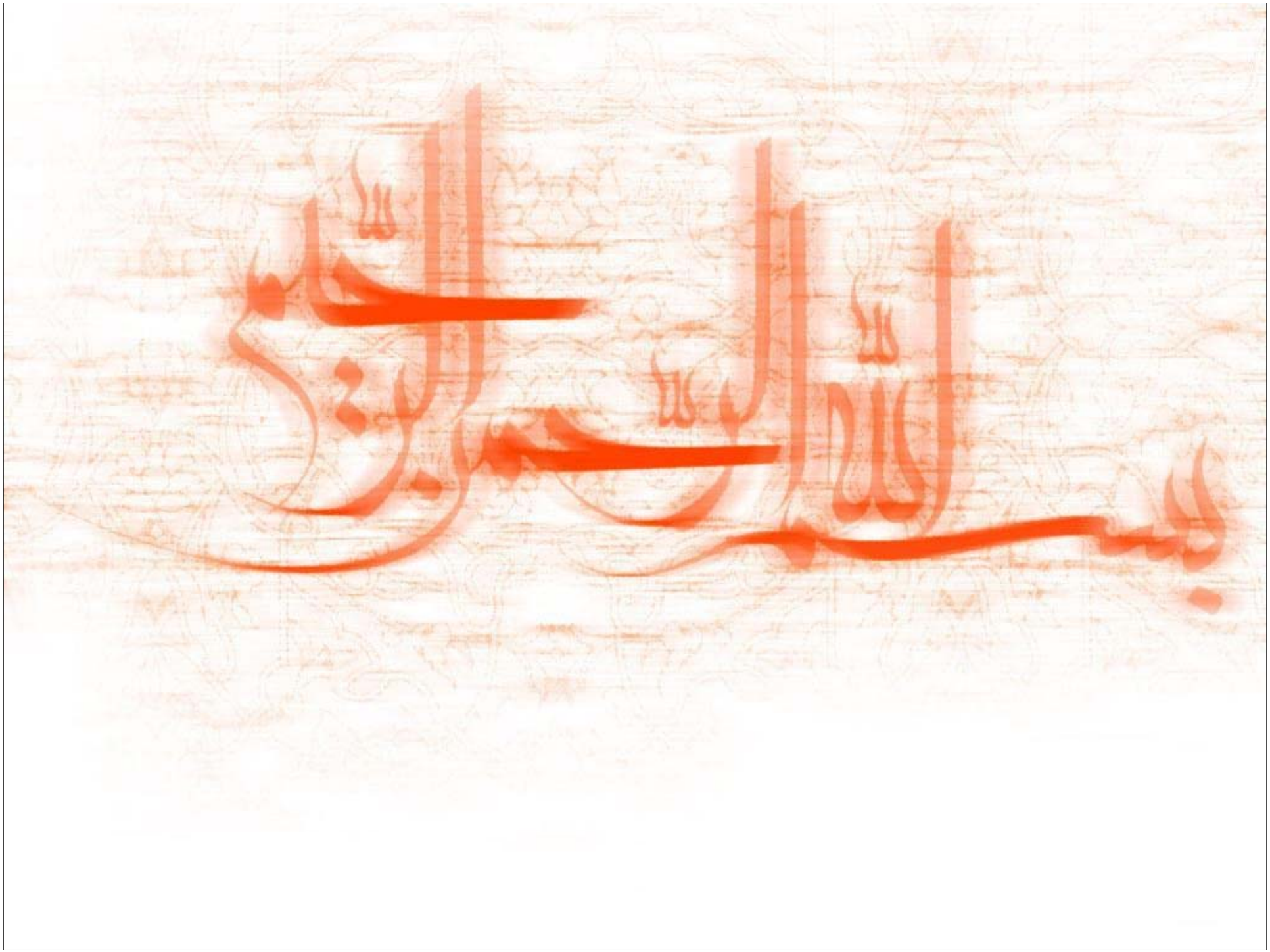
endbook.net



endbook.net

# مبانی کامپیوتر، الگوریتمها
# و
# مقدمه ای بر برنامه نویسی

رضا ترکاشون

endbook.net

# سر فصل ها

- مبانی کامپیوتر
- الگوریتم و فلوچارت

endbook.net

# منابع مربوط به مبانی و الگوریتم

- مبانی کامپیوتر و الگوریتم ها،عین اله جعفرنژاد قمی-انیس کریم پور، انتشارات علوم رایانه
- مبانی کامپیوتر و الگوریتم ها ،امیر علیخانزاده ، انتشارات پرتو نگار
- مبانی کامپیوتر، سعید سعادت
- مبانی کامپیوتر، احمد غفوری
- آشنایی با اصول و مبانی کامپیوتر و برنامه نویسی ، جواد توسلی
- و.....

endbook.net

# مبانی کامپیوتر

- مقدمه
- سازمان کامپیوتر
- طبقه بندی کامپیوترها
- برنامه و زبانهای برنامه نویسی
- انواع نرم افزارها
- اصول ریاضی کامپیوتر ها
- چگونگی نگهداری داده ها در کامپیوتر

endbook.net

# مقدمه

- **لزوم آشنایی با مبانی**

هرچند بدون داشتن دانش صحیحی از اجزاء داخلی کامپیوتر ، نحوه ارتباطات و عملکرد کامپیوتر ، باز هم میتوان در بسیاری موارد از کامپیوتر استفاده نمود ، اما داشتن یک برداشت صحیح هر چند مختصر و کلی می تواند به استفاده بهینه از کامپیوتر کمک نماید.

- **کامپیوتر؟**

*ماشینی است که :*

داده ها و دستورات را از خارج خودش دریافت کرده، درحافظه اش ذخیره می کند، دستورات را بر روی داده ها اجرا کرده ، نتایج را تحویل می دهد و این کارها را تحت کنترل خاصی انجام می دهد.

- **ویژگیهای کامپیوتر**

✓ سرعت ، دقت و صحت انجام کار

✓ مقادیر نامحدود ذخیره سازی داده ها

✓ سرعت بالای دستیابی به اطلاعات

endbook.net

# کاربردهای کامپیوتر

## ❖ طراحی

ارائه طرح -> محاسبه پارامترها-> شبیه سازی

**ارائه طرح**

از قابلیت های ذهن بشر است هرچند فعالیتهایی در جهت ابداع کامپیوترهای هوشمند وجود دارد.

**محاسبه پارامترها و شبیه سازی**

محاسبات توسط کامپیوتر و با دقت و سرعت مناسبی انجام می شود. و شبیه سازی به منظور پیش بینی عملکرد محصول در شرایط مختلف روی کامپیوتر انجام می شود و از هزینه و خسارات احتمالی جلوگیری می شود.

**مثال :** طراحی بدنه اتومبیل ، بدنه هواپیما ، سازه های مختلف ، مدارات الکترونیک و...

نمونه ای از نرم افزارهای مدلسازی در مکانیک:

**Fluent , Ansys, Thermoflow, Piping,Abaqus, OpenFoam,...**

## ❖ کنترل کامپیوتری

نمونه برداری->انجام محاسبات -> اعمال دقیق کنترل

**مثال :** کنترل موشکها ، کنترل ترافیک ، کنترل دما در نیروگاههای هسته ای و کوره ها

## ❖ ساخت به کمک کامپیوتر

کامپیوتر طرح از پیش آماده شده را دریافت کرده و از روی آن به اعمال کنترلهای لازم روی ماشین می پردازد.بدین ترتیب سرعت ، دقت و عدم خستگی کامپیوتر در پیاده سازی طرح روی ماشین آنرا جایگزین کارگر میکند.

**مثال :** دستگاههای CNC(ماشین تراش کامپیوتری)

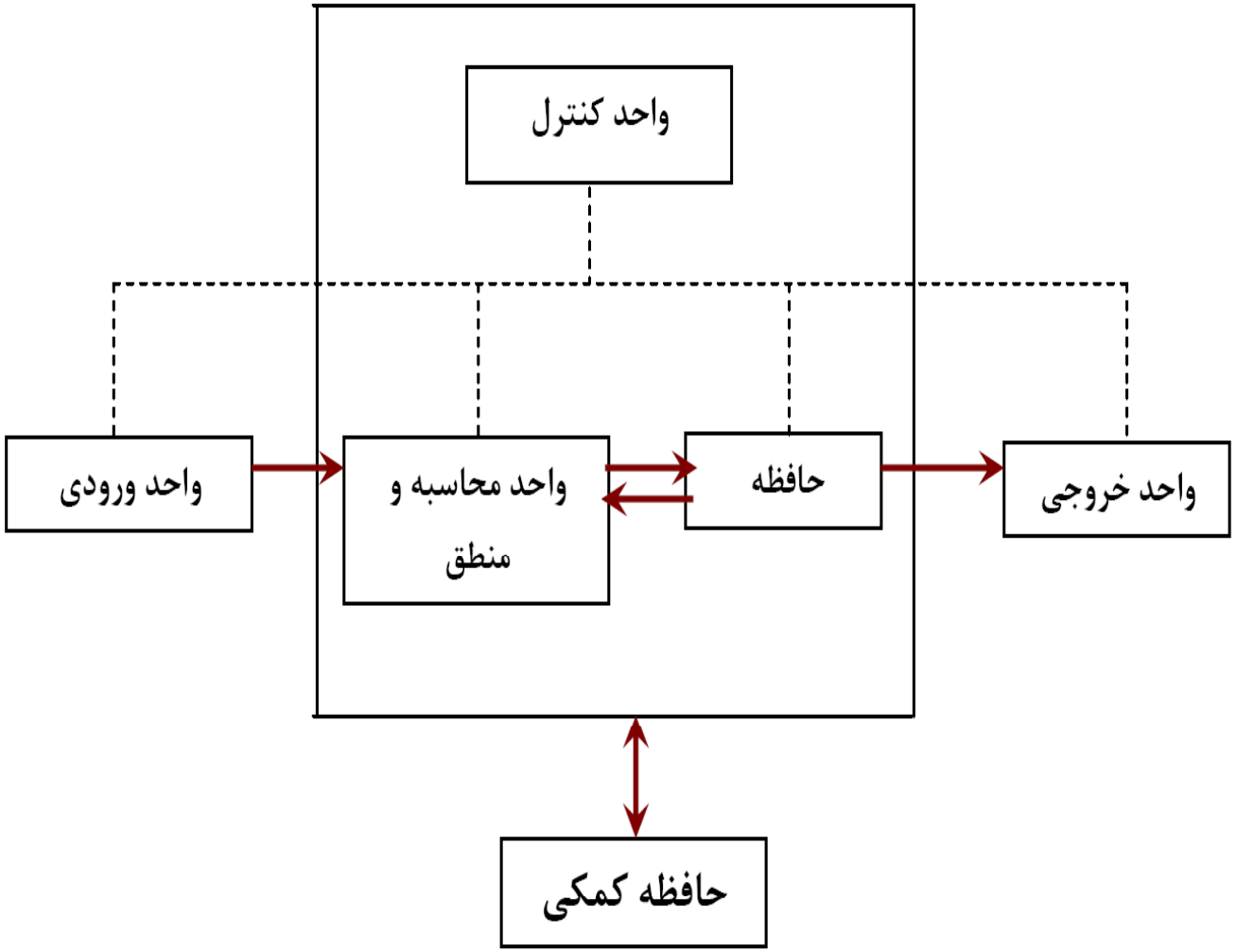
## ❖ پردازش اطلاعات به منظوراستخراج اطلاعات جدید

**مثال :** پردازش اطلاعات جوی به منظور پیش گویی وضع هوا

## ❖ بایگانی اطلاعات، آموزش و تحقیقات ، پزشکی ، متحرک سازی و.....

endbook.net

# سازمان کامپیوتر



- **واحد ورودی** : مجموعه ای از دستگاههاست که داده ها را از خارج از کامپیوتر گرفته وارد آن می نماید. یکی از وظایف عمده دستگاههای ورودی ، تغییر شکل داده ها به صورتی است که کامپیوتر آنها را درک نماید.

**واحد خروجی :** مجموعه ای از دستگاههاست که داده ها را از کامپیوتر گرفته به خارج آن منتقل می نماید. یکی از وظایف عمده دستگاههای خروجی ، تغییر شکل داده های ذخیره شده در کامپیوتر به صورتی است که انسان آنها را درک نماید.

## واحد محاسبه و منطق

اعمالی از قبیل چهار عمل اصلی ، مقایسه دو مقدار و... در این واحد انجام شده و مدار هر کدام در این قسمت وجود دارد.

## واحد کنترل

مسئولیت اجرای اتوماتیک عملیات را بر عهده دارد و علاوه بر کنترل دستگاهها، دستورالعملها را به ترتیب مورد بررسی قرار داده و دستگاههای واسطه را به قبول و اجرای دستورالعملها هدایت میکند. به عبارت دیگر واحد کنترل ، پنج بخش موجود در سازمان کامپیوتر را جهت اجرای وظایف تحت کنترل دارد.

- **حافظه** : محل نگهداری داده ها و اطلاعات در کامپیوتر است.

- **حافظه RAM** : حافظه ای خواندنی و نوشتنی است. هر نوع نرم افزار و یا برنامه کاربردی باید قبل از اجرا در حافظه اصلی کامپیوتر قرارگیرد . این حافظه در اختیار کاربر است که میتواند اطلاعاتی را به طور موقت در آن ذخیره کرده و یا از آن استفاده نماید. اطلاعات اینگونه حافظه ها با قطع منبع تغذیه از بین می رود.

- **حافظه ROM** : این حافظه حاوی دستورالعملهایی است که کارخانه سازنده کامپیوتر آنها را می نویسد. این دستورالعملها برای راه اندازی کامپیوتر لازم میباشد. اطلاعات این حافظه بر خلاف RAM با قطع جریان برق از بین نمی رود.

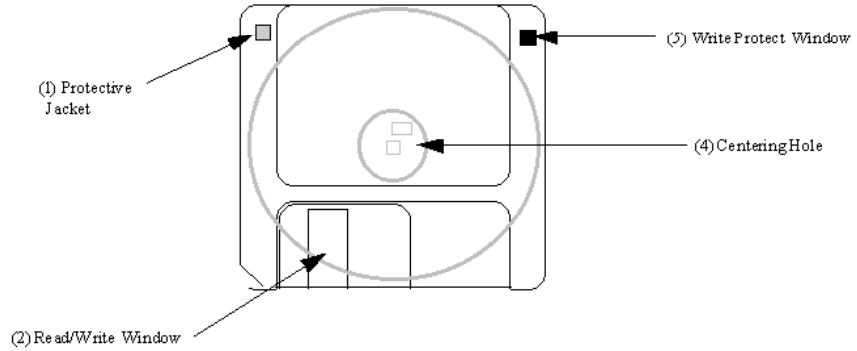
- **حافظه جانبی**

# علل استفاده از حافظه های جانبی:

۱-حافظه RAM محدود است و اطلاعات آن با قطع جریان برق از بین می رود.
۲- بعضی از اطلاعات به صورت دوره ای مورد استفاده قرار می گیرد.
۳ - حافظه جانبی برای انتقال اطلاعات از یک نقطه به نقطه دیگر به کار می رود.

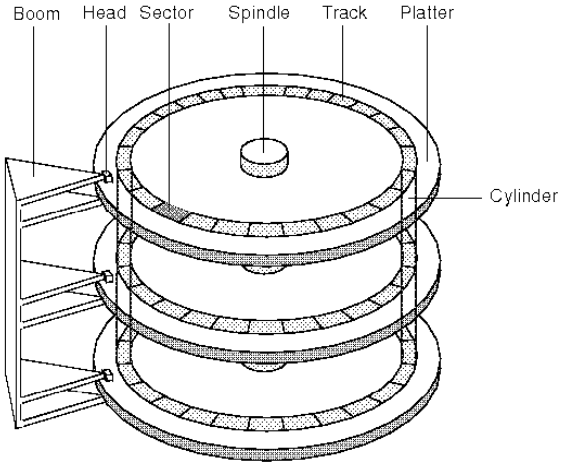
## انواع حافظه های جانبی:

۱-فلاپی دیسک (مغناطیسی)



## ۲- دیسک سخت (Hard disk)

معمولا بر روی دستگاه نصب شده و قابل حرکت نیست.جهت خواندن و نوشتن اطلاعات از دیسک گردان استفاده می شود.چند دیسک سخت در یک محفظه قرار گرفته و دیسک فشرده را تشکیل می دهند.



## ۳-دیسکهای نوری

CD,DVD…

endbook.net

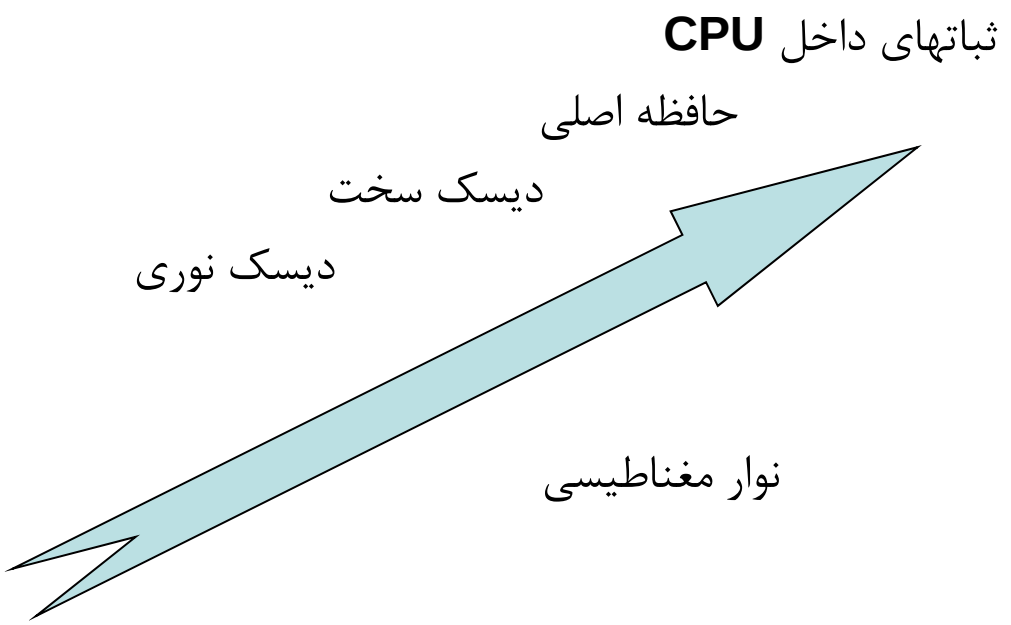
## ۴-نوارهای مغناطیسی

در گذشته به علت ارزان بودن و قابلیت حمل آسان از نوار مغناطیسی استفاده فراوان به عمل می آمد.امروزه اغلب برای تهیه پشتیبان یا آرشیو داده ها از نوار مغناطیسی استفاده می گردد.

برای دستیابی اطلاعات در نوار گاه ممکن است که لازم باشد که کل نوار باز شود . به همین دلیل سرعت دسترسی به آن پایین است.

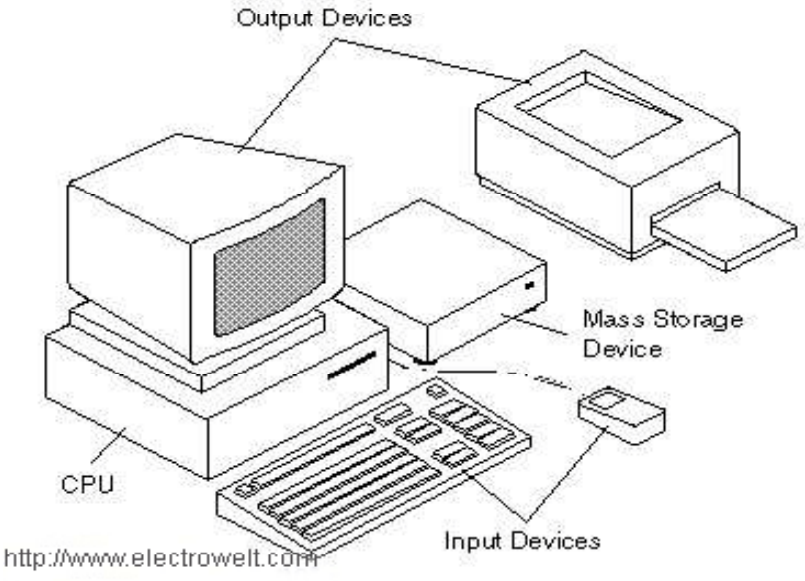
endbook.net

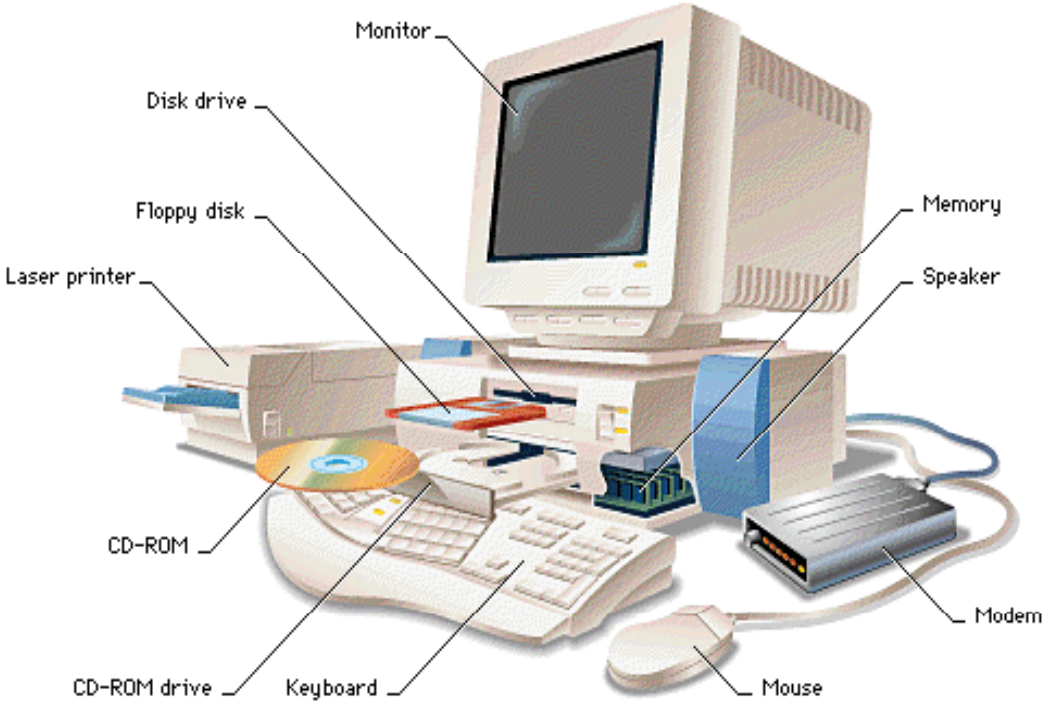
# ترتیب سرعت دسترسی به اطلاعات



endbook.net

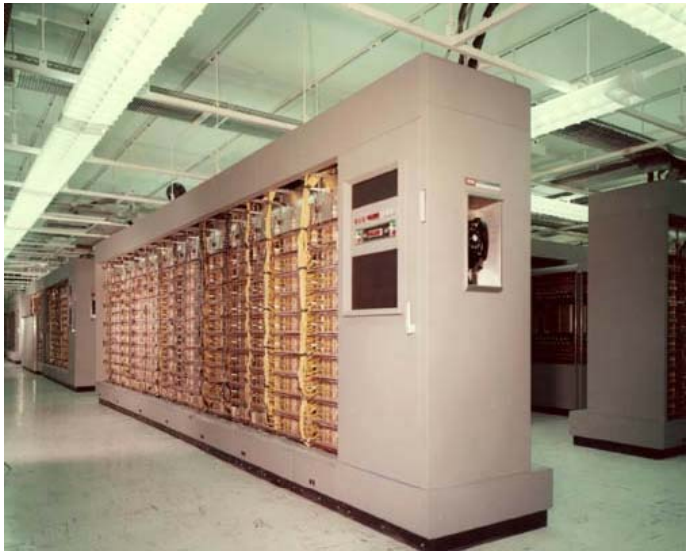
- نمونه ای از کامپیوتر با اجزای آن



endbook.net

# طبقه بندی کامپیوترها



- **ابر کامپیوتر ها(Super computer)**

**سرعت محاسباتی:**

۱۰۰تا ۱۳۰۰ میلیون دستورالعمل در ثانیه (MIPS)

**کاربرد :**

عملیات فضایی و پروژه های عظیم علمی و تحقیقاتی

- **کامپیوتر های بزرگ (Main frame)**

**کاربرد :**

مراکز آموزشی ، شرکت های بزرگ تجاری ،

دانشگاههاو.. که با حجم اطلاعات بالا سر و کار دارند.

**سرعت محاسباتی :** ۲ تا ۱۰ MIPS

endbook.net

- **کامپیوترهای کوچک (mini computer)**

**کاربرد :** مراکز تجاری ، دولتی ،
دانشگاهی با حجم اطلاعات متوسط

**سرعت :** ۰,۷ تا ۳ MIPS

- **کامپیوترهای شخصی یا ریز کامپیوترها(micro or personal computer)**

**کاربرد :** منازل ، ادارات ... که با حجم اطلاعات کمی سر و کار دارند

**سرعت :** تقریبا ۱MIPS

endbook.net

# برنامه و زبانهای برنامه نویسی

- **برنامه :**

به مجموعه ای از دستورالعملها که جهت تشریح کاری به ترتیب مناسبی آماده شده باشد ، برنامه گفته می شود.

- **زبانهای کامپیوتری:**

به دو دسته کلی زیر تقسیم می شوند:

✓ **زبانهای سطح پایین :**

این زبانها به زبانهای قابل فهم کامپیوتر که توسط سخت افزار آن قابل اجرا باشد بسیار نزدیک است. این زبانها خود به دو دسته زبان ماشین و زبان اسمبلی تقسیم می شوند:

⟵ **زبان ماشین :**

مستقیما و بدون واسطه توسط سخت افزار کامپیوتر قابل اجراست. دستورالعمل های این زبان توسط رشته ای از صفر و یک طبق ضوابط مشخص بیان می شود. برنامه نویسی به وسیله این نوع زبان بسیار دشوار است.

### ← زبان اسمبلی :

بسیار شبیه به زبان ماشین است با این تفاوت که برنامه های آن به جای اعداد دودوئی از علائم و اعداد مبنای ۱۰ استفاده می کند. برنامه ای که به زبان اسمبلی نوشته شده است باید ابتدا به زبان ماشین تبدیل شود و سپس توسط کامپیوتر اجراء شود. این عمل توسط نرم افزار اسمبلر انجام می شود. برنامه نویسی به زبان اسمبلی مشکل است اما نسبت به زبان ماشین به علت سمبلیک و نمادین بودن دستورالعملهای آن راحت تر است.

### ✓ زبانهای سطح بالا :

این زبان به زبان محاوره ای انسان نزدیک است و از زبان قابل فهم کامپیوتر دور است. برای اینکه بتوانیم برنامه هایی که به وسیله این زبانها نوشته شده است را توسط کامپیوتر اجرا نماییم احتیاج به نرم افزار خاصی جهت ترجمه زبان سطح بالا به زبان سطح پایین داریم.

مثال :BASIC,PASCAL,FORTRAN,C,...

# انواع نرم افزارها:

- **سیستم عامل (Operating system) :**

نرم افزاری است که به عنوان رابطی بین برنامه های استفاده کنندگان و منابع کامپیوتر عمل می کند. امروزه استفاده از آن آنچنان عمومیت یافته است که جدا نمودن آن از سخت افزار غیر ممکن است. از سیستم عامل می توان به عنوان مدیر منابع کامپیوتری همچون حافظه ، پردازشگر ، دستگاههای ورودی ، خروجی و... یاد کرد.

- **مترجم یا کامپایلر (Compiler):**

عمل ترجمه یا تبدیل زبان سطح بالا به زبان سطح پایین را مترجم یا کامپایلر انجام می دهد. هر زبان سطح بالا ، کامپایلر مخصوص به خود را دارد. کامپایلرها بجز عمل ترجمه ، خطاهای املائی برنامه را نیز پیدا کرده و گزارش می دهند.

- **برنامه های کاربردی آماده (Application package)**

بسیاری از برنامه ها و نرم افزارهای کامپیوتری به طور کلی طراحی شده و می توانند در موسسات و سازمانهای مختلف به کار گرفته شوند.

- **برنامه های استفاده کنندگان :**

برنامه هایی که به صورت شخصی یا برای کاربردی خاص توسط کاربران کامپیوتر طراحی و نوشته می شوند. این برنامه ها معمولا برای یک شخص یا موسسه خاص نوشته شده و جنبه عمومی ندارد.

endbook.net

# اصول ریاضی درکامپیوترها

endbook.net

# سیستم اعداد

**در سیستم اعداد در حالت کلی میتوان هر عدد را به صورت های زیر نمایش داد:**

$$\begin{cases} N = (a_{n-1}a_{n-2}.....a_1a_0 \cdot a_{-1}a_{-2}.....a_{-m})_B \\ N = a_{n-1}B^{n-1} + .... + a_0B^0 + a_{-1}B^{-1} + ..... + a_{-m}B^{-m} \\ N = \sum_{k=-m}^{n-1} a_k B^k \end{cases}$$

در روابط فوق $B$ مبنای سیستم، $a_k$ ( $0 \le a_k \le B-1$ ) ضرائب ، $n$ تعداد ارقام صحیح ، $m$ تعداد ارقام اعشاری می باشد.

endbook.net

# سیستم اعداد دهدهی یا مبنای ده

یکی از متداولترین سیستمهاست که همگان روزانه با آن سروکار دارند.

$$\begin{cases} B = 10 \\ 0 \le a_k \le B-1 \Rightarrow 0 \le a_k \le 9 \end{cases}$$

**مثال :** اعداد ۲۸۳۲۱ و ۱۲۳,۴۵ را به صورت سری و در مبنای ده نمایش دهید.

$$\begin{cases} 28321 = 2\times10^4 + 8\times10^3 + 3\times10^2 + 2\times10^1 + 1\times10^0 \\ 28321 = (28321)_{10} \end{cases}$$

.....

$$\begin{cases} 123.45 = 1\times10^2 + 2\times10^1 + 3\times10^0 + 4\times10^{-1} + 5\times10^{-2} \\ 123.45 = (123.45)_{10} \end{cases}$$

endbook.net

# سیستم اعداد دودویی

$$\begin{cases} B = 2 \\ 0 \le a_k \le B-1 \Rightarrow 0 \le a_k \le 1 \end{cases}$$

**مثال** : عدد دودوئی زیر را بسط داده و به مبنای ده ببرید.

$$\begin{cases} (10111)_2 = 1\times 2^4 + 0\times 2^3 + 1\times 2^2 + 1\times 2^1 + 1\times 2^0 = 23 \\ (10111)_2 = (23)_{10} \end{cases}$$

# سیستم اعداد هشتایی

$$\begin{cases} B = 8 \\ 0 \le a_k \le B-1 \Rightarrow 0 \le a_k \le 7 \end{cases}$$

endbook.net

# سیستم اعداد شانزده تایی

$$\begin{cases} B = 16 \\ a_k = 0,1,2,3,4,.....,9, A, B, C, D, E, F \end{cases}$$

**مثال :** عدد زیر را بسط دهید.

$$\begin{cases} (21E2)_{16} = 2\times16^3 + 1\times16^2 + 14\times16^1 + 2\times16^0 = (4578)_{10} \\ (21E2)_{16} = (4578)_{10} \end{cases}$$

endbook.net

# تبدیل مبناها

چون در سیستمهای کامپیوتری با مبناهای ۲ سروکار داریم لازم است چگونگی تبدیل مبناها به یکدیگر را مطالعه کنیم. بنابراین در این جا تبدیلات مبناها به یکدیگر را مورد مطالعه قرار می دهیم.

✓ **تبدیل اعداد دهدهی به دودویی و باالعکس با چند مثال :**

$$\begin{cases}(11001)_2 = (?)_{10} \\ (11001)_2 = 1\times 2^4 + 1\times 2^3 + 0\times 2^2 + 0\times 2^1 + 1\times 2^0 = (25)_{10}\end{cases}$$

endbook.net

از تقسیمات متوالی بر ۲ استفاده می شود تا آنجا که خارج قسمت به صفر برسد.

$$
\begin{array}{r|l}
25 & 2 \\
\hline
-24 & 12 \\
\hline
1 & 12 \\
\hline
 & 0 \\
\end{array}
\qquad
\begin{array}{r|l}
12 & 2 \\
\hline
6 & 2 \\
\hline
6 & 3 \\
\hline
0 & 2 \\
\hline
2 & 1 \\
\hline
1 & 0 \\
\hline
 & 1
\end{array}
\qquad
\begin{array}{r|l}
2 & \\
\hline
1 & 2 \\
\hline
0 & 0
\end{array}
$$

$$\Rightarrow (25)_{10} = (11001)_2$$

$$
\begin{cases}
(1110\cdot 01)_2 = (?)_{10} \\
(1110\cdot 01)_2 = 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 0\times 2^0 + 0\times 2^{-1} + 1\times 2^{-2} = (14.25)_{10}
\end{cases}
$$

endbook.net

$(14.25)_{10} = (?)_2$

قسمت صحیح :

| | | | | | |
|---|---|---|---|---|---|
| 14 | 2 | | | | |
| -14 | 7 | 2 | | | |
| 0 | 6 | 3 | 2 | | |
| | 1 | 2 | 1 | 2 | |
| | | 1 | 0 | 0 | |
| | | | 1 | | |

$\Rightarrow (14)_{10} = (1110)_2$

قسمت اعشاری :

$$\begin{cases} 0.25 \times 2 = 0.5 \\ 0.5 \times 2 = 1 \end{cases} \Rightarrow (0.25)_{10} = (0.01)_2$$

.....

و در نهایت داریم :

$(14.25)_{10} = (1110.01)_2$

برای قسمت اعشاری از ضرب های متوالی در ۲ استفاده می کنیم تا جایی که عدد اعشاری به عدد صحیح برسد. اگر این اتفاق نیفتد عمل ضرب را تا پر شدن کلمه حافظه ادامه می دهیم . ضمنا ضربهای متوالی را روی قسمت اعشاری عدد به دست آمده از مرحله قبل انجام میدهیم.

endbook.net

$$(47.7524)_{10} = (?)_2$$

برای قسمت صحیح مانند بخشهای پیشین داریم :

$$(47)_{10} = (101111)_2$$

و برای قسمت اعشاری داریم :

$$\begin{cases} 0.7524 \times 2 = 1.5048 \\ 0.5048 \times 2 = 1.0096 \\ 0.0096 \times 2 = 0.0192 \Rightarrow (0.7524)_{10} = (0.1100...)_2 \Rightarrow (47.7524)_{10} = (101111.1100)_2 \\ 0.0192 \times 2 = 0.0384 \\ ..... \end{cases}$$

endbook.net

## ✓ تبدیل اعداد هشتایی به دودویی و بالعکس

| تبدیل اعداد مبنای دو و هشت به یکدیگر | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| اعداد درمبنای ۸ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| اعداد در مبنای ۲ | 000 | 001 | 010 | 011 | 100 | 101 | 110 | 111 |

هر عدد مبنای ۸ با سه عدد در مبنای ۲ نشان داده می شود . بنابراین مبنای هشت برای نمایش خلاصه تر و خواناتر یک عدد دودوئی به کار می رود. در نهایت آنچه در حافظه نگهداری و بازیابی می شود اعداد مبنای دو هستند.

برای تبدیل اعداد مبنای دو به هشت و بالعکس از جدول فوق استفاده می نماییم.

برای تبدیل اعداد مبنای دو به هشت ، در قسمت صحیح اعداد را ۳ تا ۳ تا از سمت راست جدا کرده و اگر تعداد اعداد صحیح مضربی از ۳ نباشد به تعداد لازم صفر به انتهای سمت چپ عدد اضافه می کنیم. در قسمت اعشاری اعداد را ۳ تا ۳ تا از سمت چپ جدا کرده و اگر تعداد اعداد اعشاری مضربی از ۳ نباشد به تعداد لازم صفر به انتهای سمت راست عدد اضافه می کنیم.

به مثالهای زیر توجه شود :

$$(0110100101)_2 = (?)_8$$
$$\Rightarrow (000,110,100,101)_2 = (0645)_8$$

.....

$$(2137)_8 = (?)_2$$
$$\Rightarrow (2137)_8 = (010,001,011,111)_2$$

.....

endbook.net

$$(25.34)_8 = (?)_2$$
$$\Rightarrow (25.34)_8 = (010,101 \cdot 011,100)_2$$

**تبدیل اعداد شانزده تایی به دودویی و بالعکس :**

این تبدیلات مشابه تبدیلات مبنای ۲ به ۸ و بالعکس می باشند. با این تفاوت که هر رقم مبنای ۱۶ معادل ۴ رقم مبنای ۲ است و هر۴ رقم مبنای ۲ معادل یک رقم مبنای ۱۶ است. به این ترتیب یک عدد دودوئی را میتوان باز هم خلاصه تر نوشت.

| تبدیل اعداد مبنای دو وشانزده به یکدیگر | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| مبنای ۱۶ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| مبنای ۲ | 0000 | 0001 | 0010 | 0011 | 0100 | 0101 | 0110 | 0111 | 1000 | 1001 | 1010 | 1011 | 1100 | 1101 | 1110 | 1111 |

$$(1111101.0110)_2 = (?)_{16}$$
$$\Rightarrow (0111,1101.0110)_8 = (7D.6)_{16}$$

.....

$$(F25.03)_{16} = (?)_2$$
$$\Rightarrow (F25.03)_{16} = (1111,0010,0101.00000011)_2$$

# نمایش داخلی اطلاعات

نحوه نگهداری اطلاعات در حافظه کامپیوتر را نمایش داخلی اطلاعات گویند.

داده ها و اطلاعاتی که باید در کامپیوتر ذخیره و نگهداری شوند باید به صورت **اعداد و یا کدهای دودوئی** تبدیل شوند تا قابلیت نگهداری در حافظه اصلی و یا در حافظه های جانبی را پیدا کنند.در این بخش ، برخی از روشهای معمول در نگهداری اعداد و اطلاعات غیر عددی مورد بررسی قرار خواهند گرفت.

## واحدهای حافظه

| |
|---|
| 1 Bite : کوچکترین واحد حافظه(بیت) |
| 1 Byte =8 Bite |
| 1 Kilo-Byte(KB)= $2^{10}$ Byte |
| 1 Mega-Byte(MB)= $2^{10}$ (kB) |
| 1 Giga-Byte(GB)= $2^{10}$ MB |

## ❖ نمایش داخلی اعدا صحیح

**مثال۱) عدد دهدهی 121 را در یک بایت قرار دهید.**

$$(121)_{10} = (1111001)_2 \rightarrow$$

| 0 | 1 | 1 | 1 | 1 | 0 | 0 | 1 |
|---|---|---|---|---|---|---|---|

**عدد دودوئی حاصل هفت رقمی است. یک رقم صفر به سمت چپ آن اضافه می کنیم تا فضای یک بایت را پر نماید.**

**مثال ۲) بزرگترین عددی را که می توان در چهار بیت نمایش داد چند است ؟**

| 1 | 1 | 1 | 1 |
|---|---|---|---|

$$= 1\times 2^3 + 1\times 2^2 + 1\times 2^1 + 1\times 2^0 = 15$$

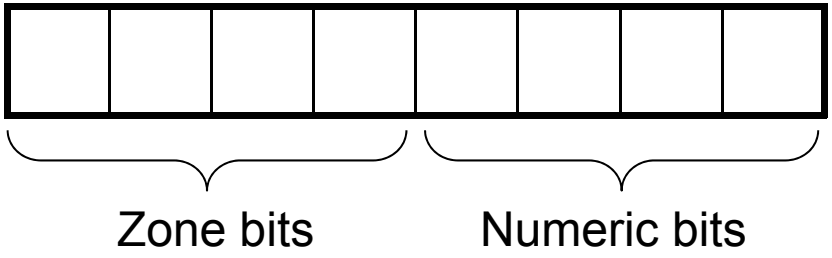
***نکته:*** **به طور کلی، بزرگترین عددی را که میتوان در n بیت نمایش داد عبارت است از:**

$$Max.value = 2^n - 1$$

## ❖ نمایش اطلاعات به صورت کد اسکی(ASCII) و کد ابسدیک (EBCDIC)

در اوایل دوران تولید کامپیوتر، برای نمایش اطلاعات کاراکتری ، کدهای مختلفی طراحی شد به طوریکه هر شرکت سازنده کامپیوتر، کد مربوط به خود را به کار می گرفت. انتخاب کد در ساخت کامپیوتر موثر بوده و بعد از ساختن کامپیوتر تغییر کد به سادگی ممکن نبود. به دلیل عدم وجود یک کد استاندارد، انتقال اطلاعات از کامپیوتری به کامپیوتر دیگر با مشکل مواجه بود. برای رفع این مشکل کمیته استاندارد تبادل اطلاعات آمریکا (ASCII) تصمیم گرفت که استانداردی را تولید نماید. این کد، کد اسکی نام گرفت. این کد در ابتدا از ۷ بیت تشکیل می شد. یعنی هر کاراکتر، ۷ بیت را اشغال می کرد. لذا با ۷ بیت ۱۲۸ کاراکتر مختلف قابل نمایش است (2^7=128). با توسعه این سیستم ، کد اسکی ۸ بیتی به وجود آمد. با این کد، ۲۵۶ کاراکتر مختلف قابل نمایش است.

کد اسکی ۸ بیتی به دو بخش کلی تقسیم می شود.



## جداول مربوط به کد اسکی در مراجع مختلف موجود است.

**مثال۳) با توجه به جدول اسکی زیر @ را با چه کدی نشان میدهید و ۰۱۰۰۰۰۱۱ مربوط به چه کاراکتری است؟**

| Zone bits | | | | | | Numeric bits |
|---|---|---|---|---|---|---|
| 0100 | 0101 | 0010 | 0011 | 0110 | 0111 | |
| @ | P | sp | 0 | / | p | 0000 |
| A | Q | ! | 1 | a | q | 0001 |
| B | R | .. | 2 | b | r | 0010 |
| C | S | # | 3 | c | s | 0011 |

@ ⟵ | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |

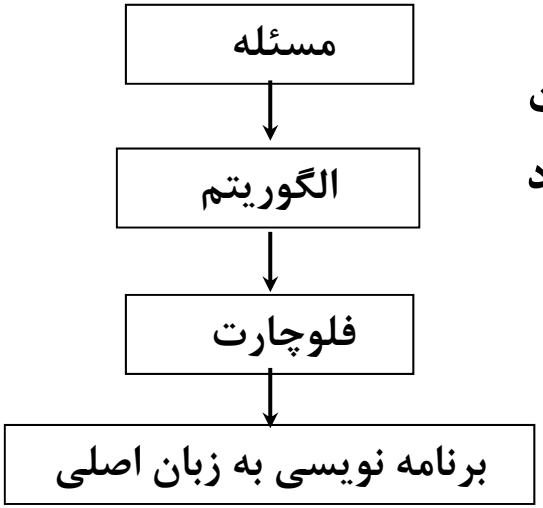
**01000011 ⟵ C**

**کد ابسدیک نسبت به کد اسکی رواج کمتری دارد و غالبا برای کامپیوترهای بزرگ IBM به کار می رود که چیزی شبیه کد اسکی است.**

endbook.net

# الگوریتم و فلوچارت

- الگوریتم در لغت به معنای روش حل مسائل می باشد و از نام دانشمند و ریاضی دان ایرانی ، ابو جعفربن محمد خوارزمی گرفته شده است.
- الگوریتم مجموعه دستورالعملهایی است که اگر دنبال گردد حاصل کار موجب حل مسئله ای خاص می شود.
- الگوریتم مجموعه دستورالعملهایی است که مراحل مختلف کاری را به زبان دقیق و با جزئیات کافی بیان نماید و در آن ترتیب مراحل و شرط خاتمه عملیات کاملا مشخص باشد.
- الگوریتم برنامه ایست که روی کامپیوتر قابل پیاده سازی بوده و تنها یک قدم به سمت نوشتن برنامه اصلی باقی است.
- برای حل یک مسئله خاص، باید آنرا به صورت الگوریتم که روش حل یک مسئله است بنویسیم ، آنگاه آنرا به فلوچارت تبدیل کرده و فلوچارت را به زبان برنامه نویسی مورد نظر خود تبدیل کنیم.



# نکات برجسته و مهم در نوشتن الگوریتم ها

## زبان دقیق

اگر از یک جمله برداشتهای متفاوتی شود و یا اینکه جمله مبهم باشد ، گوییم بیان آن جمله یا عبارت دقیق نیست.

## جزئیات کافی

عملیات ذکر شده در یک الگوریتم باید برای مجری آن شناخته شده باشد. نویسنده الگوریتم باید از چیزهایی که مجری الگوریتم(زبان برنامه نویسی مورد نظر) می داند آگاه باشد و هیچگونه فرضی را در مورد مجری الگوریتم منظور نکند.

## ترتیب مراحل

مراحل انجام دستورالعملها باید به ترتیب انجام گیرد و در غیر اینصورت دستورالعملها قابل اجرا نخواهد بود.

## خاتمه عملیات

شرط یا شروط خاتمه عملیات باید در الگوریتم ذکر شود بخصوص وقتی که عملیات تکراری هستند.

endbook.net

# مراحل ضروری در نوشتن الگوریتم

**۱- تعریف دقیقی از مسئله**

**۲-تعیین کلیه پارامترها یا عوامل موجود در مسئله**

**۳-ورودی و خروجی های مسئله**

**۴-انتخاب یک راه حل مناسب برای حل مسئله**

# روشهای بیان الگوریتم

## ▪ بیان الگوریتم با جملات فارسی

در این روش ، الگوریتمها بدون استفاده از نمادهای خاصی ذکر می شوند.

**مثال : الگوریتمی که مجموع دو عدد را محاسبه کند.**

۱-دو عدد را بگیر

۲-آنها را با هم جمع کن

۳-پایان

اشکالات این روش: الگوریتمها طولانی می شوند و گاه از دستورات الگوریتم تفسیرهای گوناگونی می شود.

## ▪ بیان الگوریتم به زبان ریاضی

نسبت به زبان فارسی دقیق تر است.

**مثال : الگوریتمی که سه مقدار عددی را از ورودی خوانده، میانگین آنها را چاپ نماید.**

۱-A,B,C را بخوان

۲-SUM←A+B+C

۳-AVE←SUM/3

۴-AVE را چاپ کن

۵-پایان

***نکته : مجری مفهوم خواندن ، جمع ، چاپ کردن و پایان را می داند.***

***مجری مفهوم میانگین را نمیداند ، بلکه راه حل محاسبه میانگین را نویسنده الگوریتم باید مشخص کند.***

***علامت ← به معنی انتساب است و مقدار سمت راست را در متغیر سمت چپ قرار می دهد.***

## ▪ بیان الگوریتم توسط شکل ها

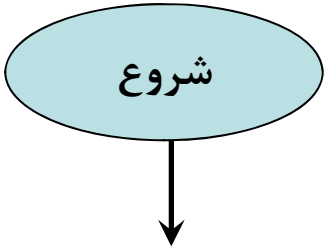
در بیان ریاضی الگوریتم ، مراحل الگوریتم به صورت دستورالعملهای متوالی تنظیم می شوند. وقتی الگوریتم طولانی باشد و یا پیچیدگی الگوریتم بیشتر شود، دنبال کردن مراحل الگوریتم دشوار می شود.

بیان الگوریتم توسط اشکال به روشهای متفاوتی امکانپذیر است که یکی از این روشها **فلوچارت (Flowchart) یا نمودار گردشی** نام دارد .در بیان الگوریتم توسط فلوچارت، از تعدادی اشکال خاص استفاده می شود که به شرح آنها می پردازیم:

***۱-شروع و پایان***

***۲- علامت اتصال***

برای اتصال بخشهای مختلف یک فلوچارت از یک فلش استفاده می شود.



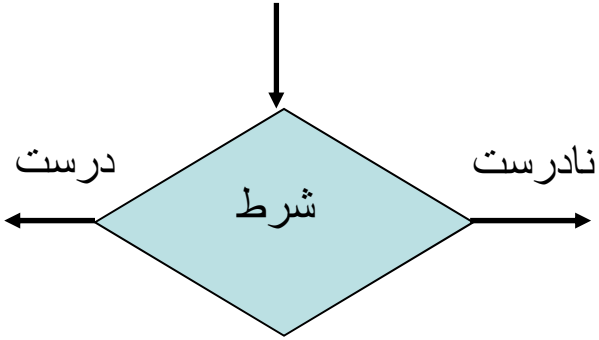
### ۳-علامت انتساب و محاسباتی

برای نمایش دستورات انتسابی و محاسباتی، از یک شکل مستطیلی استفاده می شود.

### ۴- علامات ورودی و خروجی :

برای نمایش دستورات ورودی و خروجی از یک متوازی الاضلاع استفاده می شود.

### ۵- علامتهای شرطی:



# حلقه های تکرار

**در اغلب الگوریتم ها دستوراتی وجود دارد که اجرای آنها باید چندین بار تکرار شود.**

**مثال: الگوریتمی که تعداد ۱۰۰ عدد را از ورودی خوانده و آنها را با هم جمع کند:**

- ✓ یک راه حل این است که ۱۰۰ دستور خواندن در الگوریتم تدارک دیده شود و در هر دستور یک عدد از ورودی خوانده شود.
- ✓ راه حل دیگر این است که به کامپیوتر تفهیم شود که عمل خواندن را ۱۰۰ بار تکرار کند.

**راهکار تکرار اجرای دستورات را *حلقه تکرار* گویند.**

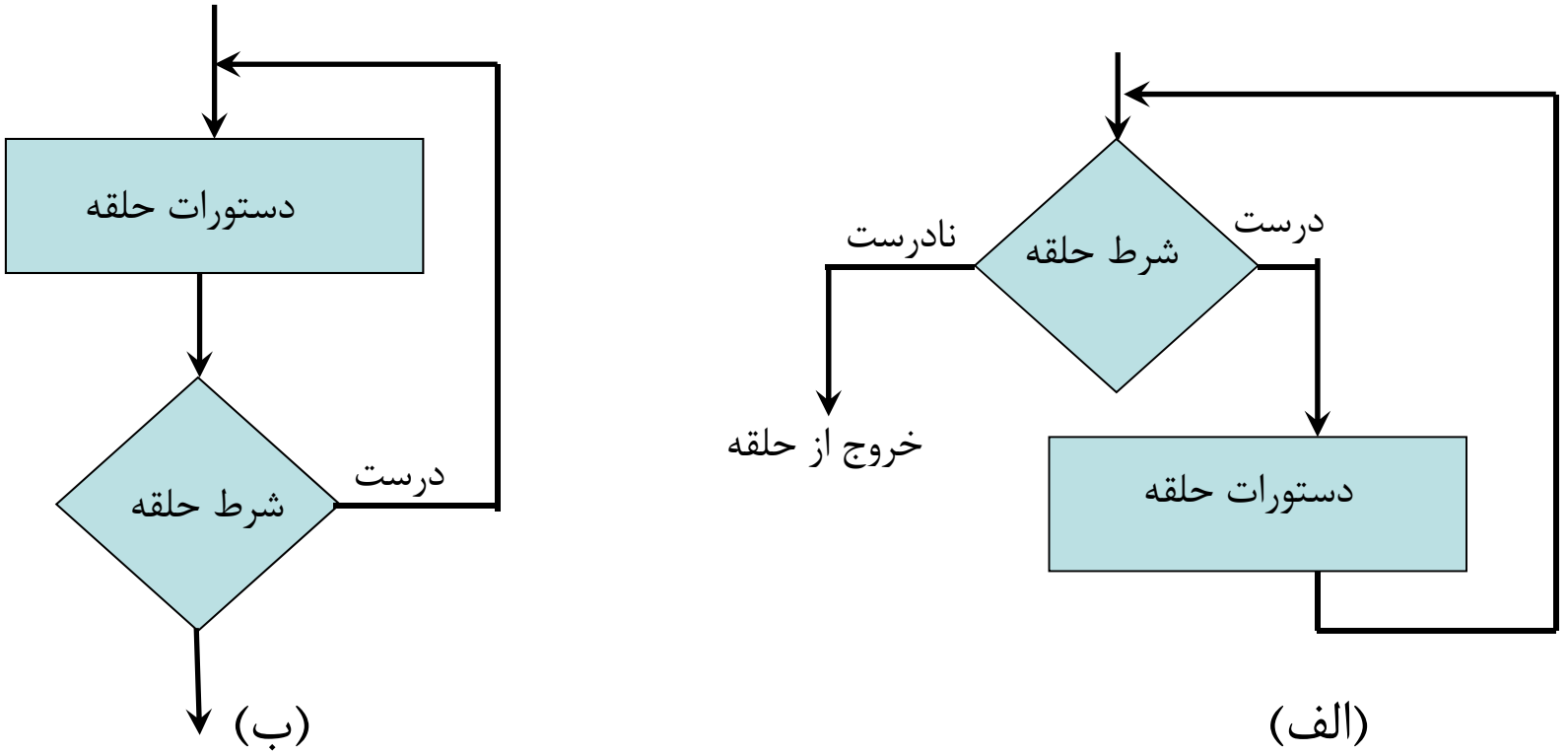
❖ **در حلقه تکرار باید سه چیز مشخص باشد:**

**-*شرط حلقه تکرار* که مشخص می کند حلقه تکرار چه زمانی خاتمه می یابد.**

**-*شمارنده حلقه تکرار* که مشخص می کند تا کنون حلقه چند بار تکرار شده است.**

**-*گام افزایش به حلقه تکرار* که مقداری است که به ازای هر بار اجرای حلقه به شمارنده حلقه اضافه می شود.**

❖ **به طورکلی حلقه تکرار بر اساس شرط حلقه بنا می شود. دستورات حلقه ممکن است به دو صورت در فلوچارت ظاهر شوند.**




- در روش الف ابتدا شرط حلقه بررسی می شود و در صورت درست بودن شرط، دستورات حلقه اجرا می شود و کنترل به ابتدای حلقه میرود و تا آنجا ادامه می یابد که شرط حلقه نادرست شود.
- در روش ب ابتدا دستورات حلقه اجرا شده و سپس شرط حلقه تست می شود و در صورت درست بودن ، کنترل الگوریتم به ابتدای حلقه می رود وگرنه خاتمه می یابد.
- ***تفاوت دو حالت :*** در روش ب دستورات حلقه حداقل یکبار تکرار می شوند.

# اثبات درستی الگوریتم

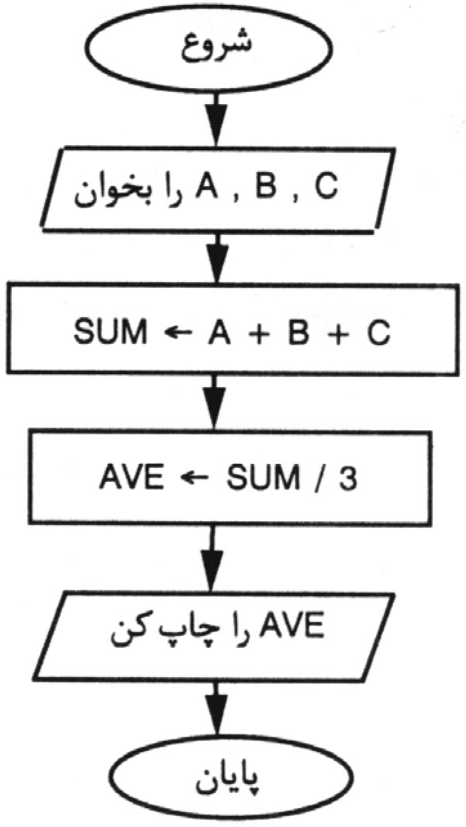
یکی از مراحل مهم در طراحی الگوریتم، اثبات درستی آن است. به عبارت دیگر، باید الگوریتم را به روش دستی و با داده های آزمایش امتحان کرد تا مشخص شود که آیا درست عمل می کند یا نه. اگر درست عمل نکرد باید اشکالات موجود را یافته و آنها را برطرف نموده و سپس الگوریتم را مجددا امتحان نمود.

endbook.net

# مثالهایی از الگوریتم و فلوچارت

**مثال۱ : الگوریتمی که سه مقدار عددی را از ورودی خوانده، میانگین آنها را چاپ نماید.**

۱-A,B,C را بخوان
۲-SUM←A+B+C
۳-AVE←SUM/3
۴-AVE را چاپ کن
۵-پایان



endbook.net

**مثال۲) الگوریتمی که شعاع دایره را از ورودی خوانده ، محیط و مساحت آنرا محاسبه کند.**

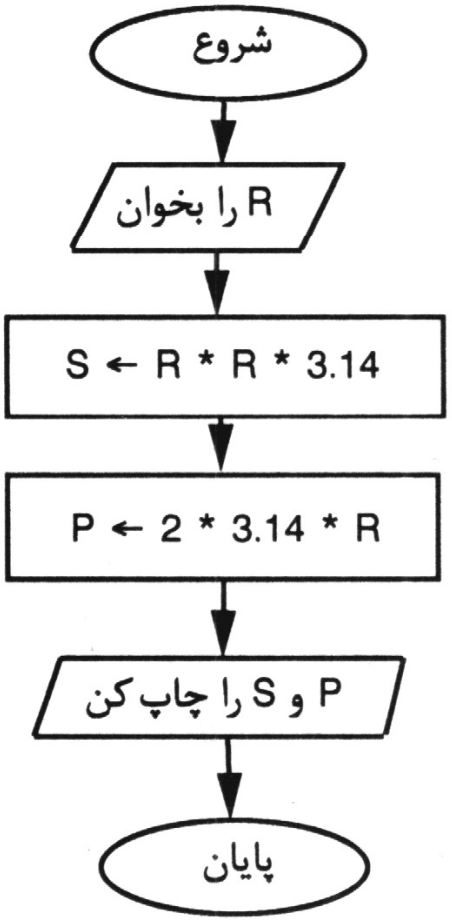
۱-R را بخوان

۲-S←R*R*3.14

۳-P←2*3.14*R

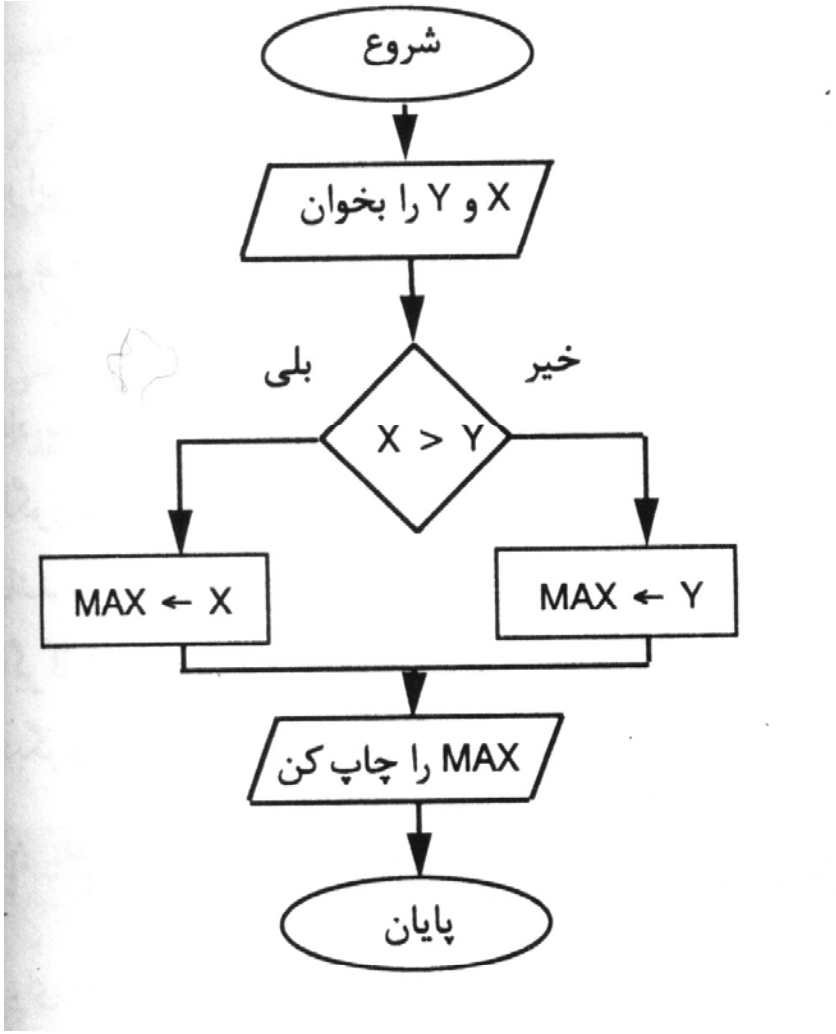
۴-S,Pراچاپ کن

۵-پایان



**مثال ۳) الگوریتمی که دو مقدار را از ورودی خوانده و مقدار بزرگتر را چاپ کند.**

۱-x,y را بخوان
۲-اگر x>y آنگاه Max←x وگرنه Max←y
۳- Max را چاپ کن
۴-پایان



endbook.net

**مثال۴) الگوریتمی که عددی مثل xرا از ورودی خوانده ، اگر زوج باشد عبارت 2x+5 و اگر فرد باشد عبارت 2x-5 را محاسبه کرده و به خروجی ببرد.**

۱-x را بخوان

۲-D←[x/2]

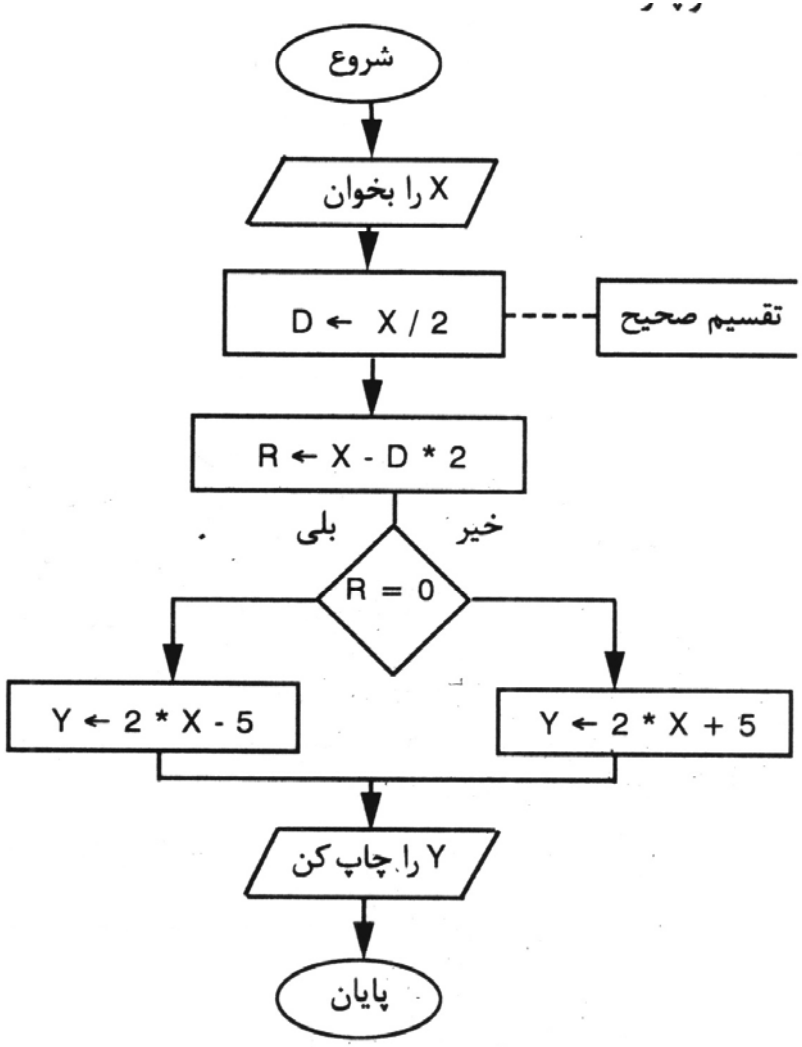
۳-R←x-D*2

۴-اگر R=0 آنگاه Y←2x+5

وگرنه Y←2x-5

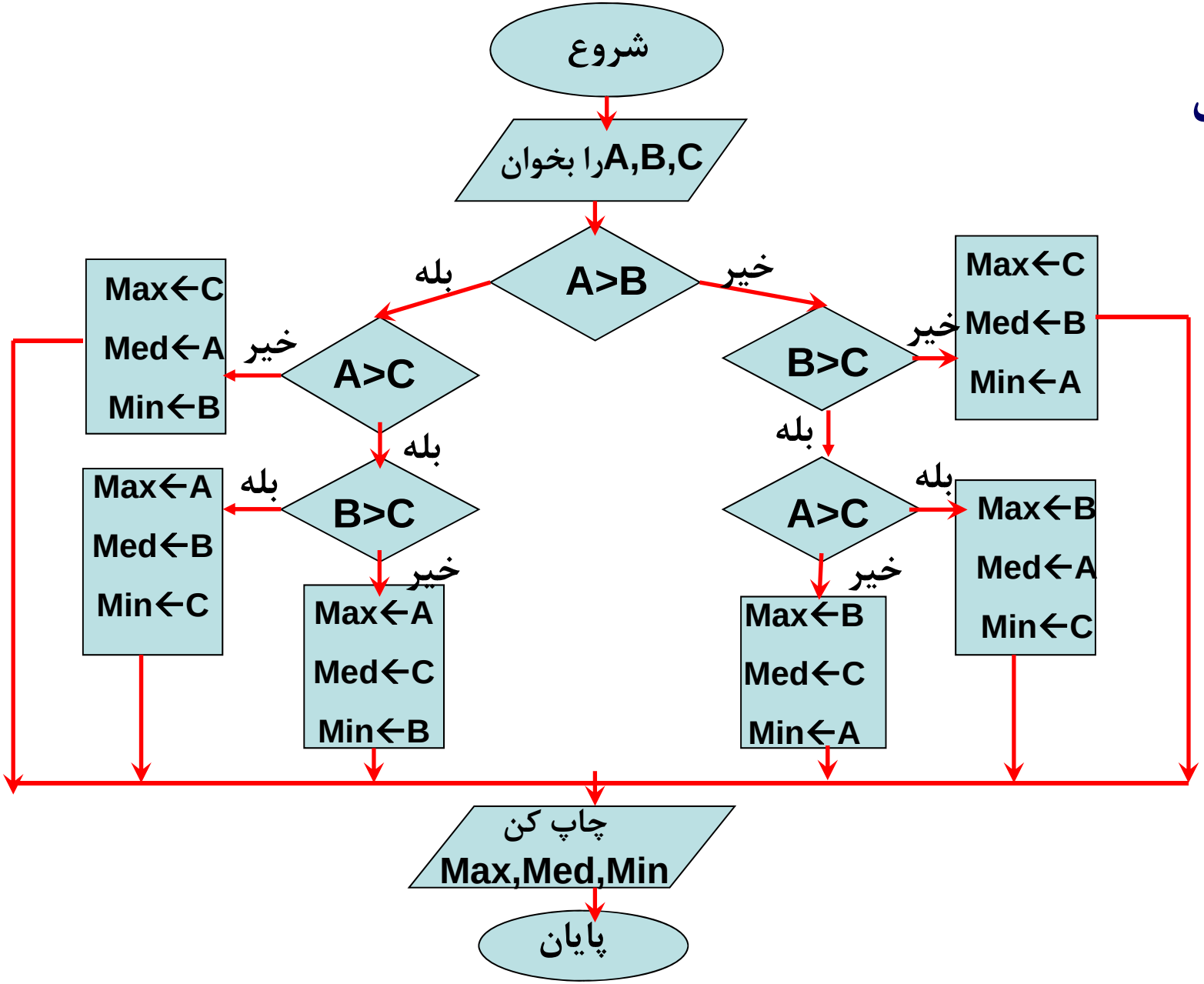
۵-Y را چاپ کن

۶-پایان

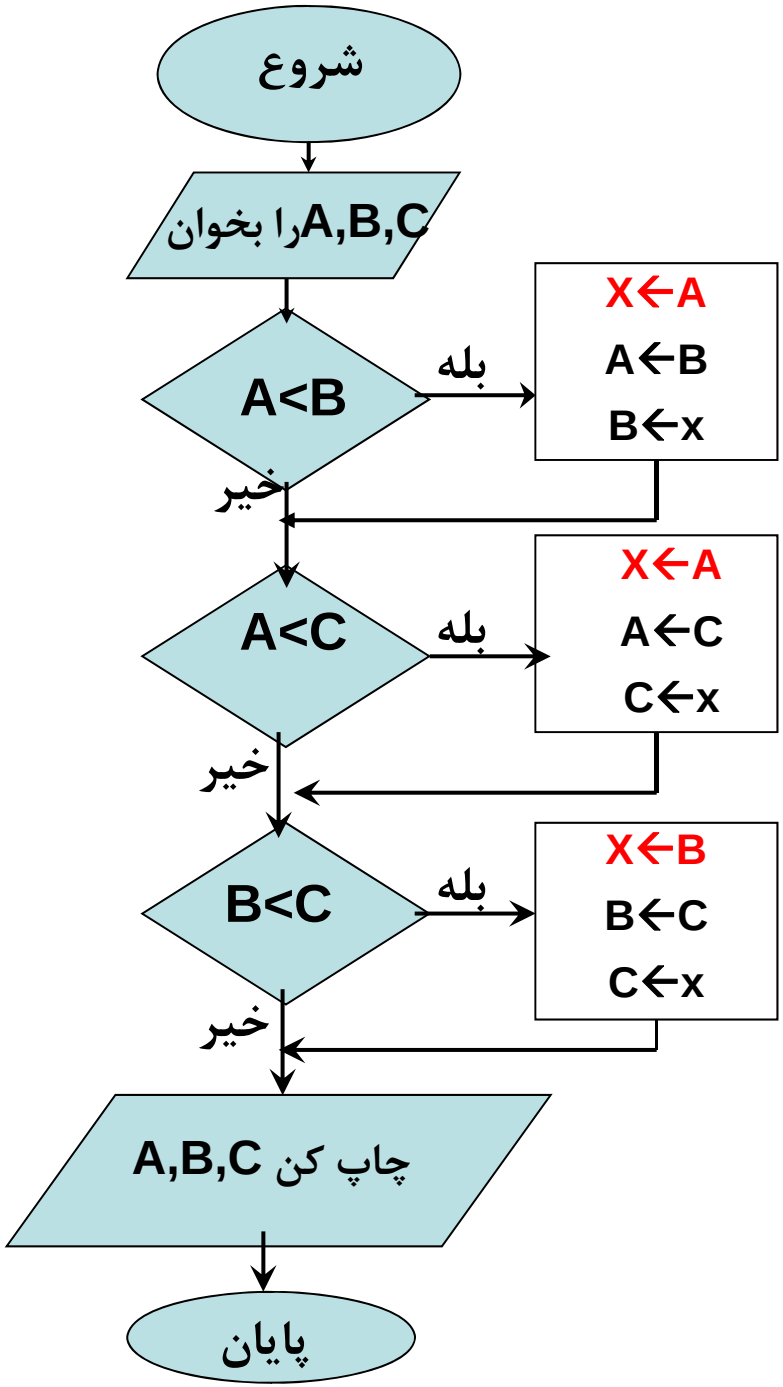


**مثال ۵)** الگوریتمی که مقدار سه عدد را خوانده و آنها را به ترتیب از بزرگ به کوچک چاپ کند.

## روش اول



# روش دوم


شروع
A,B,Cرا بخوان
A<B
بله
X←A
A←B
B←x
خیر
A<C
بله
X←A
A←C
C←x
خیر
B<C
بله
X←B
B←C
C←x
خیر
A,B,C چاپ کن
پایان

endbook.net

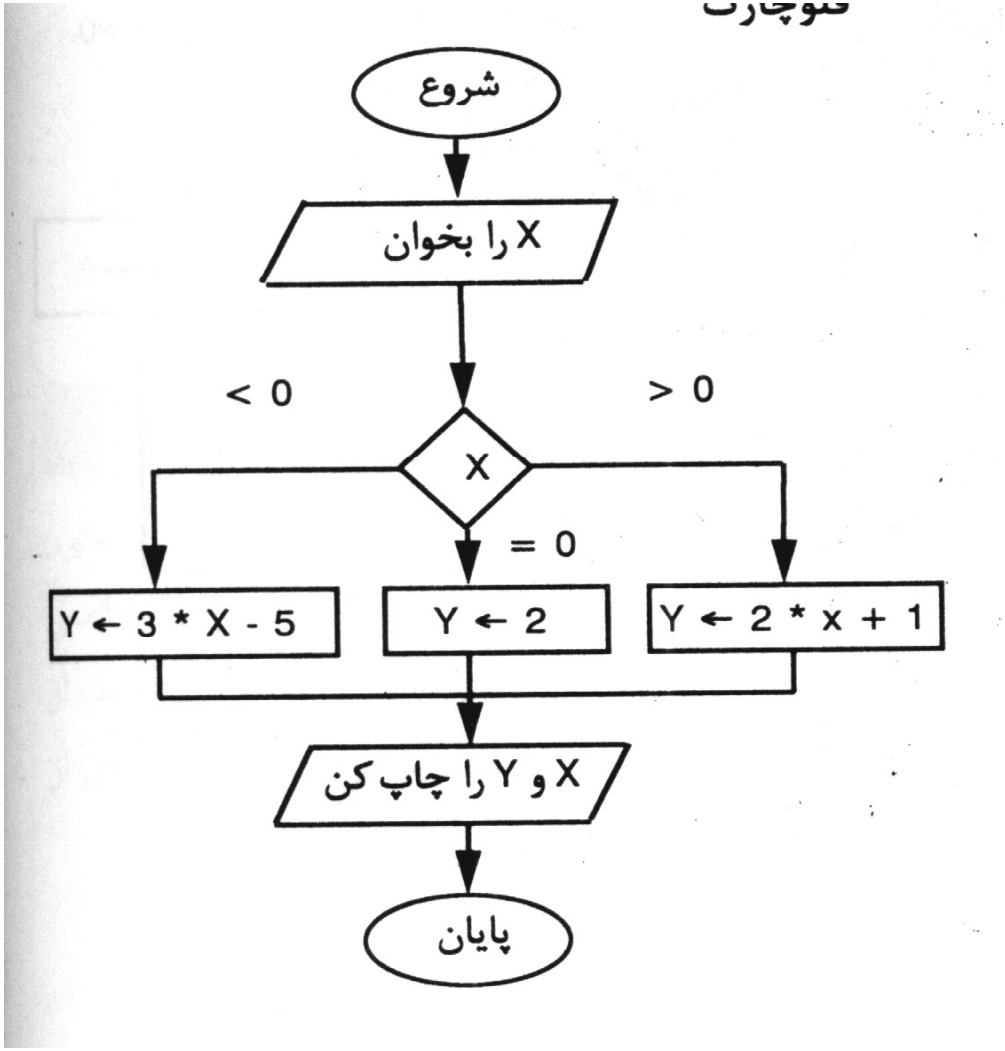
### *نکات مثال ۵:*

**در فلوچارت دوم از ۳ عملیات شرطی و در فلوچارت اول از ۵ عملیات شرطی استفاده نموده ایم. بنابراین فلوچارت دوم کارایی بالاتری دارد.**

**پس برای حل یک مسئله گاه می توان از چندین الگوریتم استفاده کرد و الگوریتمی بهتر است که کاراتر بوده، اجرای آن زمان کمتری بخواهد و یا هنرمندانه تر نوشته شده باشد.**

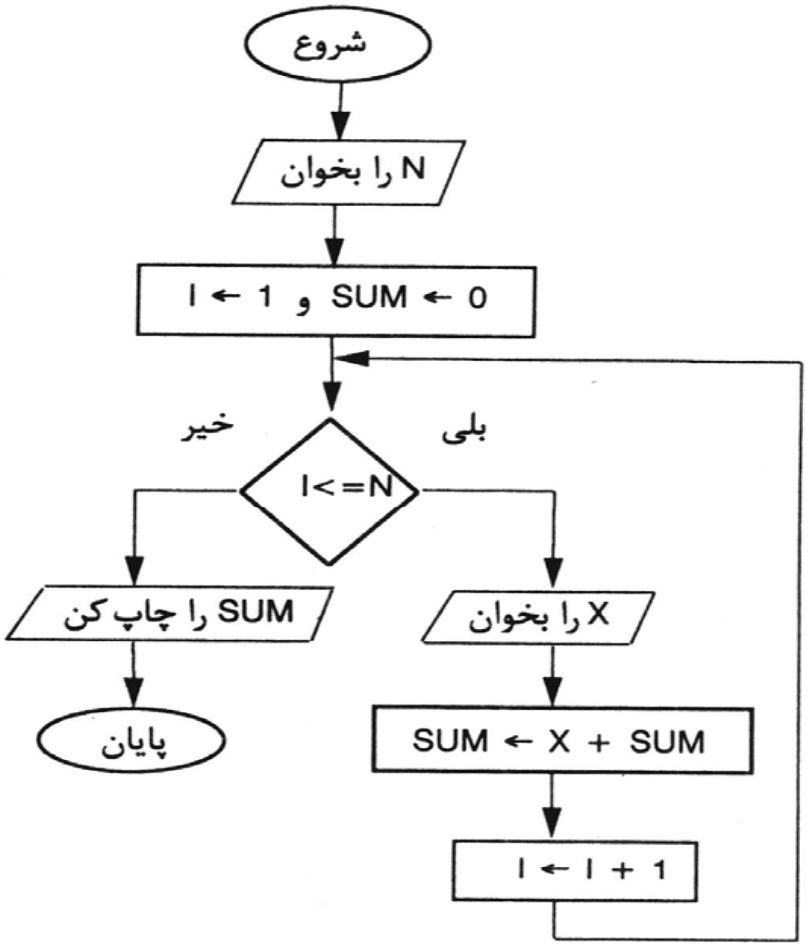
**مثال ۶: الگوریتمی که عددی مثل X را از ورودی می خواند و مقدار Y را به صورت زیر محاسبه می کند:**

**Y=3x-5 x<0**
**Y=2 x=0**
**Y=2x+1 x>0**



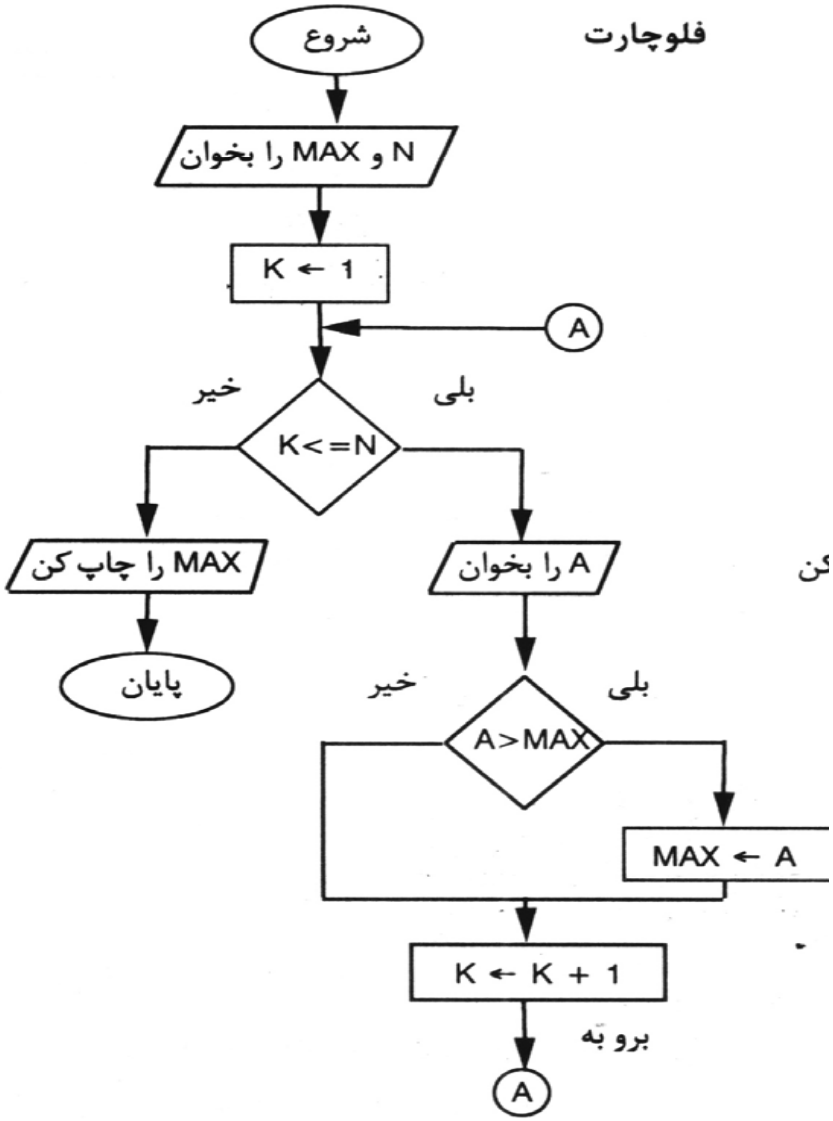
endbook.net

**مثال ۷) الگوریتمی که تعداد N عدد را از ورودی خوانده و مجموع آنها را محاسبه و چاپ کند.**



**مثال ۸) الگوریتمی که تعدادی عدد را خوانده و بزرگترین عدد را پیدا و چاپ کند.**

فلوچارت



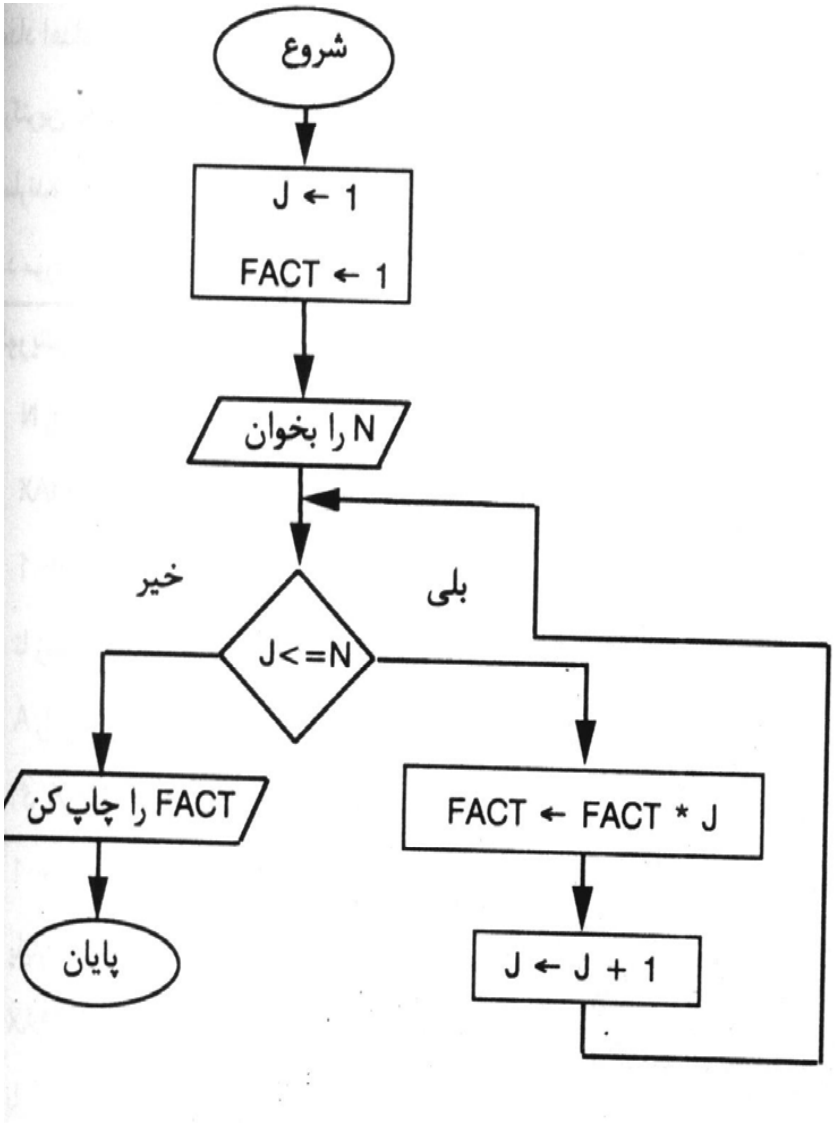

N: تعداد اعداد

Max: بزرگترین عدد

K: شمارنده

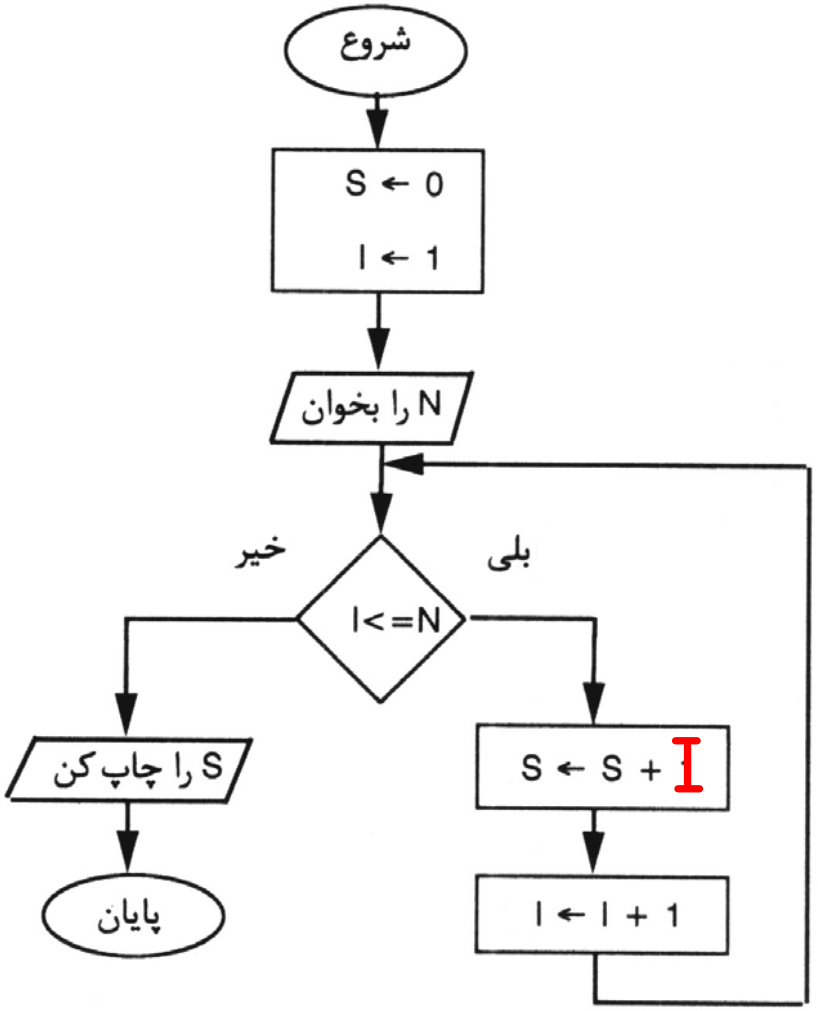
A: عدد مورد بررسی

**مثال ۹) الگوریتمی که عدد صحیح و مثبت N را از ورودی خوانده و فاکتوریل آن را محاسبه کند.**



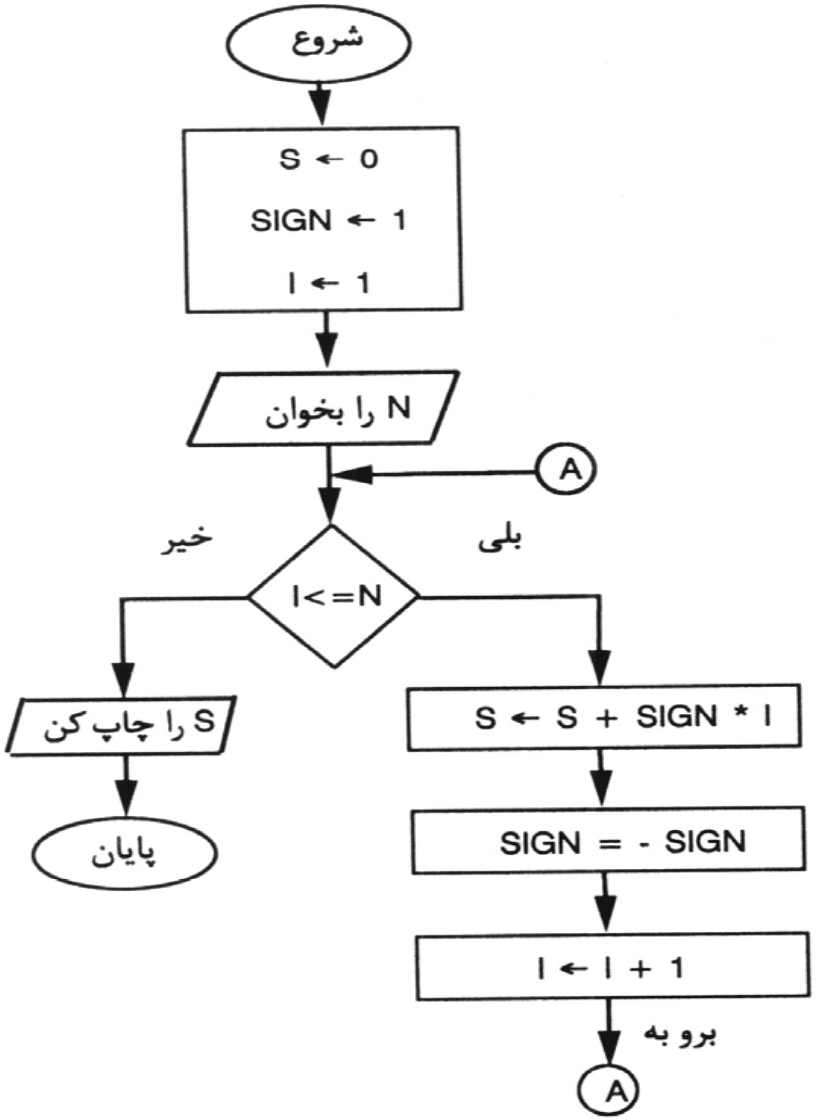
**مثال۱۰) الگوریتمی که مجموع N جمله از دنباله زیر را محاسبه کرده و به خروجی ببرد.**

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



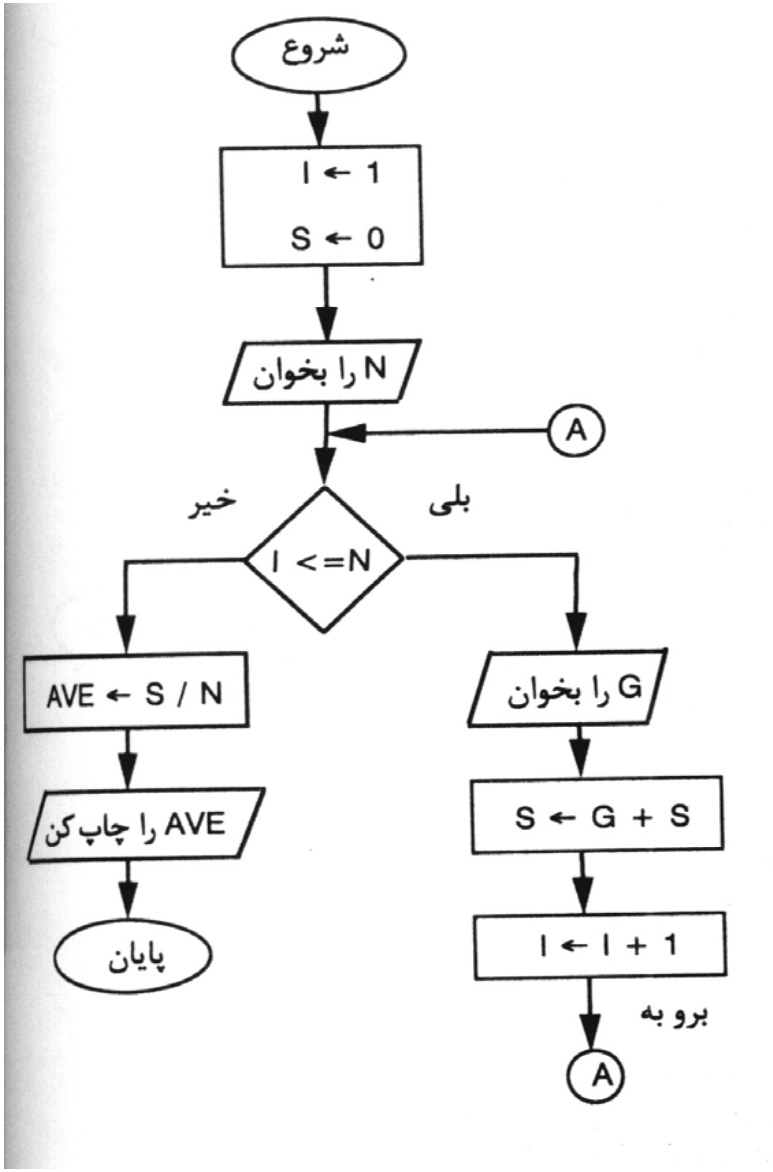
**مثال۱۱) الگوریتمی که مجموع N جمله از دنباله زیر را محاسبه کرده و به خروجی ببرد.**

**S=1-2+3-4+…+N**



endbook.net

**مثال ۱۲) الگوریتمی که نمرات دانش آموزی را از ورودی خوانده و معدل وی را محاسبه کند.**




**N: تعداد نمرات**

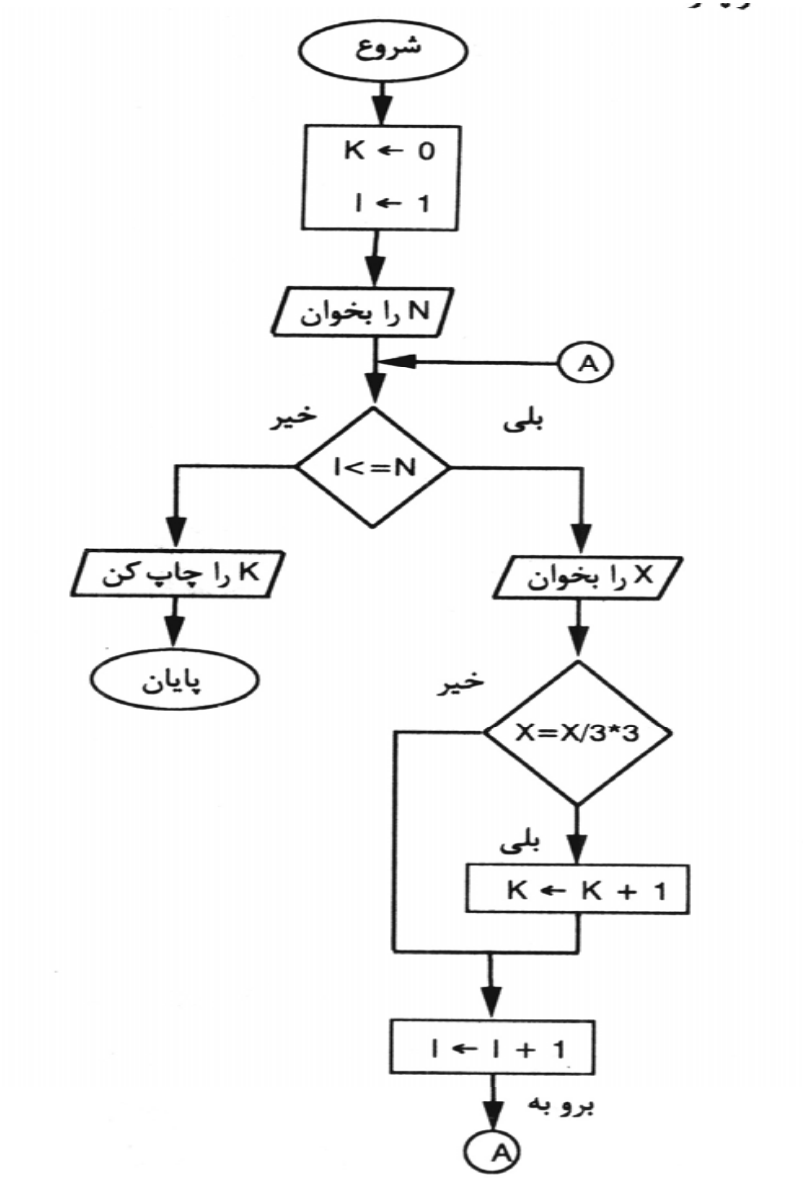
**S: جمع نمرات**

**AVE: معدل**

**I: شمارنده**

**G: نمره دانش آموز**

**مثال ۱۳) الگوریتمی که تعداد N عدد صحیح و مثبت را خوانده و تعداد اعدادی را که بر ۳ قابل قسمت هستند را مشخص نماید.**




**K:** تعداد اعدای که بر ۳ قابل تقسیم اند

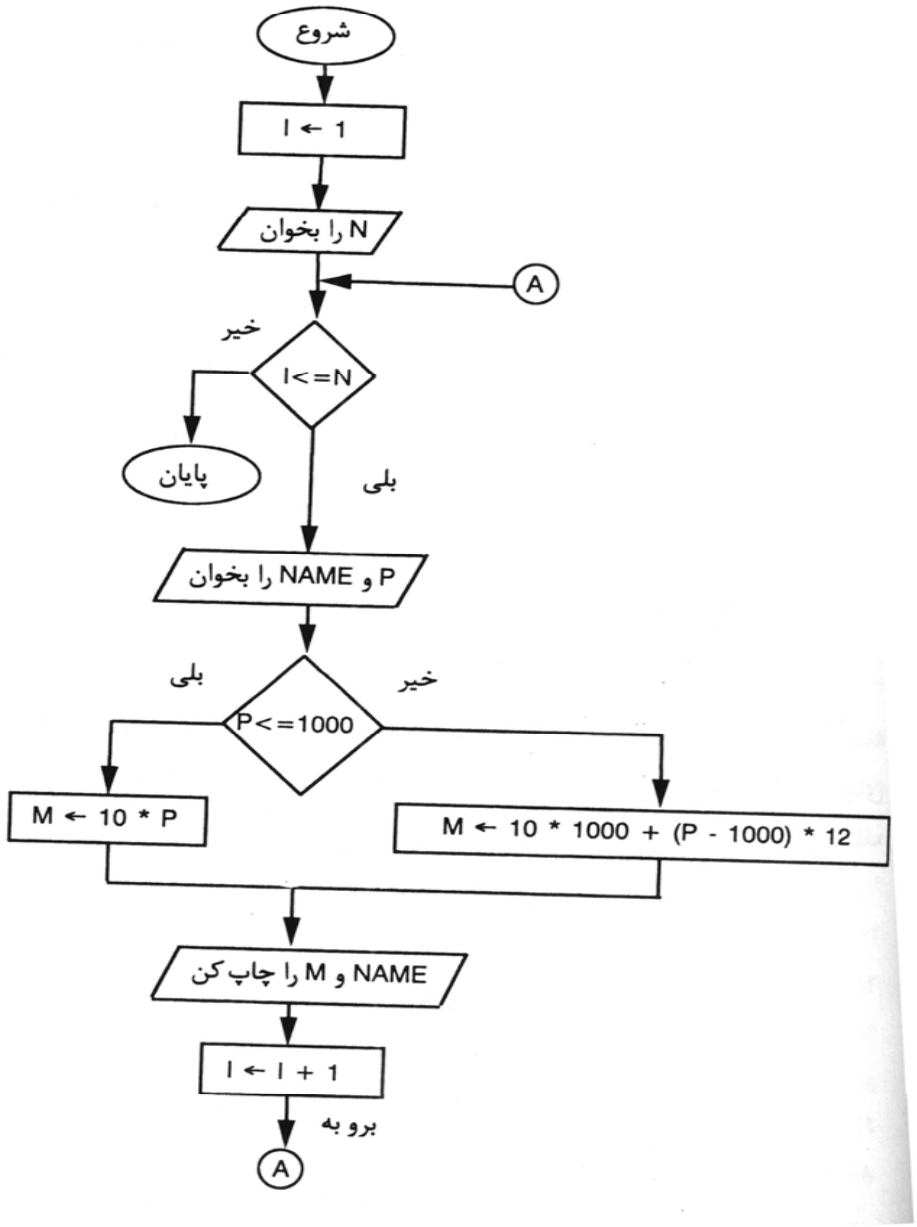
**N:** تعداد اعداد

**X:** عدد مورد بررسی

**I:** شمارنده حلقه

endbook.net

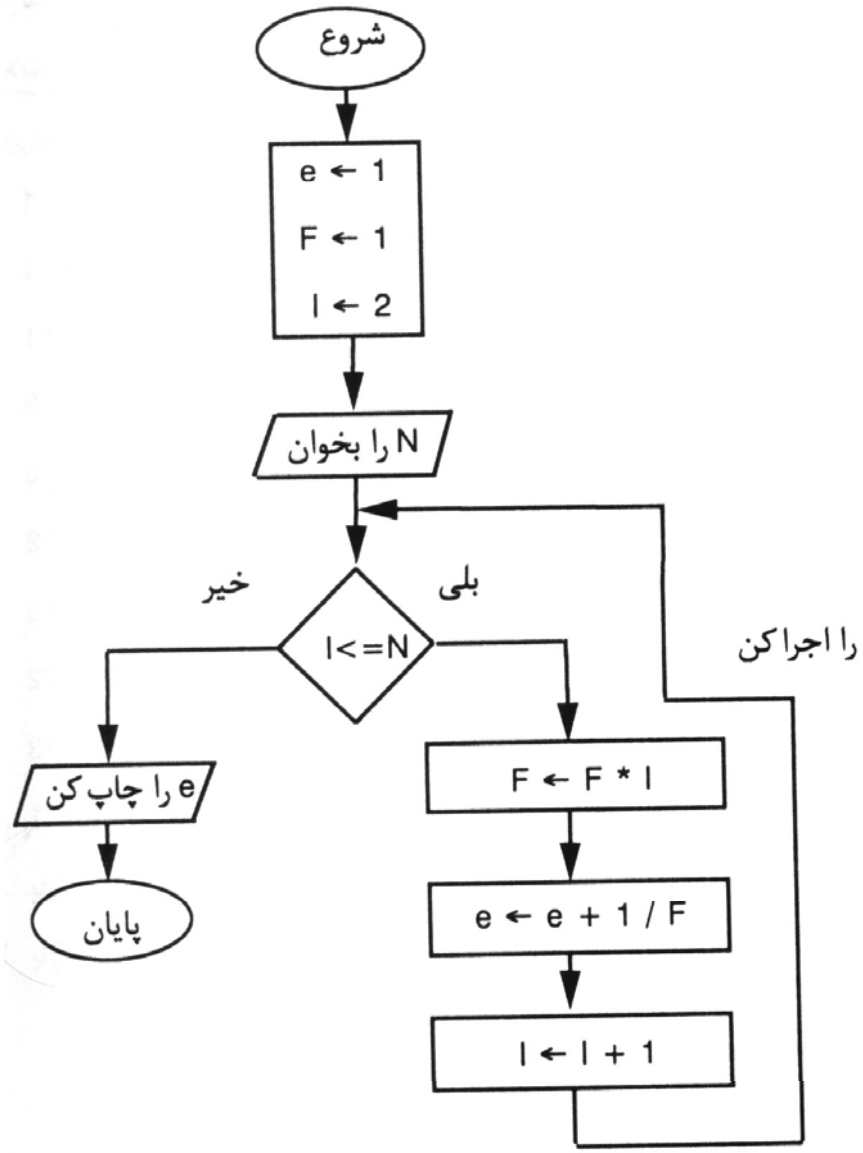
**مثال ۱۴) الگوریتمی که تعداد پالسهای تلفن مشتریان مخابرات را به همراه نام آنها خوانده و هزینه تلفن آنها را به شرح زیر محاسبه نماید:**
**از ۱ تا ۱۰۰۰ پالس، هر پالس ۱۰ ریال و**
**به ازای پالس های مازاد هر پالس ۱۲ ریال**




| | |
|---|---|
| P: | تعداد پالس |
| M: | هزینه تلفن |
| N: | تعداد مشتری |
| I: | شمارنده حلقه |

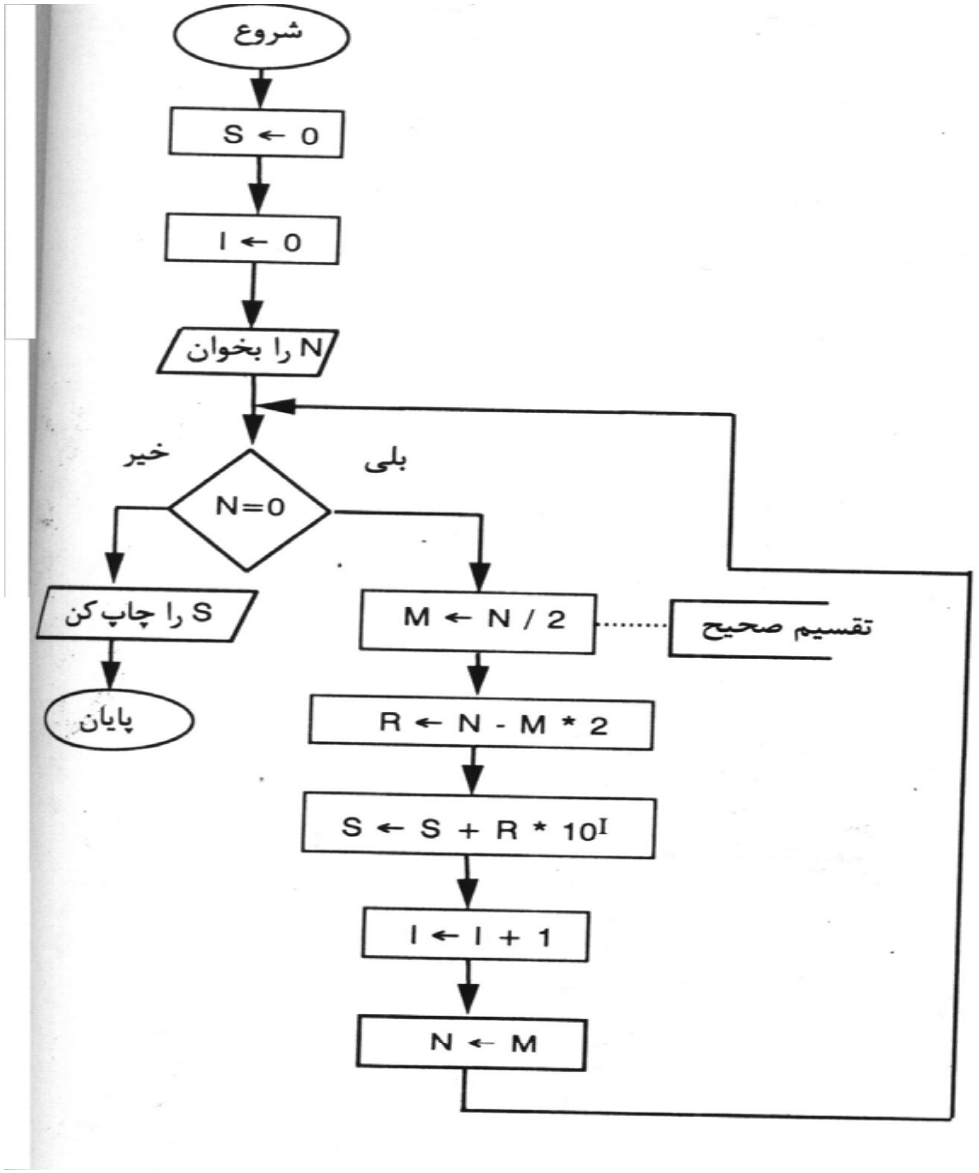
**مثال ۱۵) الگوریتمی که مقدار e را با استفاده از رابطه زیر محاسبه نماید:**

$$e = 1 + \frac{1}{2!} + \frac{1}{3!} + \ldots + \frac{1}{N!}$$



endbook.net

**مثال۱۶) الگوریتمی که یک عدد مبنای ۱۰ را خوانده و آن را به مبنای ۲ برده و نتیجه را چاپ نماید.**




**N: عدد مبنای ۱۰**

**I: شمارنده**

**M: خارج قسمت تقسیم بر ۲**

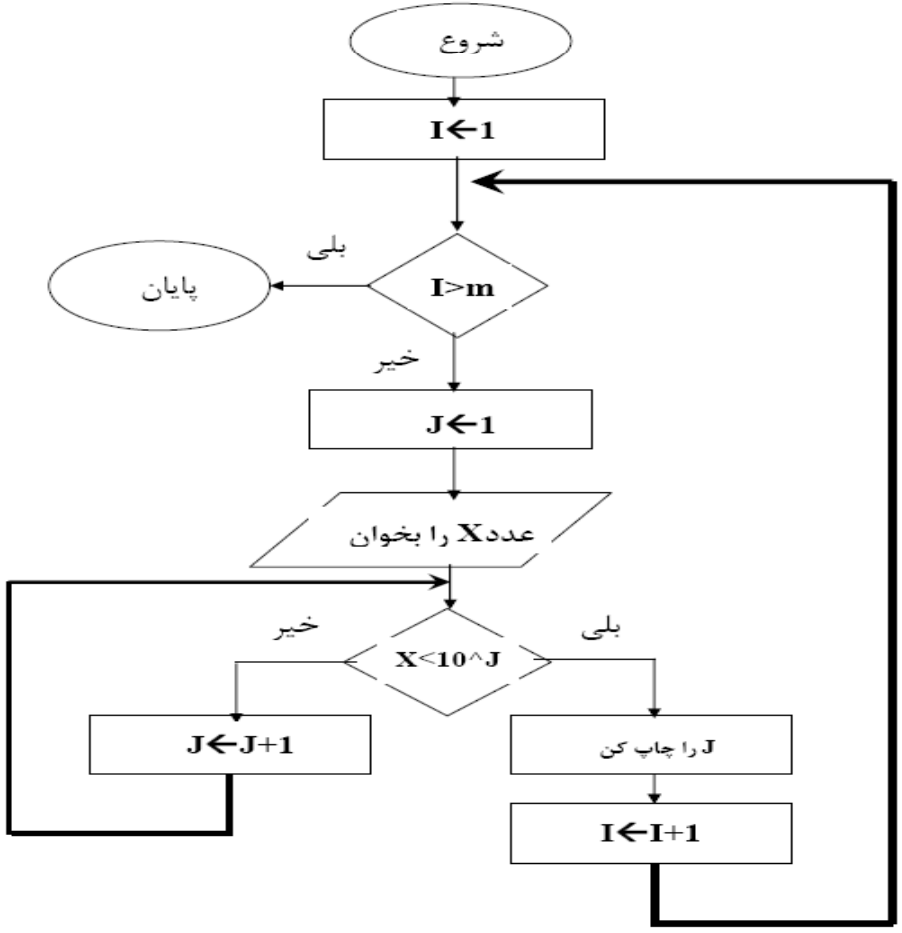
**R: باقیمانده**

**S:۲ عدد مبنای**

## حلقه های تودرتو

در الگوریتمهایی که تا کنون نوشته شد، از یک حلقه تکرار استفاده شده است. ولی گاه ممکن است در داخل یک حلقه تکرار، حلقه تکرار دیگری وجود داشته باشد. در این صورت ، این گونه حلقه های تکرار را حلقه های تودرتو گویند. راهکار اجرای حلقه های تودرتو این است که به ازای هر بار اجرای حلقه بیرونی، حلقه درونی به طور کامل اجرا شود.

**مثال ۱۷) الگوریتمی که تعداد m عدد طبیعی را گرفته و مشخص کند که هر یک چند رقمی است.**



endbook.net

# تمرینها

۱-الگوریتم و فلوچارت برنامه ای را بنویسید که n عدد را بگیرد و مجموع اعداد منفی و مثبت را به طور جداگانه چاپ نماید.

۲-الگوریتم و فلوچارت برنامه ای را بنویسید که قد n نفر را بگیرد و بگوید چند نفر قد بلند (قد>۱۷۵)، چند نفر قد متوسط(۱۷۵>قد>۱۵۰ ) و چند نفر کوتاه قد هستند (قد<۱۵۰).

۳- الگوریتم و فلوچارت برنامه ای را بنویسید که عدد طبیعی n را بگیرد و بگوید چندعدد طبیعی کوچکتر از n وجود دارد که بر ۵ یا ۷ بخش پذیر است.

۴- الگوریتم و فلوچارت برنامه ای را بنویسید که یک عدد ۴ رقمی را خوانده و معکوس آن را محاسبه و چاپ کند.

۵- الگوریتم و فلوچارت برنامه ای را بنویسید که عددی را در مبنای ۲ گرفته و به مبنای ۱۰ تبدیل نماید.

۶- الگوریتم و فلوچارت برنامه ای را بنویسید که x و k رابخواند و سری زیر را محاسبه کند.

$$\sum_{i=1}^{k} (x+1)^i$$

endbook.net

**۷-تبدیلات زیر را انجام دهید:**

۱۷۳ =( ؟ )۲

۲۳,۲۵=( ؟ )۲

۳,۷۳=( ؟ )۲

(۱۰۰۱)۲=( ؟ )۱۰

(۰۱۱۰۱۱,۰۰۱)۲=( ؟ )۱۰

(AB1F)۱۶=( ؟ )۱۰

(۷۳,۱۵)۸=( ؟ )۲

(۱۰۱۱,۱۱۱۰۱)۲=( ؟ )۸

(F2.3D)۱۶=( ؟ )۲

(۱۰۱۱۰۰,۱)۲=( ؟ )۱۶

# آرایه و کاربرد آن در الگوریتم

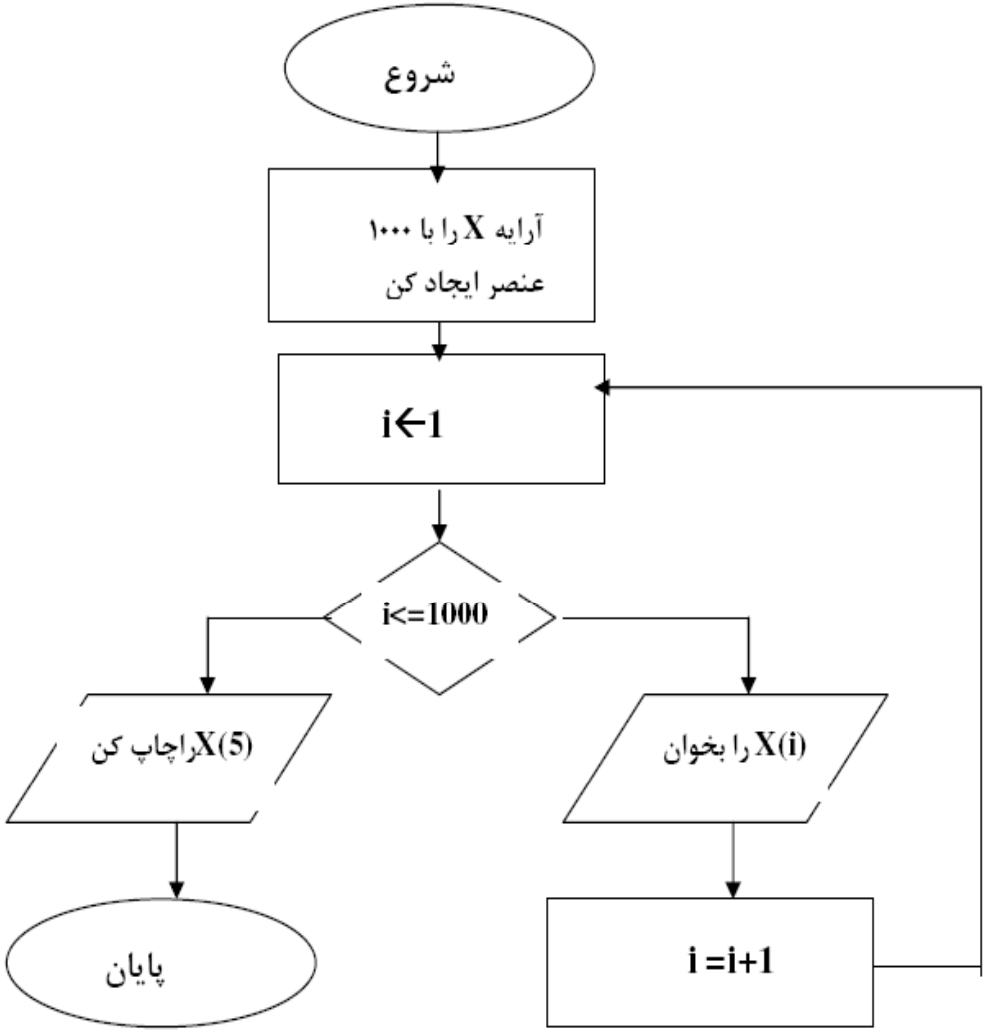
در الگوریتم ها یی که تا کنون نوشته شد، برای ذخیره هر مقدار یک متغیر تعریف گردید و آن مقدار به متغیر نسبت داده شد. متغیر نامی برای کلمه حافظه است که برای ذخیره و بازیابی اطلاعات در آن کلمه مورد استفاده قرار می گیرد. این روش استفاده از کلمات حافظه (ذخیره و بازیابی اطلاعات ) در بسیاری از کاربرد ها جوابگو نیست.

برای حل این مشکل می توان مجموعه ای از متغیرهای همنوع را در نظر گرفت که دارای یک نام بوده و برای دسترسی به هر یک از آن متغیر ها از اندیس استفاده کرد. به این مجموعه از متغیرها آرایه یا لیست گویند.

آرایه به نام X با ۵ عنصر در شکل زیر مشاهده می شود.

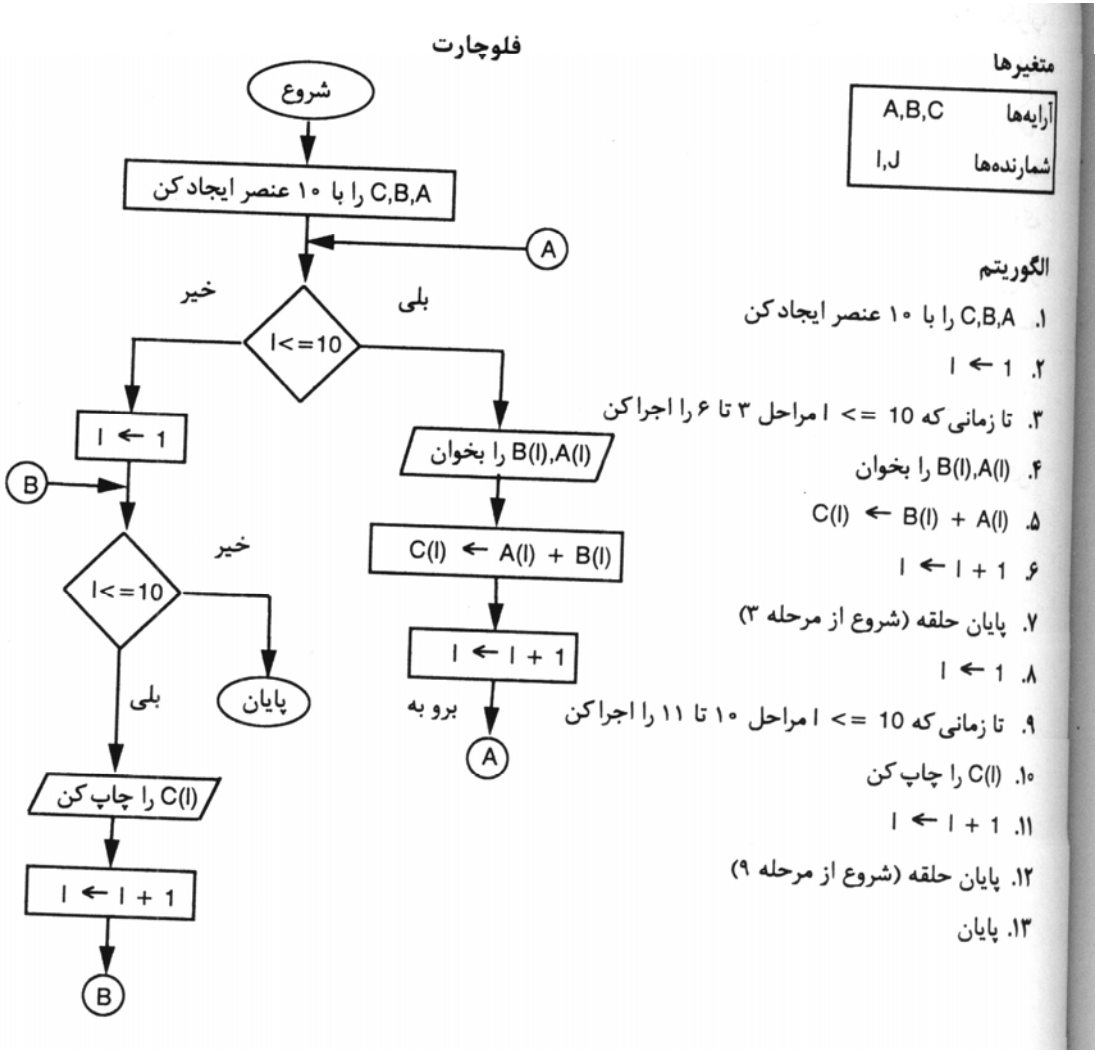
| X(1) | X(2) | X(3) | X(4) | X(5) |
|---|---|---|---|---|

**مثال۱ : الگوریتمی که تعداد ۱۰۰۰ عدد را خوانده و در آرایه ای قرار داده و پس از اتمام خواندن تمامی اعداد، عدد ۵ ام را چاپ نماید.**



endbook.net

**مثال ۲) الگوریتمی که دو آرایه ۱۰ عنصری را از ورودی خوانده ، عناصر متناظر آنها را جمع کرده و در آرایه سومی قرار دهد و به خروجی ببرد.**



endbook.net

مثال ۳) الگوریتمی که یک آرایه ۱۰ عضوی را خوانده و مشخص کند که عددی مثل X چند بار در آن تکرار شده است.

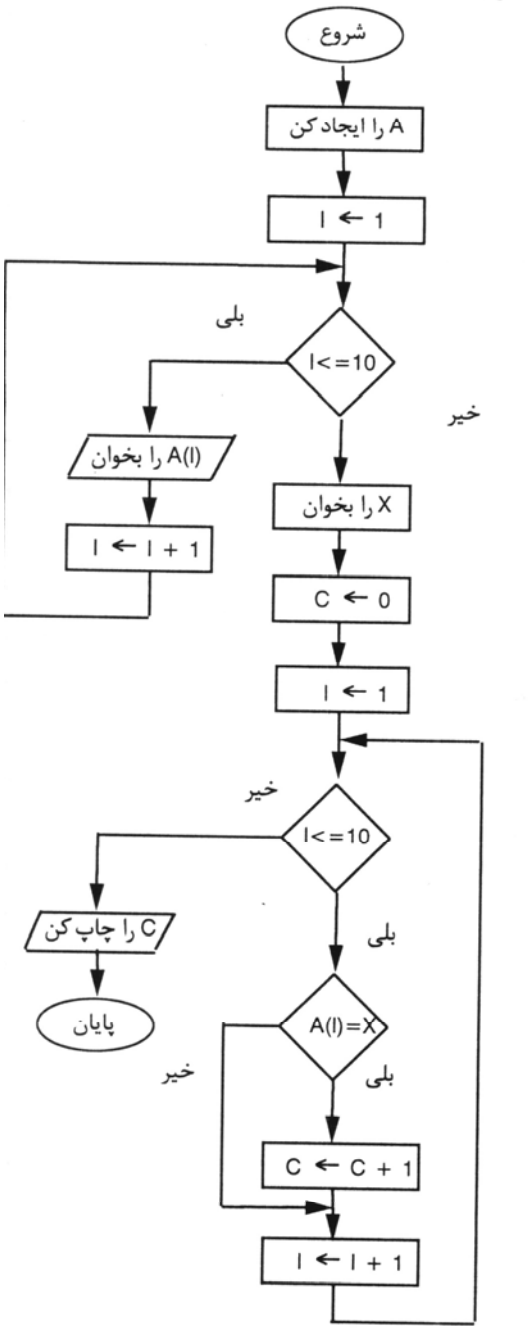
**متغیرها**

| | |
|---|---|
| آرایه | A |
| شمارنده حلقه | I |
| عدد موردنظر | X |
| شمارنده تعداد وقوع X | C |

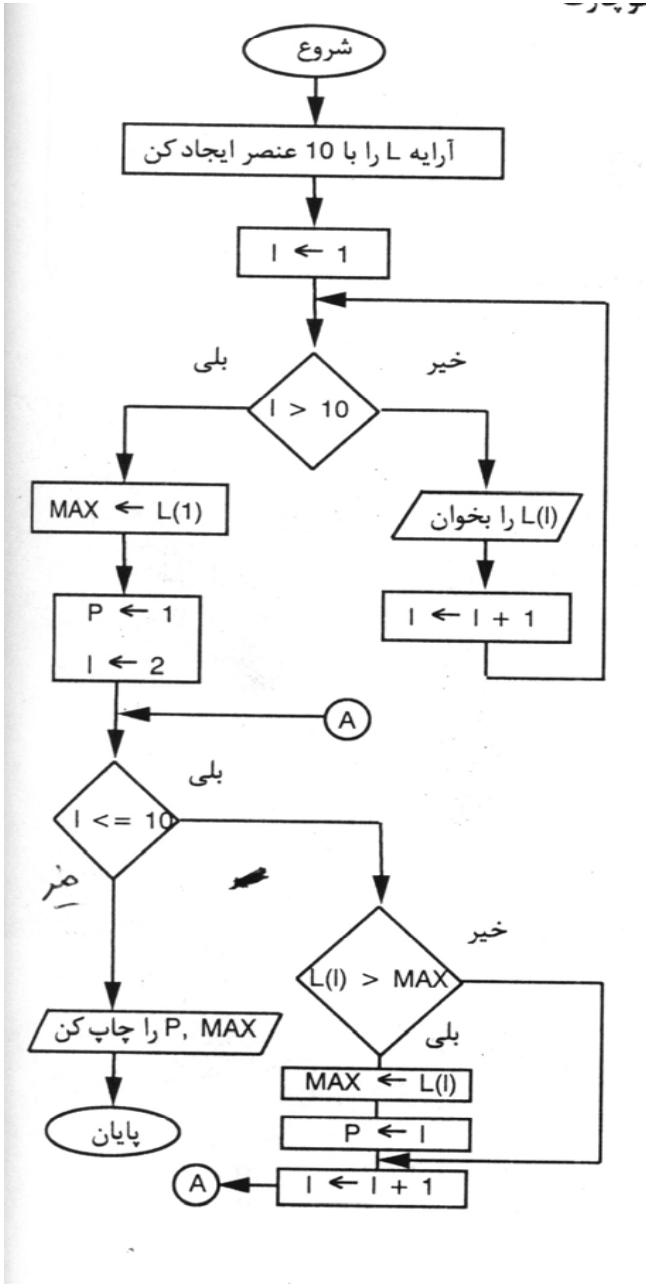
**الگوریتم**

۱. A را با ۱۰ عنصر ایجاد کن
۲. I ← 1
۳. تا زمانی که I <= 10 مراحل ۴ تا ۵ را اجرا کن
۴. A(I) را بخوان
۵. I ← I + 1
۶. پایان حلقه
۷. X را بخوان
۸. C ← 0
۹. I ← 1
۱۰. تا زمانی که I <= 10 مراحل ۱۱ تا ۱۲ را اجرا کن
۱۱. اگر A(I) = X آنگاه C ← C + 1
۱۲. I ← I + 1
۱۳. پایان حلقه
۱۴. C را چاپ کن
۱۵. پایان

**فلوچارت**



**مثال ۴: الگوریتمی که ۱۰ عدد را خوانده و بزرگترین عدد و مکان آنرا مشخص نماید.**



endbook.net

**آرایه های چند بعدی :**

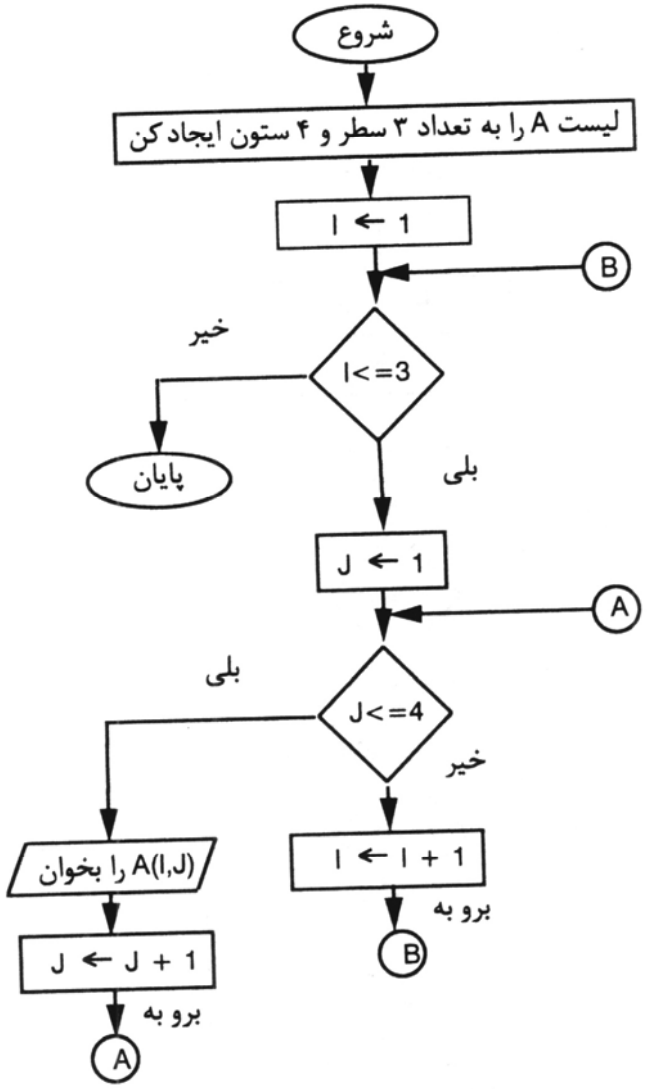
**بعضی از مسائل وجود دارند که با آرایه های یک بعدی قابل حل نبوده و باید آرایه های دیگری را به کار گرفت که بیش از یک اندیس داشته باشند. این آرایه ها را آرایه های چند بعدی گویند. اگر آرایه دارای دو اندیس باشد، دو بعدی و اگر دارای ۳ اندیس باشد ۳ بعدی نامیده می شود.**

**مثال ۱: آرایه ای دو بعدی به نام Aکه دارای ۳ سطر و ۴ ستون است و همچنین عنصر A(2,3) در شکل زیر نشان داده شده است.**

| | | | |
|---|---|---|---|
| | | | |
| | | A(2,3) | |
| | | | |

endbook.net

**مثال ۲) الگوریتمی که عناصر یک آرایه ۴*۳ را بخواند.**



endbook.net

## مثال ۳)الگوریتمی که یک ماتریس ۶*۶ را بخواند و مجموع عناصر قطر اصلی را محاسبه نماید.

**متغیرها**

| | |
|---|---|
| جدول ۶ × ۶ | M |
| شمارنده‌ها | I,J |
| مجموع عناصر قطر | S |

**الگوریتم**

۱. جدول M را ایجاد کن
۲. SUM ← 0 , I ← 1
۳. تا زمانی که I <= 6 مراحل ۴ تا ۱۰ را اجرا کن
۴. J ← 1
۵. تا زمانی که J <= 6 مراحل ۶ تا ۸ را اجرا کن
۶. M(I,J) را بخوان
۷. اگر I = J آنگاه S ← S + M(I,J)
۸. J ← J + 1
۹. پایان حلقهٔ J <= 6
۱۰. I ← I + 1
۱۱. پایان حلقهٔ I <= 6 (شروع از مرحله ۳)
۱۲. S را چاپ کن
۱۳. پایان

**فلوچارت**