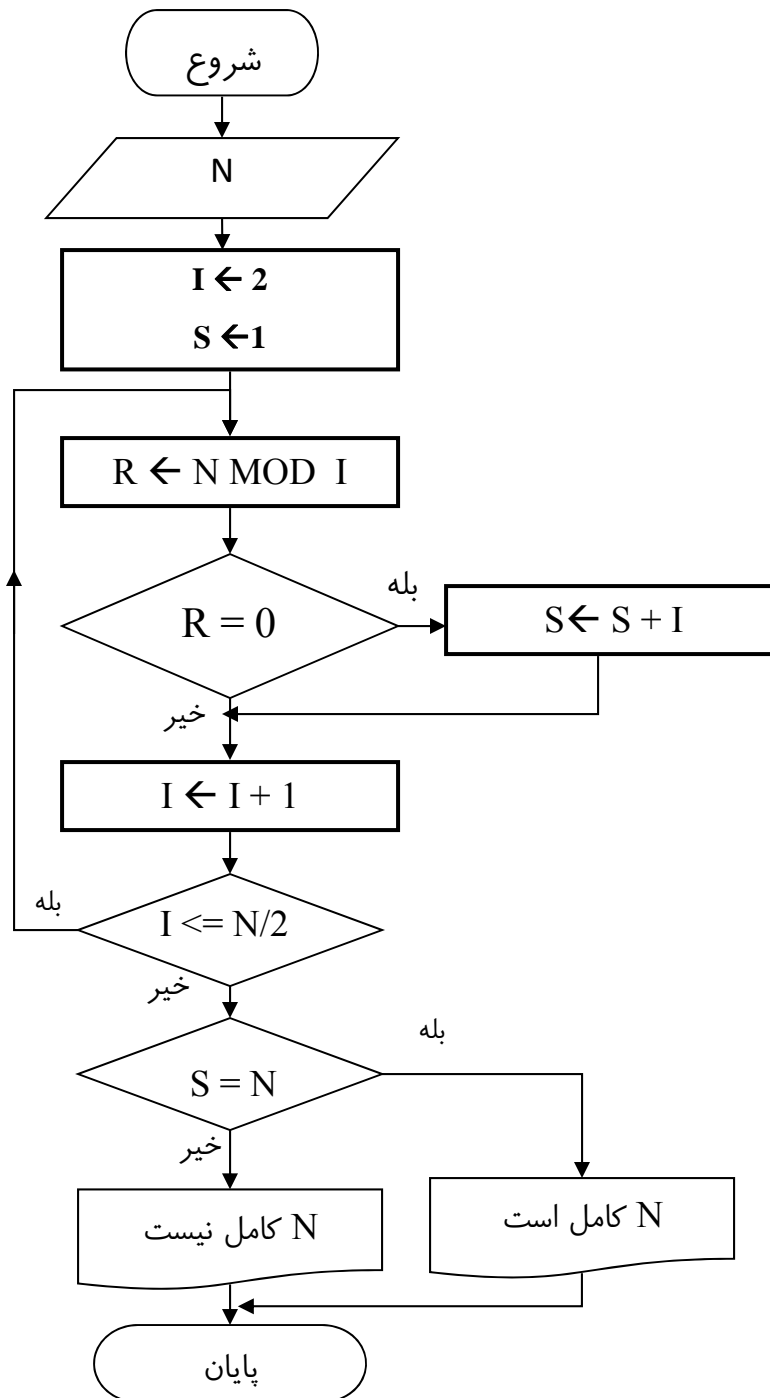
$$6 = 1 + 2 + 3$$

$$28 \rightarrow 1, 2, 4, 7, 14$$

$$28 = 1 + 2 + 4 + 7 + 14$$

$$18 \rightarrow 1, 2, 3, 6, 9$$

$$18 \neq 1 + 2 + 3 + 6 + 9$$



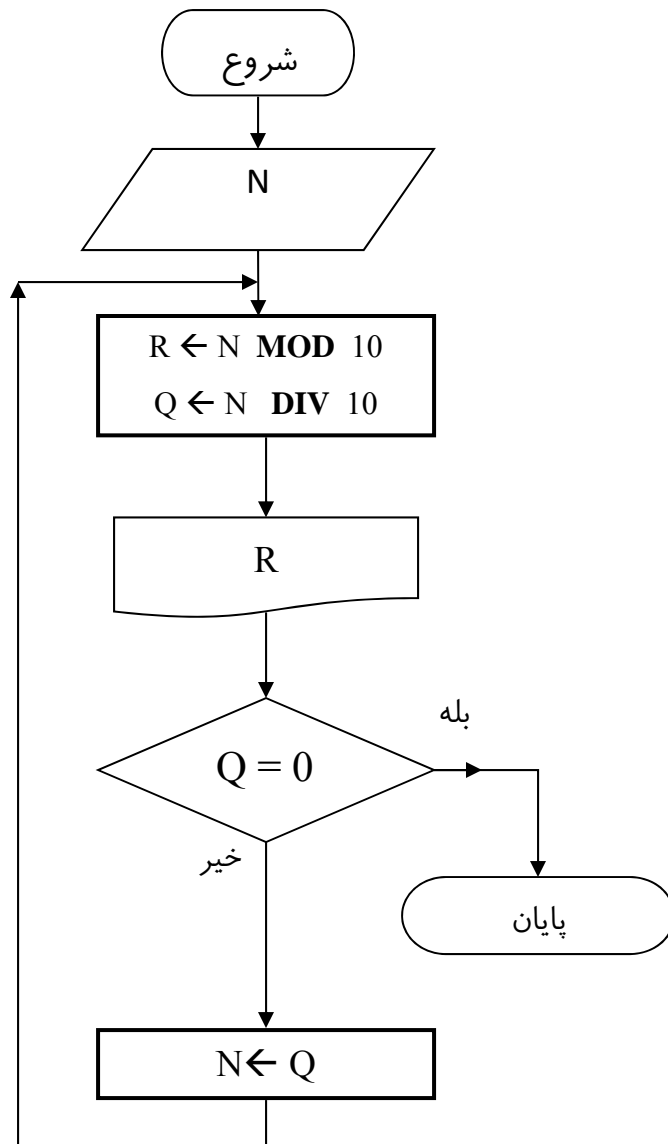


## مثال

- عدد طبیعی  $N$  را دریافت کند و مقلوب آنرا در خروجی چاپ کند.

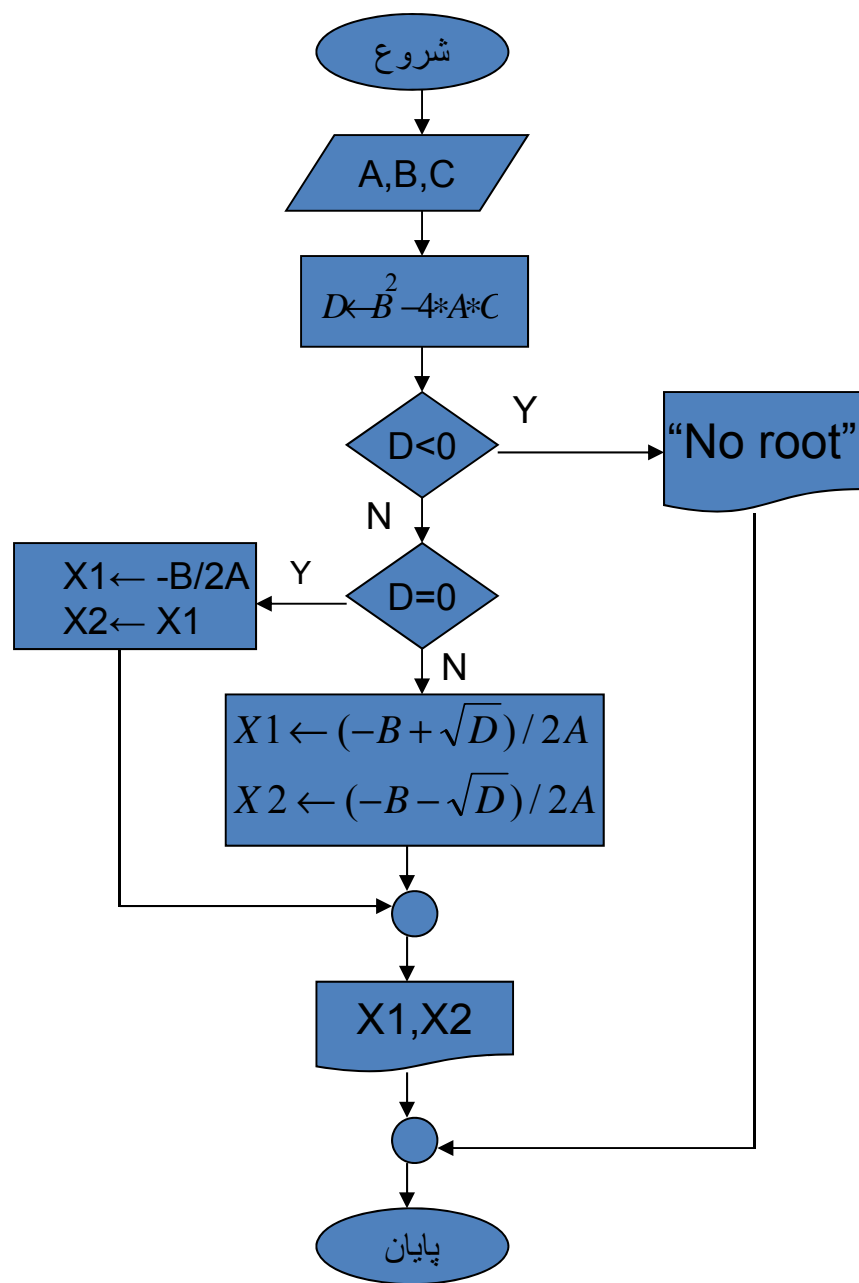
• بطور مثال :

**9527 → 7259**



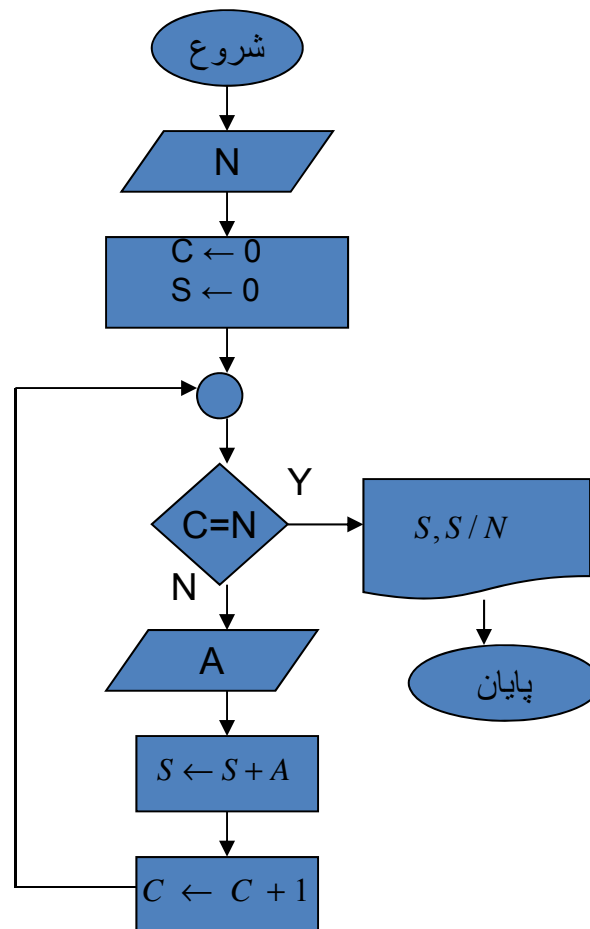
## مثال : ریشه های یک معادله درجه دوم

- $AX^2+BX+C=0$
- $D=B^2-4AC$
- اگر  $D<0$ ، معادله ریشه ندارد
- اگر  $D=0$ ، حاصل عبارت  $-B/2A$  را در  $X_1$  و  $X_2$  قرار بده
- حاصل عبارت  $(-B+\sqrt{D})/(2A)$  را در  $X_1$  قرار بده
- حاصل عبارت  $(-B-\sqrt{D})/2A$  را در  $X_2$  قرار بده
- مقادیر  $X_1$  و  $X_2$  را نمایش بده

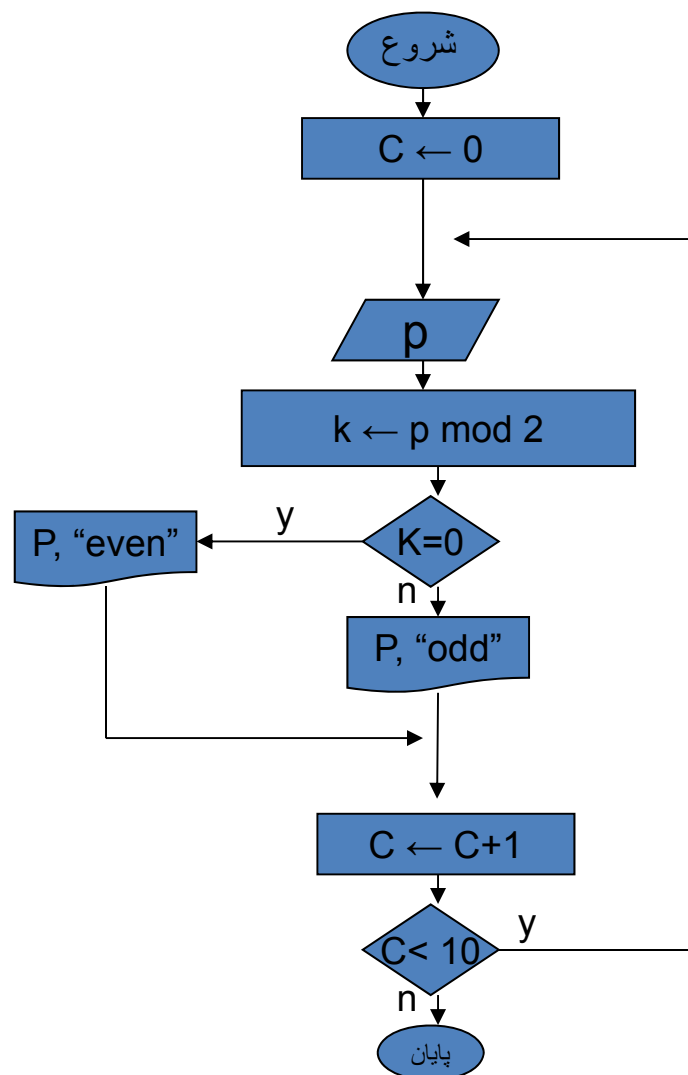


**مثال :** فلوجارتی رسم کنید که یک عدد بزرگتر از صفر را خوانده سپس به تعداد آن عدد، اعداد دیگری را خوانده مجموع و میانگین آنها را نمایش دهد.

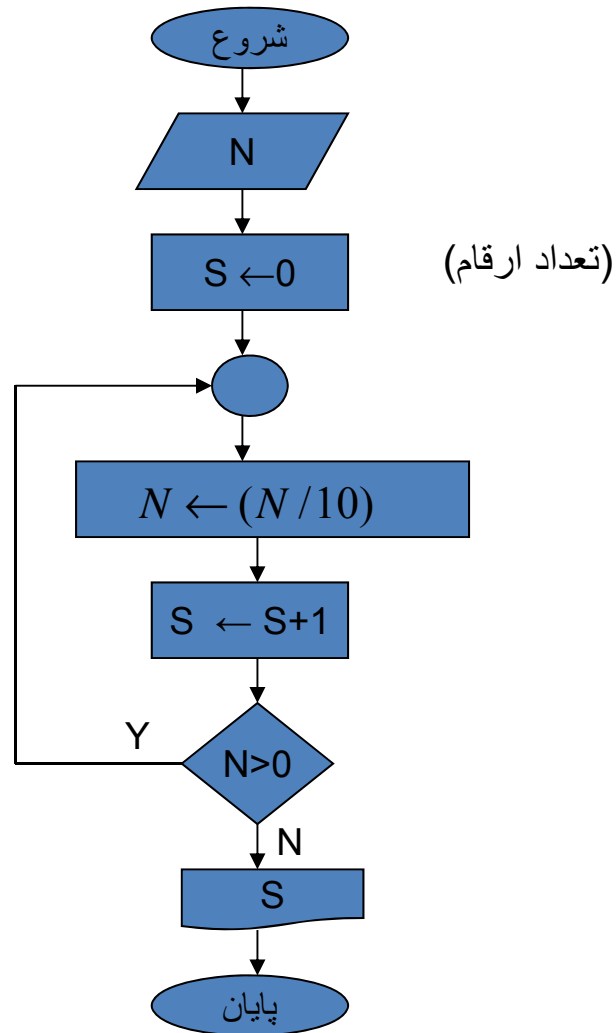
- $N$  عدد خوانده شده
- $C$  شمارنده
- $S$  مجموع
- $A$  اعداد



**مثال 8:** الگوریتم برنامه ای را بنویسید که 10 عدد را گرفته و تعیین کند کدام زوج و کدام فرد است.



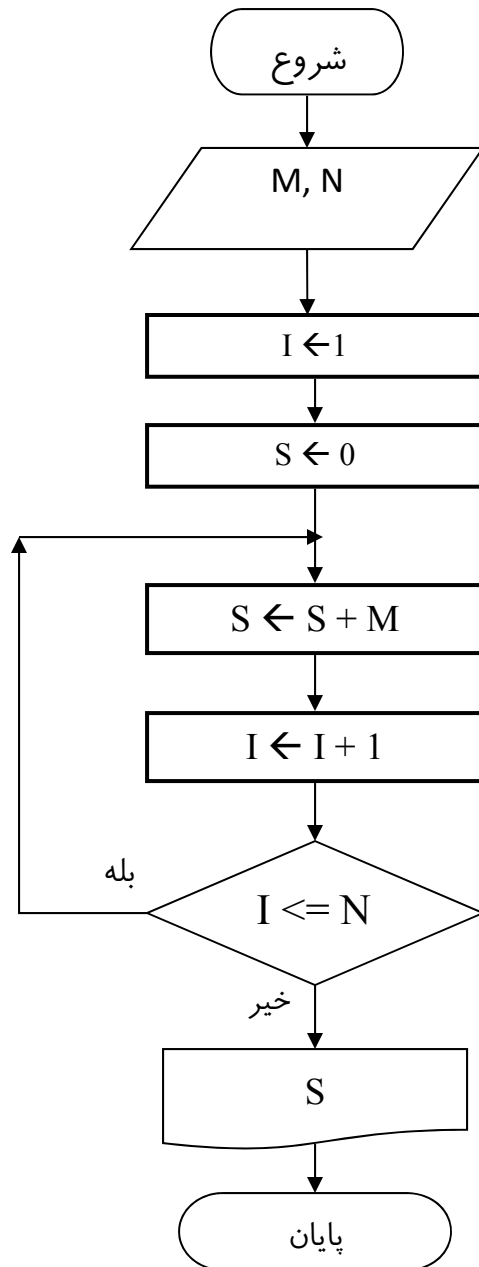
**مثال :** الگوریتم برنامه ای را بنویسید که یک عدد مثبت را خوانده و تعداد ارقام عدد را نشان دهد.



## مثال

به کمک جمع های متوالی، حاصل ضرب دو عدد طبیعی  $M, N$  را بدست آورده و چاپ کند.

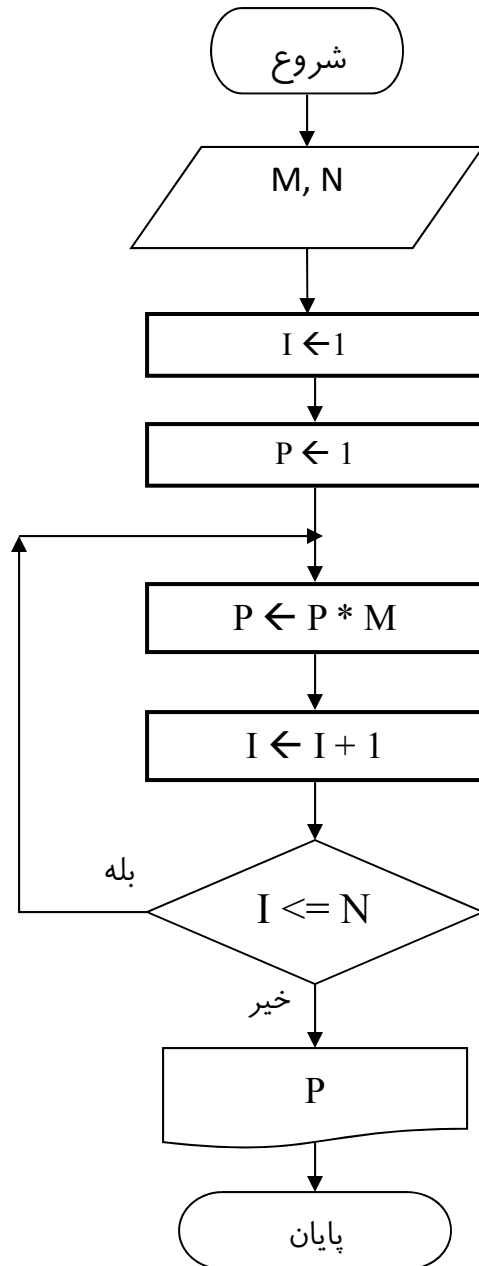
$$M * N = \underbrace{M + M + \dots + M}_{N \text{ مرتبه}}$$



## مثال

به کمک ضربهای متوالی مقدار توان  $M^N$  را بدست آورده و چاپ کند.  
( $M, N$  اعداد طبیعی می باشند)

$$M^N = \underbrace{M * M * \dots * M}_{N \text{ مرتبه}}$$



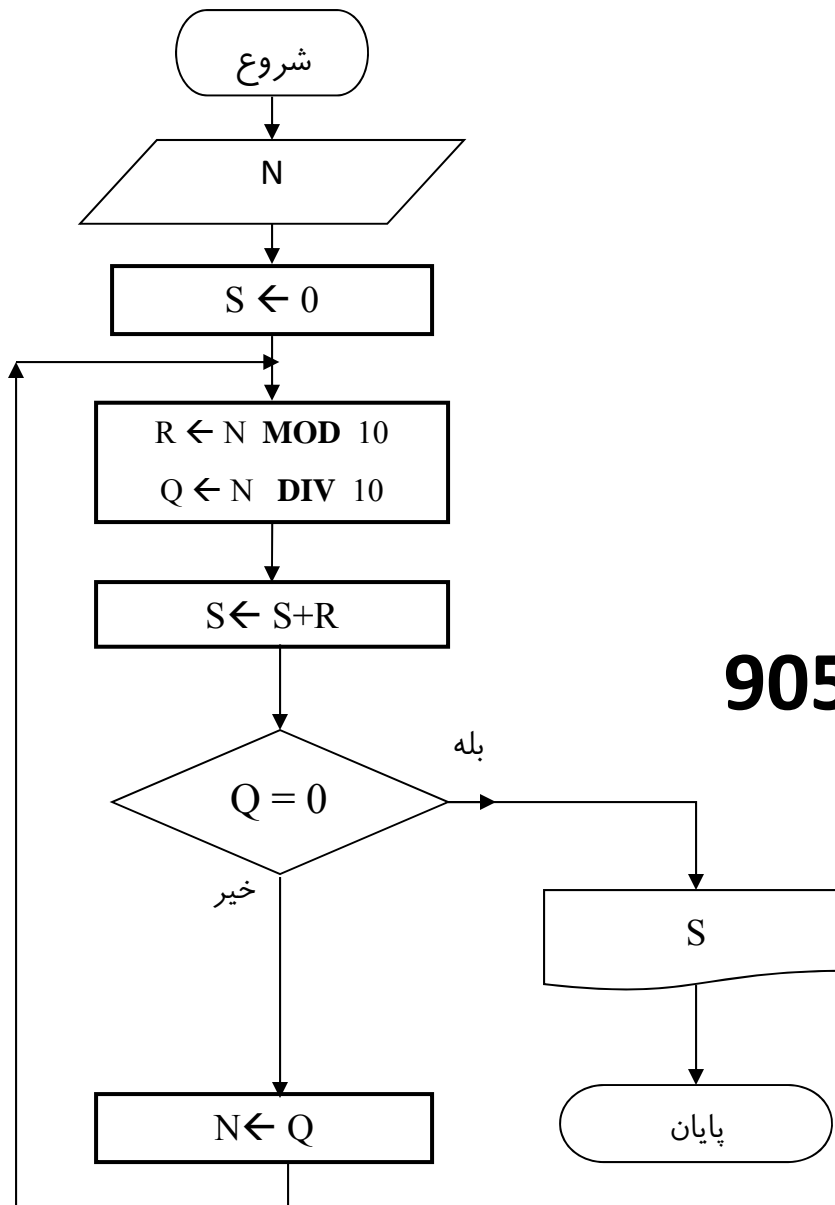


## مثال

- عدد طبیعی  $N$  را دریافت کند و مجموع ارقام آنرا در خروجی چاپ کند.

• بطور مثال :

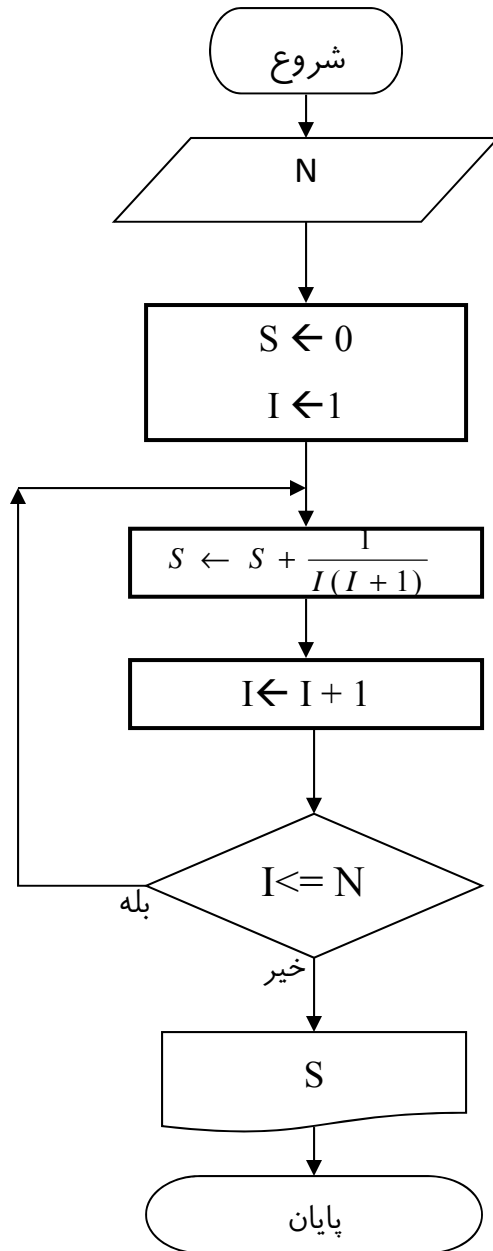
$$90527 \rightarrow 9+0+5+2+7$$



## مثال

- عدد طبیعی N را دریافت کند و مقدار سری زیر را بدست آورد.

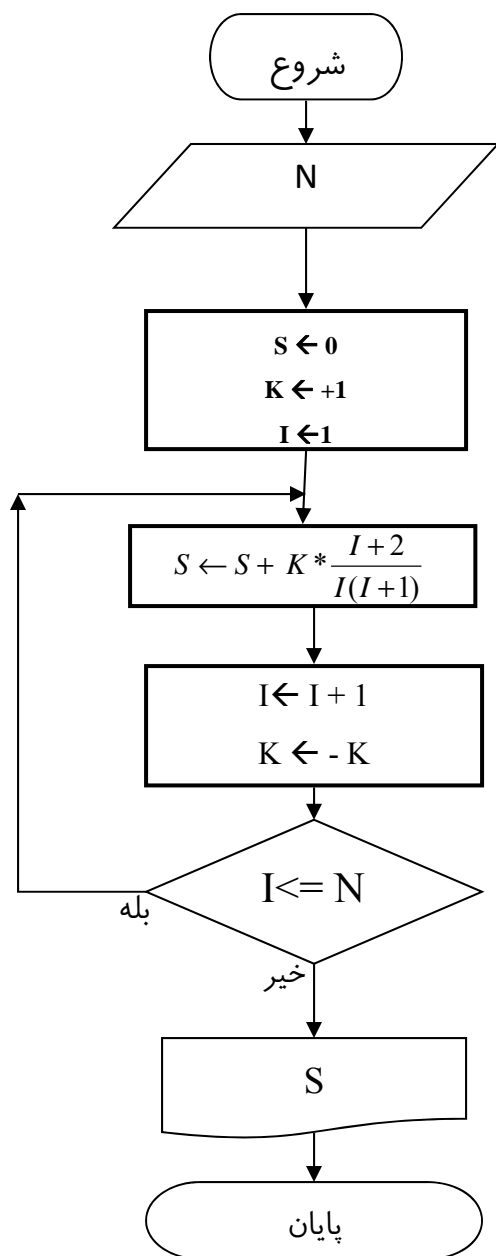
$$S = \frac{1}{1*2} + \frac{1}{2*3} + \frac{1}{3*4} + ..... + \frac{1}{N(N+1)}$$



## مثال

- عدد طبیعی N را دریافت کند و مقدار سری زیر را بدست آورد.

$$S = \frac{3}{1*2} - \frac{4}{2*3} + \frac{5}{3*4} - \frac{6}{4*5} + \dots + \frac{N+2}{N(N+1)}$$



**مثال : الگوریتم برنامه ای را بنویسید که یک عدد مثبت را خوانده فاکتوریل آن را نمایش دهد.**

- فاکتوریل یک عدد یعنی حاصل ضرب اعداد ۱ تا آن عدد
- فاکتوریل عدد صفر برابر یک می باشد.

- شمارنده A

