

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

مبانی کامپیوتر

Introduction to Fortran

90/95

مشخصات درس

- نام: مبانی برنامه‌سازی کامپیوتر (فرترن)
- تعداد واحد: 3
- پیش‌نیاز: ریاضی 1
- اهداف درس:
 - آشنایی با مبانی برنامه‌سازی
 - تحلیل یک مساله
 - تهیه الگوریتم آن
 - تبدیل آن نمودن به کد (code)
 - آشنایی با یک زبان برنامه‌نویسی (فرترن)
 - حل مسایل علمی و مهندسی به شیوه عددی

مراجع و ماخذ

- مبانی کامپیوتر و الگوریتم‌ها، نوشته جعفر نژادقمی و کریم‌پور، ناشر: علوم رایانه
- فرترن 90، نوشته دکتر مشعل، انتشارات جهاد دانشگاهی
- فرترن 90 برای رشته‌های علوم و مهندسی، ترجمه: دکتر محمود صالح، ناشر: دانشگاه امام حسین
- کتاب آموزشی Fortran 90/95 ترجمه امید ناصرقدسی و امید ممتاز، انتشارات ناقوس
- برنامه‌نویسی به زبان Fortran 90/95 برای رشته‌های علوم و مهندسی تالیف: موسوی‌ندوشنی، ناشر: دانشگاه صنعت آب و برق

اهداف کلی این درس

دانشجو پس از مطالعه این درس باید بتواند:

الگوریتمی برای حل مسئله ارائه دهد

اصول و مبانی اولیه نرم افزار و سخت افزار را بشناسد

اهداف و مفاهیم زبان های برنامه نویسی را بداند

مفاهیم اولیه برنامه نویسی ساخت یافته را بداند و اصول لازم را در مرحله اجراء بکار ببرد

دستورات زبان فرترن را در برنامه ها بکار ببرد

از توابع و روال های استاندارد زبان فرترن در صورت لزوم استفاده نماید

• از توابع ، روال ها برای جدا کردن قطعات برنامه استفاده کند

• فایل ها را به عنوان محلی برای ذخیره داده ها در برنامه استفاده کند

• با استفاده از قابلیت های زبان فرترن برنامه ای را برای یک سیستم بنویسد.

❖ جایگاه این درس

این درس نقطه شروعی برای ورود به

دنیای جالب برنامه نویسی و علم و فن کامپیوتر هست . بنابراین یاد گیری

اصول اولیه برنامه نویسی در این درس از جایگاه ویژه ای برخوردار است.

این درس پایه و اساس برنامه نویسی که جزء اصول این رشته می باشد را به

فرگیران یاد می دهد .

بنابراین یادگیری دقیق این درس به همراه ارائه پروژه های عملی که لازمه

این درس می باشد جزء اهم مسائل می باشد .

مراحل ایجاد یک برنامه

- تعریف و شناخت کامل مساله
- ایجاد الگوریتم برنامه
- تبدیل الگوریتم به یک زبان برنامه‌نویسی (مثلا فرترن) یا نوشتن کدها (codes)
- ترجمه کدها به زبان ماشین (compile)
- ایجاد یک فایل اجرایی (*.exe)
- اجرای برنامه و رفع خطاهای احتمالی آن
- مستند نمودن برنامه

الگوریتم‌ها Algorithm

هدفهای کلی

شناخت حل مسئله و ارائه الگوریتم

شناخت اجزاء لازم برای حل مسئله

بررسی صحت الگوریتم

هدفهای رفتاری

دانشجو پس از مطالعه این فصل باید بتواند:

الگوریتمی را برای حل مسئله ارائه دهد.

الگوریتم های مختلف برای یک مسئله را مقایسه کند.

شرط ها و حلقه ها را در الگوریتم بکار ببرد .

مقدمه

در زندگی روزمره، انسان با مسائل مختلفی روبروست و برای هر کدام از این مسائل (حل مشکلات) راه حلی و روشی را برمیگزیند. 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 روش انجام کار یا حل مسئله را الگوریتم آن مسئله می‌نامند

❖ تعریف الگوریتم

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

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



❖ مراحل الگوریتم

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

مخواندن داده‌ها

انجام محاسبات

مخروجی‌ها

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

ورودیها

انجام محاسبات

خروجیها

a , b

جمع دو عدد

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: 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

0- شروع

1- سه عدد از ورودی بخوان

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

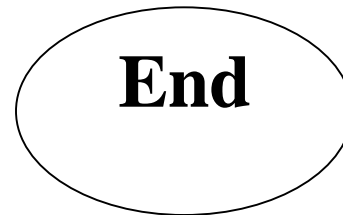
3- sum را بر سه تقسیم کرده، در ave قرار بده.

4- sum , ave را در خروجی چاپ کن.

5- پایان.

معمولا درك يك الگوریتم با شكل راحتتر از نوشتن آن بصورت متن می باشد. لذا الگوریتم را با فلوچارت (flowchart) نمایش می دهند. فلوچارت از شكلهاي زیر تشکیل می شود.

● علامت های شروع و پایان: که معمولا از یک بیضی استفاده می کنند:



● علامت های ورودی و خروجی: که معمولا از متوازی الاضلاع استفاده می شود:



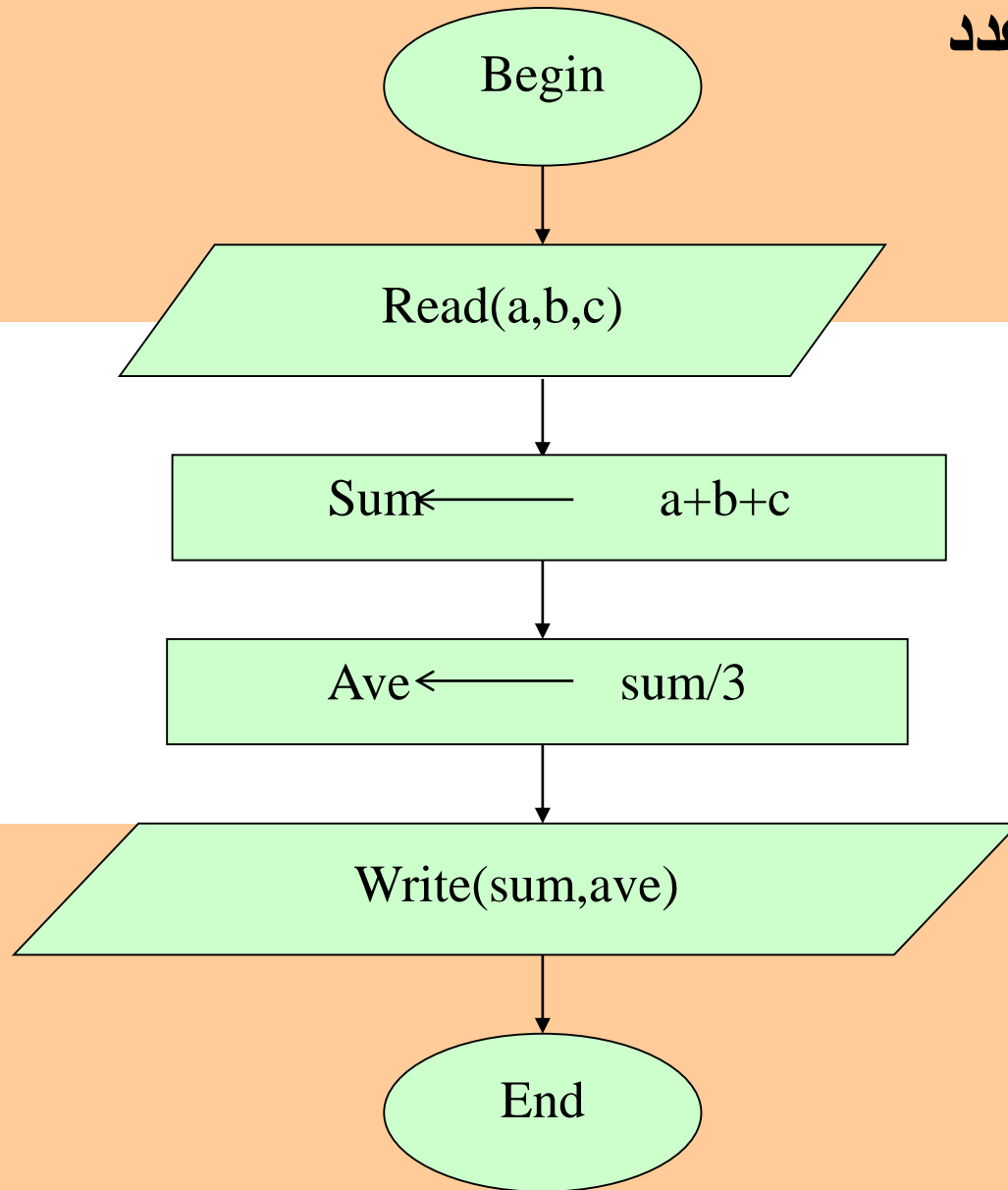
• علامتهای محاسباتی و جایگزینی: برای نمایش دستورات جایگزینی و محاسباتی از مستطیل استفاده می کنند:

جایگزین یا
محاسبات

• علامت شرط: برای نمایش شرط از لوزی استفاده می شود.

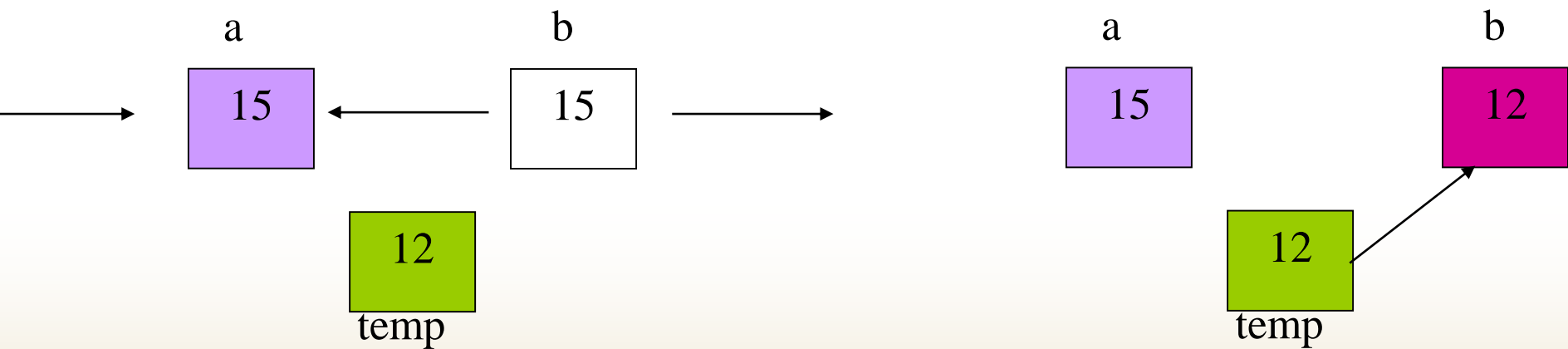
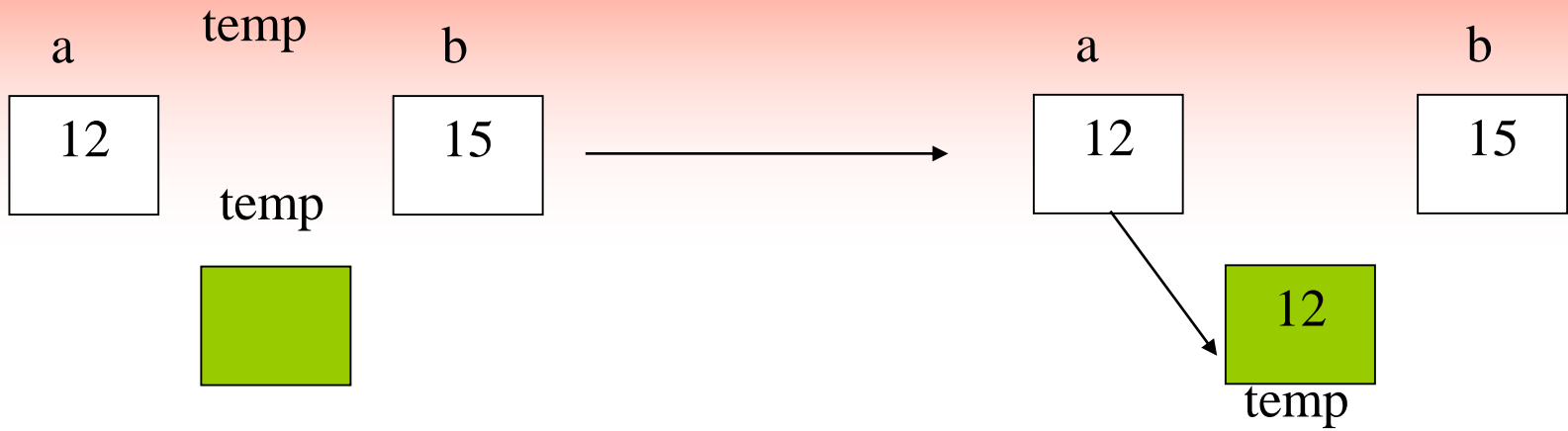
• علامت اتصال: برای اتصال شکل های مختلف بهم از فلش های جهت دار استفاده می کنند.

• فلوجارت مجموع سه عدد

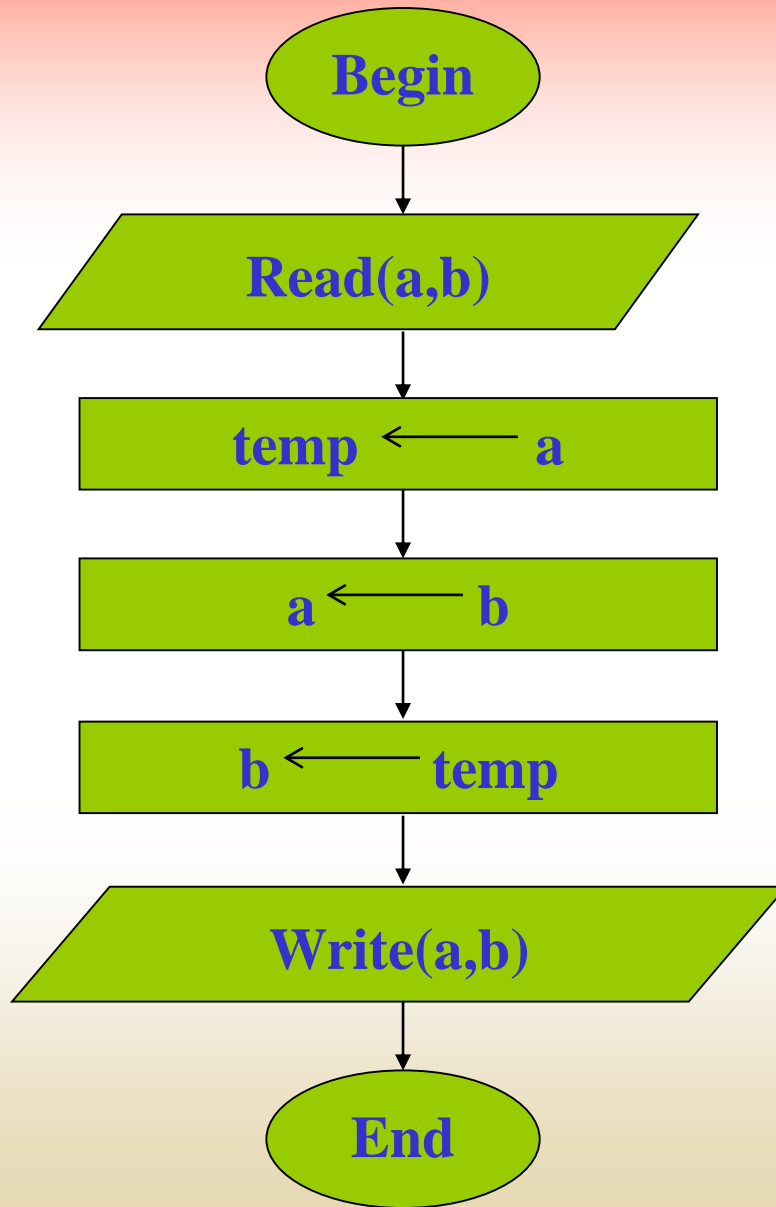


مثال: فلوجارتی رسم نمائید که دو عدد از ورودی دریافت کرده سپس محتویات دو عدد را با هم جابجا نماید.

برای حل این مسئله a , b را دو متغیر که در آنها دو عدد خوانده شده، قرار می‌گیرند در نظر می‌گیریم. سپس با استفاده از یک متغیر کمکی محتویات این دو عدد را جابجا می‌کنیم :



فلوچارت مسئله بالا بصورت زیر خواهد بود:



❖ تمرین

1- فلوجارتي رسم نمائيد که طول و عرض مستطيل را از ورودی دریافت کرده محیط و مساحت آنرا محاسبه و چاپ کند.

2- فلوجارتي رسم نمائيد که شعاع دایره‌ای را از ورودی دریافت کرده، محیط و مساحت آنرا محاسبه و چاپ نماید.

3- فلوجارتي رسم کنید که سه عدد **Third , second, first** را از ورودی دریافت کرده، محتویات آنها را جابجا نموده، حاصل را در خروجی چاپ کند.

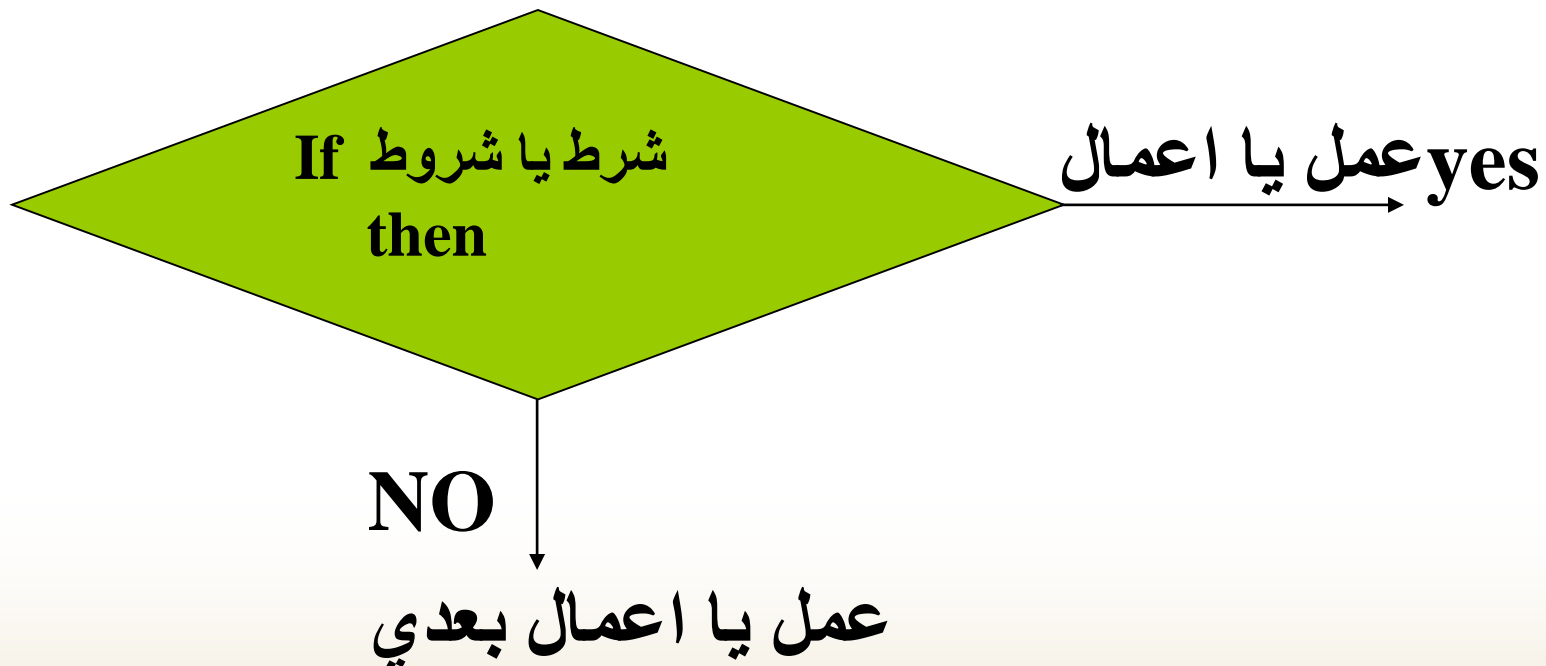
4- فلوجارتي رسم نمائيد كه عددي (درجه حرارت بر حسب سانتیگراد) را از ورودی دریافت کرده سپس آنرا به درجه فارنهایت تبدیل کند.

❖ دستورالعمل‌های شرطی

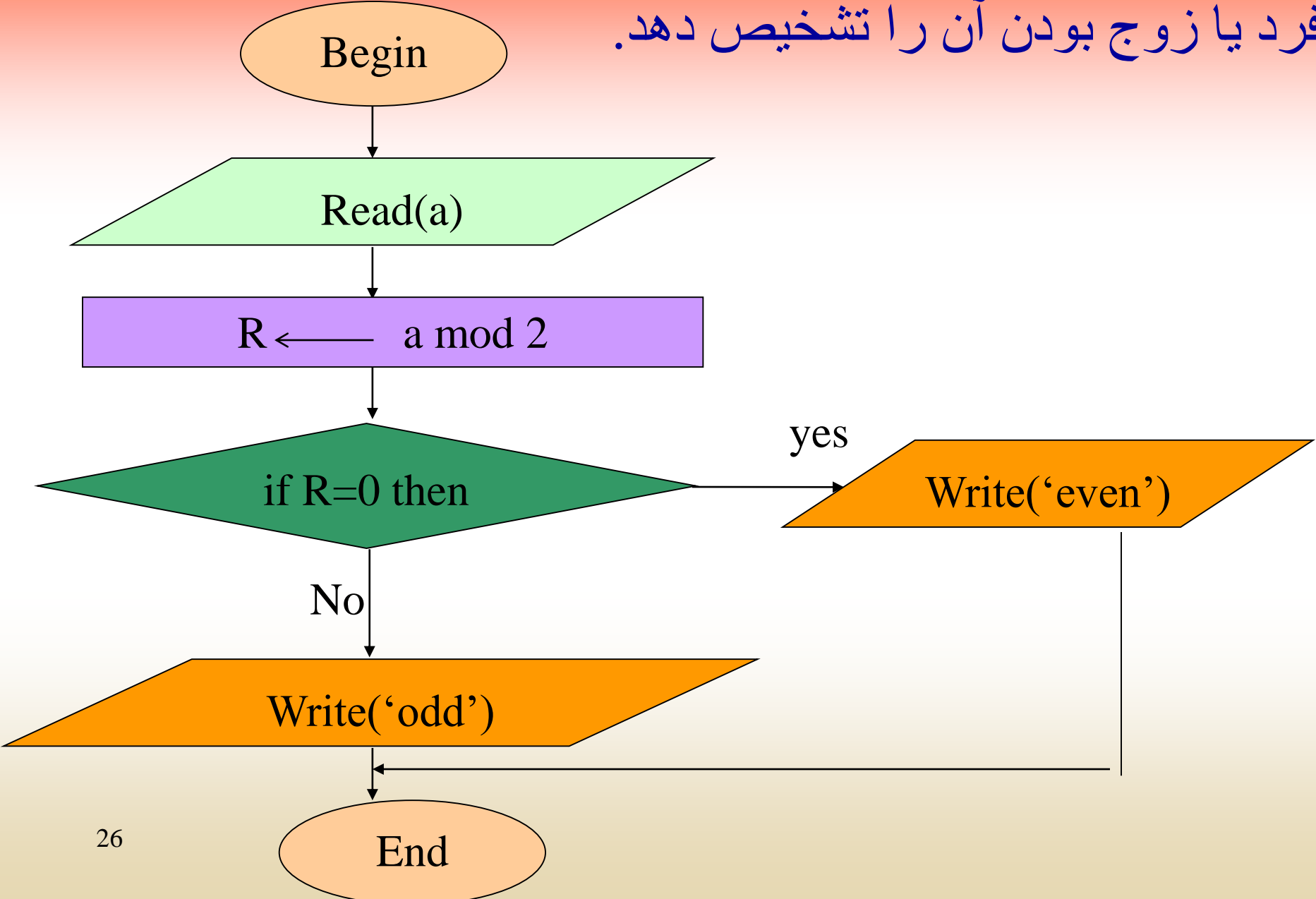
در حل بسیاری از مسائل یا تقریباً تمام مسائل نیاز به استفاده از شروط جزء، نیازهای اساسی محسوب می‌شود. همانطور که ما خودمان در زندگی روزمره با این شرطها سرکار داریم. بطور مثال اگر هوا ابری باشد ممکن است چنین سخن بگوییم:

- اگر هوا بارانی باشد سپس چتری برمی‌دارم.
در غیر اینصورت چتر برنمی‌دارم.

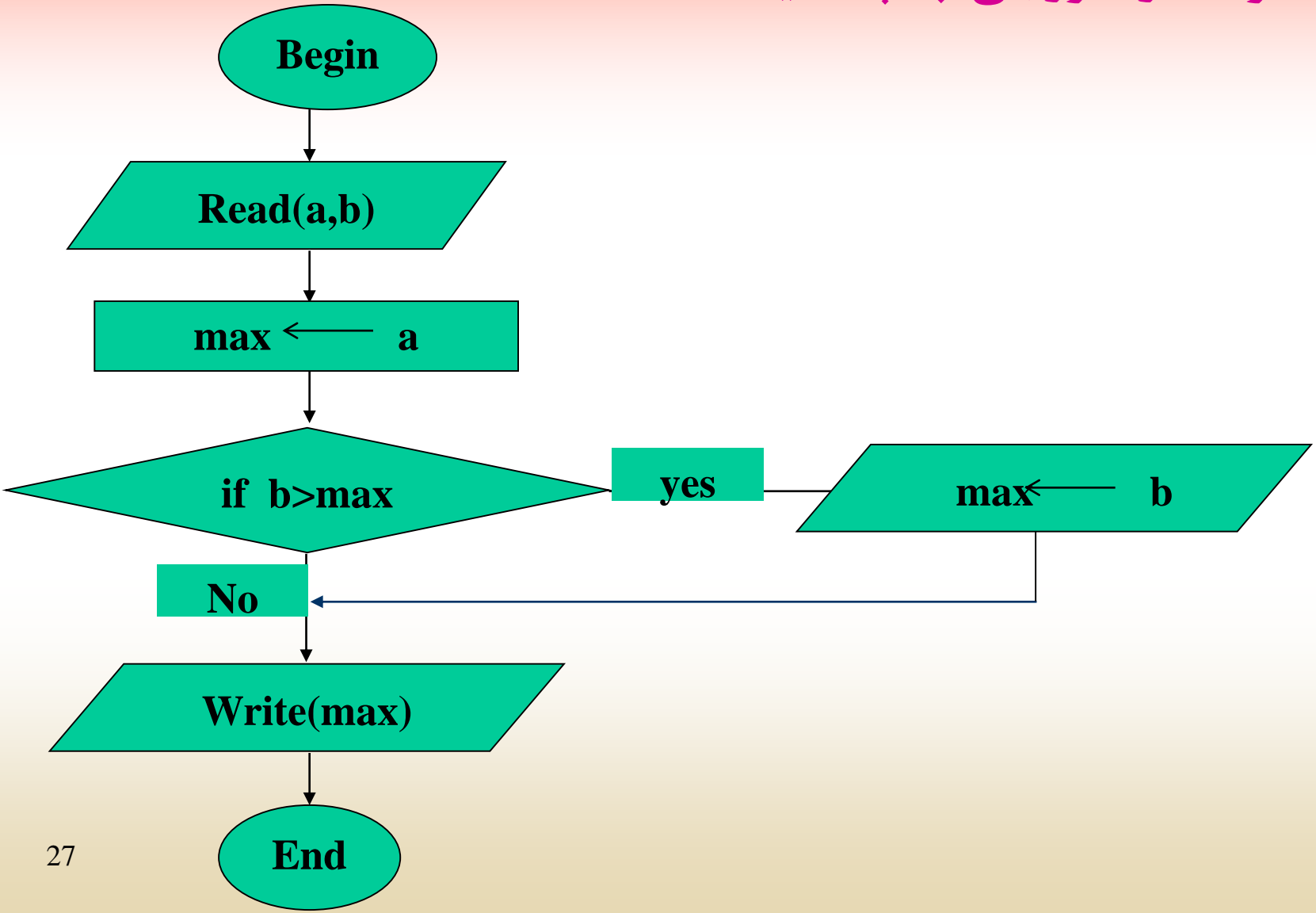
در حالت کلی شرط را بصورت زیر نمایش می‌دهند:



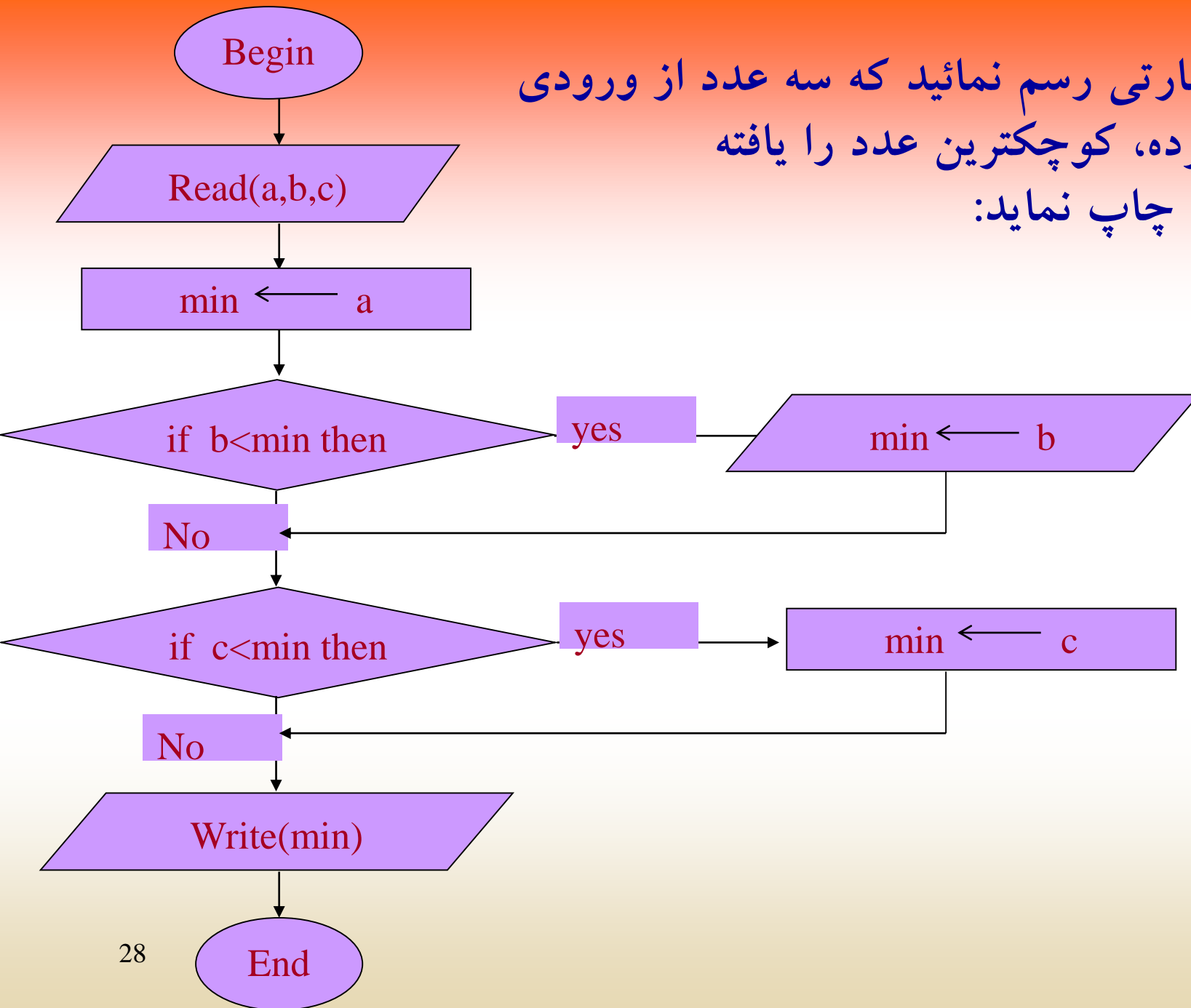
مثال : فلوچارتي رسم نمائيد که عددي را از ورودی دریافت کرده، فرد یا زوج بودن آن را تشخیص دهد.



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:



نمونه اجراي فلوچارت بالا بصورت زیر می باشد:

	a	b	c	Min	خروج
1	12	11	17		11
2				12	
3				11	
4				11	
5				11	

❖ تمرین

- 1- فلوجارتي رسم کنید که عددي را از ورودي دریافت کرده، قدر مطلق عدد را در خروجي چاپ کند.
- 2- فلوجارتي رسم نمائید که عددي از ورودي دریافت کرده مثبت، منفي یا صفر بودن عدد را تشخیص داده، در خروجي با پیغام مناسب چاپ کند.
- 3- فلوجارتي رسم نمائید که عددي را از ورودي دریافت کرده، بخشپذيري آن بر 3 و 5 را بررسی نماید.
- 4- فلوجارتي رسم نمائید که ضرایب يك معادله درجه دوم را از ورودي دریافت کرده، ریشه‌هاي آن را محاسبه در خروجي چاپ کند.

❖ حلقه‌ها

در حل بسیاری از مسائل با عملیاتی روبرو می‌شویم ، که نیاز به تکرار دارند و عمل تکرار آنها به تعداد مشخصی انجام می‌گیرد. فرض کنید، بخواهیم میانگین 100 عدد را محاسبه کنیم، در اینصورت منطقی بنظر نمی‌رسد که 100 متغیر مختلف را از ورودی دریافت کنیم سپس آنها را جمع کنیم.

• انواع حلقه ها

• حلقه های با تکرار مشخص

• حلقه های با تکرار نامشخص

• حلقه های با تکرار مشخص

در این نوع حلقه ها تعداد تکرار مشخص می باشد این حلقه از اجزاء زیر تشکیل می شود:

1- اندیس حلقه

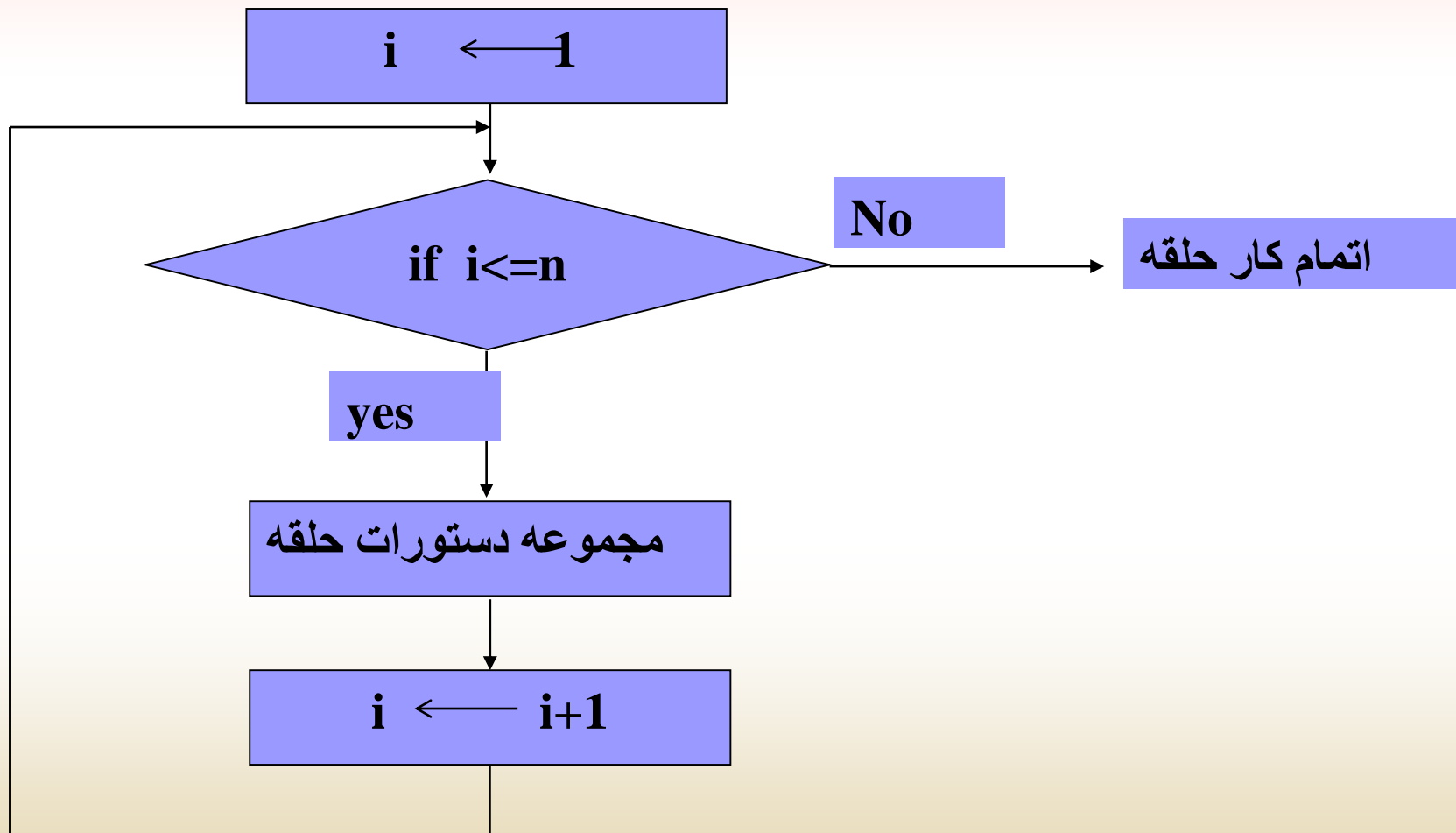
2- مقدار اولیه برای اندیس حلقه

3- مقدار افزایش برای اندیس حلقه (معمولاً يك واحد در هر مرحله)

4- مقدار نهایی (تعداد تکرار حلقه)

5- شرطی برای کنترل تعداد تکرار حلقه

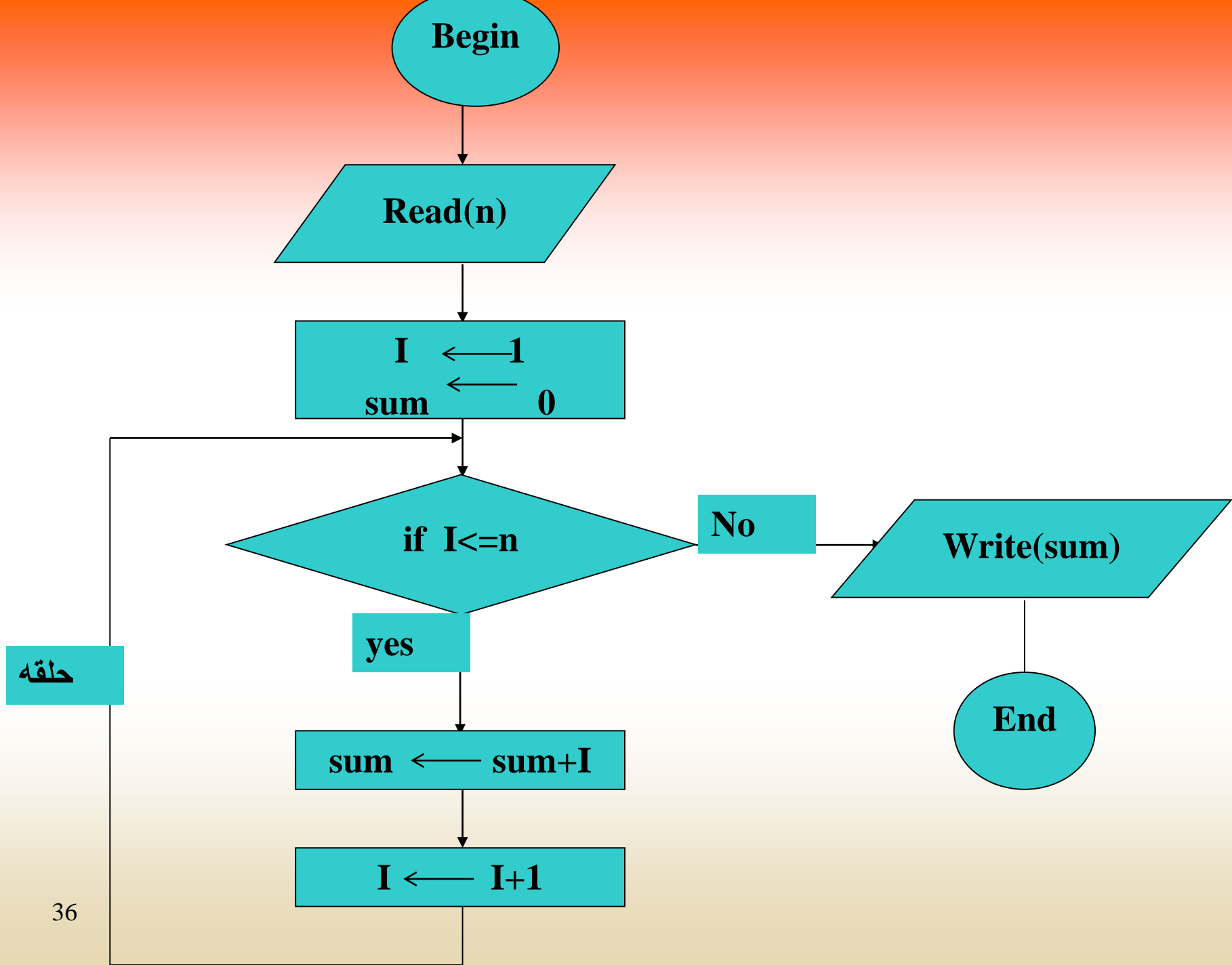
این حلقه‌ها را غالباً با فلوجارت بصورت زیر نمایش می‌دهند:



مثال : فلوچارتي رسم نمائيد كه عدد n را از ورودي دريافت کرده،
مجموع اعداد از يك تا n را محاسبه کند.

n مقدار نهایی

i اندیس حلقه



نمونه اجراي فلوچارت بالا بصورت زير است:

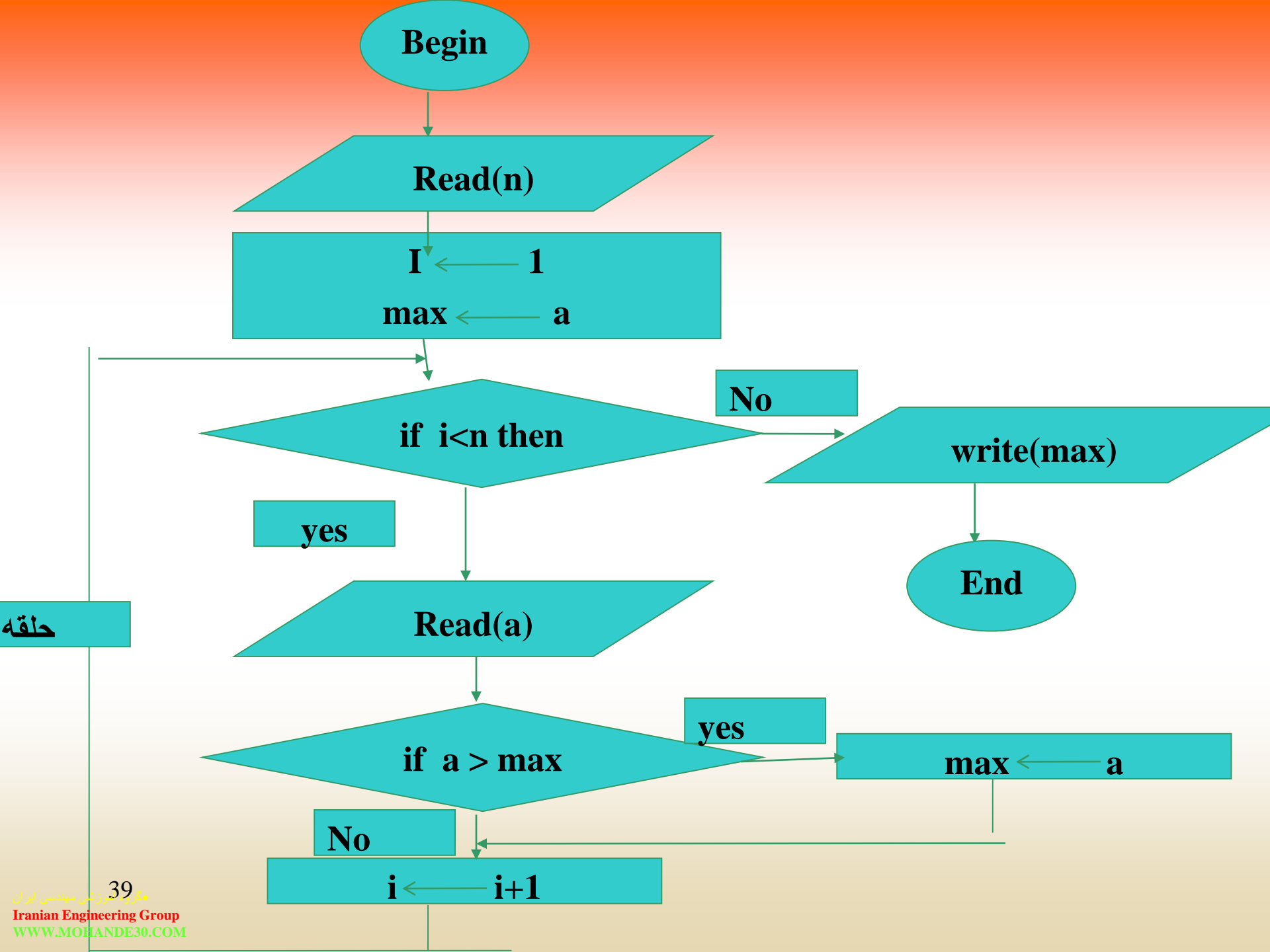
	N	I	sum	خروجي
1	5	1	1	15
2		2	3	
3		3	6	
4		4	10	
5		5	15	

مثال : فلوچارتي رسم ڪنيد ڪه n عدد از ورودي دريافت ڪرده،
بزرگترين مقدار از بين n عدد را پيدا ڪرده در خروجي چاپ نمايد.

i انديس حلقه

n مقدار نهايي

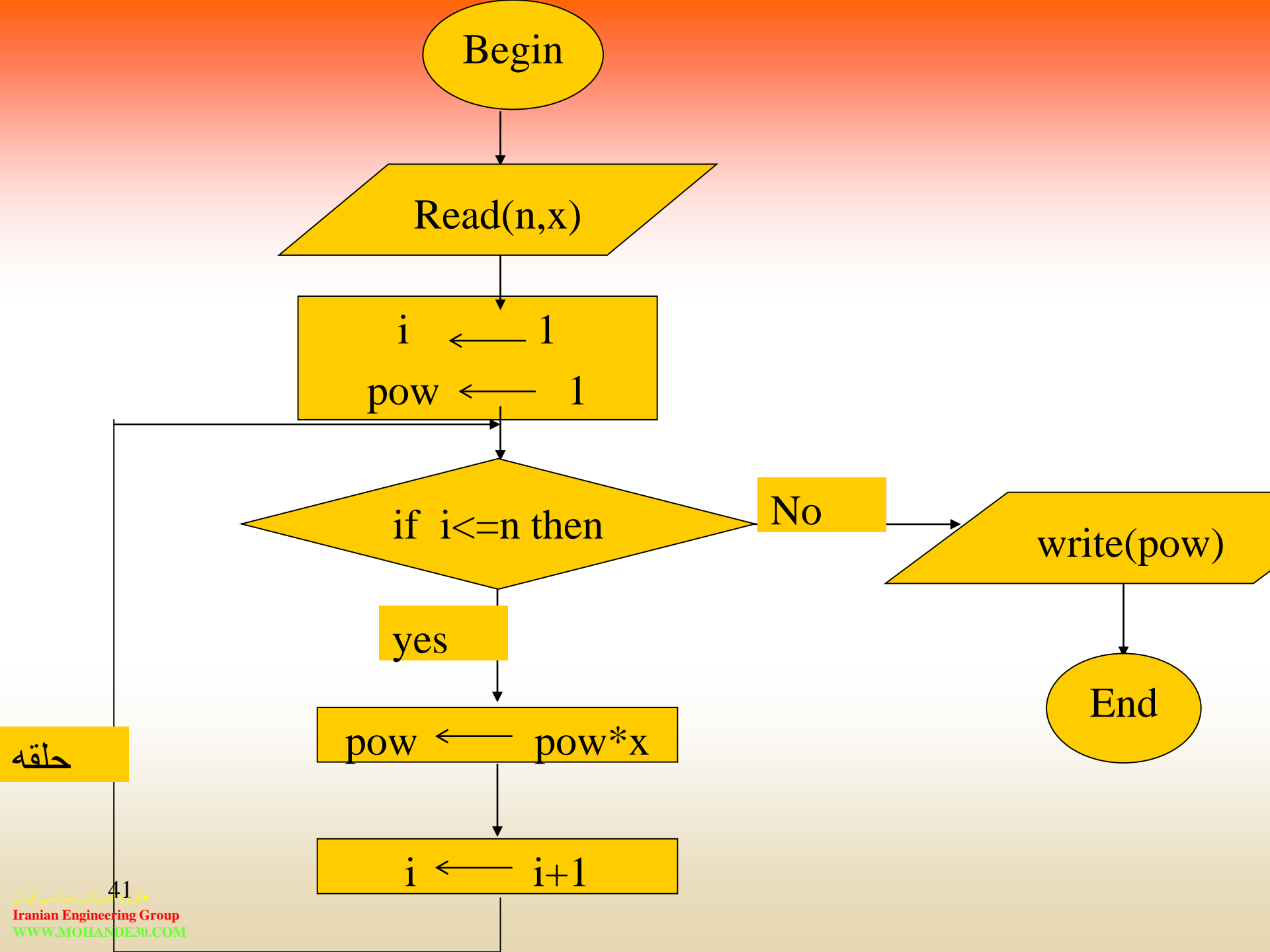
Max بزرگترين مقدار



حلقه

مثال : فلوچارتي رسم نمائيد كه x , n ، دو عدد صحيح مثبت را از ورودی دریافت کرده سپس x به توان n را محاسبه کند.

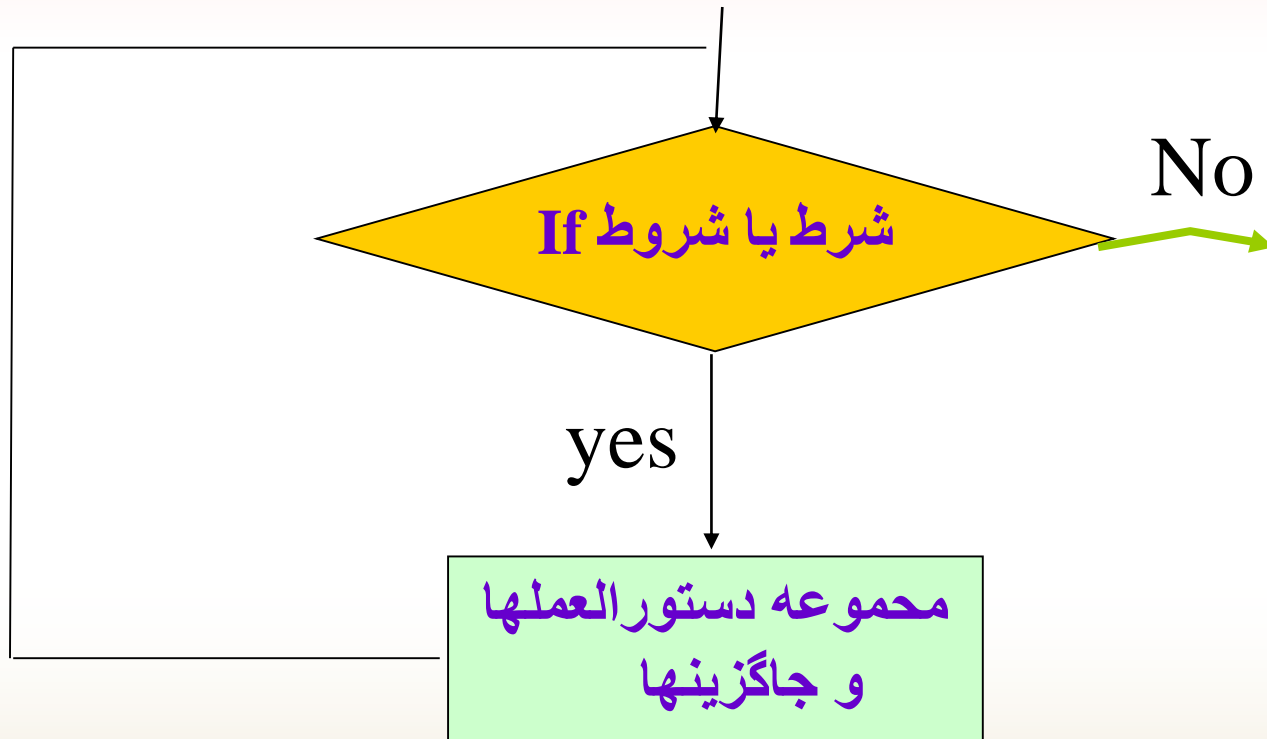
i	• اندیس حلقه
n	• مقدار نهایی
pow	• عدد به توان n



- حلقه‌هایی که تعداد تکرار آنها مشخص نیست (به حلقه while مشهورند).

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 هستند که تا زمانی که برقرار باشد حلقه اجرا می‌شود.

در حالت کلی این نوع حلقه‌ها بصورت زیر نمایش داده می‌شوند:



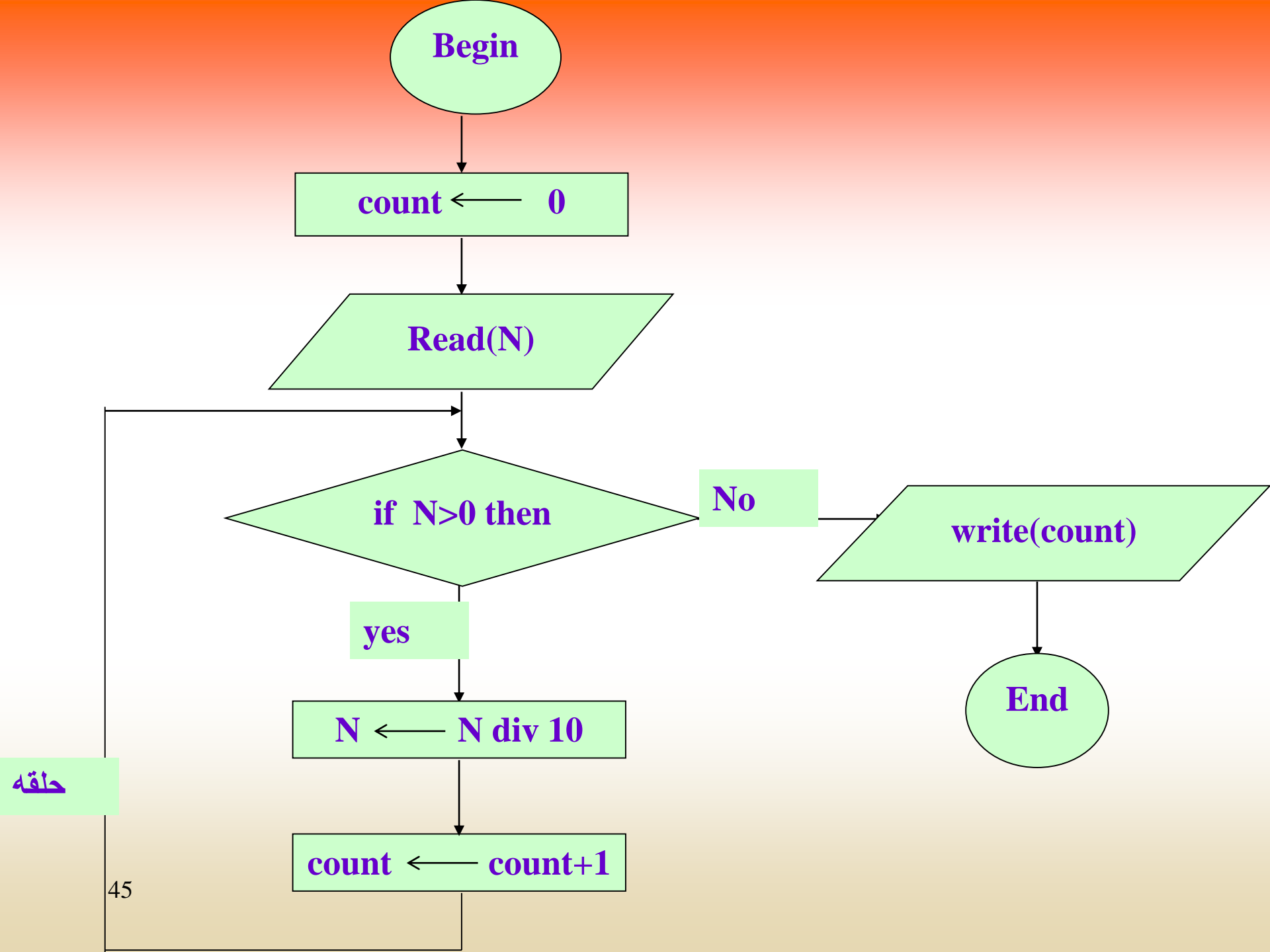
مثال: فلوجارتی رسم کنید که عددی را از ورودی دریافت کرده سپس تعداد ارقام آن را شمرده در خروجی چاپ نماید.

N

• عدد خوانده شده

count

• تعداد ارقام



❖ تمرین

1- فلوجارتي رسم نمائيد که عددي از ورودی دریافت کرده، کامل بودن آنرا بررسی نماید. (عدد کامل، عددی است که مجموع مقسوم علیه‌های آن با خودش برابر باشد. همچنین به‌طور هم ارز، یک عدد کامل، عددی است که نصف مجموع همه مقسوم علیه‌های خود باشد)

2- فلوجارتي رسم نمائيد که دو عدد M, N را از ورودی خوانده، بزرگترین مقسوم علیه مشترك دو عدد را محاسبه و چاپ کند.

6, 28

❖ حلقه‌های تودرتو

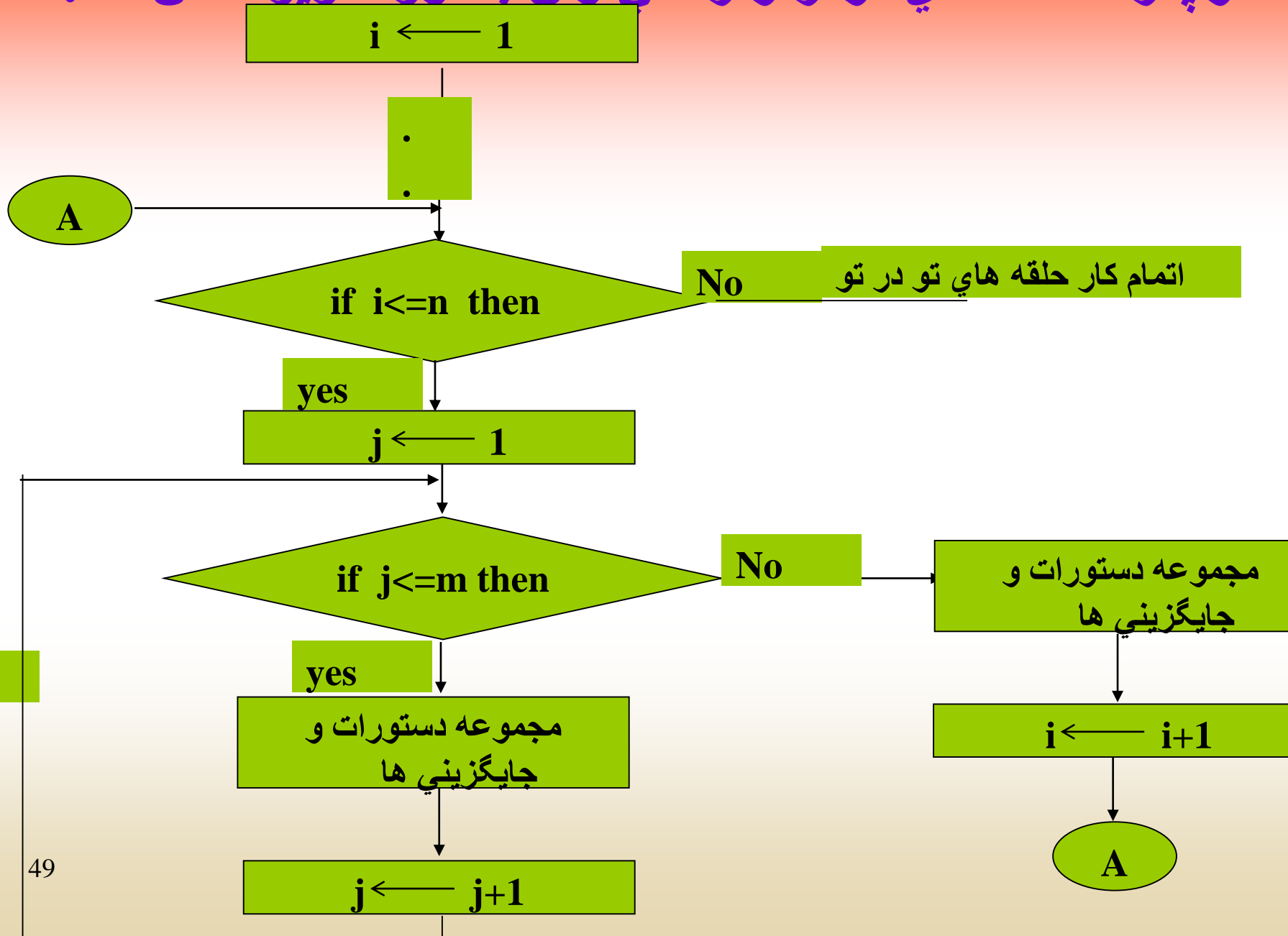
الگوریتم‌هایی که تا حال بکار بردیم، فقط شامل يك حلقه بودند. در صورتی که در بسیاری از مسائل ممکن است نیاز به استفاده از چند حلقه در داخل هم باشیم. در این نوع حلقه‌ها باید دقت بیشتری به خرج دهیم، تا مشکلی پیش نیاید. اگر از حلقه‌های نوع اول بصورت تودرتو استفاده کنیم در اینصورت برای هر حلقه شرط‌نهایی و اندیس اولیه جداگانه باید تعریف کنیم.

در حلقه‌های تو در تو 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 تو در تو را می‌توان بصورت زیر نشان داد:

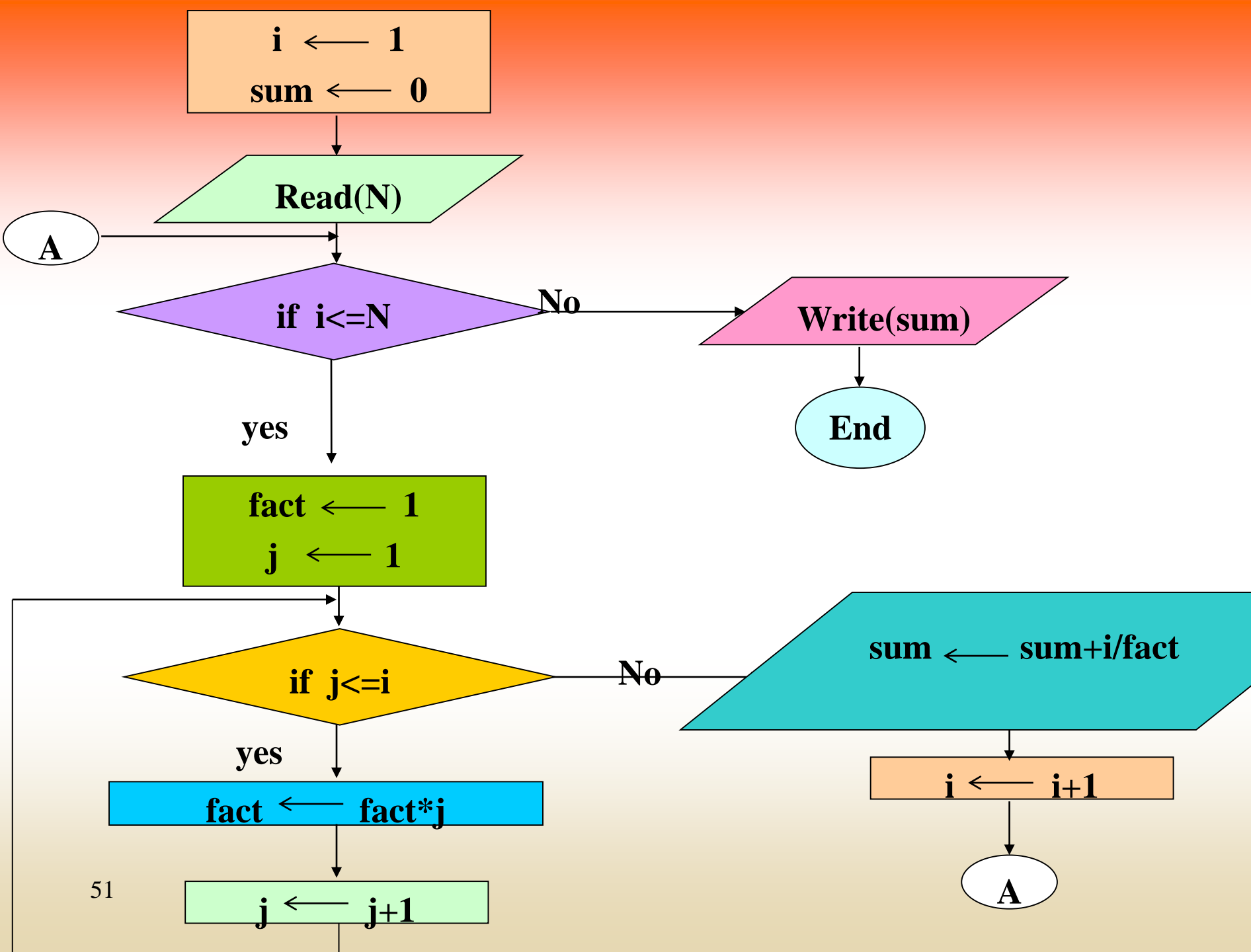


حلقه

مثال : فلوچارتي رسم نمائيد که N را از ورودي دريافت کرده، مجموع سري زیر را محاسبه نمايد:

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

I	• اندیس حلقه اول
N	• ورودي
fact	• محاسبه فاکتوریل
j	• اندیس حلقه داخلي
Sum	• مجموع



❖ تمرینات آخر فصل

1- فلوجارتي رسم نمائید که N عدد از ورودی دریافت کرده تعداد اعداد اول و کامل را شمرده در خروجی چاپ نماید.

2- فلوجارتي رسم نمائید که N, X را از ورودی خوانده مقدار سری زیر را محاسبه کند:

$$S = 1 + \frac{x^2}{2!} + \frac{x^4}{4!} + \dots + \frac{x^n}{N!}$$

3- فلوجارتي رسم نمائيد که عددي را از ورودي دريافت کرده مقلوب عدد را محاسبه و در خروجي چاپ کند.

4- فلوجارتي رسم کنید که تاريخ تولد شخصي را از ورودي خوانده، سن شخص را با تاريخ روز، محاسبه نموده در خروجي چاپ کند.