

Algorithms

جعفر تنها
دانشگاه تبریز

آبان ۱۳۹۹

فهرست مطالب

- الگوریتم
- فلوچارت
- دستورات شرطی
- مثال

الگوریتم ها

هدف های کلی

- تعریف دقیق یک مسئله
- روش حل آن
- راه حل های مختلف

هدف های رفتاری

دانشجو پس از مطالعه این فصل باید بتواند:

- مسئله خود را دقیقا درک کند
- راه حل های مختلف برای آن بیابد
- الگوریتم و فلوچارت برای مسئله ارائه دهد

تعریف الگوریتم

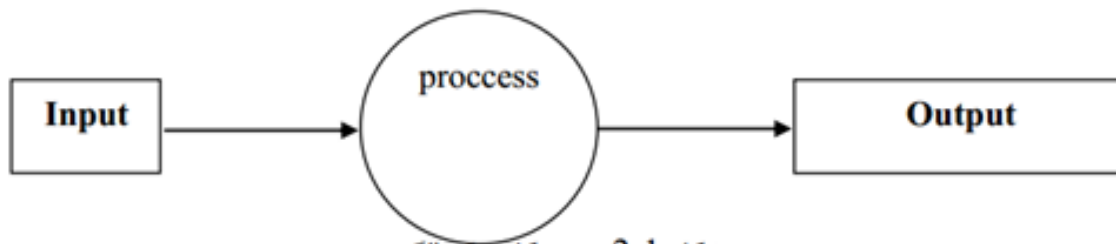
الگوریتم مجموعه‌ای از دستورات دستورات عمل‌ها، برای حل مسئله می‌باشد که شرایط زیر را باید دارا باشد:

- دقیق باشد
- جزئیات کامل حل مسئله را داشته باشد.
- پایان‌پذیر باشد.

طبق تعریف بالا برای الگوریتم، باید مراحل الگوریتم بدقت بررسی شده و تمام جزئیات رعایت شود و همچنین باید ترتیب مراحل اجرا در الگوریتم مشخص باشد و در نهایت باید الگوریتم پایان‌پذیر باشد و بعد از طی مراحل و ارائه راه‌حل به پایان برسد. الگوریتم‌ها، توسط زبانهای برنامه‌نویسی پیاده‌سازی می‌شوند. و هر الگوریتم توسط یک برنامه (program) ارائه می‌شود (با هر زبان برنامه‌نویسی).

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

مثال

ورودیها

a, b

انجام محاسبات

جمع دو عدد

خروجیها

مجموع دو عدد

0- شروع

1- a, b را بخوان.

2- مجموع a, b را محاسبه و در sum قرار بده.

3- sum را در خروجی چاپ کن

4- پایان

مثال

<u>ورودیها</u>	<u>انجام محاسبات</u>	<u>خروجیها</u>
a	محاسبه مجموع	چاپ مجموع
b	محاسبه میانگین	چاپ میانگین
c		

الگوریتم بصورت زیر می باشد:

0- شروع

1- سه عدد از ورودی بخوان

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

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

4- sum , ave را در خروجی چاپ کن.

5- پایان.

فلوچارت

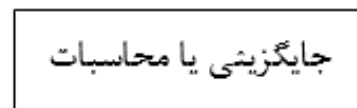
- علامتهای شروع و پایان: که معمولاً از یک بیضی استفاده می‌کنند:



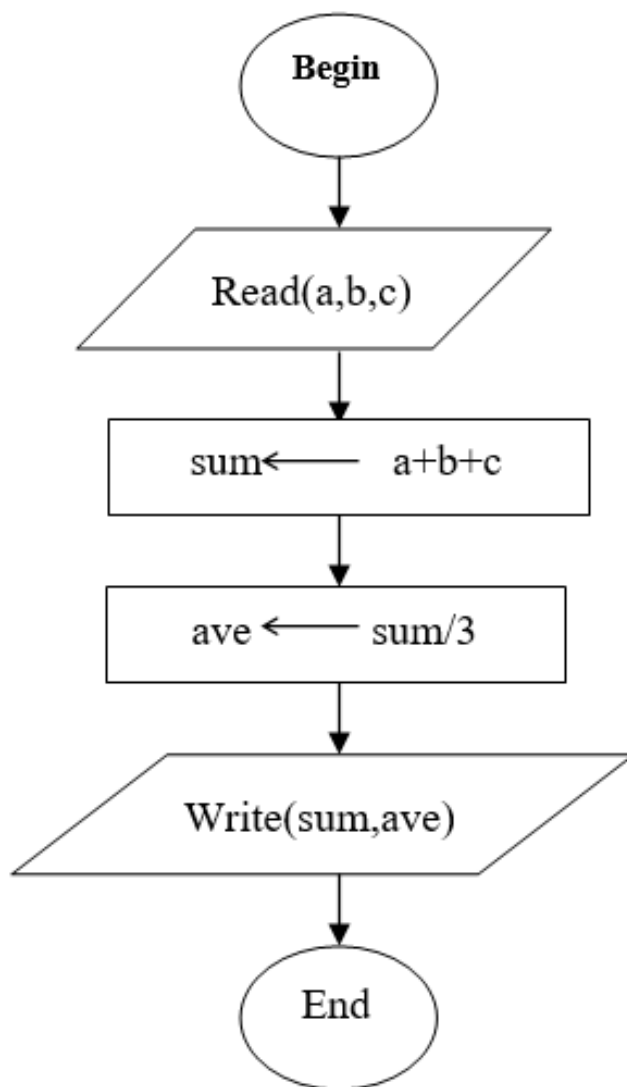
- علامتهای ورودی و خروجی: که معمولاً از متوازی‌الاضلاع استفاده می‌شود:



- علامتهای محاسباتی و جایگزینی: برای نمایش دستورات جایگزینی و محاسباتی از مستطیل استفاده می‌کنند:



- علامت شرط: برای نمایش شرط از لوزی استفاده می‌شود:
 - علامت اتصال: برای اتصال شکل‌های مختلف به هم از فلش‌های جهت‌دار استفاده می‌کنند.
- نکته: برای اینکه با دستورات زبان پاسکال آشنا شویم در داخل فلوچارت‌ها از دستورات پاسکال استفاده می‌کنیم.

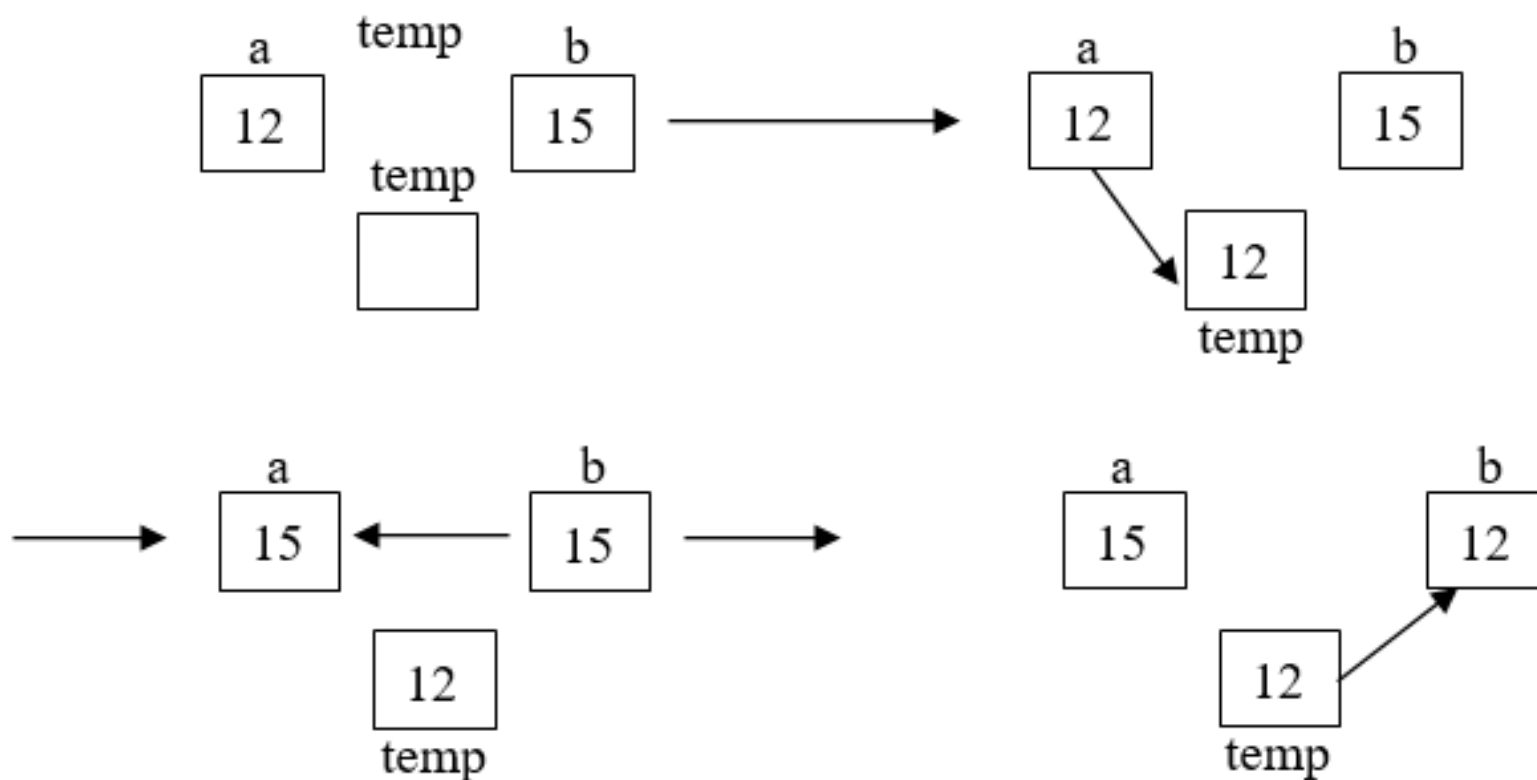


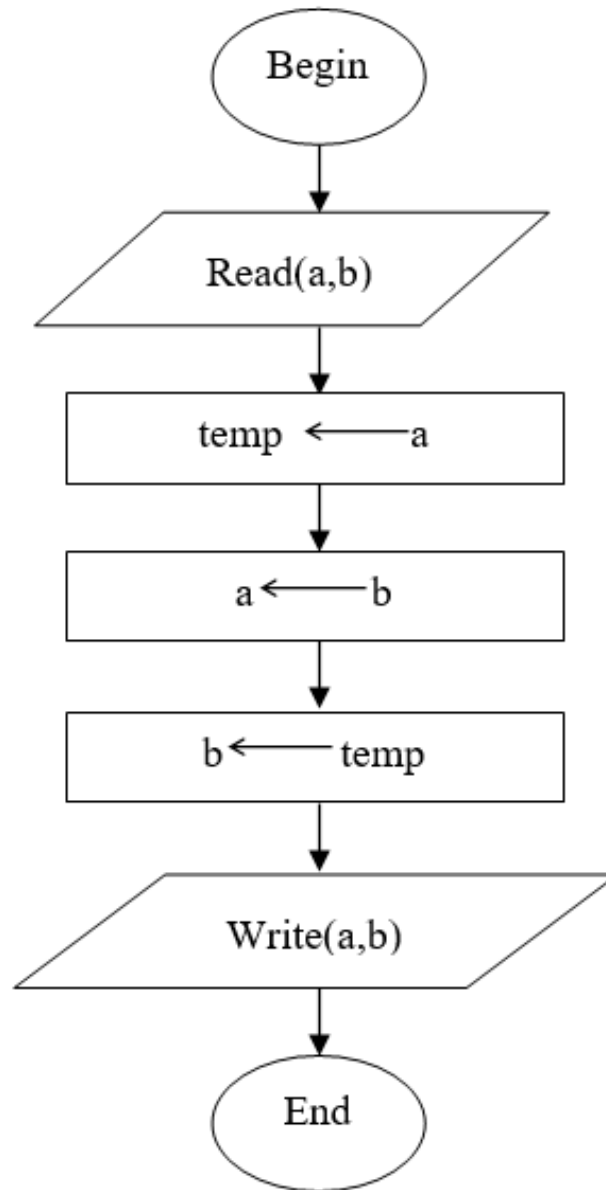
به منظور اجرای درستی الگوریتم معمولاً آن را با تعدادی ورودی تست می‌کنند.

<u>a</u>	<u>b</u>	<u>c</u>	<u>sum</u>	<u>ave</u>
12	3	6	21	7.00

مثال

فلوچارتی رسم نمائید که دو عدد از ورودی دریافت کرده، سپس محتویات دو عدد را با هم جابجا نماید.





مثال

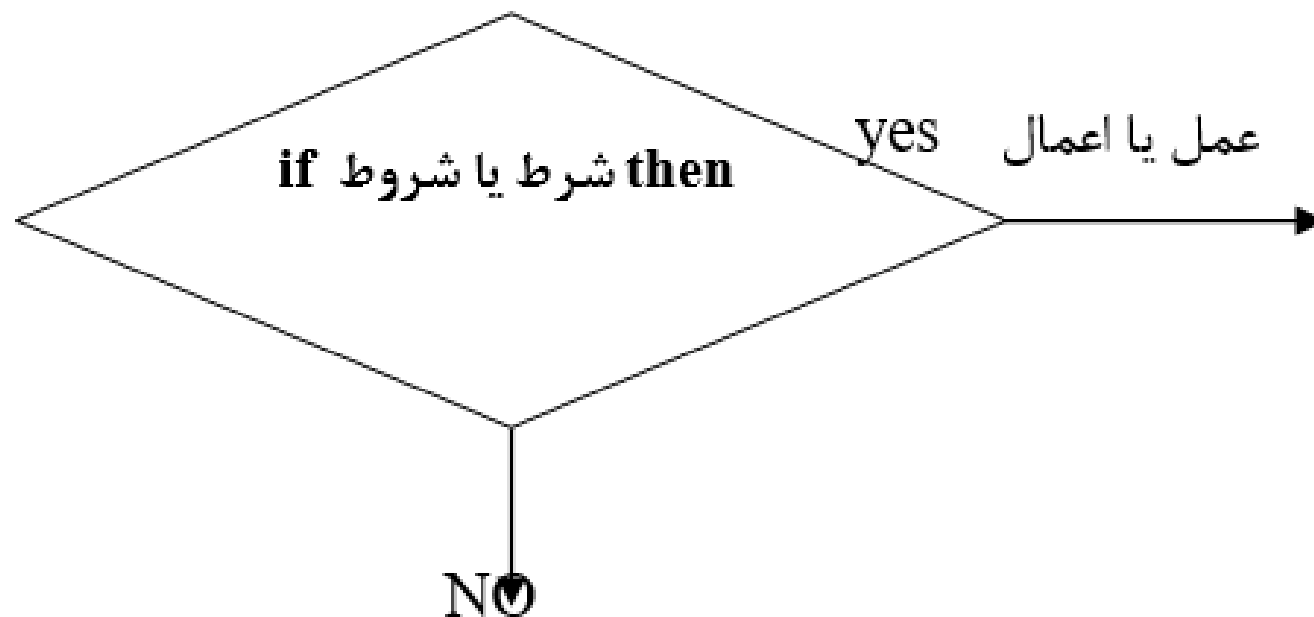
فلوچارتی رسم نمائید که دو عدد از ورودی دریافت کرده، سپس محتویات دو عدد را با هم جابجا نماید (بدون استفاده از یک خانه اضافی)

۱-فلوچارتی رسم نمائید که سه عدد از ورودی دریافت کرده، سپس محتویات سه عدد را با هم جابجا نماید.

۲- بدون استفاده از یک خانه اضافی

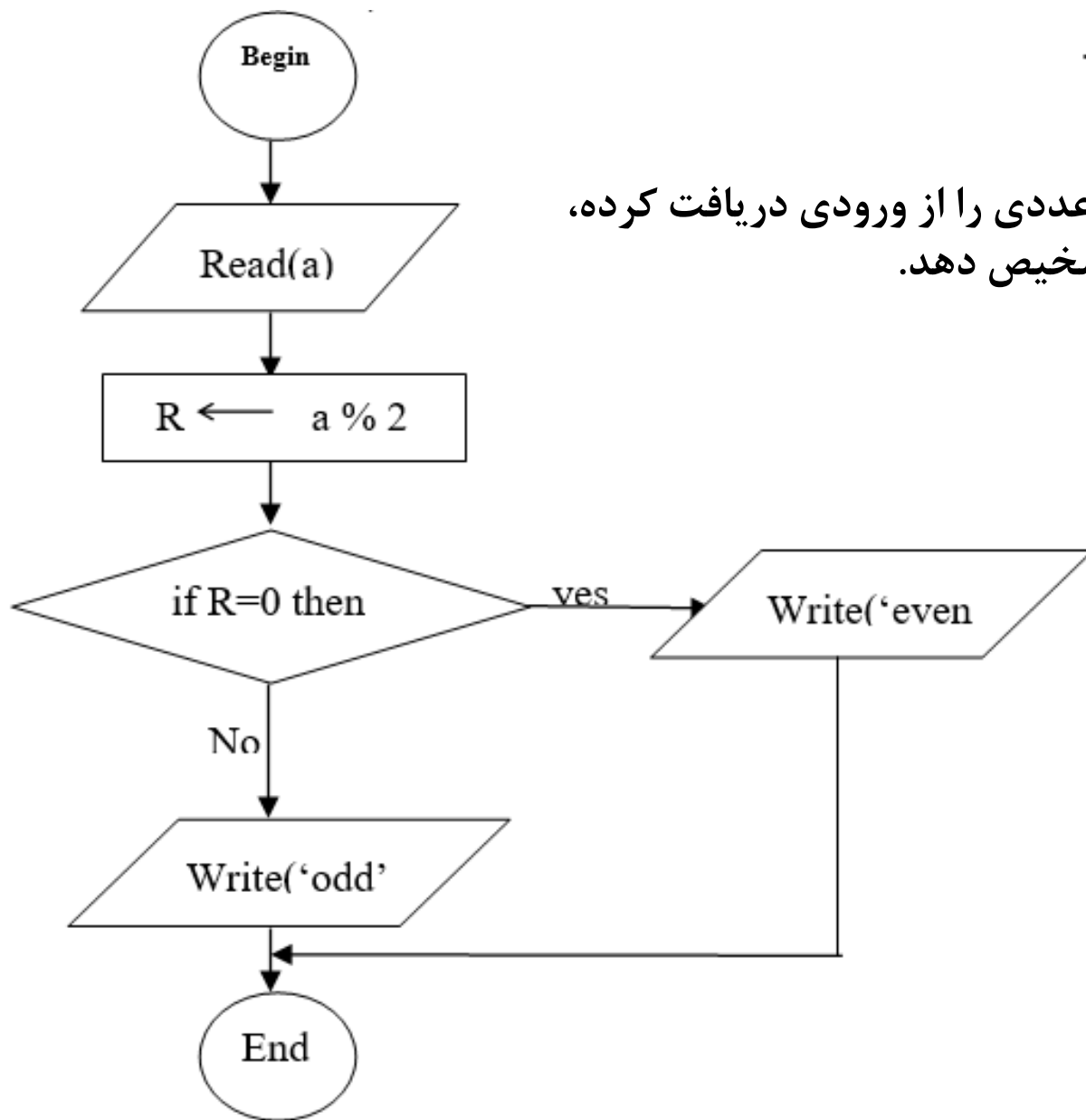
دستور العمل های شرطی

در حالت کلی شرط را بصورت زیر نمایش می دهند:



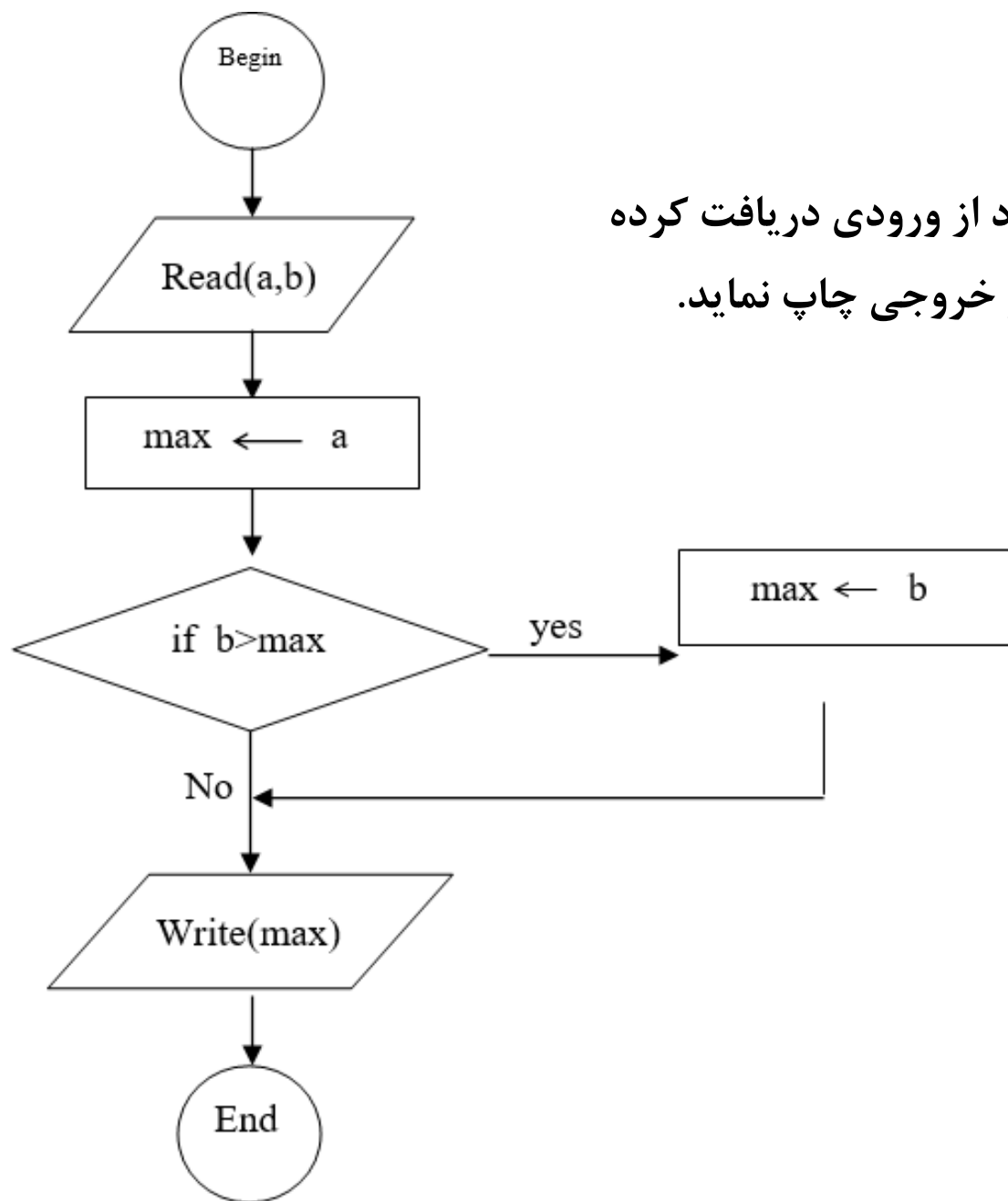
عمل یا اعمال بعدی

فلوچارتی رسم نمائید که عددی را از ورودی دریافت کرده، فرد یا زوج بودن آن را تشخیص دهد.



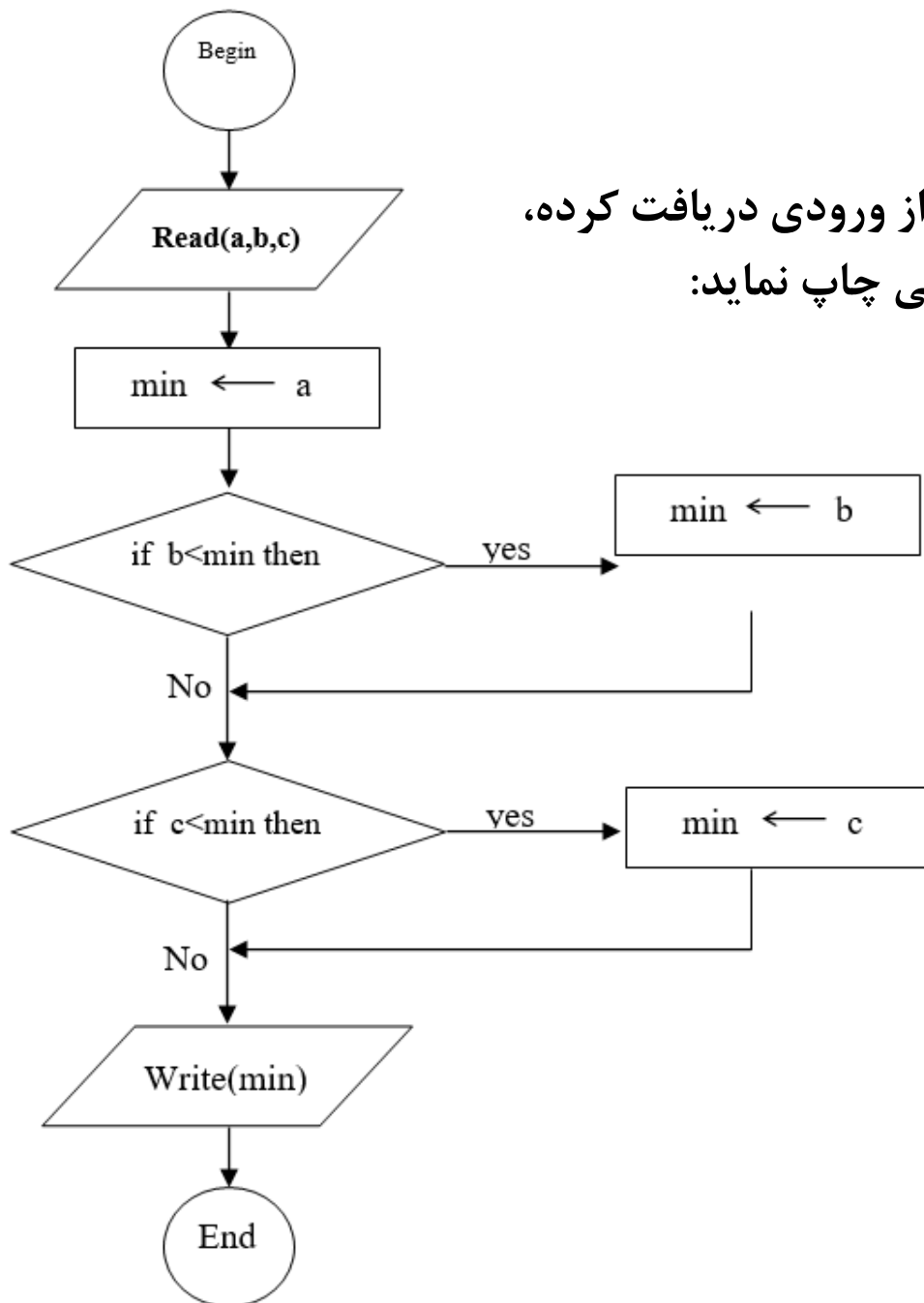
مثال

فلوچارتی رسم کنید که دو عدد از ورودی دریافت کرده
بزرگترین عدد را پیدا کرده در خروجی چاپ نماید.



مثال

فلوچارتی رسم نمائید که سه عدد از ورودی دریافت کرده،
کوچکترین عدد را یافته در خروجی چاپ نماید:



مثال

فلوچارتی رسم نمائید که دو عدد و یک عملگر را از ورودی دریافت کرده کار یک ماشین حساب ساده را شبیه‌سازی نماید.

حلقه ها

- در حل بسیاری از مسائل با عملیاتی روبرو می شویم، که نیاز به تکرار دارند و عمل تکرار آنها به تعداد مشخصی انجام می گیرد. فرض کنید، بخواهیم میانگین ۱۰۰ عدد را محاسبه کنیم، در اینصورت منطقی بنظر نمی رسد که ۱۰۰ متغیر مختلف را از ورودی دریافت کنیم سپس آنها را جمع کنیم.
- در چنین مسائلی از حلقه ها استفاده می کنند. حلقه ها انواع مختلفی دارند که در این بخش به دو نوع آنها اشاره می کنیم.

حلقه‌های نوع اول

حلقه for

در این نوع حلقه‌ها تعداد تکرار مشخص می‌باشد و هدف حلقه تکرار تعدادی از دستورالعمل‌ها به تعداد معین می‌باشد. این حلقه از اجزاء زیر تشکیل می‌شود:

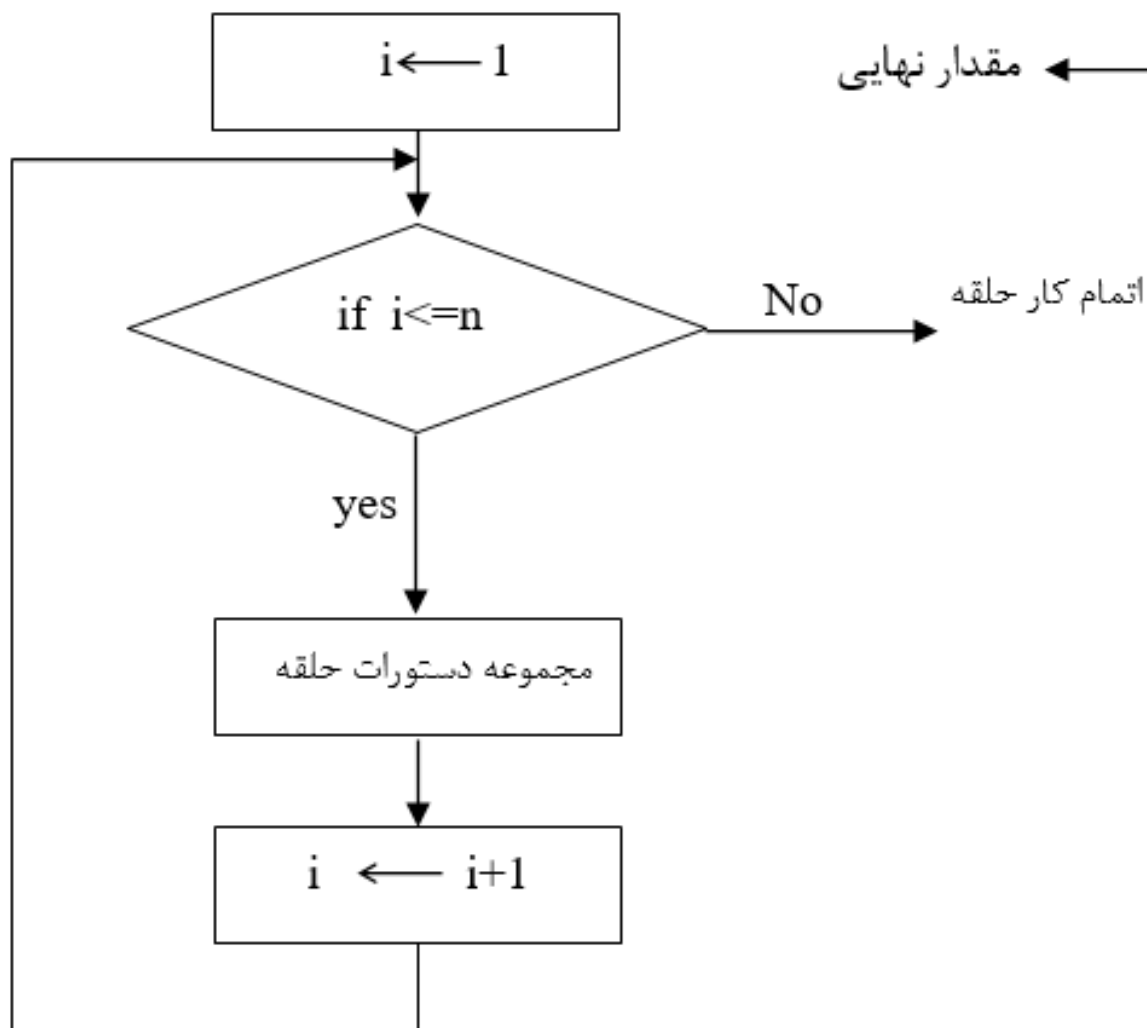
- اندیس حلقه
- مقدار اولیه برای اندیس حلقه
- مقدار افزایشده برای اندیس حلقه (معمولا یک واحد در هر مرحله)
- مقدار نهایی (تعداد تکرار حلقه)
- شرطی برای کنترل تعداد تکرار حلقه

مثال

این حلقه‌ها را غالباً با فلوجارت بصورت زیر نمایش می‌دهند:

$i \leftarrow$ اندیس حلقه

$n \leftarrow$ مقدار نهایی

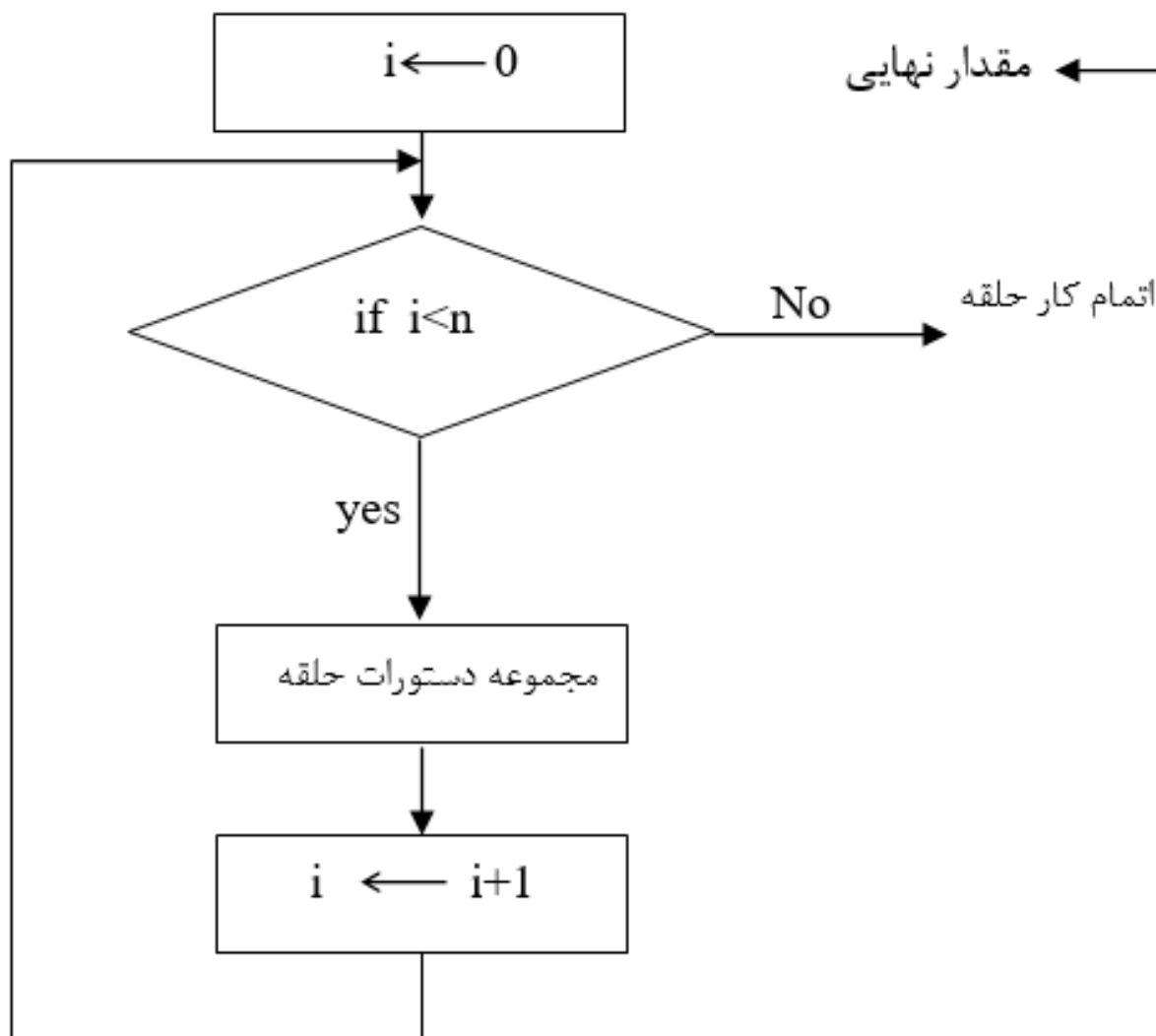


مثال

این حلقه‌ها را غالباً با فلوجارت بصورت زیر نمایش می‌دهند:

$i \leftarrow$ اندیس حلقه

$n \leftarrow$ مقدار نهایی

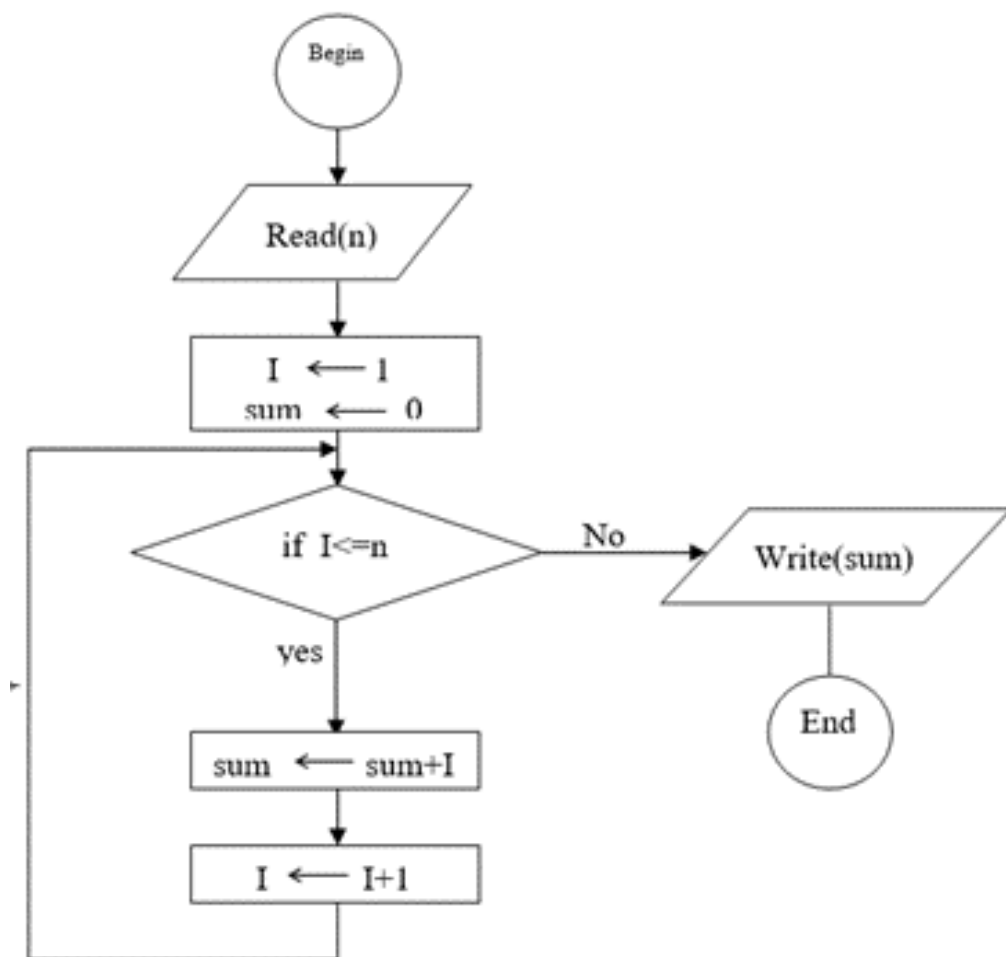


مثال

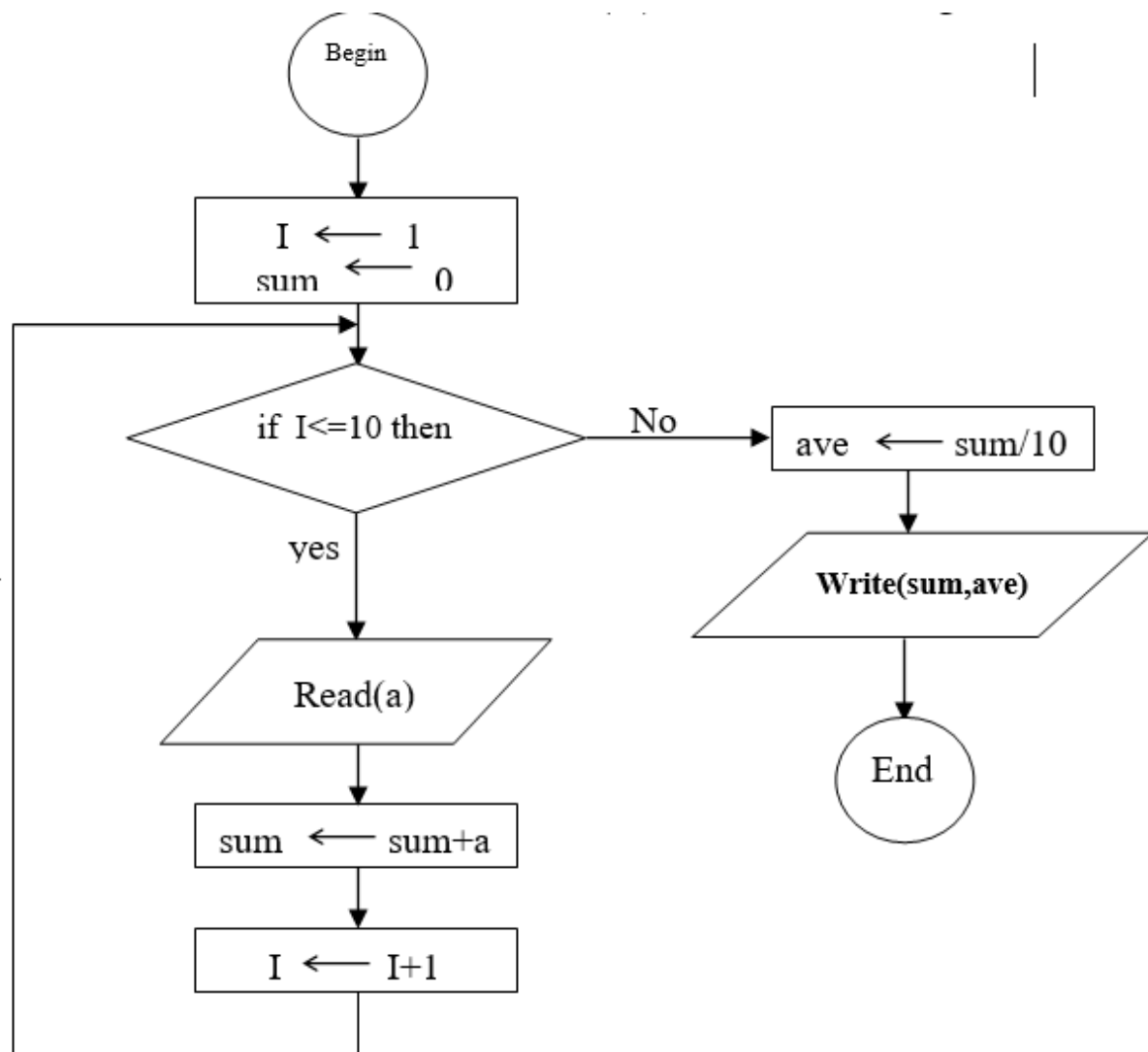
فلوچارتی رسم نمائید که عدد n را از ورودی دریافت کرده، مجموع اعداد از یک تا n را محاسبه کند.

اندیس حلقه \longrightarrow i

مقدار نهایی \longrightarrow n



فلوچارتی رسم نمائید که ۱۰ عدد از ورودی دریافت کرده، مجموع و میانگین ۱۰ عدد را محاسبه و چاپ کند.

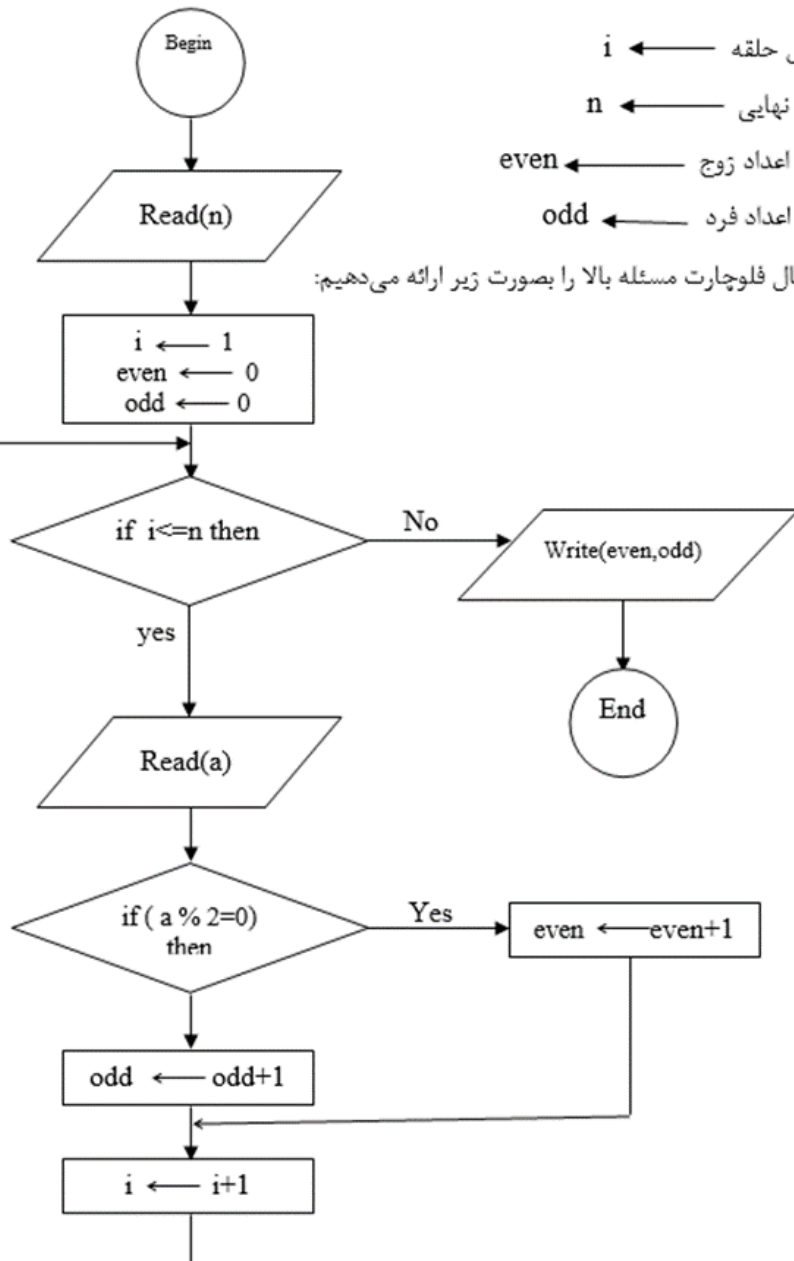


مثال

فلوچارتی رسم کنید که n عدد از ورودی دریافت کرده سپس تعداد اعداد زوج و فرد را شمرده ، در خروجی چاپ نماید.

i ← اندیس حلقه
 n ← مقدار نهایی
 $even$ ← تعداد اعداد زوج
 odd ← تعداد اعداد فرد

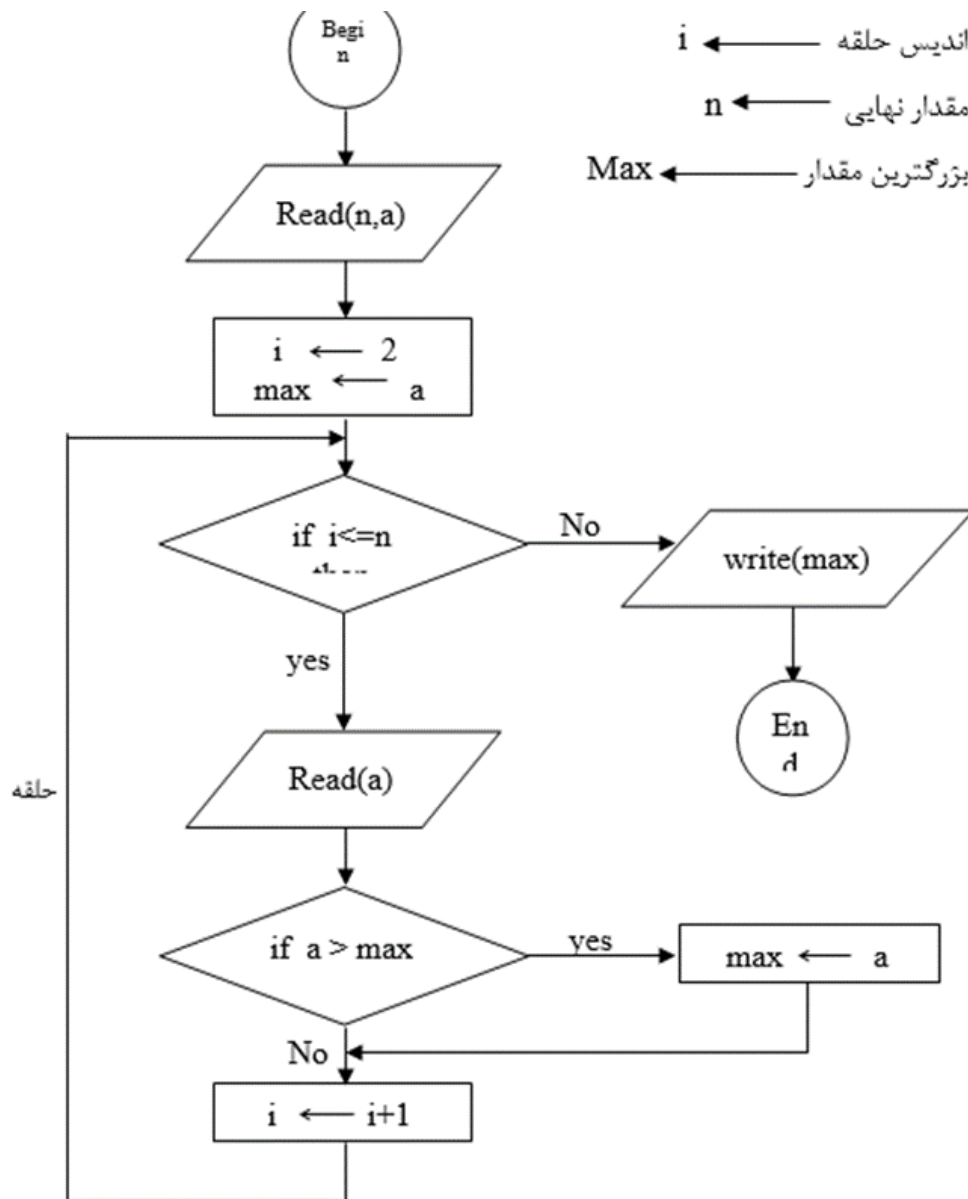
حال فلوچارت مسئله بالا را بصورت زیر ارائه می دهیم:



مثال

فلوچارتی رسم کنید که n عدد از ورودی دریافت کرده، بزرگترین مقدار از بین n عدد را پیدا کرده در خروجی چاپ نماید.

i ← اندیس حلقه
 n ← مقدار نهایی
 Max ← بزرگترین مقدار



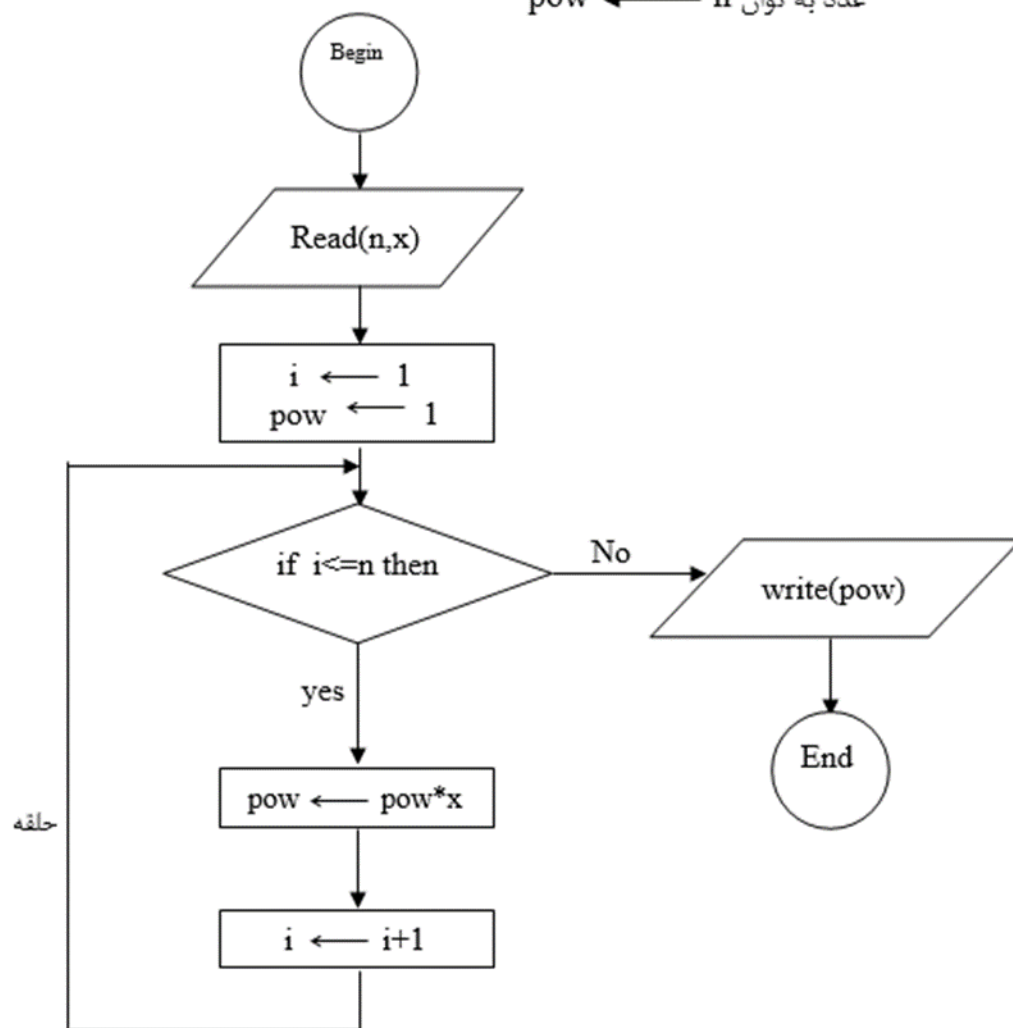
مثال

فلوچارتی رسم نمائید که n ، دو عدد صحیح مثبت را از ورودی دریافت کرده سپس x به توان n را محاسبه کند.

i ← اندیس حلقه

n ← مقدار نهایی

pow ← عدد به توان n



مثال

فلوچارتی رسم کنید که عدد N را از ورودی دریافت کرده، فاکتوریل آنرا محاسبه نماید.

فاکتوریل یک عدد در حالت کلی برابر است با:

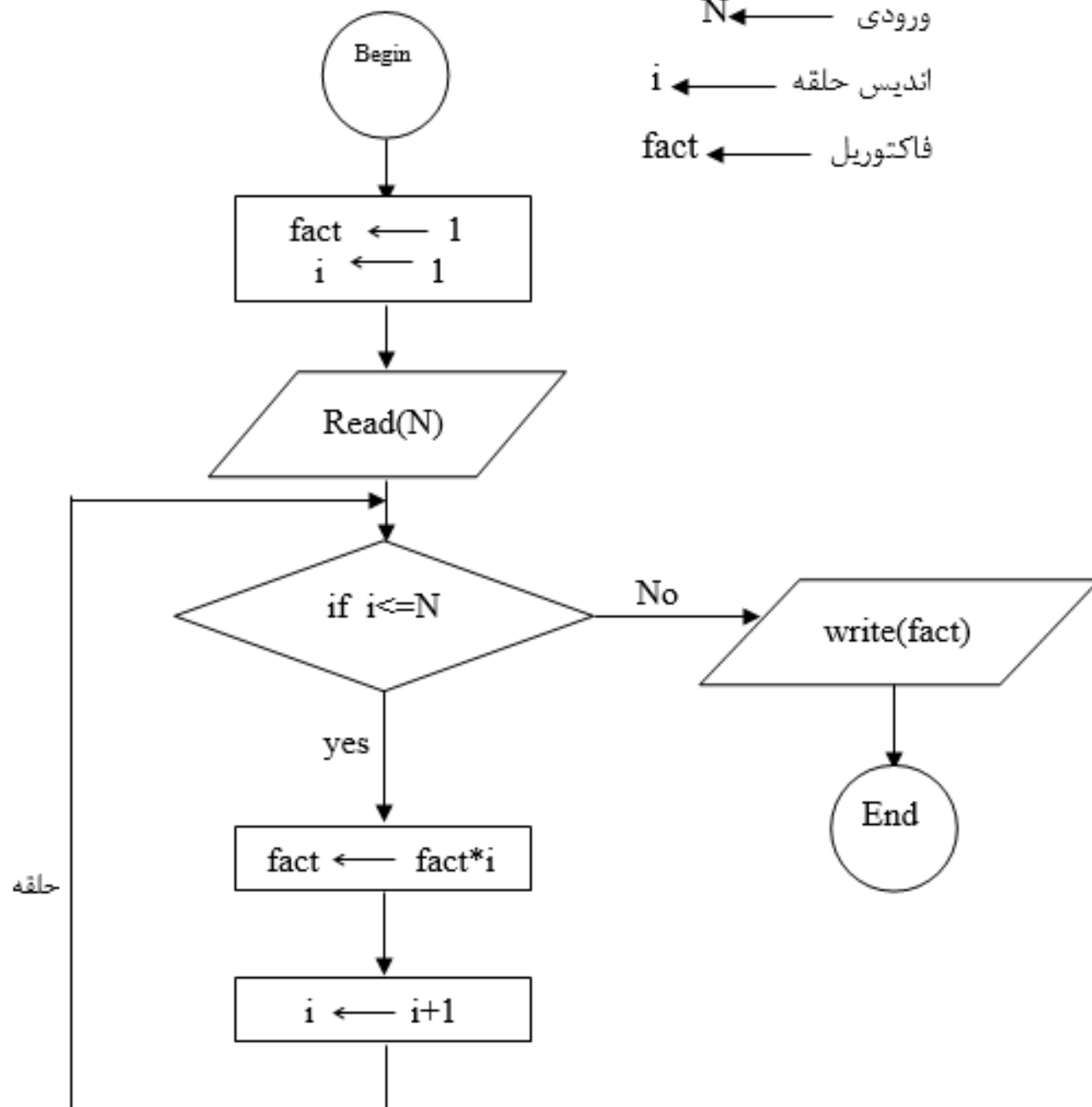
$$N! = 1 \times 2 \times \dots \times (N-1) \times N$$

مثال

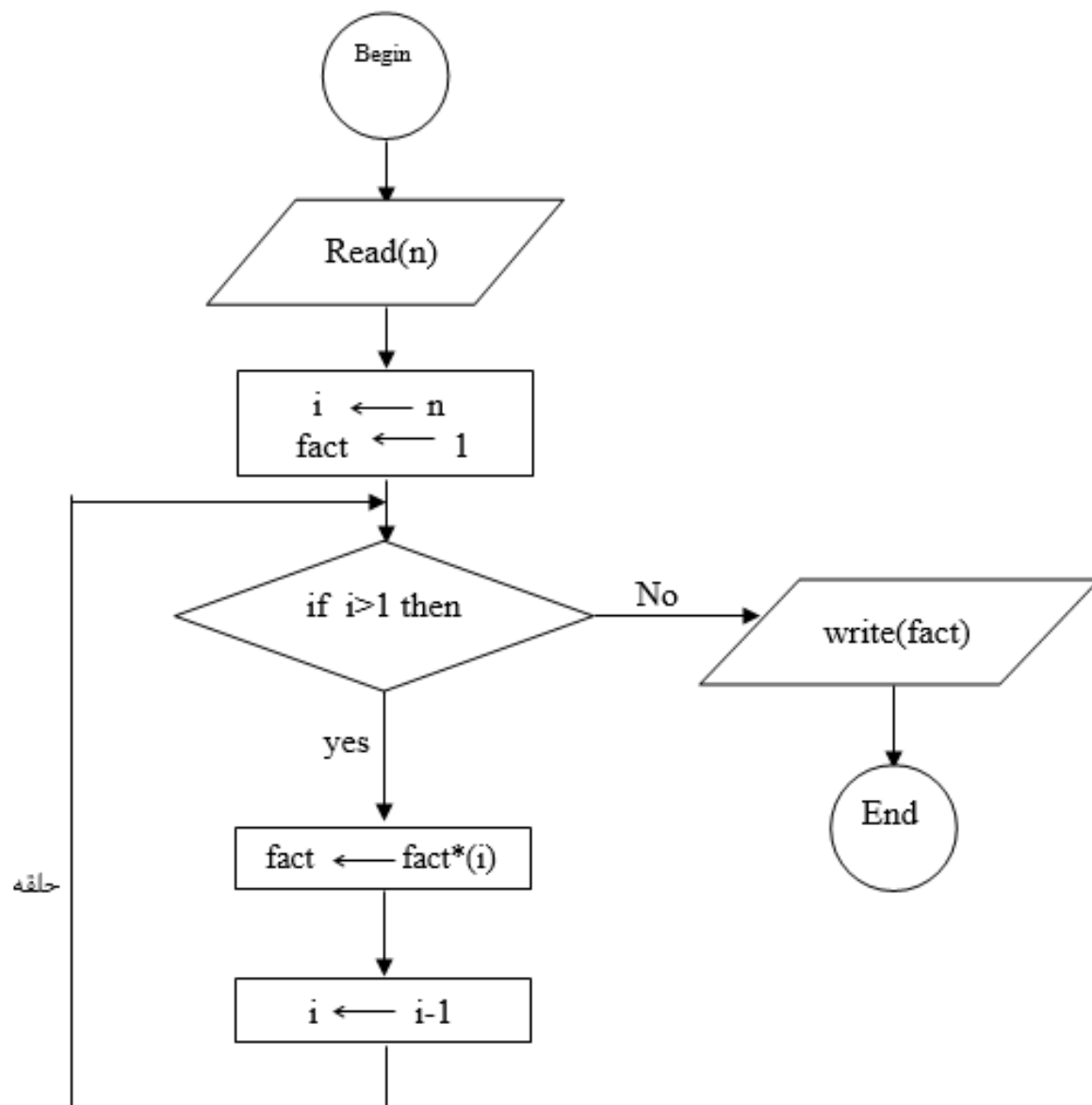
N ← ورودی

i ← اندیس حلقه

fact ← فاکتوریل



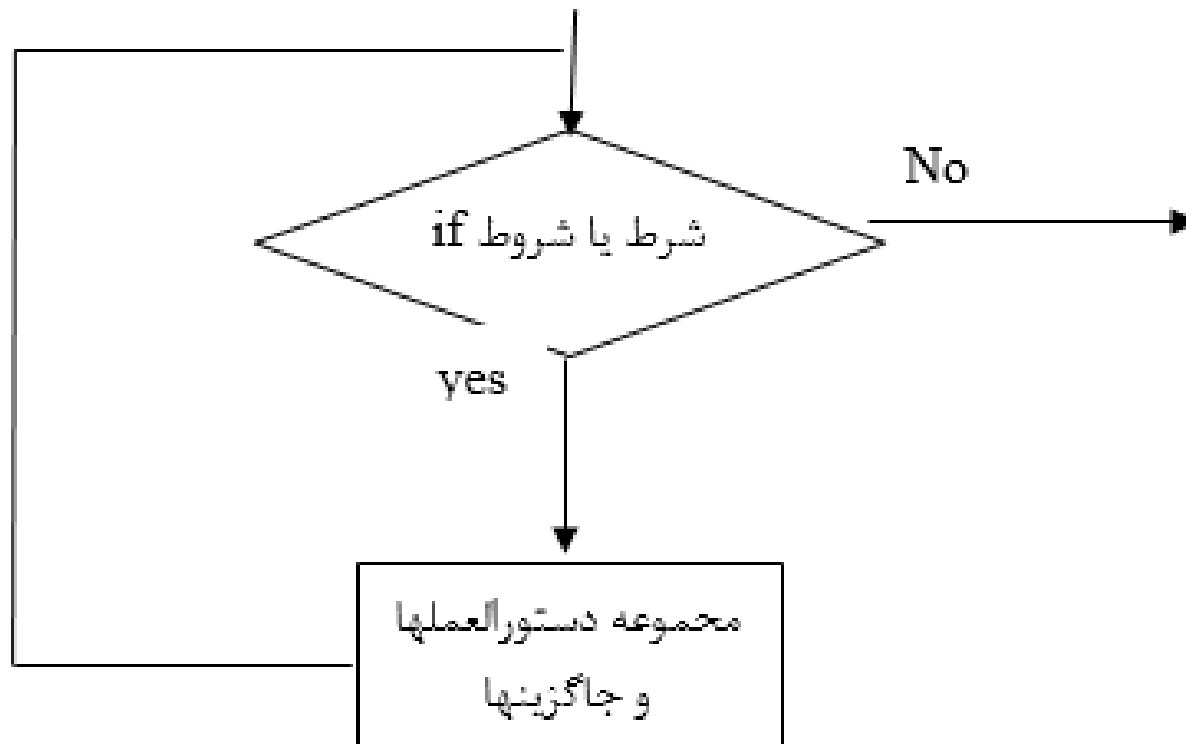
فلوچارت بالا را می‌توان بصورت زیر نیز می‌توان نوشت:



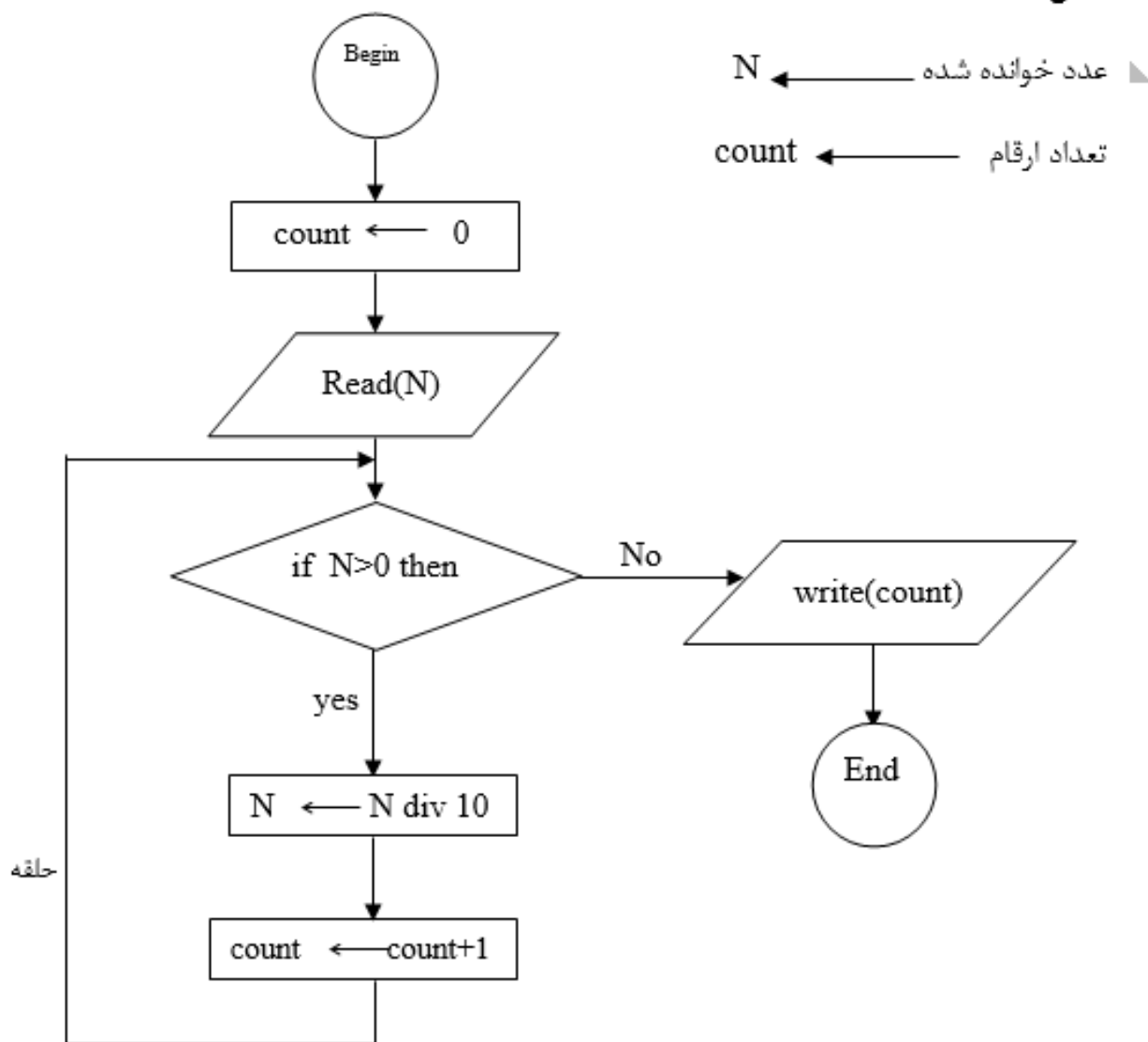
حلقه‌هایی که تعداد تکرار آنها مشخص نیست

حلقه while

در این حلقه‌ها با توجه به ورودی، تعداد تکرار مشخص می‌شود. و دقیقاً نمی‌توان تعداد تکرار حلقه را بدون ورودی معین کرد. این حلقه‌ها فقط شامل شرطی یا شرطی هستند که تا زمانی که آنها برقرار باشند حلقه اجرا می‌شود. در حالت کلی این نوع حلقه‌ها بصورت زیر نمایش داده می‌شوند:



فلوچارتی رسم کنید که عددی را از ورودی دریافت کرده سپس تعداد ارقام آن را شمرده در خروجی چاپ نماید.



مثال

فلوچارتی رسم نمائید که عددی از ورودی دریافت کرده، سری فیبوناچی قبل از آنرا تولید نماید.

سری بصورت زیر می باشد :

0 1 1 2 3 5 8 13 ...

و در حالت کلی جملات سری بصورت:

$$f_k = f_{k-1} + f_{k-2}$$

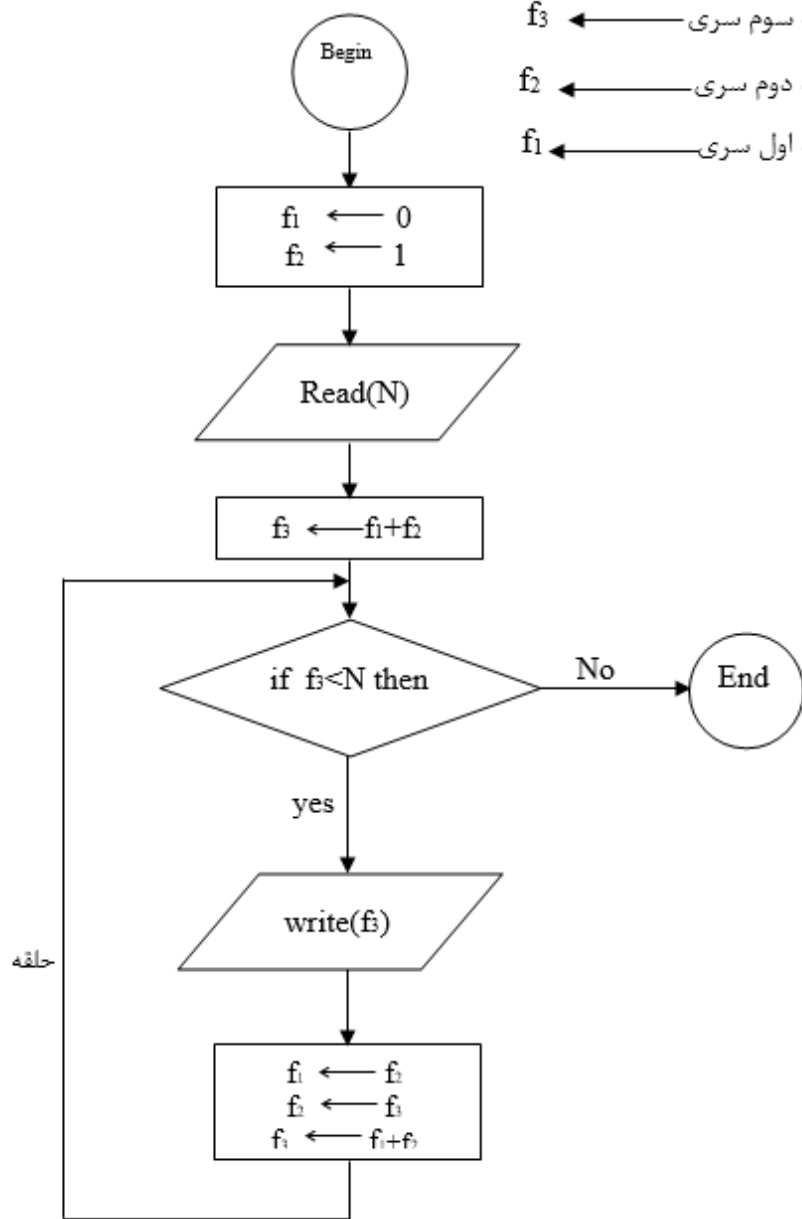
مثال

عدد خوانده شده $\leftarrow N$

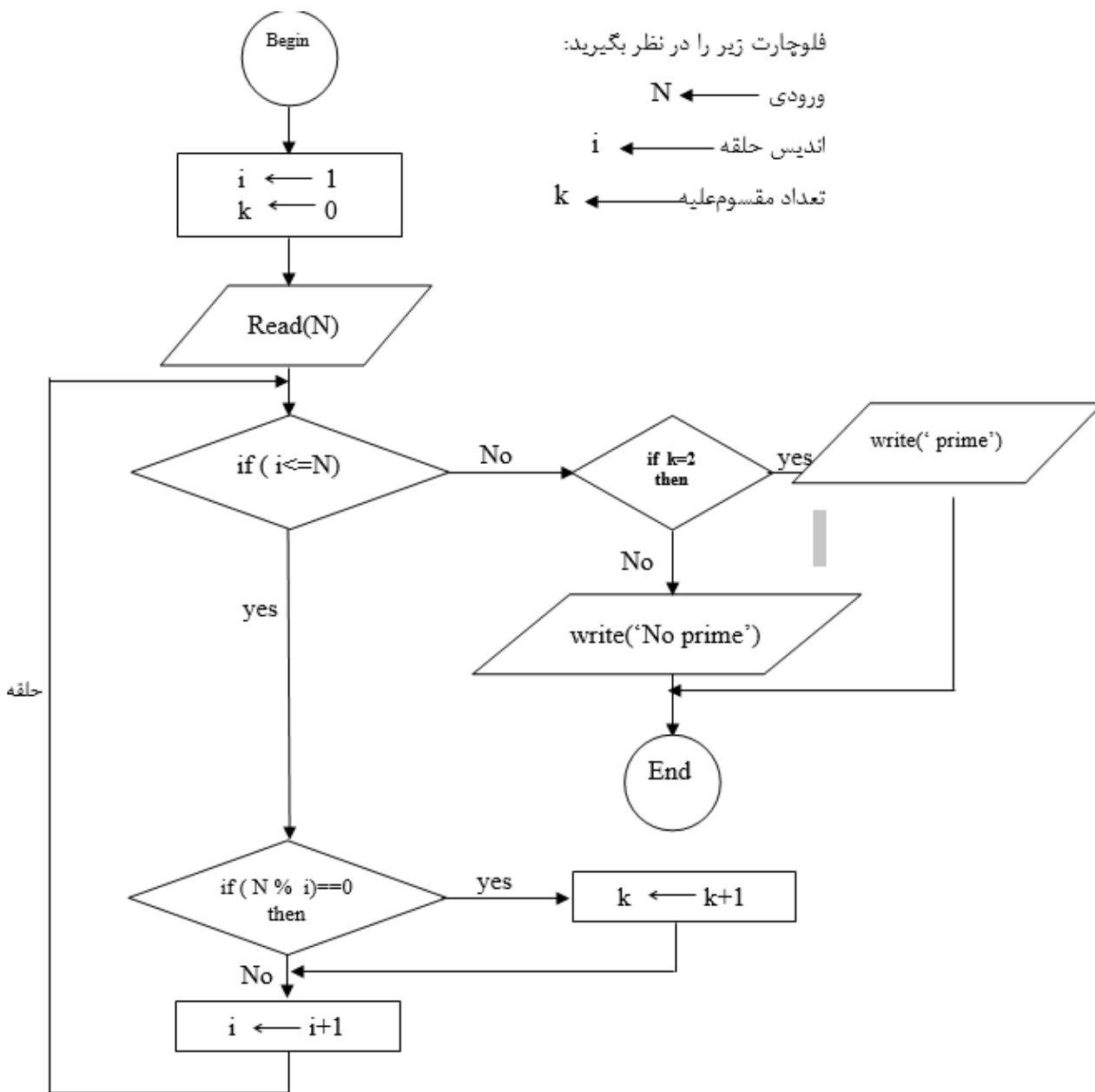
جمله سوم سری $\leftarrow f_3$

جمله دوم سری $\leftarrow f_2$

جمله اول سری $\leftarrow f_1$

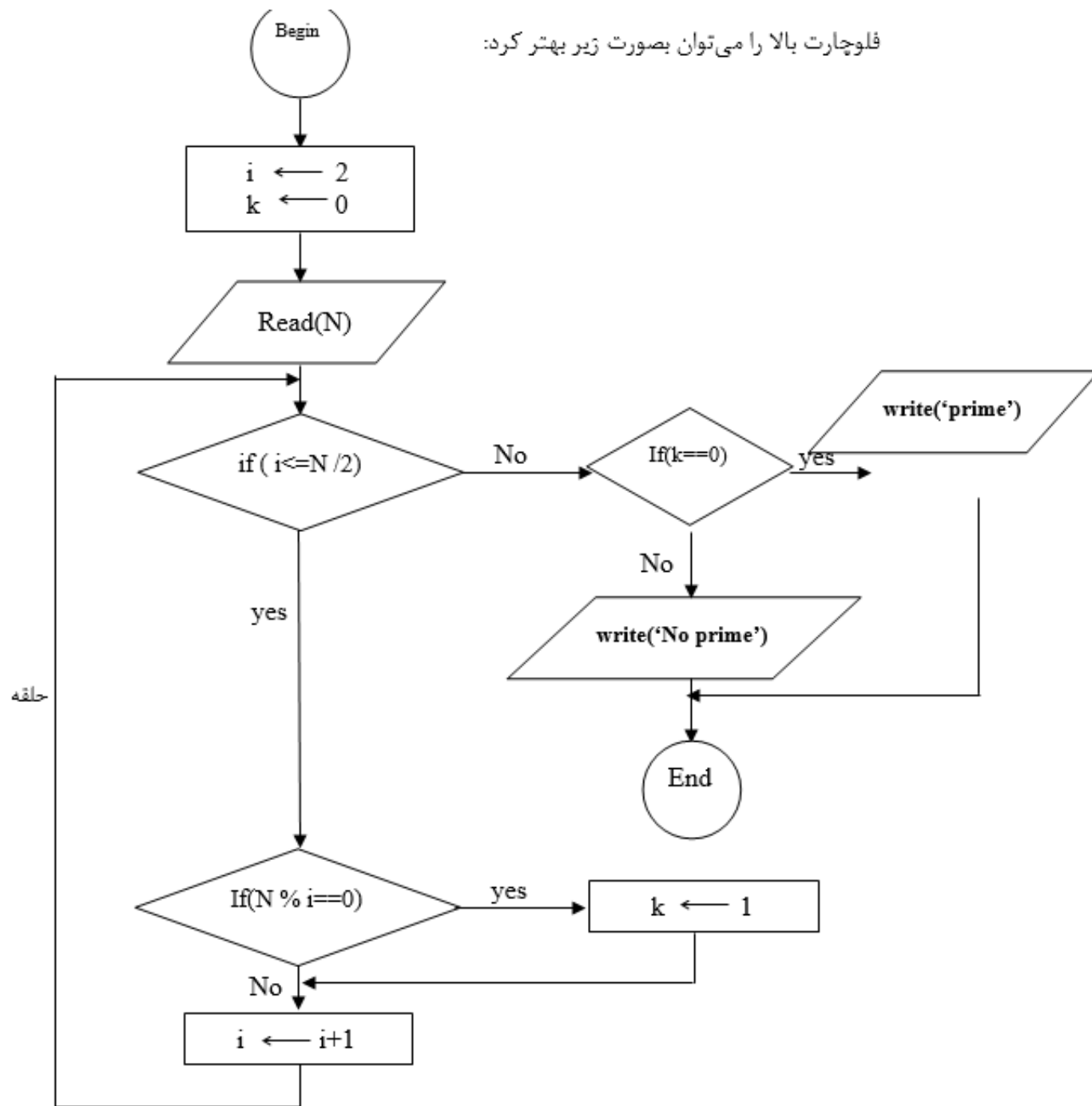


فلوچارتی رسم نمائید که عددی از ورودی دریافت کرده، اول بودن عدد را بررسی نماید.



مثال

فلوچارت بالا را می توان بصورت زیر بهتر کرد:



مثال

- فلوچارتی رسم نمائید که عددی از ورودی دریافت کرده، کامل بودن آنرا بررسی نماید. (عدد کامل، عددی است که مجموع مقسوم‌علیه‌های آن با خودش برابر باشد).
- فلوچارتی رسم نمائید که عددی را از ورودی دریافت کرده، تشخیص دهد که عدد خوانده شده جزء سری فیبرناچی هست یا نه؟
- فلوچارتی رسم نمائید که دو عدد M , N را از ورودی خوانده، بزرگترین مقسوم‌علیه مشترک دو عدد را محاسبه و چاپ کند.

مثال

- فلوچارتی رسم نمائید که N عدد از ورودی دریافت کرده، بزرگترین مقدار از بین N عدد و تعداد تکرار آن را محاسبه و چاپ نماید.
- فلوچارتی رسم کنید که عددی از ورودی خوانده، آن را به مبنای ۲ ببرد.
- فلوچارتی رسم کنید که، عدد N را از ورودی خوانده، تشخیص دهد عدد خوانده شده فاکتوریل چه عددی است.

حلقه های تودرتو

الگوریتم‌هایی که تا حال بکار بردیم، فقط شامل یک حلقه بودند.

در صورتی که در بسیاری از مسائل ممکن است نیاز به استفاده از چند حلقه در داخل هم باشیم. در این نوع حلقه‌ها باید دقت بیشتری به خرج دهیم، تا مشکلی پیش نیاید. اگر از حلقه‌های نوع اول بصورت تودرتو استفاده کنیم در اینصورت برای هر حلقه شرط نهایی و اندیس اولیه جداگانه باید تعریف کنیم. در حلقه‌های تودرتو به ازای یکبار تکرار حلقه اولیه، حلقه داخلی به اندازه مقدار نهایی خود تکرار می‌شود. در کل اگر حلقه اولیه n بار تکرار شود و حلقه داخلی m بار، در اینصورت کل حلقه $n \times m$ بار تکرار خواهد شد.

فلوچارت حلقه‌های تودرتو را می‌توان بصورت زیر نشان داد:

اندیس حلقه اول $\leftarrow i$

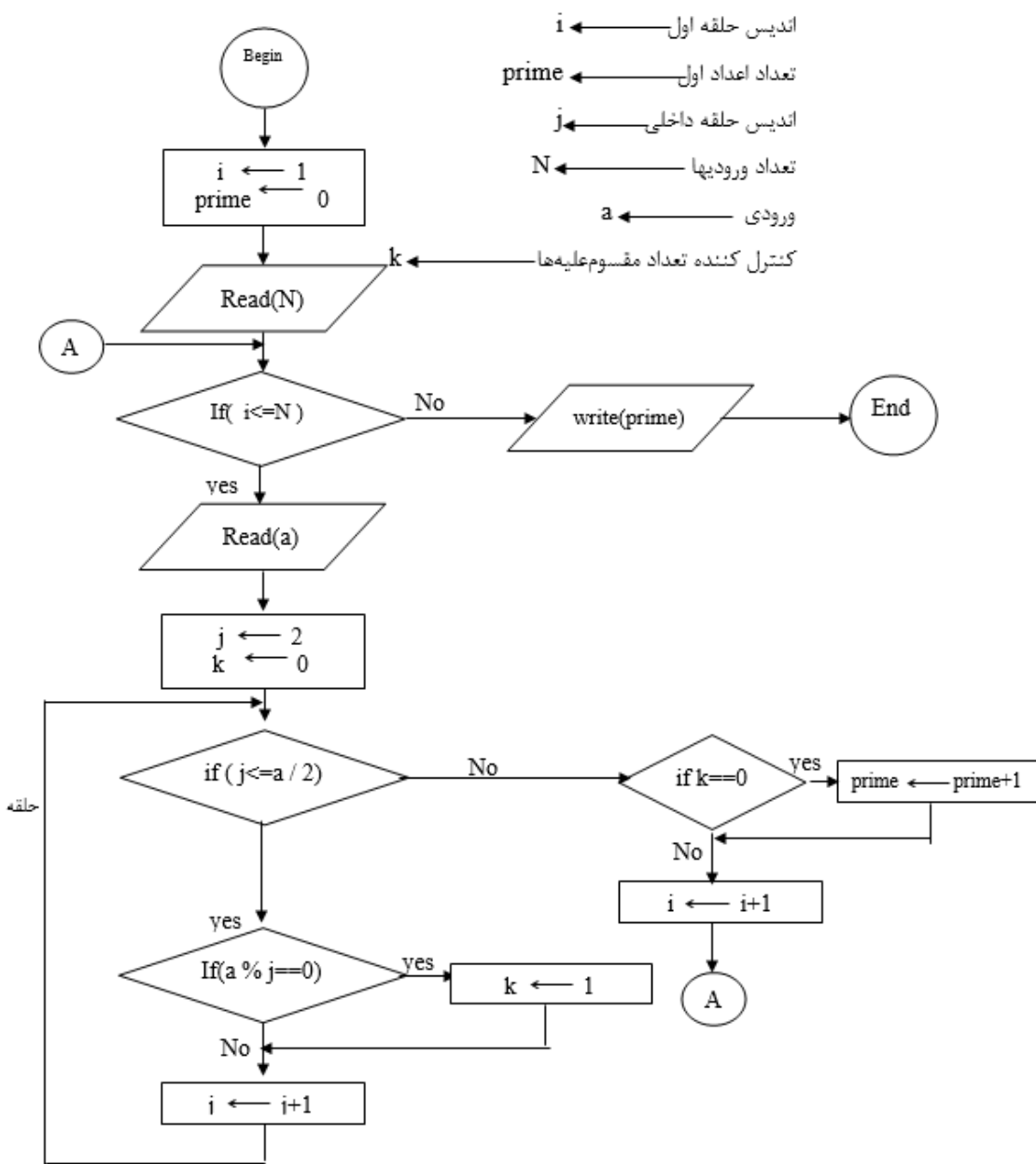
اندیس حلقه داخلی $\leftarrow j$

مقدار نهایی حلقه اول $\leftarrow n$

مقدار نهایی حلقه داخلی $\leftarrow m$

مثال

فلوچارتی رسم نمائید که N عدد از ورودی دریافت کرده تعداد اعداد اول را شمرده در خروجی چاپ نماید.



مثال

فلوچارتی رسم نمائید که N را از ورودی دریافت کرده، مجموع سری زیر را محاسبه نماید:

$$S=1+\frac{2}{2!}+\frac{3}{3!}+\dots+\frac{N}{N!}$$

مثال

