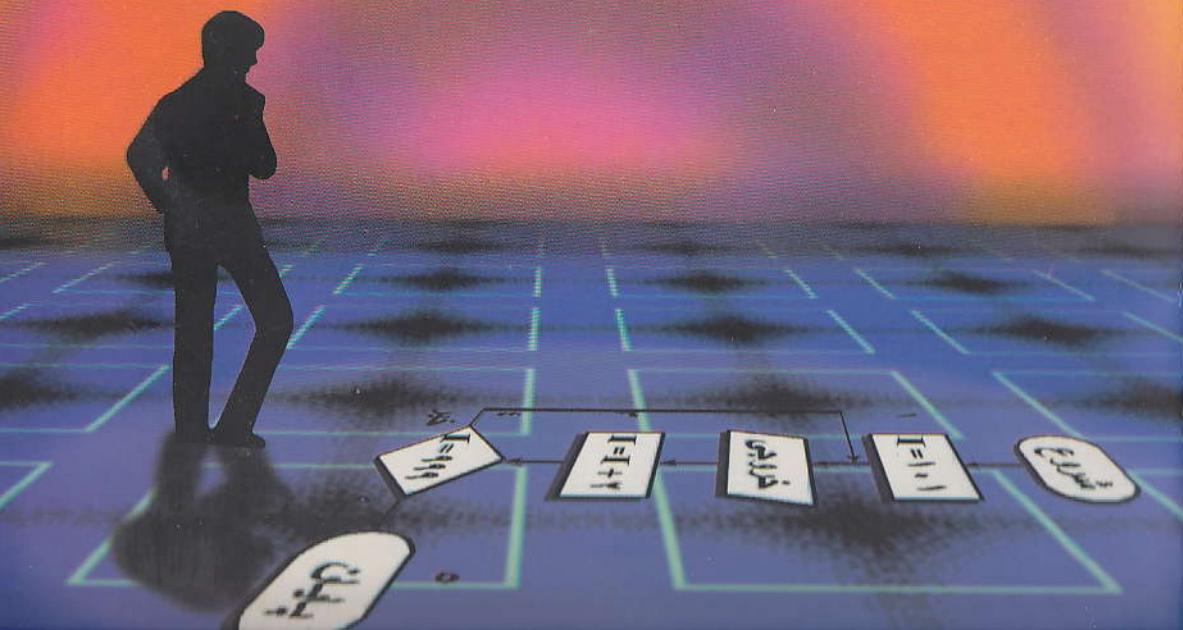




KETABZ.COM
کتابز - مرجع نسخه چاپی

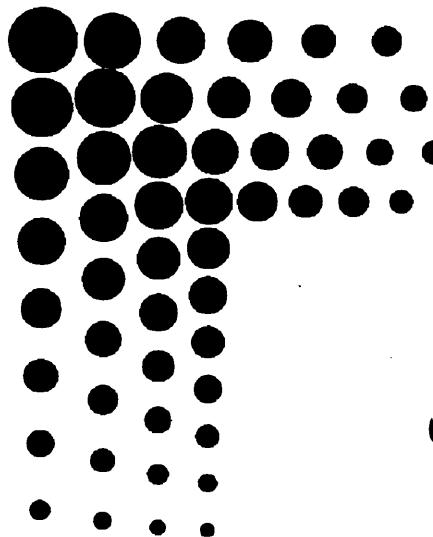
الکوریتم و فلوچارت





ISBN 964-6761-19-4





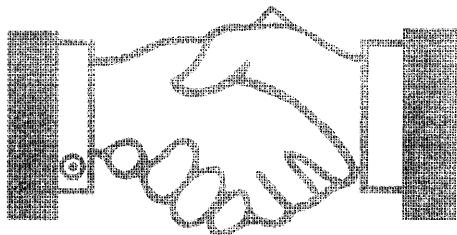
به کتاب

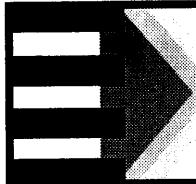
الگوریتم و فلوچارت

خوشن آمدید

"الگوریتم" و "فلوچارت" به عنوان دو مبحث بنیادین در امر برنامه‌نویسی از اهمیت ویژه‌ای برخوردارند. الگوریتم، همان راه حل مسئله و فلوچارت شیوه نمایش نموداری الگوریتم است که برای

درک بهتر و سریعتر الگوریتم ارایه می‌شود و در واقع وجه مشترک بین برنامه‌نویسان کلیه زبانهای کامپیوتراست. لذا با توجه به اهمیت این دو مقوله، در این کتاب ضمن فراگیری روش‌های ارایه الگوریتم‌های مناسب برای حل مسائل کوناکون و بررسی فلوچارت‌های مربوطه، با مفاهیم اساسی برنامه‌نویسی از قبیل: حلقه‌های تکرار، آرایه‌ها، برنامه‌های فرعی (زیر الگوریتم‌ها) روش‌های مرتب نمودن اطلاعات و مواردی از این قبیل آشنا خواهید شد. با توجه به مطالب فوق و تنوع مسائل و مثالهای ارایه شده در کتاب حاضر، مطالعه و به کارگیری این اثر را به کلیه برنامه‌نویسان مبتدی و مدرسین گرامی توصیه می‌نماییم.





هرگونه چاپ و تکثیر از محتویات این کتاب بدون اجازه کتبی
ناشر ممنوع است. متخلفان به موجب قانون حمایت حقوق
مؤلفان، مصنفان و هنرمندان تحت پیگرد قانونی قرار می‌گیرند.

الگوریتم و فلوچارت

مؤلفان : بهرام غلامی - علیرضا جباریه

ناشر : مؤسسه فرهنگی هنری دبیاگران تهران

حروفچینی و صفحه‌آرایی : مجتمع فنی تهران

طرح روی جلد : مجتمع فنی تهران

چاپ : سعیدنو

تاریخ چاپ اول : تیر ماه ۱۳۷۷

تاریخ چاپ هشتم : بهمن ماه ۱۳۸۴

تیراز : ۳۰۰۰ نسخه

قیمت : ۲۱۰۰ ریال

شابک : ۹۶۴-۶۷۶۱-۱۹

ISBN: 964-6761-19-4

غلامی، بهرام، ۱۳۵۱-

الگوریتم و فلوچارت / مؤلفین: بهرام غلامی، علیرضا جباریه، --.

تهران: مؤسسه فرهنگی هنری دبیاگران تهران، ۱۳۷۷.

ص: مصور، نمودار. ۲۲۵

ISBN 964-6761-19-4

فهرستنويسي براساس اطلاعات فيبا.

چاپ قبلی: مجتمع فنی تهران، ۱۳۷۷

چاپ هشتم: ۱۳۸۴

۱. الگوریتم‌های کامپیوتری. ۲. نمودار جریان کار. ۳. بیرنامه‌نویسی

الف. جباریه، علیرضا، ۱۳۴۷- ب. عنوان.

۰۰۵/۱۲۰۲۸

QA۷۶/۹/۷۷۸

۱۳۷۷

م۷۷-۱۵۶۲۳

كتابخانه ملی ايران

آدرس: سعادت آباد، میدان کاج . خ سرو شرقی. رویه روی خ علامه . ساختمان شماره ۹۷

تلفن: ۰۷-۱۴۳۳۵/۹۴۳

تلفن: ۰۷-۲۲۰۹۸۴۴۶

فهرست مطالب

فصل اول : الگوریتم

۹	چگونگی حل مسائل
۱۱	توضیح تاریخی الگوریتم
۱۱	تعريف الگوریتم
۱۲	تعريف الگوریتم از دیدگاه ماشین
۱۴	تعريف تابع
۱۹	الگوریتم محاسبه میانگین سه عدد
۲۲	انواع جملات در الگوریتم نویسی
۲۷	تعريف شمارنده
۲۹	الگوریتم محاسبه بزرگترین عدد از بین سه عدد مفروض
۳۴	الگوریتم محاسبه تام بودن یک عدد طبیعی
۳۴	الگوریتم محاسبه اول بودن یک عدد طبیعی
۳۵	الگوریتم محاسبه فاکتوریل یک عدد طبیعی
۳۹	الگوریتم محاسبه «ب . م» و «ک . م» دو عدد
۳۹	طریقہ جابجا کردن محتویات دو متغیر
۴۰	الگوریتم محاسبه تعداد ارقام و مجموع ارقام یک عدد طبیعی
۴۱	الگوریتم تبدیل عدد از مبنای ۲ به مبنای ۱۰

فصل دوم : تعریف فلوچارت

۵۵	علامیم فلوچارت
۵۵	تعریف فلوچارت
۵۶	علامت شروع و پایان
۵۶	علامت جایگزینی و انتساب
۵۷	علامت ورودی
۵۷	علامت شرطی
۵۷	علامت چاپ
۵۸	علامت حلقه
۵۹	فلوچارت جمع دو عدد A و B
۶۰	فلوچارت محاسبه مقدار یک تابع چند ضابطه‌ای
۶۲	فلوچارت حل معادله درجه اول یک مجھولی
۶۳	فلوچارت حل معادله درجه دوم یک مجھولی
۶۴	فلوچارت محاسبه مقسوم علیه‌ها ، تعداد و مجموع مقسوم علیه‌های یک عدد طبیعی
۷۳	فلوچارت محاسبه اول بودن یک عدد طبیعی
۷۶	اصل تجزیه
۸۰	فلوچارت محاسبه عدد e با تقریب دلخواه
۸۱	فلوچارت محاسبه کسینوس یک زاویه برحسب بسط تیلور آن حول نقطه صفر
۸۳	تعریف دنباله
۸۴	فلوچارت تولید اعداد دنباله فیبوناچی

۸۷	تعريف فایل
۸۸	فلوچارت تعیین اسم روز n هفته
۹۵	فلوچارت محاسبه اولین و دومین بزرگترین عدد از بین n عدد
۱۰۳	فلوچارت محاسبه نام روز n سال با توجه به نام روز اول سال
۱۰۴	فلوچارت محاسبه مقلوب یک عدد n رقمی

فصل سوم : آرایه‌ها

۱۲۸	الگوریتم تبدیل عدد از مبنای ۱۰ به ۲
۱۲۹	تعريف اندیس
۱۲۹	تعريف متغیر اندیس دار
۱۳۶	فلوچارت محاسبه جمع ، تفریق و ضرب دو بردار (لیست) n بعدی
۱۴۰	فلوچارت تبدیل عدد از مبنای ۱۰ به مبنای ۱۶
۱۴۲	تعريف لیست
۱۴۹	فلوچارت محاسبه تعداد عناصر تکراری در یک لیست n عنصر
۱۵۲	فلوچارت حذف عناصر تکراری از یک لیست n عنصری
۱۵۵	فلوچارت یافتن بزرگترین عدد در میان یک لیست n عنصری
۱۵۸	فلوچارت مرتب کردن یک لیست n عنصری به ترتیب صعودی
۱۶۰	فلوچارت محاسبه عناصرهای مشترک یک لیست n عنصری و یک لیست m عنصری
۱۶۱	فلوچارت ادغام دو لیست N_1 و N_2 عنصری
۱۶۷	تعريف ماتریس
۱۶۷	فلوچارت محاسبه پر کردن خانه‌های یک ماتریس
۱۷۱	فلوچارت محاسبه یک ماتریس بالا مثلثی
۱۷۴	فلوچارت جمع و تفریق دو ماتریس $m \times n$
۱۷۷	فلوچارت محاسبه حاصلضرب دو ماتریس
۱۸۲	فلوچارت محاسبه ترانهاده یک ماتریس $m \times n$

فصل چهارم : مرتب کردن و جستجوی اطلاعات

۱۹۰	روشهای جستجو
۱۹۰	روش جستجوی خطی
۱۹۰	روش جستجوی دو دویی
۱۹۱	الگوریتم بدست آوردن موقعیت یک عدد (از نظر وجود داشتن یا نداشتن) در یک لیست
۱۹۳	الگوریتم محاسبه مبلغ جریمه برای ماشین
۱۹۵	الگوریتم جستجوی یک عنصر از میان یک لیست n عنصری مرتب شده
۱۹۷	روشهای مرتب سازی
۱۹۷	مرتب سازی حبابی
۱۹۸	الگوریتم روش مرتب سازی حبابی
۱۹۹	الگوریتم روش مرتب سازی حبابی با استفاده از سویچ

۲۰۰	روش مرتب سازی جایگزینی
۲۰۰	الگوریتم روش مرتب سازی جایگزینی
۲۰۱	روش مرتب سازی SHELL
۲۰۲	الگوریتم روش مرتب سازی SHELL

فصل پنجم: زیر الگوریتم‌ها

۲۰۸	تعريف زیر الگوریتم
۲۰۸	الگوریتم محاسبه ترکیبات m عضوی n عنصر
۲۰۹	تعريف فاکتوریل یک عدد طبیعی
۲۱۰	مشخصات یک زیر الگوریتم
۲۱۲	الگوریتم محاسبه تام بودن اعداد با استفاده از زیر الگوریتم
۲۱۳	الگوریتم محاسبه سینوس یک زاویه بر حسب بسط تیلور آن با استفاده از زیر الگوریتم
۲۱۵	الگوریتم محاسبه n سطر از مثلث خیام - پاسکال با استفاده از زیر الگوریتم
۲۱۸	الگوریتم محاسبه تعداد اعداد اول دو رقمی با استفاده از زیر الگوریتم
۲۱۹	الگوریتم محاسبه مجموع ارقام یک عدد با استفاده از زیر الگوریتم

مقدمه مؤلفین

بشر در طول زندگی خود همواره با مسائلی روبرو بوده که هر کدام را به شیوه‌ای حل نموده است . برای حل ساده‌ترین مسائل تا پیچیده‌ترین آنها ، علاوه بر روش‌های خاص ، از ابزار خاصی نیز استفاده کرده است . حال اگر نگاهی به عصر حاضر بیندازیم می‌بینیم که امروزه بشر بیشتر مسائل را به وسیله کامپیوتر حل می‌کند و بطور کلی لازمه پیشرفت در هر زمینه علمی استفاده از کامپیوتر است . اما چگونه این ماشین می‌تواند طیف گسترده‌ای از مسائل را حل نماید ؟

این سؤالی است که اغلب افراد در برخورد اولیه با کامپیوتر با آن مواجه می‌شوند . در پاسخ به این سؤال باید گفت که کامپیوتر براساس روش مرحله به مرحله حل یک مسئله ، که به آن الگوریتم گفته می‌شود ، مسائل را حل می‌کند . لذا برای حل یک مسئله توسط کامپیوتر ، باید آن را به شیوه مرحله به مرحله بیان نماییم . کتاب حاضر چگونگی حل یک مسئله و سپس بیان مرحله به مرحله آن را به شما آموزش می‌دهد تا با استفاده از روندهای بیان شده در آن بتوانید هرنوع مسئله‌ای را حل نمایید . توصیه ما به شما دانش پژوهان عزیز آن است که مطالب این کتاب را بطور دقیق مطالعه و درک نمایید ، چرا که پایه حل هر مسئله‌ای توسط کامپیوتر نوشتن الگوریتم آن است . «دونالد کنوت» یکی از دانشمندان بزرگ علم کامپیوتر می‌گوید : «علم کامپیوتر چیزی نیست بجز مطالعه الگوریتم‌ها » .

این کتاب حاصل تجربه سالها تدریس مؤلفین در درس الگوریتم و فلوچارت می‌باشد و در نگارش آن دقت فراوانی مبذول گشته است . لذا از تمامی اساتید ، همکاران و دست‌اندرکاران گرامی خواهشمندیم که بر ما منت نهاده و از راهنماییهای خود در جهت رفع عیوب کتاب دریغ ننمایند . همچنین از جناب آقای مهندس سعید سعادت مدیرعامل محترم مجتمع فنی تهران ، که راهنماییهایشان در جهت تألیف این اثر بسیار مفید و مؤثر بوده است و سایر عزیزانی که به نحوی مارا یاری نموده‌اند صمیمانه سپاسگزاریم .

تهران – اردیبهشت ۷۷

بهرام غلامی – علیرضا جباریه

فصل اول

چگونگی حل مسائل
و
الگوریتم

(

این فصل با بررسی روش‌های حل یک مسئله و مفهوم الگوریتم از دیدگاه‌های مختلف آغاز می‌شود. این بررسی و به دنبال آن مهارتی که در الگوریتم سازی کسب می‌کنید، کمک مؤثری در برنامه نویسی به شما خواهد کرد. مفهوم الگوریتم نویسی برای حل یک مسئله مطالب جدیدی برای خوانندگان به شمار می‌رود و عموماً در ابتدا مشکل بنظر می‌رسد ولی اطمینان داشته باشید که با مطالعه دقیق این فصل و حل تمرینات مربوط به آن، تا حدودی با نوشتمن الگوریتم های ساده آشنا خواهید شد.

اهداف کلی :

- ۱- آشنایی با چگونگی حل یک مسئله اعم از روش، استراتژی، تعمیم و ...
- ۲- آشنایی با مفهوم الگوریتم و الگوریتم سازی.
- ۳- کسب مهارت در نوشتمن الگوریتم.

هدفهای رفتاری:

دانشجو پس از مطالعه این فصل باید بتواند :

- ۱- مراحل مختلف حل یک مسئله را به طور کامل تشریح کند.
- ۲- برای حل یک مسئله راه حل های مختلفی را به طور دقیق مشخص کند.
- ۳- مفهوم الگوریتم را یادگرفته و تعریف دقیقی از آن ارایه دهد.
- ۴- مفهوم برداشت یگانه در مورد یک دستورالعمل را توضیح دهد.
- ۵- برای بعضی از مسائل روزمره الگوریتم بنویسد.
- ۶- منظور از مجری الگوریتم را بیان کند.
- ۷- برای بیان مسائل مختلف الگوریتم نوشته و آنها را آزمایش نماید.

چگونگی حل مسائل :

عموماً برای حل مسائل نمی‌توان یک راه حل کلی و عمومی ارایه داد. یعنی برای حل هر مسئله باید از شیوه‌ها و رهیافت‌های خاصی استفاده کرد که انتخاب شیوه‌های مناسب برای حل مسئله بستگی به ابتکار، خلاقیت و تجربه هر شخص دارد.

به طور کلی پیرامون اتخاذ روش‌های مناسب جهت حل مسائل سه عامل مهم زیر را می‌توان در نظر گرفت:

- ۱- شناخت دقیق مسئله

- ۲- طرح نقشه حل مسئله
۳- آنالیز(تحلیل) کردن مسئله

۱- شناخت مسئله :

در برخورد اولیه بایک مسئله باستی تمام عوامل زیر را با دقت مورد بررسی قرار داد:

- ۱-۱- داده‌ها : در یک مسئله منظور از داده‌ها ، فرض‌ها و نیز نتایجی است که باید از داده‌ها حاصل شود.

۱-۲- مجھولها : منظور از مجھولها مقادیری هست که مسئله در جستجوی آن می‌باشد.

- ۱-۳- ارتباط بین داده‌ها و مجھولها : منظور از ارتباط ، رابطه منطقی است که توسط آن می‌توان از داده‌ها به مجھولات دست یافت که در این میان از ابزارهای ریاضی می‌توان استفاده نمود.
اکنون با ذکر یک مثال عوامل فوق توضیح داده می‌شوند.

مثال (۱) فرض کنید می‌خواهیم میانگین دو عدد ۱۰ و ۲۰ را محاسبه کنیم.

الف) داده‌ها : دو عدد ۱۰ و ۲۰ .

ب) مجھول : میانگین دو عدد ۱۰ و ۲۰ .

ج) ارتباط بین داده‌ها و مجھول : فرمول میانگین دو عدد یعنی مجموع آنها تقسیم بر ۲ .

۲- طرح نقشه حل مسئله : (ارایه راه حل)

پس از شناختن مسئله باید برای حل آن (بدست آوردن مجھولات) نقشه ای طرح کرد. به طور کلی انسان برای حل مسائل به دو صورت منطقی و غیرمنطقی (الگوریتمی و غیرالگوریتمی) عمل می‌کند.

در روش غیرمنطقی برای حل مسائل از تفکر جانبی استفاده می‌شود که در قسمت بعد تعریف آن ارایه شده است.

تفکر جانبی: نمونه ای تفکر است که از مجموعه راه حل‌های موجود، ساده‌ترین روش را با شیوه ای نامتعارف بر می‌گزیند .

در روش منطقی از شیوه الگوریتمی استفاده می‌شود قبل از بیان کامل این روش که یکی از قوی ترین روش‌های حل مسئله می‌باشد ابتدا به حل یک مثال به دوروش منطقی و غیرمنطقی می‌پردازیم :

- مثال (۲) ۲۰ شطرنج باز در یک دوره مسابقات یک حذفی (در هر بازی نفر بازنش از دور مسابقات حذف می‌شود). شرکت دارند . معین کنید در این دوره از مسابقات چند بازی انجام گرفته است؟
مراحل زیر را برای حل این مسئله بکار می‌بریم :
شناخت مسئله :

داده‌ها : ۲۰ شطرنج باز و یک حذفی بودن مسابقات

مجھول : تعداد بازیهای انجام شده

ارتباط بین داده‌ها و مجھول : ارایه راه حل، این ارتباط را مشخص می‌کند.

حل به روش غیر منطقی : چون هر بازی شطرنج فقط یک برنده دارد و جمماً ۲۰ نفر شرکت داشته‌اند و

یک نفر برنده و یک نفر بازنده شده‌اند پس نتیجه می‌گیریم که جمماً ۱۹ بازی انجام گرفته است.

حل به روش منطقی : در روش منطقی می‌توان تعداد بازیها را به فرم زیر بدست آورد : ابتدا ۱۰ بازی

صورت می‌گیرد ، سپس ۵ بازی و ...

$$\text{جمع کل بازیهای انجام شده} : ۱۹ = ۱ + ۱ + ۵ + ۲ + ۱ + ۱ + ۱ + ۱ + ۱ \rightarrow ۱ + ۲ + ۵ + ۱۰$$

۳- تحلیل راه حل مسئله :

منظور از تحلیل کردن راه حل مسئله عموماً بررسی و تجزیه راه حل و درنهایت تعمیم دادن آن است. توضیح بیشتر در مورد این قسمت را به بعد موکول می‌کنیم.

مسئله مهمی که در مورد حل مسئله قابل بیان می‌باشد، این است که شناخت و پیدا کردن راه حل مناسب به دلیل پیچیدگی و یا بزرگ بودن و یا ...، به سادگی امکان پذیر نیست. به همین دلیل توصیه می‌کنیم مسئله را به شکل زیر به قسمتهای کوچکتری تجزیه کنید :

الف) در بالاترین سطح یا سطح اول، صورت مسئله را قرار دهید.

ب) در سطح میانی یا سطح دوم، مسئله را به چند زیر مسئله ساده تجزیه کنید.

ج) در سطح آخر یا سطح سوم هر کدام از زیر مسئله‌ها را تک تک بررسی و در صورت پیچیده بودن به زیر مسئله‌های ساده‌تر تجزیه کنید و این عمل را آنقدر ادامه دهید تا دیگر نیازی به تقسیم کردن نباشد.

د) هر کدام از زیر مسئله‌ها را حل کرده و با بهم پیوستن راه حل‌ها، مسئله اصلی را حل کنید.

الگوریتم (Algorithm)

توضیح تاریخی الگوریتم : به احترام ریاضیدان بزرگ ایرانی ابوموسی خوارزمی و روشهای او در حل مسائل به شیوه منطقی به کار برده این روش را الخوارزمی می‌نامیدند که بعد از معرفت شدن این واژه بصورت الگوریتم بکاربرده شده است.

تعريف الگوریتم : به مجموعه‌ای از یک یا چند دستور العمل که اجرای آنها با ترتیب تعیین شده منجر به انجام یک کار به ویژه حل یک مسئله گردد، الگوریتم گفته می‌شود.

به عبارت دیگر مجموعه‌ای از دستورالعملها که با یک ترتیب معین و خاص و همچنین مرحله به مرحله انجام شده و موجب حل مسئله‌ای می‌گردد، الگوریتم گفته می‌شود.

بنابر این ویژگیهای یک الگوریتم عبارتند از :

- ۱- تعداد دستور العمل‌ها مشخص باشد.
- ۲- ابتدا و انتهای داشته باشد (از مرحله‌ای شروع و به مرحله‌ای ختم گردد).
- ۳- هریک از دستورالعملهای آن عاری از هرگونه ابهام و پیچیدگی باشد (برداشتهای متفاوتی از دستورالعملهای آن نتیجه نشود).
- ۴- هریک از دستورالعمل‌ها قابل فهم و قابل اجرا باشد.
- ۵- از اجرای همه آنها به هدف خاص و معینی بررسیم.

تعريف دقیق الگوریتم : به مجموعه ای از دستورالعمل‌ها که مراحل مختلف کاری (حل یک مسئله) را به زبان دقیق و با جزئیات کافی بیان کرده و در آن ترتیب مراحل و خاتمه پذیر بودن عملیات کاملا مشخص باشد، الگوریتم گفته می‌شود.

منظور از زبان دقیق این است که از یک دستور العمل برداشتهای متفاوتی نتیجه گرفته نشده و یا سؤال برانگیز نباشد. پس ویژگی برداشت یگانه در مورد دستورات یک الگوریتم امری ضروری است.

جزئیات کافی : از انجائیکه دستورات الگوریتم را عموماً ماشین اجرا می‌کند ، لذا باید دستورات را با تمامی جزئیات در اختیار ماشین قرار دهیم، بنابراین هر کدام از دستورات بایستی کامل باشند.

ترتیب مراحل : باید ترتیب اجرای دستورالعمل‌ها به طور کامل مشخص شود به همین دلیل برای هریک از آنها شماره‌ای را در نظر می‌گیریم.

خاتمه پذیر بودن : بدین معنی که باید مشخص کنیم که الگوریتم چه زمانی یا تحت چه شرایطی خاتمه پذیرفته و اجرای آن متوقف می‌شود.

نکته مهمی که از تعریف فوق نتیجه می‌شود این است که الگوریتم‌های نوشته شده می‌توانند توسط انسان یا ماشین اجرا شوند و منظور از اجرای یک الگوریتم یعنی اجرای کلیه دستورالعمل‌ها و دنبال کردن آنها ، برای اخذ نتیجه مطلوب است . به همین دلیل برای الگوریتم یک مجری در نظر گرفته و آنرا مجری الگوریتم می‌نامیم. اگر مجری ماشین باشد عموماً به آن کامپیوتر می‌گویند .

به همین دلیل مجدداً تعریف دیگری از الگوریتم را می‌آوریم که به آن از دیدگاه یک ماشین توجه شده است که آن ماشین لزوماً کامپیوتر نیست، بلکه ماشینی کاملاً فرضی است که اعمال محدودی را می‌تواند انجام دهد.

تعريف الگوریتم از دیدگاه ماشین :

الگوریتم، یک ماشین ساده است که قابلیتهای زیر را دارد:

- ۱- دریافت یک یا چند داده به عنوان ورودی.
- ۲- ارسال یک یا چند مقدار به عنوان خروجی.

۳- انجام عملیات مقایسه بین دو داده دلخواه.

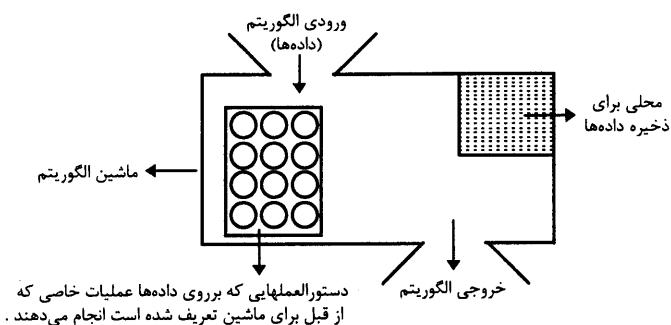
۴- ذخیره داده‌ها و اطلاعات در قسمتی از ماشین.

۵- انجام اعمال ریاضی خاص (از قبیل جمع، تفریق، ضرب، تقسیم و ...) روی داده‌ها.

توجه : بین دو داده عددی همواره یکی از سه حالت زیر برقرار است:

$$\left\{ \begin{array}{ll} a < b & \text{برای دو داده} \\ a = b & \text{یا} \\ a > b & \text{یا} \\ a = 0 & \text{برای یک داده} \\ a > 0 & \text{یا} \\ a < 0 & \text{یا} \end{array} \right.$$

که با توجه به تعریف فوق می‌توانیم شکل زیر را برای ماشین الگوریتم در نظر بگیریم :



توضیح : در مورد هر مسئله‌ای شیوه منطقی در حل کردن آن ، الگوریتمیک فکر کردن درباره آن مسئله است . یعنی ابداع روشی برای حل آن مسئله بوسیله ماشین به شیوه مرحله به مرحله ، که برای رسیدن به این هدف از روش ها و روندهای مختلف در ریاضیات که ابزار بسیار قوی در حل مسائل می‌باشند، استفاده می‌کنیم .

برای فهم دقیق یک مسئله باید از قواعدی که در قسمت‌های قبل توضیح داده شده است ، استفاده کنید یعنی برای مسئله مدل ریاضی خاصی ایجاد و آن را در قالب یک فرمول ریاضی بیان کنید. به همین منظور بعضی از مطالب ریاضی مورد استفاده در این فصل را به عنوان یادآوری ذکر می‌شوند .

مجموعه اعداد طبیعی :

یک مجموعه نامحدود است که از عدد ۱ شروع شده و تا بی‌نهایت ادامه دارد و اختلاف هر عدد از عدد قبل، یک است. و آن را با نماد N نمایش می‌دهیم .

$$N = \{ 1, 2, 3, \dots \}$$

مجموعه اعداد صحیح :

این مجموعه از دو طرف نامحدود است که کلیه اعداد طبیعی و عدد صفر و نیز قرینه آنها را شامل می شود و آن را با نماد Z نمایش می دهیم .

$$Z = \{0, \pm 1, \pm 2, \dots\} \quad \text{یا} \quad Z = \{\dots, -1, 0, +1, \dots\}$$

مجموعه اعداد گویا :

این مجموعه نیز نامحدود بوده و شامل تمام اعداد کسری که صورت و مخرجشان اعداد صحیح و مخرج مخالف صفر داشته باشند، می باشد و آن را با نماد Q نمایش می دهیم .

$$Q = \left\{ \frac{a}{b} \mid b \neq 0, a, b \in \mathbb{Z} \right\}$$

مجموعه اعداد حقیقی :

این مجموعه نیز مجموعه ای نامحدود بوده و شامل اعداد طبیعی، اعداد صحیح، اعداد گویا و همچنین اعداد گنگ (مثل $\sqrt{2}$ و $\sqrt{3}$ و ...) بوده و نماد آن R می باشد.

تعریف تابع

فرض کنید A و B دو مجموعه باشند. هر رابطه ای که اعضاء مجموعه A را به مجموعه B مربوط کند تابعی از A به B نامیده می شود ، به شرط آنکه به ازاء هر عضو از مجموعه A فقط یک عضو از مجموعه B وجود داشته باشد.
برای نمایش یک تابع از نمادهای زیر استفاده می شود.

$$\begin{array}{ccc} f : A \rightarrow B & & \\ \begin{array}{c} \text{ف} \\ \text{A} \xrightarrow{\hspace{1cm}} \text{B} \end{array} & \text{یا} & \\ \begin{array}{c} \text{ف} \\ \text{A} \xrightarrow{\hspace{1cm}} \text{B} \end{array} & \text{یا} & \end{array}$$

تابع جزء صحیح :

جزء صحیح نیز تابعی است از مجموعه اعداد حقیقی به مجموعه اعداد صحیح به طوریکه به هر عدد حقیقی ، بزرگترین عدد صحیح کوچکتر از آن را مربوط می کند .

$$f : R \rightarrow Z$$

اگر بخواهیم جزء صحیح یک عدد را تعریف کنیم آنرا بصورت زیر می نویسیم .

تعریف : فرض کنید x یک عدد حقیقی بوده و بتوان آن را به شکل زیر نوشت :

$$x = n + p$$

$$z = \begin{cases} n & \text{ صحیح متعلق به } \\ 0 \leq p < 1 & \text{ که در این رابطه داریم} \end{cases}$$

آنگاه n را جزء صحیح عدد حقیقی x می‌گوئیم و آنرا با نماد (x) INT یا $[x]$ نمایش می‌دهیم.
برای مثال :

(الف) $[2.3] = 2$

زیرا می‌توان 2.3 را بصورت زیر نوشت :

$$2.3 = 2 + 0.3 \rightarrow [2.3] = 2$$

↓ ↓
متصل به بین صفر و یک

(ب) $[2] = 2$

زیرا می‌توان 2 را بصورت زیر نوشت :

$$2 = 2 + 0 \rightarrow [2] = 2$$

↓ ↓
متصل به بین صفر و یک

پس جزء صحیح اعداد صحیح مساوی خود عدد است .

(ج) $[-1] = -1$

زیرا -1 را می‌توان بصورت زیر نوشت :

$$-1 = -1 + 0 \rightarrow [-1] = -1$$

↓ ↓
متصل به بین صفر و یک

(د) $[-1.3] = -2$

زیرا می‌توان -1.3 را به شکل زیر نوشت :

$$-1.3 = -2 + 0.7 \rightarrow [-1.3] = -2$$

توجه : جزء صحیح اعداد منفی یک واحد کمتر از خود عدد می‌باشد. به عبارت دیگر برای تمام اعداد ، جزء صحیح یک عدد از خود عدد کوچکتر یا مساوی آن است .

توضیحی درباره تقسیم :

قبلًا با مفهوم تقسیم و عوامل آن یعنی مقسوم، مقسوم‌علیه، خارج قسمت و باقیمانده آشنا شده‌اید و برای پیداکردن هر کدام از قاعده مشخصی استفاده می‌کردید، در اینجا همین مطلب را به روش دیگری بیان کرده و طریقه محاسبه خارج قسمت و باقیمانده را بر حسب مقسوم و مقسوم‌علیه بدست می‌آوریم، البته تقسیم را در Z یعنی مجموعه اعداد صحیح بررسی می‌کنیم .

تعریف تقسیم: فرض کنید a و b متعلق به مجموعه اعداد صحیح باشند ($b \neq 0$) ، تقسیم a بر b را به صورت زیر تعریف می کنیم .

$$\frac{a}{b} = q \quad \left| \begin{array}{l} a \text{ مقسوم} \\ b \text{ مقسوم علیه} \\ q \text{ خارج قسمت} \\ R \text{ باقیمانده} \end{array} \right\} \text{که در این رابطه :}$$

که نماد \in بمعنای تعلق داشتن است

زمانی تقسیم پایان یافته است که باقیمانده مساوی صفر یا از مقسوم علیه کوچکتر باشد. پس اگر تقسیم تمام شده باشد، همواره $R < b$ است .

همچنین برای امتحان درستی عمل تقسیم باید رابطه زیر برقرار باشد :

$$a = b \cdot q + R \quad (1)$$

برای بدست آوردن خارج قسمت و باقیمانده از رابطه (2) استفاده می کنیم :

$$R = a - b \cdot q \quad (2)$$

طرفین را برابر b که مخالف صفر است تقسیم می کنیم .

$$\frac{R}{b} = \frac{a - b \cdot q}{b} \Rightarrow \frac{R}{b} = \frac{a}{b} - \frac{b \cdot q}{b} \Rightarrow \frac{R}{b} = \frac{a}{b} - q \quad (3)$$

از طرفین رابطه (3) جزء صحیح می گیریم .

$$\left[\frac{R}{b} \right] = \left[\frac{a}{b} - q \right] \quad (4)$$

چون q صحیح است، از داخل جزء صحیح بیرون می آید.

$$\left[\frac{R}{b} \right] = \left[\frac{a}{b} \right] - q \quad (5)$$

وچون $R < b$ است لذا $\frac{R}{b} < 1$ ، همچنین جزء صحیح اعداد بین صفر و یک همواره مساوی صفر

است پس خواهیم داشت :

$$0 = \left[\frac{a}{b} \right] - q \rightarrow q = \left[\frac{a}{b} \right] \quad (6)$$

اگر در رابطه (2) بجای q مقدار مساوی آنرا از فرمول (6) قراردهیم، مشاهده می کنید که :

$$R = a - b \left[\frac{a}{b} \right] \quad (7)$$

بنابراین رابطه (6) و (7) فرمول محاسبه خارج قسمت و باقیمانده را بر حسب مقسوم و مقسوم علیه بیان می کنند.

حال چند مثال را در رابطه با تقسیم حل می کنیم :

(1) باقیمانده و خارج قسمت تقسیم ۱۵ بر ۴ را حساب کنید .

$$q = \left[\frac{15}{4} \right] = [3.75] = 3$$

$$R = 15 - 4 \times \left[\frac{15}{4} \right] = 15 - 4 \times 3 = 15 - 12 = 3 \quad \text{به طریق دستی اگر عمل کنیم داریم که}$$

$$\begin{array}{r} 15 \\ \hline 12 \end{array} \begin{array}{r} 4 \\ \hline 3 \end{array} \rightarrow \left. \begin{array}{l} \text{خارج قسمت} \\ \text{باقیمانده} \end{array} \right\}$$

که حاصل ازنتایج بدست آمده در بالا برابر است.

(۲) باقیمانده و خارج قسمت تقسیم ۲ بر ۳ را حساب کنید.

$$q = \left[\frac{2}{3} \right] = [0.\overline{6}] = 0$$

$$R = 2 - 3 \times \left[\frac{2}{3} \right] = 2 - 3 \times 0 = 2 - 0 = 2$$

$$\begin{array}{r} 2 \\ \hline 0 \end{array} \begin{array}{r} 3 \\ \hline 0 \end{array} \rightarrow \left. \begin{array}{l} \text{خارج قسمت} \\ \text{باقیمانده} \end{array} \right\}$$

به طریقه دستی خواهیم داشت :

(۳) باقیمانده و خارج قسمت تقسیم ۱ بر ۲ را بدست آورید.

$$q = \left[\frac{1}{2} \right] = [0.5] = 0$$

$$R = 1 - 2 \times \left[\frac{1}{2} \right] = 1 - 2 \times 0 = 1 - 0 = 1$$

به طریقه دستی خواهیم داشت :

$$\begin{array}{r} 1 \\ \hline 0 \end{array} \begin{array}{r} 2 \\ \hline 0 \end{array} \rightarrow \left. \begin{array}{l} \text{خارج قسمت} \\ \text{باقیمانده} \end{array} \right\}$$

تذکر : همواره باید توجه داشت که این تقسیم‌ها در Z انجام می‌شود نه در R (مجموعه اعداد حقیقی) که عموماً حاصل تقسیم ۱ بر ۲ را $2/5$ می‌نویسند که غلط است چونکه اصلاً $2/5$ در Z وجود ندارد.

تعريف بخش پذیری : فرض کنیم a و b دو عدد متعلق به Z باشند و $(b \neq 0)$ ، a را بر b بخش پذیر گوییم، هر گاه باقیمانده تقسیم a بر b مساوی صفر شود یا :

$$a \text{ بر } b \text{ بخش پذیر است} \Leftrightarrow \left(\frac{a}{b} = \left[\frac{a}{b} \right] \right)$$

توجه : اگر عدد a را بر b تقسیم کنیم ($0 \neq b$) ، در این صورت باقیمانده این تقسیم یکی از اعداد زیر است :

(b-1) ... یا (2) یا (1) یا (0)

تا این مرحله الگوریتم را از دیدگاه‌های مختلف بررسی کردیم حال به سما تووصیه می‌کنیم در علم الگوریتم نویسی نکات زیر رعایت فرمایید .

.الف) در ابتدای هر الگوریتم کلمه شروع و در انتهای آن کلمه پایان قرار دهید .

.ب) حتماً برای هر یک از دستورالعمل‌ها حتماً شماره‌ای در نظر بگیرید .

.ج) در علم الگوریتم نویسی ، برای محاسبه یک عبارت ریاضی یا انجام عملیات روی چند داده، حتماً مکانی برای ذخیره داده‌ها و نتایج حاصله در نظر بگیرید .

.د) برای قراردادن مقداری در یک خانه یا یک مکان از حافظه ماشین فرضی الگوریتم از علامت (\leftarrow) استفاده کنید .

مثال : مفهوم هر یک از عبارات زیر را برای ماشین الگوریتم بنویسید :

1) $A \leftarrow 2$

این عبارت بدین معنی است که خانه ای را به اسم A در نظر بگیرید و مقدار عددی ۲ را در آن ذخیره کنند.

2) $B \leftarrow 2^*3-5$

این عبارت بدین معنی است که ابتدا ۲ را در ۳ ضرب کن و سپس ۵ را از حاصل آن کم کرده و در خانه B قرار بده.

3) $C \leftarrow B+1$

این عبارت به این معناست که به مقدار خانه B یک واحد اضافه کن و حاصل را در خانه ای به اسم C ذخیره کن .

4) $I \leftarrow I+1$

این عبارت بدین معنی است که به مقدار قبلی خانه I یک واحد اضافه کن و سپس مقدار نهایی را دوباره در خانه I ذخیره کن .

بحث را با ذکر مثالی آدامه می‌دهیم.

مثال ۱) الگوریتمی بنویسید که سه عدد ۵، ۳، ۲ را در سه خانه A و B و C ذخیره کرده و سپس میانگین آنها را محاسبه و چاپ نماید.

با توجه به مطالبی که درباره چگونگی حل یک مسئله بیان کردیم نتایج زیر را داریم :

الف) شناخت مسئله : داده‌های مسئله اعداد ۵، ۳ و ۲ هستند و نتیجه حاصل از آنها میانگین سه عدد است.

ب) طرح نقشه مسئله : فرمول محاسبه میانگین سه عدد :

$$\text{مجموع سه عدد} = \frac{\text{میانگین سه عدد}}{3}$$

باتوجه نه توضیحات بالا الگوریتم مسئله بصورت زیر است :

- ۱- شروع
- ۲- عدد ۲ را در خانه A قرار بده .
- ۳- عدد ۳ را در خانه B قرار بده .
- ۴- عدد ۵ را در خانه C قرار بده .
- ۵- خانه‌های A و B و C را باهم جمع کن و حاصل را در خانه S قرار بده .
- ۶- مقدار خانه S را بر ۳ تقسیم کن و در AVE قرار بده .
- ۷- مقدار خانه AVE را بنویس .
- ۸- پایان .

عموما برای ساده تر شدن الگوریتم و قابل فهم تر بودن آن از فرم نوشتاری در ریاضیات برای دستور العملها ، بجای فرم فارسی استفاده می‌کنیم . به عنوان نمونه اگر الگوریتم مثال ۱ را بخواهیم بفرم ریاضی بنویسیم، داریم:

۱- شروع

$A \leftarrow 2$ - ۲

$A \leftarrow 3$ - ۳

$C \leftarrow 5$ - ۴

$S \leftarrow C+B+A$ - ۵

$AVE \leftarrow \frac{S}{3}$ - ۶

AVE را بنویس .

- پایان .

اگر بخواهیم راه حل این مسئله را تعمیم دهیم به طوریکه برای سه عدد دلخواه عمل میانگین رالجام دهد ان را به شکل زیر می نویسیم :

(مثال ۲)

۱- شروع .

۲- سه عدد را به عنوان ورودی بگیر .

۳- حاصل جمع سه عدد را در S قرار بده .۴- S را بر ۳ تقسیم کن و در AVE قرار بده .

۵- AVE را بنویس .

۶- پایان .

فرم ساده شده

 $S \leftarrow A+B+C$ -۳

$$AVE \leftarrow \frac{S}{3}$$
 -۴

-۵ AVE را بنویس .

-۶ پایان .

برای اینکه همواره مراحل مختلف اجرای یک الگوریتم را بهتر درک کنید، آنرا بطریقه دستی اجرا کنید. برای این منظور به هریک از متغیرها مقداری داده و برای خروجی نیز محلی را در نظر گرفته و با توجه به روند الگوریتم آنرا اجرا کنید.

اجرای دستی الگوریتم مثال ۱ بصورت زیر است :

$\frac{A}{2}$	$\frac{B}{3}$	$\frac{C}{5}$	$\frac{S}{10}$	$\frac{AVE}{\frac{10}{3}}$	چاپ
---------------	---------------	---------------	----------------	----------------------------	-----

برای مثال ۲ نیز:

$\frac{A}{4}$	$\frac{B}{6}$	$\frac{C}{2}$	$\frac{S}{12}$	$\frac{AVE}{\frac{12}{3}}$	چاپ
---------------	---------------	---------------	----------------	----------------------------	-----

دلیل اجرای دستی الگوریتم بررسی صحت درستی آن می باشد. در ادامه، الگوریتم ها را بفرم خلاصه شده نوشته و سعی می کنیم اسامی متغیرها را طوری انتخاب کنیم که تناسبی با راه حل مسئله و مقداری که قرار است در آن ذخیره شود، داشته باشند .

توجه : تساوی در ریاضیات به مفهوم برابر بودن دو مقدار است اما در کامپیوتر به مفهوم جایگزینی است به همین دلیل از علامت \leftarrow برای جایگزینی استفاده می کنیم. نکته دیگری که حائز اهمیت است، اسامی خانه ها یا متغیر هایی است که داده ها یا نتایج محاسبات در آنها ذخیره می شوند. این اسامی باید ترکیبی از حروف و ارقام بوده و حتماً اولین کاراکتر آن یکی از حروف الفباء باشد و از کاراکتر های خاص مثل (+ ، - ، \times ، \div ، ? ، ! و ... برای نامگذاری استفاده نکنید.

مثلاً اگر بخواهیم مقدار ۵ را در خانه I ذخیره کنیم حتماً باید بنویسید :

 $I \leftarrow 5$

چرا که اگر برعکس آن یعنی $I \leftarrow 5$ را بنویسید بدین مفهوم است که مقدار I را در خانه ای بنام ۵ قرار بده که نادرست است. بدلیل اینکه ۵ نام آن خانه است که با توجه به قواعد گفته شده نادرست می باشد.

ویا $b^2 - 4ac \leftarrow D$ که درست بوده ولی بر عکس آن یعنی $D \leftarrow b^2 - 4ac$ نادرست می‌باشد چرا که $b^2 - 4ac$ نام خانه می‌باشد که با توجه به قواعد گفته شده نادرست می‌باشد.

مثال ۳ حقوق کارگری A ریال است و هر ماه ۱۰٪ از حقوق وی بابت بیمه و ۵٪ آن بابت حق مسکن کسر می‌گردد. الگوریتمی بنویسید که حقوق یک کارگر را به عنوان ورودی دریافت کرده و حقوق خالص وی را با کسر بیمه و حق مسکن محاسبه و چاپ نماید.

$$\left. \begin{array}{l} \text{مقدار بیمه} \\ \text{مقدار حق مسکن} \\ \text{مقدار حقوق خالص} \end{array} \right\} \quad \begin{array}{l} 1-\text{شروع} \\ 2-A \text{- را بگیر.} \\ B \leftarrow \frac{10A}{100} - 3 \\ M \leftarrow \frac{5A}{100} - 4 \\ S \leftarrow M+B - 5 \\ H \leftarrow A-S - 6 \\ 7-H \text{ را بنویس.} \\ 8-\text{پایان.} \end{array}$$

$$\frac{A}{10000} \quad \frac{B}{1000} \quad \frac{M}{500} \quad \frac{S}{1500} \quad \frac{H}{8500} \quad \frac{\text{چاپ}}{8500}$$

اجرای آن :

۸۵۰۰ ریال حقوق خالص است.

مثال ۴) الگوریتمی بنویسید که زمان T بر حسب ثانیه را به عنوان ورودی دریافت نموده و معین کند که چند ساعت، چند دقیقه و چند ثانیه است. با فرض اینکه هر ساعت ۶۰ دقیقه و هر دقیقه ۶۰ ثانیه است. حل :

برای بدست آوردن ساعت ابتدا T را بر ۳۶۰۰ تقسیم کرده و جزء صحیح آنرا بدست می‌آوریم که معرف ساعت است، سپس این مقدار را در ۳۶۰۰ ضرب کرده از T کم کرده و حاصل را بر ۶۰ تقسیم و جزء صحیح آنرا بدست می‌آوریم که معرف دقیقه است و این عمل را برای ثانیه نیز انجام می‌دهیم. با توجه به توضیحات بالا الگوریتم مسئله بصورت زیر می‌باشد.

$$\left. \begin{array}{l} \text{ساعت} : H \\ \text{دقیقه} : M \\ \text{ثانیه} : S \end{array} \right\} \quad \begin{array}{l} 1-\text{شروع} \\ 2-T \text{ را بگیر.} \\ H \leftarrow \frac{T}{3600} - 3 \\ 4-H \text{ را بنویس.} \end{array}$$

$R \leftarrow T - 3600 \times H - \Delta$

$$M \leftarrow \left[\frac{R}{60} \right] - 6$$

-۷ M را بنویس.

-۸ $S \leftarrow R - 60 \times M$

-۹ S را بنویس.

-۱۰ پایان.

اجرای دستی آن :

T	H	R	M	S	چاپ
4210	1	610	10	10	ساعت 1
					دقیقه 10
					ثانیه 10

انواع جملات

در الگوریتم نویسی جملات به چهار نوع تقسیم می‌شوند :

۱- جملات شرطی

۲- جملات محاسباتی

۳- جملات توضیحی

۴- جملات مربوط به ورودی و خروجی (I/O) Input/Output

جملات شرطی

این جملات به دو دسته تقسیم می‌شوند :

شرطی نوع ساده : فرم کلی این جملات به صورت زیر است :

< یک یا چند دستور > **THEN** < یک یا چند شرط >

در این گونه جملات شرطی، اگر شرط بعد از IF درست باشد دستورات مقابل THEN را اجرا و به خط بعد منتقل می‌شود. اما اگر شرط نادرست باشد دستورات جلوی THEN را انجام نداده و مستقیماً به خط بعد می‌رود.

تذکر: دستورات الگوریتم به ترتیب نوشتن آنها اجرا می‌شوند ولی ما می‌توانیم ترتیب اجرای دستورات را از خطی به خط دیگر انتقال دهیم. در دستورات شرطی اگر شرط ما درست باشد دستورات مقابل THEN اجرا شده و در این حالت می‌توانیم اجرای الگوریتم را به چند خط بالاتر و یا چند خط

پائین تر ارجاع دهیم .

مثال ۵) الگوریتمی بنویسید که اعداد زوج دو رقمی را یکی یکی محاسبه و چاپ نماید.

I	چاپ	
10	10	۱- شروع
12	12	I ← 10 - ۲
14	14	I ← I + ۲ - ۳
:	:	
.	.	
98	98	-۴ اگر 98 == I سپس برو به خط ۳
100		-۵
		-۶ پایان .

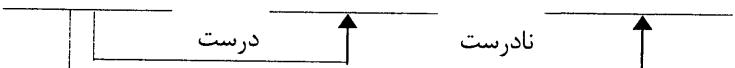
در این مثال ملاحظه می شود که مقدار نهایی I برابر 100 می باشد.

مثال ۶) الگوریتمی بنویسید که عدد طبیعی n را به عنوان ورودی دریافت و اعداد فرد کوچکتر یا مساوی عدد n را یکی یکی محاسبه و چاپ نماید .

n	I	چاپ	
8	1		۱- شروع
3	3		۲- n را بگیر
5	5		I ← I - ۳
7	7		۴- I را بنویس
9			I ← I + ۲ - ۵
			-۶ اگر n == I سپس برو به خط ۴
			-۷ پایان .

شرطی نوع دوم : فرم کلی این جملات بصورت زیراست :

If < یک یا چند دستور > یک یا چند دستور < Else > یک یا چند شرط <



یا

اگر < یک یا چند شرط > سپس < یک یا چند دستور > و گزنه < یک یا چند دستور >



تذکر : هرگاه بخواهیم دو دستور العمل را در یک شماره بنویسیم بین دو دستور العمل (و) قرار می

دهیم .

مثال ۷) الگوریتمی بنویسید که اعداد زوج بین ۱۰۰۰ و ۲۰۰۰ را یکی یکی تولید و چاپ نماید . در ضمن مجموع آنها را نیز چاپ کند.

I	S	چاپ	
۱۰۰۰	۰	۱۰۰۰	۱- شروع
	۱۰۰۰		۲- $S \leftarrow 0$ $I \leftarrow 1000$
۱۰۰۲	۳۰۰۶	۱۰۰۲	۳- بنویس I و $S \leftarrow S + I$
۱۰۰۴	.	۱۰۰۴	۴- $I \leftarrow I + 2$
.	.	.	
.	.	.	۵- اگر $I \leq 2000$ سپس برو به ۳ و گرنه S را بنویس
۷۵۰۰۰۰	۷۵۰۰۰۰	۷۵۰۰۰۰	۶- پایان
۲۰۰۲	۷۵۰۰۰۰	۷۵۰۰۰۰	

توضیح : برای فهم این مثال بهتر ، آن را با یک مثال شهودی توضیح می دهیم .

فرض کنید یک جعبه محتوی سبب دارید و تعداد سبب های درون آن را نمی دانید. حال اگر بخواهید تعداد آنها را محاسبه کنید، ابتدا یک سبد خالی در نظر گرفته، سپس یکی یکی سبب ها را از جعبه اصلی بیرون آورده و در جعبه خالی قرار دهید. به عنوان مثال وقتی اولین سبب را بیرون می آورید ، هنوز محتويات سبد خالی هیچ یا صفر است. سپس وقتی که اولین سبب را درون سبد خالی قرار می دهید مانند این است که مقدار صفر (تعداد سبب های داخل سبد در ابتدا) را با یک (اولین سبب برداشته شده از جعبه) جمع کرده و دوباره در سبد قرار داده اید. برای دومی هم وقتی آن را از داخل جعبه بیرون آورده و داخل سبد قرار می دهید، مانند این است که یک را با دو جمع کرده (تعداد سبب های داخل سبد به اضافه سبب بیرون آورده شده از جعبه) و دوباره داخل سبد قرار داده و این عملیات را برای همه سبب های داخل جعبه ادامه دهید.

با این روش کلیه سبب های داخل جعبه شمرده می شوند. به بیان دیگر سبد خالی فقط نقش یک شمارنده را در این مثال شهودی بازی می کند و هیچ نقش دیگری ندارد. در تذکر زیر مراحل منطقی این عمل را توضیح می دهیم.

تذکر : هر گاه بخواهیم مجموعی را محاسبه کنیم، ابتدا متغیری را در نظر می گیریم که مقدار اولیه آن صفر باشد(مانند S)، سپس تک تک جملاتی را که قرار است در خانه S ذخیره شوند را تولید و با مقدار قبلی S جمع و دوباره در خود S ذخیره می کنیم و این مراحل را تا پایان تولید جملاتی که قرار است تولید شوند ادامه می دهیم.

مثال ۸) الگوریتمی بنویسید که عدد طبیعی N را دریافت و مجموع زیر را محاسبه و چاپ نماید .

S = $1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{N}$	N	I	S	چاپ	
	۷	X	۰	۲.۲۹	۱- شروع
		۲	۲.۲۹		۲- N را بگیر
		۳	۱.۵		
		۴	۱.۸۴		۳- $I \leftarrow I + 1$ و $S \leftarrow S + \frac{1}{I}$
		۵	۲.۰۹		
		۶	۲.۲۹		
		۷	۲.۲۹		۴- $S \leftarrow S + \frac{1}{I}$

I \leftarrow ۱ S \leftarrow ۰ -۳

S \leftarrow S + $\frac{1}{I}$ -۴

I \leftarrow I+1 -۵

-۶ اگر I \leq N سپس برو به خط ۴ و گرنه S را بنویس
-۷ پایان .

مثال ۹) الگوریتمی بنویسید که عدد طبیعی n را دریافت و مجموع زیر را محاسبه و چاپ نماید.

$$S = \frac{1}{2} + \frac{1}{4} + \dots + \frac{1}{n}$$

n	I	S	چاپ
2	2	0	1.045
4	4	0.5	
6	6	0.75	
8	8	0.94	
10	10	1.045	

-شروع

-۲ n را بگیر

I \leftarrow ۲ و S \leftarrow ۰ -۳

S = S + $\frac{1}{I}$ -۴

I \leftarrow I+2 -۵

-۶ اگر I \leq n سپس برو به خط ۴ و گرنه S را بنویس
-۷ پایان .

مثال ۱۰) الگوریتمی بنویسید که عدد طبیعی n را دریافت و مجموع زیر را محاسبه و چاپ نماید.

$$P = \frac{1}{3^1} + \frac{2}{3^2} + \frac{3}{3^3} + \dots + \frac{n}{3^n}$$

n	I	S	چاپ
1	1	0	0.7160
2	2	0.34	
3	3	0.56	
4	4	0.67	
5	5	0.7160	

- شروع

-۲ n را بگیر

P \leftarrow ۰ I \leftarrow ۱ -۳

p = p + $\frac{I}{3^I}$ -۴

I \leftarrow I+1 -۵

-۶ اگر I \leq n سپس برو به خط ۴ و گرنه p را بنویس
-۷ پایان .

مثال ۱۱) الگوریتمی بنویسید که ۳ عدد a , b و c را دریافت و معین کند که با این سه عدد می توان یک مثلث ساخت یا خیر ؟

فرض: شرط آنکه سه عدد a و b و c طول اضلاع مثلثی باشند، آنستکه بین a و b و c روابط زیر برقرار باشند

$$\left. \begin{array}{l} a \leq b + c \\ b \leq a + c \\ c \leq a + b \end{array} \right\}$$

- ۱- شروع
- ۲- a و b و c را بگیر
- ۳- اگر $a <= b+c$ بود برو به خط ۴ و گرنه برو به ۷
- ۴- اگر $b <= a+c$ بود برو به خط ۵ و گرنه برو به ۷
- ۵- اگر $c <= a+b$ بود برو به خط ۶ و گرنه برو به ۷
- ۶- بنویس می‌توان یک مثلث ساخت و برو به ۸
- ۷- بنویس نمی‌توان یک مثلث ساخت
- ۸- پایان

مثال ۱۲) الگوریتمی بنویسید که سه عدد a ، b و c که طول اضلاع مثلث هستند، را دریافت و معین کند که مثلث قائم الزاویه است یا خیر؟

یکی از شرایط قائم الزاویه بودن اینست که برای اضلاع a و b و c یکی از سه شرط زیر برقرار باشد.

$$\left. \begin{array}{l} a^2 = b^2 + c^2 \\ \text{یا} \\ b^2 = a^2 + c^2 \\ \text{یا} \\ c^2 = a^2 + b^2 \end{array} \right\}$$

- ۱- شروع
- ۲- a و b و c را بگیر
- ۳- اگر $a^2 = b^2 + c^2$ بود برو به خط ۶ و گرنه برو به خط ۴
- ۴- اگر $b^2 = a^2 + c^2$ بود برو به خط ۶ و گرنه برو به خط ۵
- ۵- اگر $c^2 = a^2 + b^2$ بود برو به خط ۶ و گرنه برو به خط ۷
- ۶- بنویس می‌توان یک مثلث قائم الزاویه ساخت و برو به ۸
- ۷- بنویس نمی‌توان یک مثلث قائم الزاویه ساخت
- ۸- پایان.

مثال ۱۳) الگوریتمی بنویسید که ۱۰۰ عدد دلخواه را یکی یکی دریافت و چاپ نماید و درنهایت جمع آن اعداد را نیز چاپ کند.

۱- شروع

$S \leftarrow 0$ و $I \leftarrow 1$ -۲

۳- a را بگیر

۴- a را بنویس

$S \leftarrow S + a$ -۵

$I \leftarrow I+1$ -۶

۷- اگر $I=100$ بود پس برو به خط ۳

۸- S را بنویس

۹- پایان .

نتذکر : در کلیه الگوریتمهایی که روند اجرا را از مرحله ای به مرحله ای دیگر ارجاع داده ایم، دراصل یک حلقه ساخته ایم. نکته ای که در مورد این حلقه ها قابل توجه می باشد، این است که همواره متغیری در ابتدای الگوریتم وجود دارد که یک مقدار اولیه دارد سپس در حین اجرا به مقدار آن افزوده شده و دوباره در خودش ذخیره می شود (البته این عمل در داخل حلقه انجام می شود) و با یک مقدار مشخص سنجیده می شود. اگر شرط درست بود، دوباره حلقه تکرار می گردد که به این گونه متغیر ها شمارنده نیز گفته می شود.

باید دقت زمانیکه از حلقه خارج می شویم ، همواره مقدار شمارنده حلقه از مقداری که با آن سنجیده شده است، بیشتر است . (اگر شمارنده حالت نزولی داشته باشد مقدار شمارنده از مقداری که با آن سنجیده می شود ، کمتر است) .

در مورد مثال ۱۳، نقش شمارنده حلقه را دارد که مقدار اولیه آن یک است و هر بار در داخل حلقه یک واحد به آن اضافه شده و با عدد ۱۰۰ مقایسه می گردد، زمانیکه از حلقه خارج می شویم، مقدار شمارنده آن همواره از عدد ۱۰۰ بزرگتر است یعنی ۱۰۱ می باشد .

درباره هر مسئله دلخواهی این مقدار بفرم زیر محاسبه می شود :

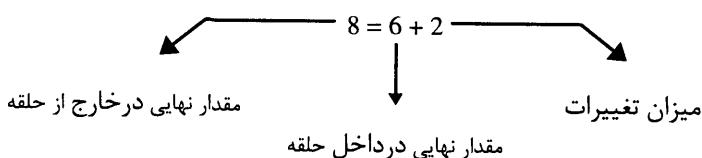
$$\text{میزان تغییرات در داخل حلقه} + \text{مقدار نهایی آن در داخل حلقه} = \text{مقدار نهایی شمارنده در خارج از حلقه}$$

مثال ۱۴) در الگوریتم زیر معین کنید ، شمارنده حلقه چه متغیری بوده و مقدار نهایی آن در داخل حلقه و خارج حلقه چقدر است ؟

- ۱- شروع
- ۲- $N \leftarrow 2$
- ۳- I و Z و k را بگیر
- ۴- $t \leftarrow i + j + k$
- ۵- t را بنویس
- ۶- $N \leftarrow N+2$
- ۷- اگر $N \leq 6$ سپس برو به ۳
- ۸- پایان

$\frac{N}{2}$	i	j	k	t	جاب
2	1	2	3	6	6
4	5	7	1	13	13
6	8	0	-1	7	7
8					

در این مثال N شمارنده حلقه است که سه بار تکرار می‌شود و مقدار نهایی آن در داخل حلقه با توجه به جدول فوق، برابر ۶ و خارج از حلقه برابر ۸ است.



از این قسمت به بعد مثالهای مختلفی را درباره مسائل مختلف مطرح و سعی می‌کنیم مرحله به مرحله با توضیحات کافی الگوریتم آنها را بنویسیم. به تواناییهای خود کم بها ندهید و مطمئن باشید که اگر فقط یکبار از روی هر کدام از الگوریتم‌ها بنویسید و بدرسی آنها را متوجه شوید، مفهوم الگوریتم و به دنبال آن برنامه سازی را فرا گرفته‌اید.

مثال ۱۵) الگوریتمی بنویسید که دو عدد A و B را به عنوان ورودی دریافت و بزرگترین و کوچکترین عدد را محاسبه و چاپ نماید. (اگر A و B مساوی بودند، دو عدد دیگر را بگیرد).

این مسئله را در دو حالت حل می‌کنیم.

حالات اول: در این قسمت فرض می‌کنیم که هیچوقت دو عدد مساوی نیستند که الگوریتم آن بصورت زیر می‌شود.

- ۱- شروع
- ۲- و A و B را بگیر
- ۳- اگر $A > B$ بود سپس بنویس A بزرگترین عدد و B کوچکترین عدد است و پایان.
- ۴- B را به عنوان بزرگترین عدد بنویس.
- ۵- A را به عنوان کوچکترین عدد بنویس.

A	B	چاپ
3	5	۵ بزرگترین و ۳ کوچکترین عدد

۶- پایان.

اجرای دستی آن بصورت زیر است .

حالت دوم : این قسمت تعییم یافته حالت اول است ، با این فرض که قبل از مقایسه کوچکتری یا بزرگتری A و B ، حالت تساوی آنها مقایسه شود و در صورت برابری ، دو عدد دیگر دریافت شود ، که الگوریتم آن بصورت زیر است .

۱- شروع

۲- A و B را بگیر

۳- اگر A=B سپس برو به خط ۲

۴- اگر A>B بود سپس A را بنویس بزرگترین و B را بنویس کوچکترین عدد و پایان

۵- B را بنویس بزرگترین عدد

۶- A را بنویس کوچکترین عدد

۷- پایان .

A	B	چاپ
6	6	3 بزرگترین و -1
3	-1	کوچکترین عدد

اجرای دستی آن بصورت زیر است :

مثال ۱۶) الگوریتم مثال ۱۵ را برای سه عدد بنویسید به طوریکه فقط بزرگترین را چاپ کند . (فرض بر این است که سه عدد با هم برابر نیستند) .

۱- شروع

۲- A و B و C را بگیر

۳- اگر A>B و A>C سپس A را بنویس و پایان

۴- اگر A>B و C>A سپس C را بنویس و پایان

۵- اگر B>A و C>B سپس B را بنویس و پایان

۶- اگر B>A و C>B سپس C را بنویس و پایان

که اجرای دستی آن بصورت زیر خواهد بود :

A	B	C	چاپ
-5	1	-3	1
A	B	C	چاپ
3	2	6	6

نکته‌ای که درباره این مثال قابل ذکر است ، این است که می توانیم شرط‌ها را با استفاده از عبارات منطقی (و) ، (یا) ، (نقیض) باهم ترکیب نموده و درستی یا نادرستی آنها را با توجه به جدول درستی ترکیب آنها که به صورت زیر است تعیین کنیم .

فرض کنیم A و B دو شرط باشند . جدول ترکیب این شرطها با اپراتورهای منطقی (و) و (یا) و (نیز) بصورت زیر است .

A	B	A و B (A \wedge B)	(A \vee B) A یا B	نقیض A (~A)	نقیض B (~B)
د	د	د	د	ن	ن
د	ن	ن	د	ن	د
ن	د	ن	د	د	ن
ن	ن	ن	ن	د	د

مثال ۱۷) الگوریتمی بنویسید که یک عدد صحیح مثبت را به عنوان ورودی دریافت و معین کند عدد زوج یا فرد می باشد .

توضیح : اگر عددی بر ۲ تقسیم شود، باقیمانده آن یکی از دو عدد صفر یا یک می باشد. اگر صفر بود عدد زوج است و اگر یک بود عدد فرد است.

- ۱- شروع
- ۲- عدد N را بگیر
- ۳- $R \leftarrow N - 2 \times \lfloor N/2 \rfloor$
- ۴- اگر $R=0$ سپس بنویس N زوج است و پایان
- ۵- بنویس N فرد است
- ۶- پایان .

که می توان آنرا به فرم زیر با دوشرط نوشت :

- ۱- شروع
- ۲- عدد N را بگیر
- ۳- $R \leftarrow N - 2 \times \lfloor N/2 \rfloor$
- ۴- اگر $R=0$ سپس بنویس N زوج است و پایان
- ۵- اگر $R=1$ سپس بنویس N فرد است
- ۶- پایان .

$$\frac{N}{13} \quad \frac{R}{1} \quad \frac{\text{چاپ}}{\text{فرد است ۱۳}}$$

که اجرای دستی آن به شکل مقابل خواهد بود :

مثال ۱۸) الگوریتمی بنویسید که عدد طبیعی N را دریافت و مجموعه مقسوم علیه های آن و همچنین ، تعداد آنها و نیز مجموع آنها را محاسبه و چاپ نماید .

توضیح : فرض کنیم N عددی طبیعی باشد، مقسوم علیه‌های N ، اعداد صحیح کوچکتر از N هستند که اگر N به هر کدام از آنها تقسیم شود باقیمانده مساوی با صفر می‌شود. برای مثال عدد ۱۲ را در نظر بگیریم که مقسوم علیه‌های آن ۱، ۶، ۴، ۳، ۲، ۱ هستند.

۱- شروع

۲- N را بگیر

۳- $I \leftarrow I - 1$

۴- $R \leftarrow N - I \times [N/I]$

۵- اگر $R=0$ سپس I را بنویس

۶- $I \leftarrow I + 1$

۷- اگر $I \leq N$ سپس برو به خط ۴

۸- پایان

توضیح : با توجه به تعریف مقسوم علیه یک عدد ، باید عدد N را که ورودی است به ترتیب به اعداد از ۱ تا N ($N, \dots, 3, 2, 1$) تقسیم کنیم و هر بار که باقیمانده مساوی صفر شد ، مقسوم علیه تقسیم را بنویسیم که در این صورت احتیاج به N عمل تقسیم داریم. برای اجتناب از چنین کاری متغیری را مانند I مساوی با یک در نظر می‌گیریم و بجای آنکه عدد N را بر ۱ تقسیم کنیم، آن را بر I تقسیم می‌نمائیم و مقدار I را در داخل یک حلقه یکی اضافه می‌کنیم تا مقدار آن به N برسد. به بیان دیگر عمل تقسیم را N بار با N عدد مختلف انجام می‌دهیم و هر کجا که باقیمانده صفر شد، مقسوم علیه تقسیم را که I نقش آن را دارد می‌نویسیم.

الگوریتم بالا فقط مقسوم علیه‌ها را حساب می‌کند برای محاسبه مجموع آنها خانه‌ای مانند S را مساوی صفر در نظر گرفته و هر بار که یک مقسوم علیه را تولید کرده و می‌نویسیم آنرا با S جمع می‌کنیم و برای تعداد W ای را مساوی صفر قرارداده و هر بار که یک مقسوم علیه را می‌نویسیم یکی به W اضافه می‌کنیم و در آخر S و W را چاپ می‌کنیم.

الگوریتم کامل مسئله بصورت زیر است :

۱- شروع

۲- N را بگیر

۳- $S \leftarrow 0$

۴- $W \leftarrow 0$

۵- $I \leftarrow I - 1$

۶- $R \leftarrow N - I \times [N/I]$

۷- اگر $R=0$ سپس I را بنویس و $W \leftarrow W + 1$ و $S \leftarrow S + I$

۸- $I \leftarrow I + 1$

N	I	R	S	W	چاپ
6	X	0	0	0	
	X	X	1		
2	0	3	2	2	
3	0	6	3	3	
4	2				
5	X				
6	0	10	4	6	
7			10		
			4		

۹- اگر $N \leq S$ سپس برو به خط ۶

۱۰- و $S > N$ را بنویس

۱۱- پایان .

در قسمت مربوط به چگونگی حل مسئله ذکر شد که بهترین راه حل برای یک مسئله راه حلی است که ساده و قابل فهم و در عین حال کوتاه نیز باشد ، در مورد الگوریتم نیز باید بگوئیم که یک الگوریتم زمانی بهترین حالت را دارد که :

ساده و قابل فهم باشد .

روان باشد .

کوتاه باشد .

همچنین از عملیاتی که وقت زیادی گرفته و در عین حال زائد هستند، استفاده نشده باشد. در مورد الگوریتم مثال ۱۸ می دانیم که برای هر عددی یک و خود عدد، مقسوم علیه های آن هستند، پس عمل محاسبه مقسوم علیه برای ۱ و N اضافه و زائد است یعنی می توان از ۲ شروع کرد و تا $N-1$ پیش رفت، حتی قویتر از آن می توان گفت که کافیست از ۲ تا نصف عدد پیش برویم چرا که قضیه زیر بیان این موضوع است که از نصف عدد تا خود عدد (منظور کوچکتر از خود عدد) هیچ مقسوم علیه ای برای عدد وجود ندارد.

قضیه : فرض کنیم N عددی طبیعی باشد، آنگاه بین $\frac{N}{2}$ و N هیچ مقسوم علیه ای برای عدد N وجود ندارد.

اثبات : (بهروش برهان خلف) فرض کنیم چنین عددی بین $\frac{N}{2}$ و N که مقسوم علیه نیز باشد وجود داشته باشد، پس داریم :

\exists نماد وجود داشتن
 \exists نماد به طوری که
 \in نماد تعلق داشتن

$$\exists q \in \mathbb{Z} \ni \frac{N}{2} < q < N \quad (1)$$

$$\Rightarrow \exists a \in \mathbb{Z} \ni a \cdot q = N \quad (2) \quad \text{چون } q \text{ یک مقسوم علیه است پس}$$

$$\Rightarrow q = \frac{N}{a} \quad (3)$$

اگر در رابطه (۱) به جای q مقدار مساوی آنرا از رابطه (۳) قرار دهیم داریم :

$$\frac{N}{2} < q < N \quad (4)$$

$$\frac{N}{2} < \frac{N}{a} < N \quad (5)$$

$$\Rightarrow \frac{2}{N} > \frac{a}{N} > \frac{1}{N} \text{ یا } \frac{1}{N} < \frac{a}{N} < \frac{2}{N} \quad (6)$$

$$\Rightarrow 2 > a > 1 \quad \text{یا} \quad 1 < a < 2 \quad (7)$$

که خلاف فرض است چرا که بین ۲ و ۱ هیچ

عدد صحیحی وجود ندارد.

پس الگوریتم مثل ۱۸ را می‌توانیم در حالت

بهینه بصورت زیر بنویسیم :

۱- شروع

۲- N را بگیر

۳- ۱ را بنویس

$W \leftarrow 2$ و $S \leftarrow N+1$ -۴

$I \leftarrow 2-5$

$R \leftarrow N - I \times [N/I]$ -۶

۷- اگر $R=0$ سپس ۱ را بنویس و $S \leftarrow S+I$ و $W \leftarrow W+1$

$I \leftarrow I+1$ -۸

۹- اگر $\frac{N}{2} < I < N$ سپس برو به ۶

۱۰- N را بنویس

۱۱- S و W را بنویس

۱۲- پایان.

چون یک و خود عدد را برای محاسبه مقسوم علیه در نظر نمی‌گیریم لذا از ابتدا مقدار S را $N+1$ و W را ۲ در نظر گرفتیم.

N	I	R	S	W	چاپ
12	2	0	15	3	1
	3	0	18	4	
	4	2	32	5	
	5	0			
	6				
	7		28	6	

تعداد ۶ مجموع 28

تذکر: اگر بخواهیم مقسوم علیه های زوج یک عدد را حساب کنیم از ۲ شروع می‌کنیم و دوتا دوتا پیش می‌رویم و اگر مقسوم علیه های فرد را بخواهیم، از یک شروع کرده و باز دوتا دوتا پیش می‌رویم و به طور کلی اگر مقسوم علیه های مضرب K عدد طبیعی N را ($K < N$) بخواهیم محاسبه کنیم از K شروع می‌کنیم و K تا K تا پیش می‌رویم.

مثال ۱۹) الگوریتمی بنویسید که عدد طبیعی N را دریافت و معین کند این عدد تام است یا نه ؟

توضیح: عدد طبیعی N را تام یا کامل می‌گوئیم هرگاه مجموع مقسوم علیه های کوچکتر از عدد N ، با خود عدد N برابر شود مانند:

$$\text{مجموع مقسوم علیه های عدد } 6 = \{1, 2, 3, 6\}$$

$$\text{مجموع مقسوم علیه های کوچکتر از } 6 = \{1, 2, 3\}$$

که مجموع انها $6 = 1+2+3$ است که با خود عدد برابر است لذا عدد 6 تام است.

N	I	R	S	چاپ	شروع
14	۱	۰	۰	۱۴	N-۲ رابگیر
				عدد ۱۴	S ← ۰-۳
	۲	۰	۳	تام نیست	I ← ۱-۴
	۳	۲	۰		$R \leftarrow N - I \times \frac{N}{I}$ -۵
	۴	۲			۶- اگر $R=0$ پس $R=0$
	۵	۲			I ← I+1-۷
	۶	۲			
7	0	10			$\frac{N}{2}$ -۸ اگر $I \leq N$ پس برو به ۸
8					

۹- اگر $N = S$ سپس بنویس عدد N تام است و پایان

۱۰- بنویس عدد N تام نیست.

۱۱- پایان

توضیح: در خط ۸ می‌توانستیم به جای $\frac{N}{2}$ بنویسیم $I=N-1$ که این شرط هم درست است . اما زمان اجرای آنرا طولانی تر می‌کند.

مثال ۲۰) الگوریتمی بنویسید که عدد طبیعی N را به عنوان ورودی گرفته و معین کند اول است یا نه ؟
تعریف عدد/اول: عدد طبیعی N را اول گوئیم هرگاه به جز یک و خودش هیچ مقسوم-علیه دیگری نداشته باشد.

مانند عدد ۱۳ که بجز ۱ و ۱۳ مقسوم علیه دیگری ندارد. با توجه به تعریف بالا هرگاه عددی بر یکی از اعداد بین یک و خودش بخش پذیر باشد، دیگر اول نیست و اگر به تمامی آنها تقسیم شده و با قیمانده در هیچ کجا صفر نشود (یعنی فقط بر یک و خودش بخش پذیر باشد) اول است.

مثال برای عدد ۱۵ داریم :

۱ | ۱، ۲، ۳، ۴، ۵، ۶، ...، ۱۳، ۱۴، ۱۵

برای این که ببینیم اول نیست باید ۱۵ را بر یکی از اعداد ۲، ۳، ۴، ...، ۱۴ بخش پذیر باشد که ۱۵ بر ۳ بخش پذیر است پس اول نیست و دیگر عمل تقسیم را ادامه نمی دهیم.

اما برای عدد ۵ داریم :

۱. $\boxed{2, 3, 4, 5}$

می بینید که عدد ۵ بر ۲ بخش پذیر نیست، بر عدد بعدی یعنی ۳ نیز بخش پذیر نیست و همین طور به ۴ هم بخش پذیر نیست، لذا ۵ اول است.

۱- شروع

N	I	R	چاپ	N	I	R	چاپ	N	R	را بگیر
6	2	0	۶ عدد	11	۷	1	۱۱ عدد	۱	-۳	I \leftarrow 2
			اول		۳	2				$R \leftarrow N - I \times \left[\frac{N}{I} \right]$
			نیست		4	.				۵- اگر $R = 0$ سپس بنویس
			.		.	.				۶- اول نیست و پایان
			.		.	.				I $\leftarrow I + 1$
				10	1					۷- اگر $I < N$ سپس برو به ۴
				11						۸- بنویس عدد N اول است .

۹- پایان .

توضیح : این الگوریتم فقط برای محاسبه اول بودن یا نبودن اعداد بزرگتر از ۲ درست عمل می کند. روشهای متنوع دیگری برای محاسبه اعداد اول وجود دارد که در فصل مربوط به فلوچارت به طور مفصل آنها را مطرح خواهیم کرد .

مثال ۲۰) الگوریتمی بنویسید که عدد طبیعی N را دریافت و فاکتوریل آن را محاسبه کند. فرض کنید که N یک عدد طبیعی باشد ، فاکتوریل N را با نماد ! N نمایش داده و مقدار آن طبق روابط زیر بدست می آید.

$$N! = \begin{cases} 1 & \text{اگر } N = 0 \\ 1 & \text{اگر } N = 1 \\ N \times (N - 1) \times (N - 2) \times \dots \times (3) \times (2) \times (1) & \text{اگر } N \neq 0, 1 \end{cases}$$

برای مثال :

1) $5! = 5 \times 4 \times 3 \times 2 \times 1 = 120$

2) $1! = 1$

3) $0! = 1$

این تعریف فاکتوریل فقط برای اعداد طبیعی و صفر درست است و برای اعداد اعشاری و منفی بطریق دیگری تعریف می شود که از حوصله این کتاب خارج است .

در این مثال باید حاصلضرب چند عدد متوالی را پیدا کرده و مانند محاسبه مجموع که خانه‌ای را صفر در نظر گرفته و مقادیر را با آن جمع کردید، خانه ای را برابر یک در نظر گرفته و اعداد را یکی یکی تولید و در آن خانه ضرب کنید و به دلیل اینکه با ضرب شدن اعداد متوالی مقادیر قبلی از بین نزود مقدار آن

را در ابتدای ک در نظر بگیرید (چرا که اگر صفر بود هر عددی که ضرب می شد حاصل را صفر می کرد).

۱- شروع

N 6	P 1	I 1	چاپ	
	$1 \times 1 = 1$			N را بگیر
	$1 \times 2 = 2$	2		P \leftarrow 1 - ۳
	$1 \times 2 \times 3 = 6$	3		I \leftarrow 1 - ۴
	$1 \times 2 \times 3 \times 4 = 24$	4		P \leftarrow P \times I - ۵
	$1 \times 2 \times 3 \times 4 \times 5 = 120$	5		I \leftarrow I + 1 - ۶
	$1 \times 2 \times 3 \times 4 \times 5 \times 6 = 720$	6	720	۷- اگر N \leq ۱ سپس برو به ۵
		7		P را بنویس
				۸- پایان .

تذکر : در کلیه مثالهای قبل به نوعی از حلقه استفاده شده است. با توجه به نوع مسئله می توان از چند حلقه استفاده کرد بفرمی که هر حلقه به طور کامل داخل حلقة دیگری قرار گرفته باشد.

مثال (۱) الگوریتمی بنویسید که عدد طبیعی N را دریافت و حاصل جمع زیر را محاسبه و چاپ نماید .
 $S = 1! + 2! + 3! + \dots + N!$

برای حل ، حلقه ای با یک شمارنده برای جمع کردن و حلقة دیگری برای محاسبه فاکتوریل تک تک جملاتی که قرار است تولید و با هم جمع شوند، در نظر بگیرید.

۱- شروع

N 3	S 0	t 1	P 1	I 1	چاپ	
		۱	۱	۱		N را بگیر
			۱	۲		S \leftarrow ۰ - ۳
		۲	۱	۱		t \leftarrow ۱ - ۴
			۱	۲		P \leftarrow ۱ - ۵
		۳	۲	۳		I \leftarrow ۱ - ۶
			۱	۱		P \leftarrow P \times I - ۷
		۳	۱	۱		I \leftarrow I + 1 - ۸
			۱	۲		
		۹	۲	۳		
			۱	۲		
		4	6	4		
					۹	۹- اگر t \leq ۱ سپس برو به ۷

S \leftarrow S + P - ۱۰

t \leftarrow t + 1 - ۱۱

۱۲- اگر N \leq ۱ سپس برو به ۵

S را بنویس

۱۴- پایان .

این فرم حلقه ها را حلقه های تودر تو می گویند که در حل مسائل متعدد، کاربرد دارد و بعد از آشنایی با فلوچارت مجدداً آنها را مورد بررسی قرار خواهیم داد.

مثال (۲۲) الگوریتمی بنویسید که کلیه مقسوم علیه های اعداد بین ۲ تا ۵۰۰ را برای هر کدام، به طور جداگانه چاپ نماید.

این مثال نیز مانند مثال ۱۸ می باشد با این تفاوت که مثال ۱۸ برای یک عدد عمل می کند. برای اینکه تمامی مقسوم علیه های اعداد بین ۲ و ۵۰۰ را محاسبه کنید بایستی یک حلقه دیگر اضافه کنید که شمارنده آن از ۲ تا ۵۰۰ تغییر کند و محاسبه مقسوم علیه ها در داخل این حلقه صورت بگیرد.

۱- شروع

۲- $t \leftarrow 2$ ۳- t را بنویس۴- $I \leftarrow 1$

$$R \leftarrow t - I \times \left[\frac{t}{I} \right] - 5$$

۵- اگر $R = 0$ سپس I را بنویس۶- $I \leftarrow I+1$ ۷- اگر $I <= t$ سپس برو به ۵۸- $t \leftarrow t+1$ ۹- اگر $t <= 500$ سپس برو به ۳

۱۰- پایان .

مثال (۲۳) فرض کنید در روز R ام از ماه شماره M هستیم. الگوریتمی بنویسید که R و M را سوال و معین کند در چند مین روز سال هستیم (R و M اعداد صحیح هستند).

۱- شروع

۲- R و M را بگیر

۳- اگر $6 <= M <= 500$ سپس برو به خط ۴ و گرنه برو به خط ۶

$$N \leftarrow (M-1) \times 31 + R - 4$$

۴- N را بنویس و پایان

۵- اگر $11 < M < 16$ سپس برو به خط ۷ و گرنه برو به خط ۹

$$N \leftarrow 186 + (M-6) \times 30 + R - 7$$

۶- N را بنویس و پایان

$$N \leftarrow 336 + R - 9$$

۷- N را بنویس

۸- پایان .

چاپ

بار اول اجرا برای ۴ و ۳	70
بار دوم اجرا برای ۸ و ۱۱	257
بار سوم اجرا برای ۶ و ۱۲	342

مثال ۲۴) الگوریتمی بنویسید که عدد صحیح زوج و مثبت N را به عنوان ورودی دریافت و مجموع زیر را محاسبه کند.

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

توضیح: در این مثال جملات یک در میان مثبت و منفی می‌شوند. برای تولید اینگونه از جملات ضربی مانند K را یک در نظر گرفته و در جمله اول ضرب کنید (اگر جمله اول منفی بود منهای یک در نظر بگیرید) و هر بار که جمله بعدی را تولید می‌کنید، قبیل از آن مقدار ضریب رادر منفی یک ضرب کنید، (اگر یک باشد منهای یک می‌شود و برعکس). بدین ترتیب به طور متناوب ضریب جملات یک و منهای یک شده و جملات را تولید می‌کنند همچنین علامت جمله آخر را مشتبه قرار دهید، زیرا N زوج و (N-1) فرد است همچنین ملاحظه می‌کنید که در این عبارت جملاتی که صورت فرد دارند ضریب‌شان برابر یک است.

باتوجه به توضیحات فوق الگوریتم آن به صورت زیر نوشته می‌شود:

۱- شروع

۲- N را بگیر

۳- اگر $\frac{N}{2} \neq [\frac{N}{2}]$ سپس برو به ۴

۴- $S \leftarrow 0$

۵- $I \leftarrow 1$

۶- $K \leftarrow 1$

۷- $S \leftarrow S + K \times \frac{I}{(I+1)}$

۸- $I \leftarrow I+1$

۹- اگر $I \leq N$ سپس $K \leftarrow K$ و برو به خط ۷

۱۰- S را بنویس

۱۱- پایان.

N	S	I	K	چاپ
5	0	X	X	-0.21
0.5	2	1		
-0.16	3	X		
0.59	4	1		
-0.21	5		1	
	6			

مثال ۲۵ الگوریتمی بنویسید که دو عدد صحیح و مثبت M و N را دریافت و بزرگترین مقسوم علیه مشترک و کوچکترین مضرب مشترک آنها را محاسبه و چاپ نماید.

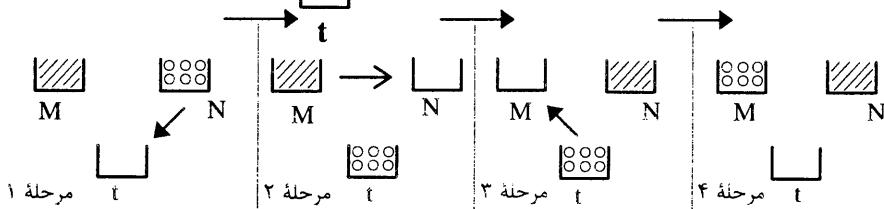
توضیح : برای محاسبه (ب . م . م) از روش غربال (اراستن) استفاده می کنیم، به این ترتیب که عدد بزرگتر را بر عدد کوچکتر تقسیم می کنیم، اگر باقیمانده صفر شد عدد کوچکتر (که مقسوم علیه تقسیم است) ب . م . م می باشد و اگر صفر نشد، مقسوم علیه را جای مقسوم و باقیمانده را بجای مقسوم علیه قرار داده و دوباره عمل تقسیم را انجام می دهیم تا زمانی که باقیمانده صفر شود که در این مرحله همواره آخرین مقسوم علیه ای که باقیمانده را صفر می کند (ب . م . م) است.

برای محاسبه (ک . م . م) از رابطه زیر استفاده می کنیم.

$$\text{حاصلضرب دو عدد} = \underset{\substack{\text{ب . م . م}}}{{\text{ک . م . م}}} \rightarrow \text{ب . م . م دو عدد} \times \text{ک . م . م دو عدد} = \text{حاصلضرب دو عدد}$$

$$|ab| = [a, b] \times (a, b) \Rightarrow [a, b] = \frac{|ab|}{(a, b)}$$

M را همواره بزرگتر از N در نظر بگیرید، در غیر اینصورت باید محتویات M و N را عوض کنید.
نکته دیگر در این مثال جا به جا کردن محتویات دو متغیر M و N می باشد که با یک مثال آنرا توضیح می دهیم، فرض کنید که محتویات دو طرف  و  را می خواهیم جابجا کنیم، برای اینکار بدین ترتیب عمل می کنیم که خانه‌ای خالی به نام t در نظر گرفته و اعمال زیر را انجام می دهیم.



با انجام ۴ مرحله فوق، محتویات دو خانه (متغیر) M و N جابجا می شوند.
بدون استفاده از متغیر سوم نیز می شود محتویات دو متغیر را جابجا کرد که به عنوان تمرین آنرا در آخر فصل آورده ایم.

حال با توضیحات داده شده الگوریتم آنرا بصورت زیر می نویسیم.

۱- شروع

۲- M و N را بگیر

۳- اگر N < M سپس N ← M و ! N ← N

$$A \leftarrow M - 4$$

$$B \leftarrow N - 5$$

$$R \leftarrow M - N \times \left[\frac{M}{N} \right] - 6$$

-۷ اگر $R = 0$ سپس برو به خط ۱۱

$$M \leftarrow N - 8$$

$$N \leftarrow R - 9$$

-۸ برو به خط ۶

-۹ N را بنویس ب . م . م

$$KMM \leftarrow \frac{A \times B}{N} - 12$$

-۱۰ چاپ کن KMM

-۱۱ پایان .

M	N	t	R	A	B	KMM	چاپ
24	30	30					
30	22		8	30	22	$\frac{22 \times 30}{2}$	۲ ب.م.م است
24	8			8			۳۳ ک.م.م است.
8	8				2		
6	2				0		

مثال (۲۶) الگوریتمی بنویسید که عدد طبیعی N را دریافت و تعداد ارقام و نیز مجموع ارقام آنرا محاسبه و چاپ نماید .

توضیح : فرض کنید عدد مورد نظر T رقمه باشد، رقم یکان آن از تقسیم عدد ، بر ۱۰ بدست می آید که همان باقیمانده تقسیم است. برای محاسبه رقم دهگان خارج قسمت تقسیم اولی (که در اصل جزو صحیح تقسیم عدد بر ۱۰ است) را دوباره بر ۱۰ تقسیم کنید(باقیمانده آن رقم دهگان است) و این عمل را تازمانی ادامه دهید که آخرین خارج قسمت صفر گردد .

مثال :

$$\begin{array}{r} 234 \quad | \quad 10 \\ 230 \quad | \quad 23 \quad | \quad 10 \\ 4 \quad | \quad 20 \quad 2 \quad | \quad 10 \\ \hline 3 \quad 0 \quad 0 \\ \hline 2 \end{array}$$

چون خارج قسمت صفر است دیگر عمل تقسیم را ادامه نمی دهیم .

-۱ شروع

-۲ N را بگیر

-۳ W \leftarrow 0

$S \leftarrow 0 - ۴$

$$R \leftarrow N - 10 \times \left[\frac{N}{10} \right] - ۵$$

 $S \leftarrow S + R - ۶$ $W \leftarrow W + 1 - ۷$

$$N \leftarrow \left[\frac{N}{10} \right] - ۸$$

۹- اگر $N > 0$ سپس برو به ۵۱۰- چاپ کن S مجموع ارقام۱۱- چاپ کن W تعداد ارقام

۱۲- پایان .

<u>N</u>	<u>W</u>	<u>S</u>	<u>R</u>	چاپ
358	0	0	8	۱۶مجموع ارقام
3	+	8		۳تعداد ارقام
5	2	13	5	
3	3	16	3	
0				

مثال (۲۷) الگوریتمی بنویسید که عددی در مبنای ۲ را گرفته و معادل آن را در مبنای ۱۰ بنویسد .
 توضیح : از آنجائی که ارقام یک عدد در مبنای ۲ فقط از ۰ و ۱ تشکیل شده است و هدف بدست آوردن تک تک ارقام آن و ضرب آن در توانهای متوالی ۲ است، بهمین دلیل برای بدست آوردن ارقام عدد مانند مثال ۲۶ عمل کنید.

<u>N</u>	<u>S</u>	<u>I</u>	<u>R</u>	چاپ
1101	0	0	1	۱۳
1	+			
1	+	+	0	
1	5	2	+	
0	13	3	1	

۱- شروع

۲- N را بگیر۳- $S \leftarrow 0$ ۴- $I \leftarrow 0$

$$R \leftarrow N - 10 \times \left[\frac{N}{10} \right] - ۵$$

 $S \leftarrow S + R \times 2^I - ۶$

$$N \leftarrow \left[\frac{N}{10} \right] - ۷$$

۸- اگر $N > 0$ سپس $I \leftarrow I + 1$ و برو به خط ۵۹- S را بنویس

۱۰- پایان .

اجراء بطریقه دستی :

$$(1101)_2 = 1 \times 2^0 + 0 \times 2^1 + 1 \times 2^2 + 1 \times 2^3 = 1 + 0 + 4 + 8 = (13)_{10}$$

اگر بخواهید از مبنای غیر از ۲ به ۱۰ ببرید، همین عملیات را انجام دهید ولی در توانهای متوالی همان عدد ضرب و تمام اعداد بدست آمده را با هم جمع کنید.

مثال ۲۸) الگوریتمی بنویسید N عدد را یکی یکی خوانده و بزرگترین عدد بین آنها را مشخص و چاپ نماید.

در اینگونه مسائل همواره اولین عدد را خوانده و آن را در خانهای به نام MAX قرار داده و عدد بعدی را می خوانید، که دو حالت اتفاق می افتد. اگر عدد دومی از MAX (همان عدد اول) بزرگتر بود، آنرا در MAX قرار داده و بعد عدد سوم را می خوانید ولی اگر کوچکتر بود مستقیماً عدد سوم را می خوانید و همین روند را ادامه می دهیم تا به آخرین عدد برسید. به عبارت دیگر همواره در بین اعداد، بزرگترین را پیدا کرده و در MAX قرار می دهیم.

N	A	B	I	MAX	چاپ	بزرگترین عدد
4	3		1	2		
		5	2	5		
		0	3	5		
		8	4	8		

- ۱- تروع
- ۲- N را بگیر
- ۳- A را بخوان
- ۴- MAX ← A

اعداد خوانده شده عبارت بودند از ۳.۵.۰.۸

۱ ← ۱ -۵

۶- اگر N > I سپس MAX را چاپ کن و پایان
۷- B را بخوان

۸- اگر B < MAX سپس MAX ← B

۹- ۱+۱ ← ۱ و برو به خط ۶

۱۰- پایان .

مثال ۲۹) الگوریتمی بنویسید که نام و ساعت کار و حقوق ساعتی کارکنان اداره‌ای را به عنوان ورودی دریافت و حقوق آنها را محاسبه و چاپ نماید.

اگر تعداد ساعت کار بیش از ۵۰ ساعت بود $\frac{3}{2}$ حقوق ساعتی به عنوان اضافه کار محاسبه شود.

M تعداد کارمندان	۱- شروع
N نام کارمند	۲- M را بگیر
H تعداد ساعت کار	۳- ۱ ← ۱-۳
HS حقوق ساعتی	۴- N را بخوان
S کل حقوق	۵- H را بخوان
	۶- HS را بخوان

۷-اگر $H \leq 50$ سپس $S \leftarrow H \times HS$ و برو به خط ۹

$$S \leftarrow 50 \times HS + (H - 50) \times \frac{3}{2} \times HS \quad ۸$$

۹- و S را چاپ کن .

$I \leftarrow I+1 \quad ۱۰$

۱۱- اگر $M \leq I$ سپس برو به خط ۴

۱۲- پایان .

مثال (۳۰) اداره‌ای دارای ۵۰۰ کارمند می‌باشد و می‌خواهد حقوق کارمندان خود را افزایش دهد .
الگوریتمی بنویسید که به حقوق کارمندانی که کمتر یا مساوی ۲۵۰۰۰ تومان باشد ۵٪ اضافه شود و به حقوق کسانی که بین ۲۵۰۰۰ و ۳۵۰۰۰ تومان است ۷٪ و بیش از ۳۵۰۰۰ تومان ۱۰٪ اضافه شود و حقوق اولیه و مقدار اضافه حقوق و مقدار حقوق جدید را برای هر کارمند محاسبه و چاپ نماید .

۱- شروع

$I \leftarrow 1 \quad ۲$

۳- N را بخوان

۴- H را بخوان

$$۵- \text{اگر } H \leq 25000 \text{ سپس } S \leftarrow H + E \leftarrow \frac{H \times 5}{100} \text{ و برو به خط ۹}$$

$$۶- \text{اگر } H < 35000 \text{ سپس } S \leftarrow H + E \leftarrow \frac{25000 \times 5}{100} + \frac{(35000 - H) \times 7}{100} \text{ و برو به خط ۹}$$

$$۷- E \leftarrow \frac{25000 \times 5}{100} + \frac{10000 \times 7}{100} + \frac{(H - 35000) \times 10}{100} \quad ۸$$

$S \leftarrow H + E \quad ۸$

۹- N را بنویس

۱۰- H را بنویس

۱۱- E را بنویس

۱۲- S را بنویس

۱۳- $I \leftarrow I+1 \quad ۱۴$

۱۴- اگر $500 \leq I$ سپس برو به خط ۳

۱۵- پایان

$$\left. \begin{array}{l} N \text{ نام کارمند} \\ H \text{ حقوق قدیم} \\ E \text{ اضافه حقوق} \\ S \text{ حقوق جدید} \end{array} \right\}$$

این فصل را بیشتر از این طولانی نمی‌کنیم و شما را با تمرينهای متنوع آن تنها می‌گذاریم تا آنها را حل کنید . اگر نتوانستید آنها را حل نمایید به این معنی نیست که الگوریتم را

نفهمیدهاید ، بلکه الگوریتم یک مفهوم مجرد است و باید با تمرینهای متعدد در ذهن بنشینند. پس دوباره به متن کتاب رجوع کنید و تک تک مثالهای حل شده را مرور نمایید . این بار حتما موفق خواهید شد .

پرسش‌های تشریحی

- ۱- سه عامل موثر برای حل یک مسئله را نام بده و هر یک را به طور مختصر توضیح دهید.
- ۲- سوال یک را درباره یک مسئله ساده بکار برد و مراحل مختلف آنرا تشریح کنید.
- ۳- تفکر منطقی و غیر منطقی را تعریف کنید.
- ۴- الگوریتم را ازدیدگاه یک ماشین تعریف کنید.
- ۵- تعریفی از الگوریتم ارایه نمایید که در آن زبان دقیق ، جزئیات کافی ، ترتیب مراحل و خاتمه پذیر بودن عملیات مشخص شده باشد .
- ۶- توضیح دهید که چرا در الگوریتم نویسی بجای تساوی ، از \leftarrow استفاده می‌شود.
- ۷- تفاوت یک انسان و یک ماشین را از نظر نحوه تفکر بیان کنید.
- ۸- منظور از مجری الگوریتم را به طور کامل توضیح دهید.
- ۹- شمارنده را در مجری الگوریتم تعریف نمایید و در مورد آن یک مثال بیاورید.
- ۱۰- مفهوم هریک از عبارات زیر را به طور کامل بنویسید .

الف) $p \leftarrow 2 \times 3$

ب) $S \leftarrow A$

ج) $T \leftarrow T + (W-2)$

پرسش‌های چهارگزینه‌ای

- ۱- کدامیک از ویژگی‌های الگوریتم محسوب نمی‌شود .
- الف) ترتیب مشخ
 - ب) خاتمه پذیر بودن
 - ج) جزئیات کافی
- ۲- الگوریتم زیر اعداد ۱۰ تا ۲۰ را روی صفحه ، نمایش می‌دهد ، خط شماره ۵ آن را با کدام گزینه باید کامل کرد .
- ۱ - شروع
 - ۲ - $I \leftarrow 9$
 - ۳ - $I \leftarrow I + 1$
 - ۴ - I را چاپ کن
 - ۵ - ؟
 - ۶ - پایان .
- الف) اگر $I <= 20$ سپس برو به خط ۳
- ب) اگر $I > 20$ سپس برو به خط ۳
- ج) اگر $I = 20$ سپس برو به خط ۳
- ۳- پس از اجرای الگوریتم زیر خروجی آن چه اعدادی هستند ؟
- ۱ - شروع
 - ۲ - $X \leftarrow 5$
 - ۳ - $Y \leftarrow X + 5$
 - ۴ - اگر $Y \leq X$ سپس برو به خط ۵ و گرنه X و Y را چاپ کن
 - ۵ - $X \leftarrow X + 2$
 - ۶ - $Y \leftarrow Y - 3$
 - ۷ - برو به خط ۴
- الف) ۱۰ و ۱۵
- ب) ۱۳ و ۱۳
- ج) ۱۶ و ۱۱
- ۴- کدام یک از گزینه‌های زیر عملکرد الگوریتم زیر را نشان می‌دهد ؟
- ۱ - شروع
 - ۲ - X را بخوان
 - ۳ - اگر $I < 10$ سپس $X \leftarrow -X$
 - ۴ - X را چاپ کن و پایان .
- الف) یک عدد را دریافت می‌کند و قسمت صحیح آن را می‌نویسد .

- ب) یک عدد را دریافت می کند و قدر مطلق آن را می نویسد.
 ج) یک عدد را دریافت می کند و اگر آن عدد منفی بود آن را می نویسد.
 د) یک عدد را دریافت می کند و همان را عیناً می نویسد.

۵- الگوریتم زیر چند ورودی و خروجی دارد ؟

۱- شروع

۲- A را بگیر

۳- A را بنویس

۴- $I \leftarrow 12$

۵- N را بگیر

۶- N+I را بنویس

۷- $I \leftarrow I + 2$

۸- اگر $I = 2$ سپس برو به خط ۴

۹- بنویس $A \times A$

۱۰- پایان .

- الف) ۱۱ ورودی و ۱۲ خروجی
 ب) ۷ ورودی و ۹ خروجی
 ج) ۲ ورودی و ۳ خروجی
 د) ۱۳ ورودی و ۱۵ خروجی
 ۶- خروجی الگوریتم زیر کدام است ؟

۱- شروع

۲- $Z \leftarrow \sqrt{2}$

۳- $I \leftarrow 1$

۴- $Z \leftarrow \sqrt{Z+2}$

۵- $I \leftarrow I + 1$

۶- اگر $I = 4$ سپس برو به خط ۴

۷- Z را بنویس

۸- پایان .

$$\sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}} \quad \text{ب)$$

$$\sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2}}}}} \quad \text{د)$$

$$\sqrt{2} + \sqrt{2} + \sqrt{2} + \sqrt{2} + \sqrt{2} \quad \text{الف)$$

$$4\sqrt{2} \quad \text{ج)$$

۷- در الگوریتم زیر معین کنید متغیر S چند بار نوشته می شود ؟

۱- شروع

$N \leftarrow 3 - ۲$

-۳ j , I را بخوان

$S \leftarrow I + j - ۴$

-۵ S را بنویس

$N \leftarrow N+1 - ۶$

-۷ اگر $6 <= N$ سپس برو به خط ۳

-۸ پایان .

الف) ۶ ب) ۳

ج) ۴ د) ۷

-۸ در الگوریتم زیر معین کنید شمارنده N چند بار تکرار می شود ؟

-۱ شروع

$N \leftarrow 0 - ۲$

-۳ I و Z را بخوان

$S \leftarrow I + j - ۴$

-۵ S را بنویس

$N \leftarrow N+2 - ۶$

-۷ اگر $7 < N$ سپس برو به خط ۳

-۸ پایان .

الف) ۴ ب) ۷

ج) ۳ د) ۵

-۹ در الگوریتم زیر معین کنید شمارنده N چند بار تکرار می شود .

-۱ شروع

-۲ I و Z را بخوان

$S \leftarrow i + j - ۳$

-۴ S را بنویس

$N \leftarrow N+2 - ۵$

-۶ اگر $7 < N$ سپس برو به خط ۲

-۷ پایان .

الف) ۵ ب) صفر

ج) ۳ د) نامعلوم

۱۰- در الگوریتم زیر به جای خط شماره ۴ کدام گزینه را باید قرار داد تا مجموع زیر در S ذخیره گردد.
 $S=1+3+5+\dots+99$

- شروع ۱

$S \leftarrow 0$ ۲

$I \leftarrow 1$ ۳

? ۴

$I \leftarrow I+2$ ۵

۶- اگر $I \leq 99$ سپس برو به خط ۴

۷- S را بنویس

۸- پایان .

ب) $I \leftarrow S$

د) $S \leftarrow S+I$

الف) $S \leftarrow I$

ج) $I \leftarrow I+S$

مسائل

۱- الگوریتمی بنویسید که چهار عدد a و b و c و d را به عنوان ورودی دریافت و اگر $a > c+d$ بود مقدار $a \times c$ و در غیر این صورت مقدار $b \times d$ را محاسبه و چاپ نماید.

۲- الگوریتمی بنویسید که مضارب عدد ۲ را تا ۱۰۰۰ یکی یکی تولید و چاپ نماید.

۳- الگوریتمی بنویسید که مضارب عدد ۹ تا ۹۰۰ را یکی یکی تولید و چاپ کرده و مجموع آنها را نیز محاسبه و چاپ نماید.

۴- الگوریتمی بنویسید که عدد طبیعی N و حقیقی X را سوال نموده و مقادیر زیر را محاسبه کند.

$$P = \frac{1}{3} + \frac{1}{6} + \frac{1}{9} + \dots + \frac{1}{N} \quad \text{(الف)}$$

$$S = 1 + X + X^2 + \dots + X^N \quad \text{(ب)}$$

$$\sin(X) = X - \frac{X^3}{3!} + \frac{X^5}{5!} + \dots + \frac{X^{(2N-1)}}{(2N-1)!} \quad \text{(ج)}$$

$$T = 1 \times 2 + 2 \times 3 + \dots + N(N-1) \quad \text{(د)}$$

۵- الگوریتمی بنویسید که درآمد سالانه یک نفر را به عنوان ورودی دریافت و مقدار مالیات را با توجه به ضوابط زیر محاسبه نماید.

الف) کمتر از ۵۰۰۰۰ تومان معاف

ب) بین ۵۰۰۰۰ و ۱۰۰۰۰۰ ، ۱۰٪ مازاد از ۵۰۰۰۰

ج) بیش از ۱۰۰۰۰۰ ، ۱۵٪ مازاد از ۱۰۰۰۰۰

الگوریتم را طوری تغییر دهید که N را به عنوان تعداد افراد گرفته و برای N نفر نیز کار کند.

۶- الگوریتمی بنویسید که عدد طبیعی N را به عنوان ورودی گرفته و معین کند آیا این عدد بر مجموع ارقامش بخش پذیر است یا نه؟

۷- الگوریتمی بنویسید که دو عدد صحیح M و N را دریافت و اعداد مضارب ۳ بین M و N را تولید و چاپ کند.

۸- الگوریتمی بنویسید که عددی از مبنای ۳ را به مبنای ۱۰ ببرد.

۹- الگوریتمی بنویسید که اعداد دو رقمی را تولید و چاپ کند که خودشان با معکوسشان برابرند.

۱۰- الگوریتمی بنویسید که تمام اعداد اول بزرگتر از ۲ و کوچکتر از ۱۰۰۰ را تولید و چاپ کند (همین طور اعداد تام).

۱۱- الگوریتمی بنویسید که عدد طبیعی N را دریافت و مقسوم علیه های مضرب ۴ آن را یکی یکی تولید و چاپ نماید، همچنین تعداد و مجموع آنها را محاسبه و چاپ کند.

۱۲- الگوریتمی بنویسید که عددی بین ۱ تا ۱۰۰۰ را که حدس زده شده است مشخص نماید.

۱۳- الگوریتمی بنویسید که بدون استفاده از متغیر سومی محتویات دو متغیر را جابجا نماید.

۱۴- الگوریتمی بنویسید که این اطلاعات را بخواند: نام شخص، موجودی، تعداد دفعات مراجعته به بانک برای برداشت و واریز پول و مبلغ آن. برای اینکه مشخص شود مبلغ مورد نظر برداشته با واریز

شده کدی در نظر گرفته شود (۱: واریز ، ۲: برداشت) سپس موجودی شخص را محاسبه و در خروجی چاپ کند.

۱۵- الگوریتمی بنویسید که اطلاعات زیر را در مورد هر یک از شرکت کنندگان یک آزمون بخواند. اطلاعات عبارتند از : نام ، شماره شرکت کننده ، کد نوع دیپلم (۰: تجربی ، ۱: انسانی ، ۲: فنی و کارودانش)، سن داوطلب ، کد رشته و کد جنسیت. سپس اطلاعات زیر را به خروجی ببرد :

- تعداد کل شرکت کنندگان پسر که دارای کد رشته ۲۴ باشند.

- تعداد شرکت کنندگان که سن آنها کمتر از ۱۸ سال است.

- درصد شرکت کنندگان که کد رشته آنها ۲۴ است.

۱۶- الگوریتمی بنویسید که این اطلاعات را برای چند فروشگاه بخواند : شماره فروشگاه و میزان فروش ماهیانه در سال اگر میزان فروش در سال کمتر یا مساوی ۵۰۰۰۰۰ باشد ، ۳٪ میزان فروش به فروشنده پرداخت می شود . اگر میزان فروش در سال بیش از ۵۰۰۰۰۰ و کمتر از ۷۰۰۰۰۰ باشد ۵٪ فروش به فروشنده داده خواهد شد . الگوریتم باید شماره فروشگاه ، میزان فروش و میزان پرداختی به فروشنده را چاپ کند و در پایان شماره فروشگاهی که بیشترین فروش را دارد در خروجی چاپ کند .

۱۷- فرض کنید در روز N ام سال هستیم . الگوریتمی بنویسید که تاریخ این روز را بنویسد (مثلاً ، اگر روز شصت و چهارم باشد ، باید ۳/۲ نوشته شود و اگر روز دویست و شانزدهم باشد ۷/۳۰) .

۱۸- الگوریتمی بنویسید که مختصات دو نقطه متمایز M و N از صفحه را بگیرد و معادله خطی را که از دو نقطه M و N می گذرد به شکل $AX+BY=C$ بنویسد .

۱۹- الگوریتمی بنویسید که عدد طبیعی N را دریافت و معین کند چند رقم آن زوج ، چند رقم آن فرد و چند رقم آن صفر می باشد .

۲۰- الگوریتمی بنویسید که اعداد ۴ رقمی را تولید کند که رقم یکان و صدگان آنها زوج ، اما رقم دهگان و هزارگان آنها فرد باشند.

۲۱- الگوریتمی بنویسید که کلیه اعداد دو رقمی را که از ارقام فرد تشکیل شده اند یکی یکی تولید و چاپ نماید .

۲۲- الگوریتمی بنویسید که کلیه اعداد سه رقمی را بنویسد که رقم یکان و هزارگان مساوی داشته و دهگان آنها زوج باشد .

۲۳- الگوریتمی بنویسید که تعداد نقاطی از داخل دایره $25 = Y^2 + X^2$ را که مختصاتشان اعداد طبیعی است، محاسبه و چاپ نماید .

۲۴- الگوریتمی بنویسید که تعیین کند یک سکه ۵۰ ریالی را به چند طریق می توان با سکه های ۲۰ ریالی ، ۱۰ ریالی و ۵ ریالی خرد کرد (لازم است از تمام سکه ها استفاده شود).

یادداشت‌های درس

فصل دوم

فلوچارت

در فصل اول یاد گرفتید که چگونه یک مسئله را حل کرده و برای آن الگوریتم بنویسید. الگوریتمهایی که نوشته‌ای از جملات فارسی برای بیان آنها استفاده شده بود، سپس آنها را خلاصه کرده و از عبارات ریاضی برای بیان جملات استفاده کردید، یعنی به دو طریق یک الگوریتم را نوشته‌ید که در هر دو، از ترکیب جملات فارسی، حروف انگلیسی و نمادهای ریاضی استفاده شده بود که این روشهای برای الگوریتمهایی که ساده بودند و تعداد دستورالعملهایشان کم بود بسیار کارآمد و مناسب بود. ولی برای مسائل پیچیده‌ای که از چند قسمت تشکیل شده‌اند و هر قسمت آن دارای تعداد زیادی دستورالعمل می‌باشد، روشهای ذکر شده دیگر مناسب نیستند همچنین اگر از شرط‌های تودرتو و پیچیده استفاده شده باشد، دیگر در ک عملکرد الگوریتم بسیار مشکل خواهد بود. به همین دلیل در این فصل از روش دیگری که تصویری است برای بیان الگوریتم استفاده می‌کنیم که به آن فلوچارت یا نمودارگردشی یا نمودار عملیاتی می‌گویند. همانطوریکه در الگوریتم از چند نوع جمله، برای بیان عملکرد آن استفاده کردیم، در فلوچارت نیز از نمادهای تصویری خاصی استفاده می‌کنیم (که به آنها مؤلفه‌های تصویری می‌گویند) که از به هم پیوستن آنها یک نمودار یا چارت خاصی بوجود می‌آید که به آن فلوچارت می‌گویند. نمادهای تصویری یا مؤلفه‌های تصویری که در ترسیم فلوچارت بکار می‌روند، حدوداً ۲۵ مؤلفه است (البته به طور استاندارد) که در قسمتهای بعدی هر یک را به‌طور کامل توضیح می‌دهیم.

تعريف فلوچارت

برای درک بهتر الگوریتم و سهولت در دنبال کردن دستورالعملهای آن از یکسری اشکال خاص برای نشان دادن الگوریتم استفاده می‌کنیم که به آن فلوچارت می‌گویند. یا به عبارت ساده‌تر :

به مجموعه‌ای از علائم ساده که الگوریتم را بصورت نمادهای تصویری یا نموداری تبدیل می‌کنند ، فلوچارت گفته می‌شود.

توضیح : اولین مرحله در حل یک مسئله با کامپیوتر، نوشتمن الگوریتم و مرحله دوم رسم فلوچارت می‌باشد(رسم فلوچارت درک روش حل را ساده‌تر می‌کند)، مرحله سوم نوشتمن برنامه به یکی از زبانهای برنامه نویسی است. لازم بذکر است که اگر بتوانید برای یک مسئله الگوریتم نوشت و فلوچارت مربوط به آن را رسم کنید، نوشتمن برنامه آن از روی فلوچارت بسیار ساده خواهد بود، لذا با دقت هر یک از فلوچارت‌هایی را که در این فصل بیان می‌شوند بررسی کنید سپس آنها را بدرستی درک نمایید و در نهایت سعی کنید حداقل یک بار از روی آنها بنویسید .

علائم فلوچارت (مؤلفه‌های تصویری)

علامت شروع و پایان

در ابتدای هر فلوچارت از علامت بیضی به مفهوم شروع و خاتمه عملیات استفاده می‌کنیم.

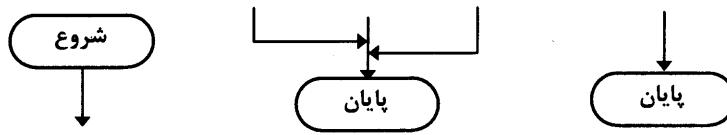


نکته‌ای قابل توجه این است که چون هر الگوریتمی فقط دارای یک شروع و پایان است لذا فلوچارت مربوط به آن نیز فقط دارای یک شروع و پایان خواهد بود.

اشتباه نکنید: در بعضی از الگوریتمهای فصل اول، در انتهای بعضی از خطوط دستور پایان را می‌نوشتید که این به مفهوم داشتن چند پایان نیست یعنی صحیح آن است که همواره در ابتدای الگوریتم شروع و در انتهای آن پایان بنویسید و در هر خطی که نیاز به دستور پایان داشتید به آخرین خط ارجاع دهید.

برای اتصال دادن مؤلفه‌های تصویری در ترسیم فلوچارت از علامت \rightarrow استفاده کنید (با توجه به این که چگونه آنها را به هم متصل کنیم جهت و اندازه آن اختیاری خواهد بود).

از توضیحات بالا در می‌یابیم که از بیضی مربوط به شروع فقط یک فلاش به بیرون منشعب شده ولی به بیضی مربوط به پایان می‌تواند چند فلاش از جاهای مختلف ممتد شود که باز متذکر می‌شویم که این به مفهوم داشتن چند پایان نیست.



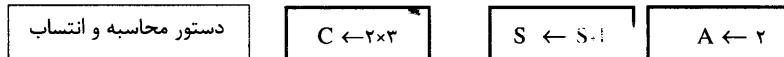
فقط یک فلاش خارج می‌شود

چند فلاش وارد شود

یک فلاش وارد شده است

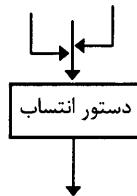
علامت جایگزینی و انتساب

از علامت مستطیل برای نمایش یک عمل، بخصوص عملیات محاسبه‌ای یا جایگزینی مقداری در یک خانه و کلاً عبارتهاي انتسابی استفاده می‌کنیم.



مانند :

در مورد این علامت نیز می‌توانید چند فلاش به آن وارد ولی فقط یک فلاش از آن خارج کنید.

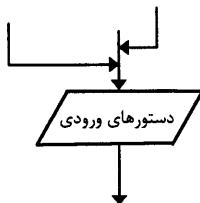


علامت ورودی

از علامت متواری الاضلاع برای نمایش عملیات مربوط به گرفتن مقادیر و داده‌ها به عنوان ورودی استفاده کنید.



در مورد این علامت نیز می‌توانید چند فلش به آن منتهی ولی فقط یک فلش از آن خارج کنید.



علامت شرطی

می‌دانید که در عبارات شرطی، انجام عملیات منوط به برقرار بودن شرط یا شرط‌هایی می‌باشد ، به همین منظور از علامت لوزی برای دستورات شرطی استفاده کنید .

به دستورات شرطی عبارتهاي تصميم‌گيری نيز گفته می‌شود. شرط یا شرط‌ها را داخل لوزی بنويسيد.



در اين علامت برخلاف علامتهاي قبلی می‌توان چند فلش را وارد کرده و دو يا سه فلش از آن با توجه به ارزش شرط(ها) (يعني درست يا نادرست بودن) خارج کرد. مانند :



سه فلش به عنوان خروجي

دو فلش به عنوان خروجي

علامت چاپ

اگر بخواهید مقداری را روی صفحه کاغذ یا صفحه نمایش، نشان دهید از علامت زیر استفاده کنید و مقادیری که می‌خواهید چاپ شوند را داخل آن بنویسید.



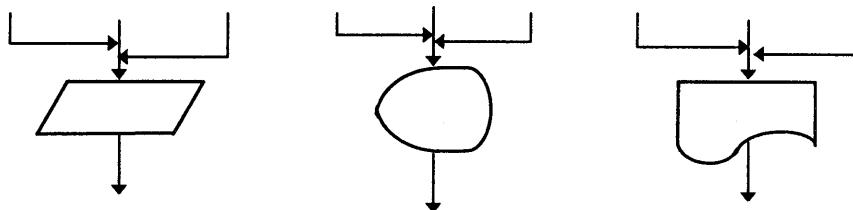
برای چاپ روی صفحه کاغذ



برای چاپ روی صفحه نمایش

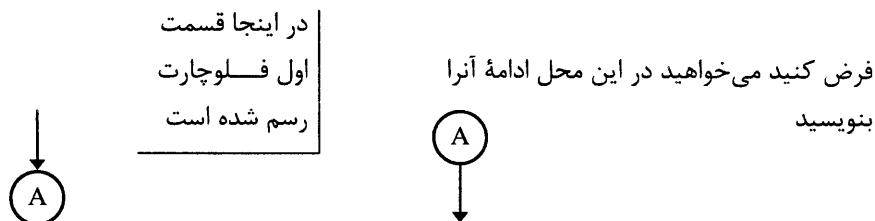
اگر هدف فقط نمایش دادن اطلاعات باشد ، می‌توانید از همان علامت متواری الاضلاع استفاده کنید .

در این علامتها مانند بیضی و مستطیل فقط یک فلش از آنها خارج ولی چند فلش به آنها وارد کرد.



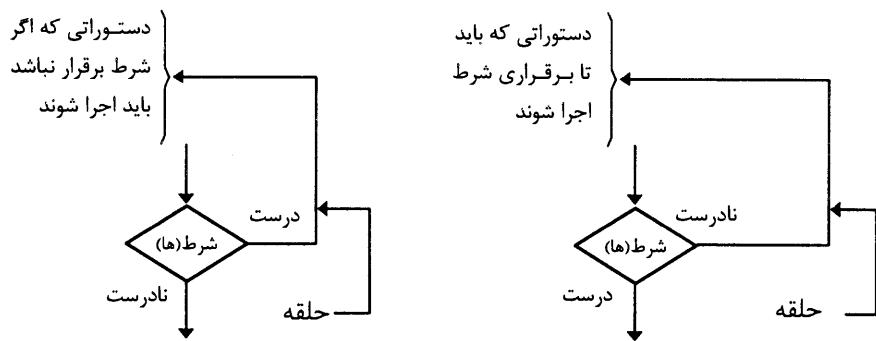
علامت ادامه

اگر فلوچارت را از یک قسمت قطع کرده و بقیه آن را در محل دیگر بنویسید، برای اتصال دادن این قسمتها به هم از علامت زیر استفاده کنید.



علامت حلقه

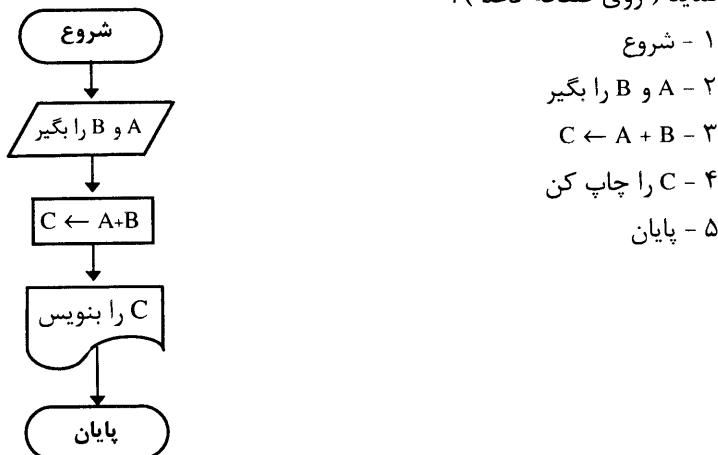
برای ساخت حلقه‌ها از ترکیب یک شرط و یک سری دستورالعمل استفاده می‌کنیم که علامت فلوچارت آن‌ها نیز بصورت زیر است.



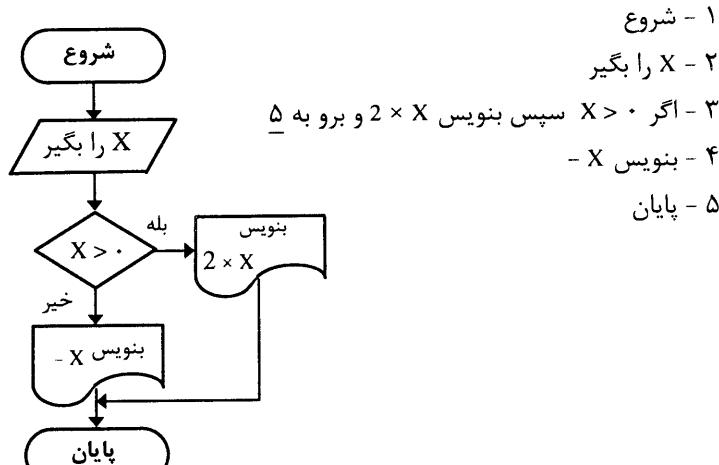
توضیح: مؤلفه تصویری خاصی برای نمایش زیر الگوریتم‌ها وجود دارد که در فصل پنجم (زیر الگوریتم‌ها) آنها را بیان و بررسی خواهیم نمود.
همچنین برای نمایش حلقه‌ها از اشکال دیگری استفاده می‌شود که بعد از آشنا شدن با این فرم نمایش، اشکال دیگر آن را معرفی خواهیم کرد.

برای روشنتر شدن مطلب روند کار را ابتدا با چند مثال ساده آغاز می‌کنیم، سپس فلوچارت هر یک از الگوریتم‌های فصل یک را رسم می‌کنیم.

مثال ۱) فلوچارتی رسم کنید که دو عدد A و B را به عنوان ورودی گرفته و حاصل جمع آنها را چاپ نماید (روی صفحه کاغذ).



مثال ۲) فلوچارتی رسم کنید که عدد X را به عنوان ورودی دریافت و اگر X مثبت بود، آنرا در ۲ ضرب کرده و چاپ نماید و در غیر اینصورت، نقیض آنرا چاپ نماید. (روی صفحه کاغذ)



مثال ۳) فلوچارتی رسم نماید که دو مقدار Y و X را به عنوان ورودی دریافت و اگر $Y > X$ بود، مجموع X و Y و اگر $Y < X$ بود، تفاضل X و Y و اگر مساوی بودند، یکی از آنها را روی صفحه کاغذ چاپ نماید.

1 - شروع

۲ - $X + Y$ را بگیر

۳ - اگر $Y > X$ سپس بنویس $Y + X$ و برو به ۶

۴ - اگر $Y < X$ سپس بنویس $X - Y$ و برو به ۶

۵ - بنویس X

۶ - پایان

چون ترکیب سه شرط و درک آن مشکل است بهتر است مقدار $Y - X$ را در متغیری به نام A ذخیره کرده و بعد روی A تصمیم-گیری نمایید.

۱ - شروع

۲ - $X + Y$ را بگیر

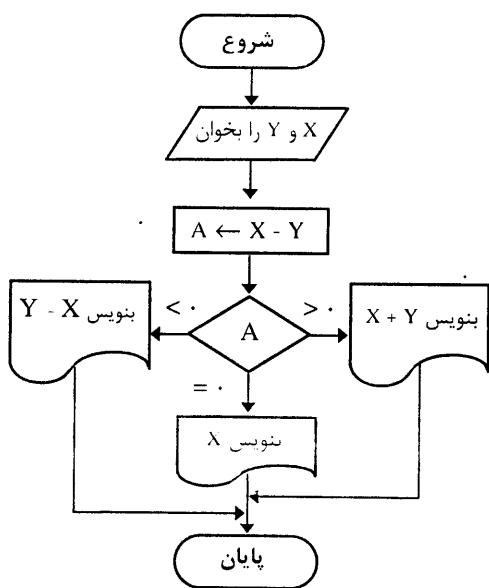
$A \leftarrow X - Y$ - ۳

۴ - اگر $A > 0$ سپس بنویس $Y + X$ و برو به ۷

۵ - اگر $A < 0$ سپس بنویس $X - Y$ و برو به ۷

۶ - بنویس X

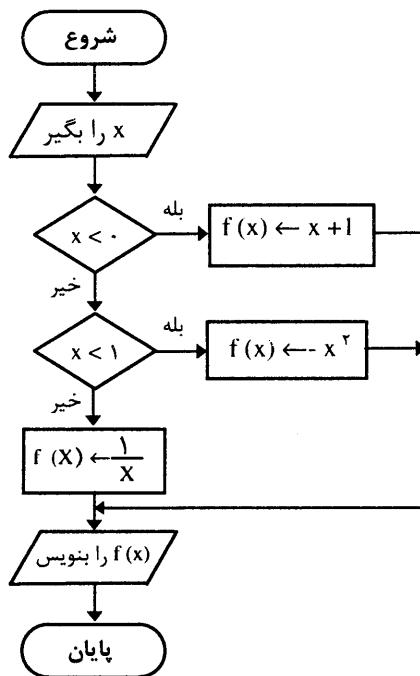
۷ - پایان



مثال ۴) فلوچارتی رسم کنید که عدد حقیقی x را دریافت و مقدار تابع چند ضابطه ای زیر را محاسبه و چاپ نماید.

$$f(x) = \begin{cases} x + 1 & x < 0 \\ -x^2 & 0 \leq x < 1 \\ \frac{1}{x} & 1 \leq x \end{cases}$$

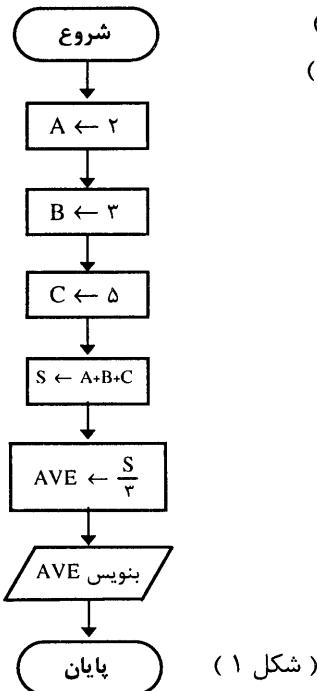
۱ - شروع

۲- x را بگیر۳- اگر $x < 0$ سپس $f(x) \leftarrow x + 1$ و برو به ۴۴- اگر $x < 1$ سپس $f(x) \leftarrow -x^2$ و برو به ۵

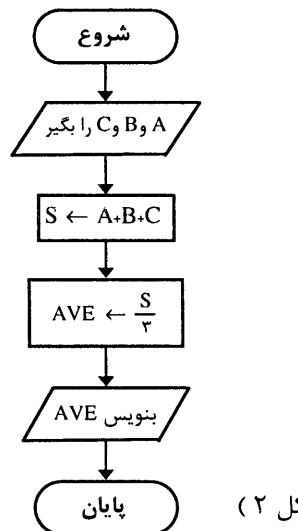
$$f(x) \leftarrow \frac{1}{x} \quad \text{۵}$$

۶- $f(x)$ را بنویس

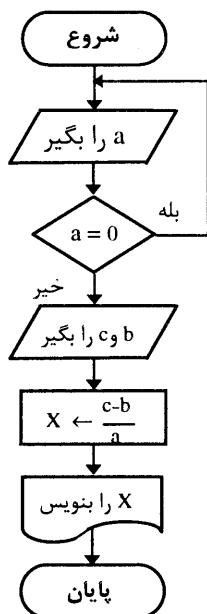
۷- پایان



مثال ۵) فلوچارت مثال ۱ فصل اول را رسم کنید. (شکل ۱)
 مثال ۶) فلوچارت مثال ۲ فصل اول را رسم کنید. (شکل ۲)



مثال ۷) فلوچارتی رسم نماید که جواب معادله درجه اول یک مجهولی $aX + b = c$ ($a \neq 0$) را محاسبه و روی صفحه کاغذ چاپ نماید.



- ۱ - شروع
- ۲ - را بگیر a
- ۳ - اگر $a = ۰$ سپس برو به ۲
- ۴ - را بگیر c و b
- ۵ - $X ← \frac{c-b}{a}$
- ۶ - X را چاپ کن
- ۷ - پایان

مثال ۸) فلوچارتی رسم کنید که جوابهای حقیقی معادله درجه دوم یک مجهولی $aX^2 + bX + c = 0$ را محاسبه و چاپ نماید.

۱ - شروع

۲ - A را بگیر

۳ - اگر $a = 0$ سپس برو به ۲

۴ - b و c را بگیر

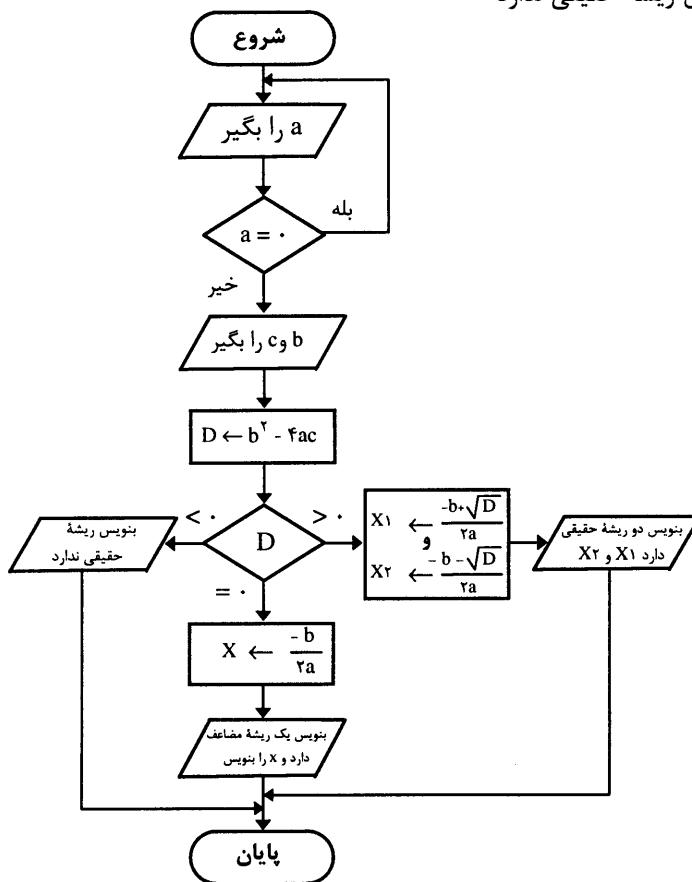
۵ - $D \leftarrow b^2 - 4ac$

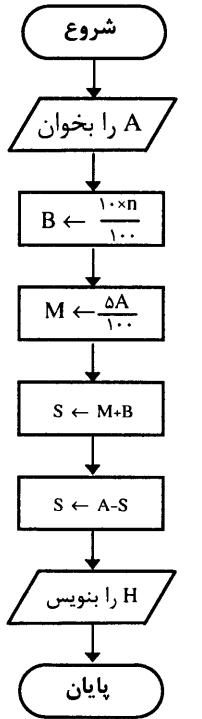
۶ - اگر $D > 0$ سپس $X_1 \leftarrow \frac{-b + \sqrt{D}}{2a}$ و $X_2 \leftarrow \frac{-b - \sqrt{D}}{2a}$ را بنویس و برو به ۹

۷ - اگر $D = 0$ سپس $X \leftarrow \frac{-b}{2a}$ و بنویس یک ریشه مضاعف دارد و X را بنویس و برو به ۹

۸ - بنویس ریشه حقیقی ندارد

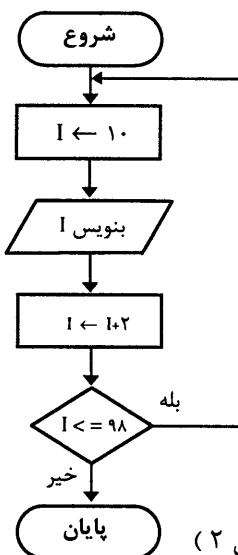
۹ - پایان





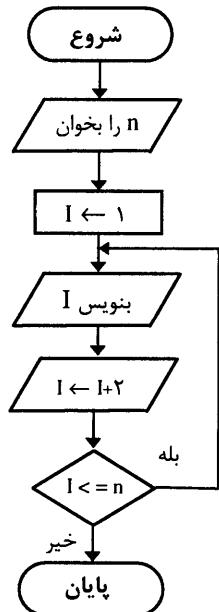
(شکل ۱)

مثال ۹) فلوچارت مثال ۳ فصل اول را رسم کنید. (شکل ۱)



(شکل ۲)

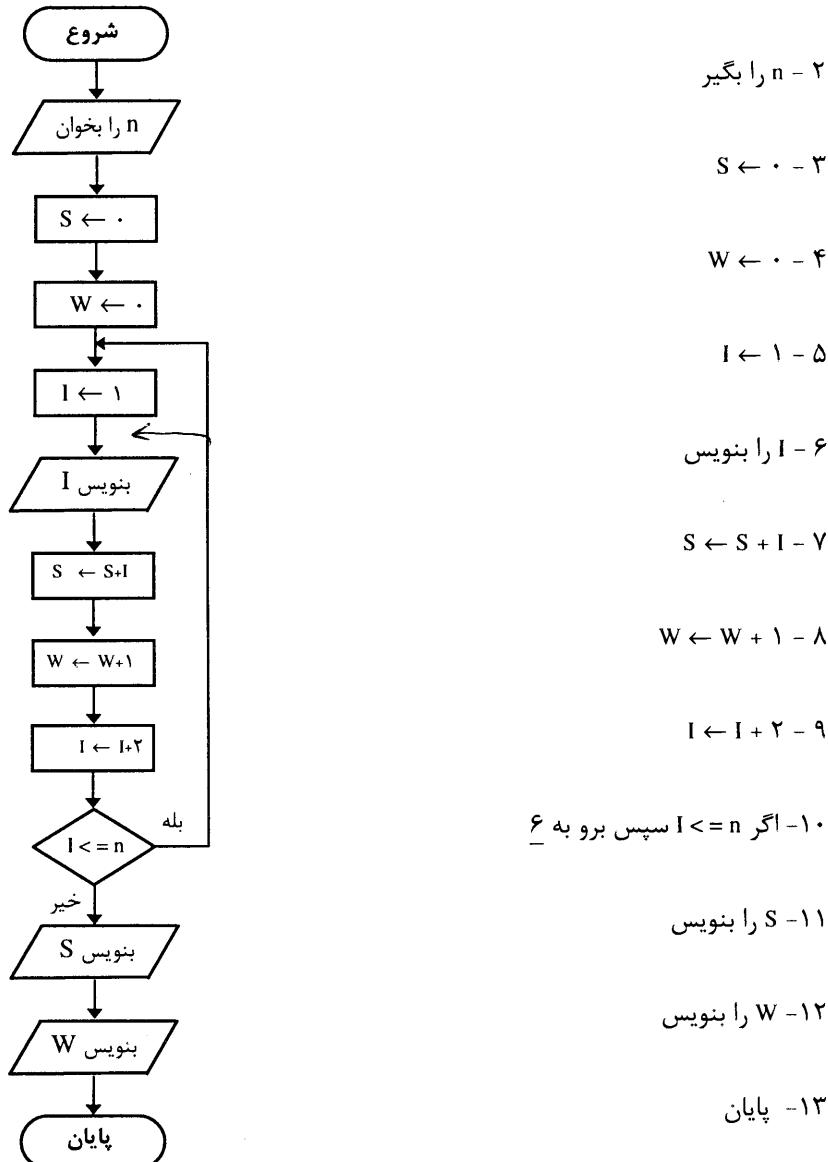
مثال ۱۰) فلوچارت مثال ۵ فصل اول را رسم کنید. (شکل ۲)



مثال ۱۱) فلوچارت مثال ۶ فصل اول را رسم نمائید.
 اگر بخواهید این مثال را طوری تغییر دهید که
 مجموع و تعداد آنها را نیز محاسبه و چاپ نماید،
 خانه‌ای به نام S برای مجموع و خانه‌ای به نام W
 برای تعداد در نظر گرفته و بعد از چاپ در داخل
 حلقه با S جمع کرده و به W یکی اضافه کنید.

حال الگوریتم و فلوچارت مثال قبل را طوری تغییر دهید که مجموع و تعداد آنها را نیز محاسبه و چاپ کند.

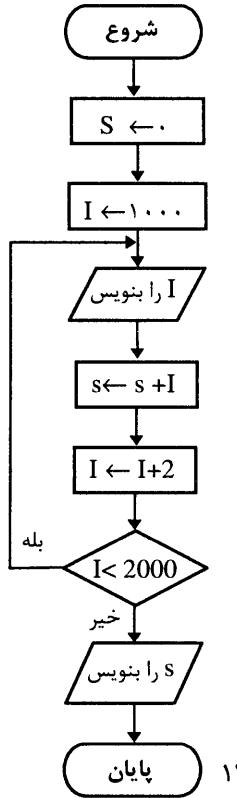
۱ - شروع



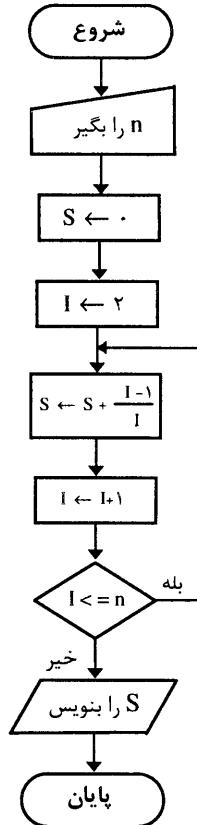
مثال ۱۲) فلوچارتی رسم کنید که عدد صحیح و مثبت N را به عنوان ورودی از طریق صفحه کلید دریافت و سپس حاصل جمع زیر را محاسبه کرده و روی صفحه کاغذ چاپ نماید.

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

توضیح: در بعضی از مسائل گفته می‌شود که مقداری را از صفحه کلید، به عنوان ورودی دریافت نماید
که به جای استفاده از علامت متوازن اضلاع از علامت استفاده کنید.



شكل مثال ۱۳



شكل مثال ۱۲

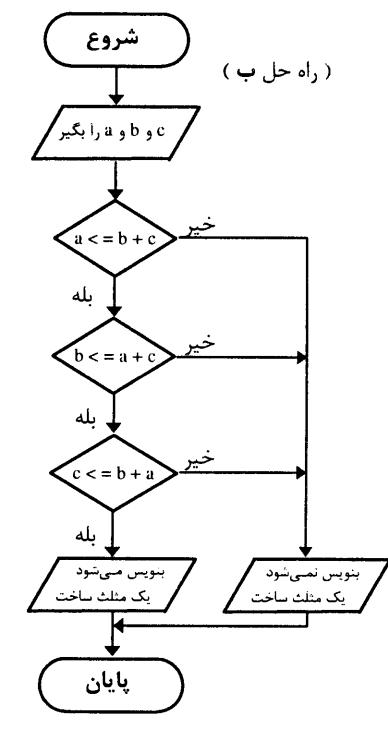
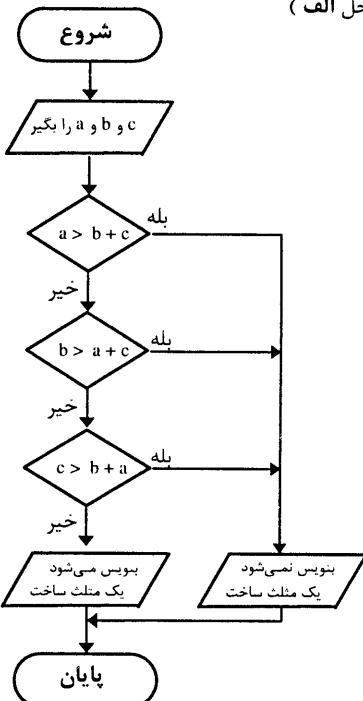
- ۱ - شروع
- ۲ - n را بگیر
- ۳ - $S \leftarrow 0$
- ۴ - $I \leftarrow 2 - 4$
- ۵ - $S \leftarrow S + \frac{I-1}{I} - 5$
- ۶ - $I \leftarrow I + 1 - 6$
- ۷ - اگر $I \leq n$ سپس برو به ۵
- ۸ - S را چاپ کن
- ۹ - پایان

مثال ۱۳) فلوچارت مثال ۷ فصل اول را رسم نماید.

توضیح: برای اینکه از تعداد علائم کمتری استفاده کنید، می‌توانید دو یا چند دستور العمل را در یک علامت قرار دهید، البته این کار باید با توجه به منطق فلوچارت انجام گیرد.

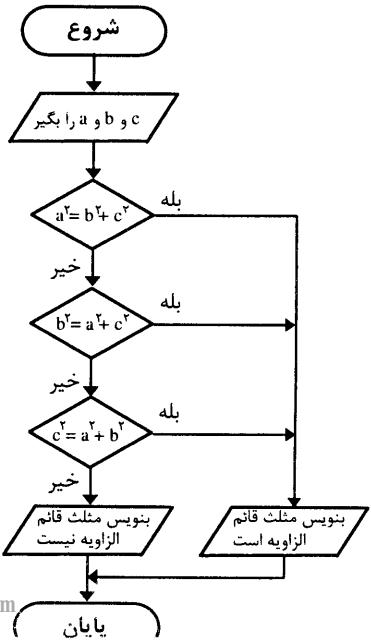
مثال ۱۴) فلوچارت مثال ۱۱ فصل اول را رسم نماید.

(راه حل الف)

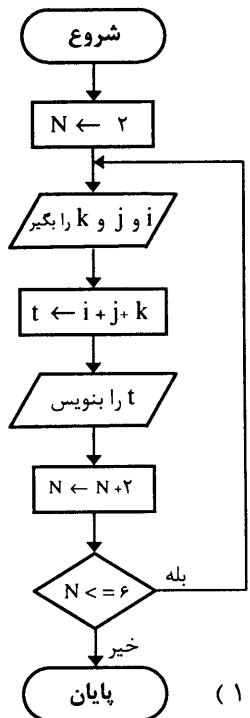


سعی کنید الگوریتم راه حل ب را خودتان بنویسید.

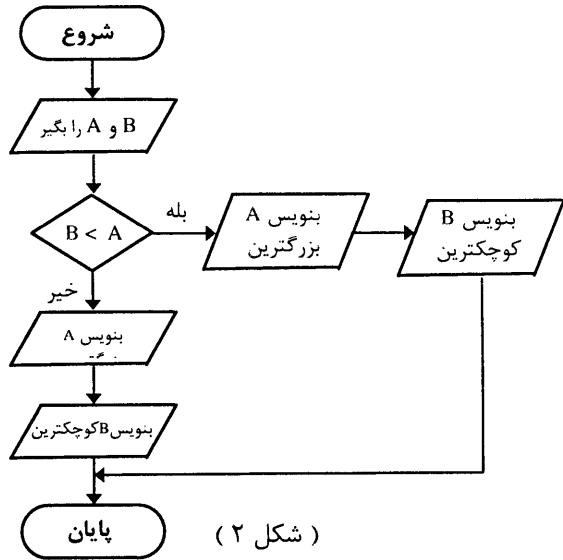
مثال ۱۵) فلوچارت مثال ۱۲ فصل اول را رسم نماید.



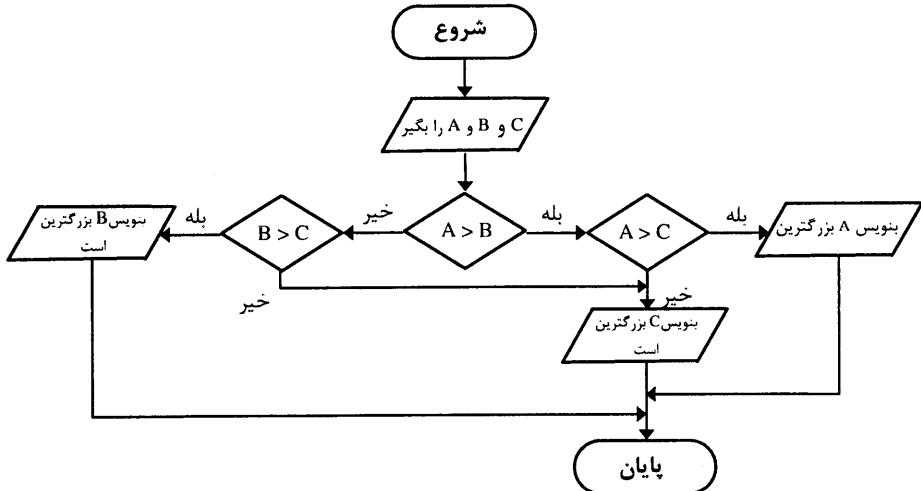
مثال ۱۶) فلوچارت مثال ۱۴ فصل اول را رسم نماید. (شکل ۱)



مثال ۱۷) فلوچارت مثال ۱۵ فصل اول را رسم نماید. (شکل ۲)



مثال ۱۸) فلوچارت مثال ۱۶ فصل اول را رسم نماید.



مثال ۱۹) فلوچارتی رسم کنید که سه عدد را دریافت و آنها را به ترتیب صعودی مرتب کند (بدون استفاده از روش‌های SORT کردن)

۱- شروع

۲- C,B,A را بگیر

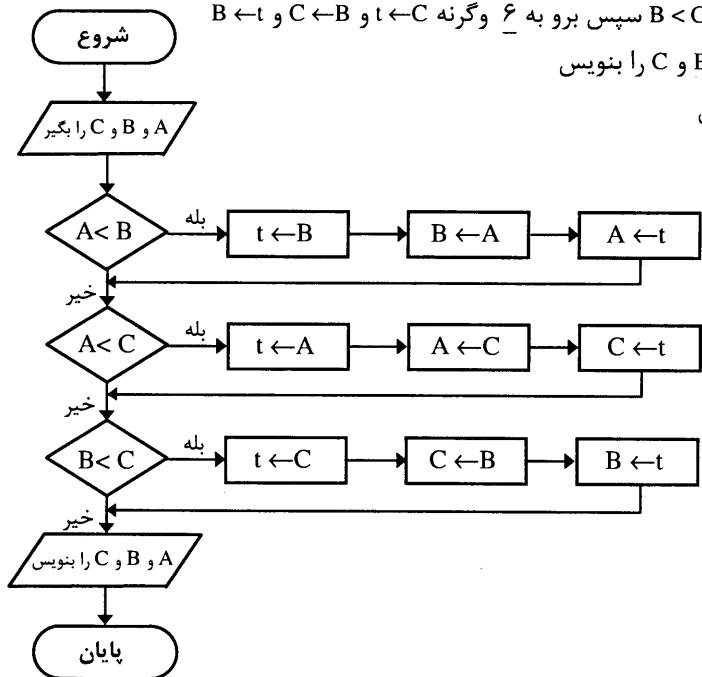
۳- اگر $A < B$ سپس برو به ۴ و گرنه $t \leftarrow B$ و $B \leftarrow A$ و $A \leftarrow t$

۴- اگر $A < C$ سپس برو به ۵ و گرنه $t \leftarrow A$ و $A \leftarrow C$ و $C \leftarrow t$

۵- اگر $B < C$ سپس برو به ۶ و گرنه $t \leftarrow C$ و $C \leftarrow B$ و $B \leftarrow t$

۶- A و B و C را بنویس

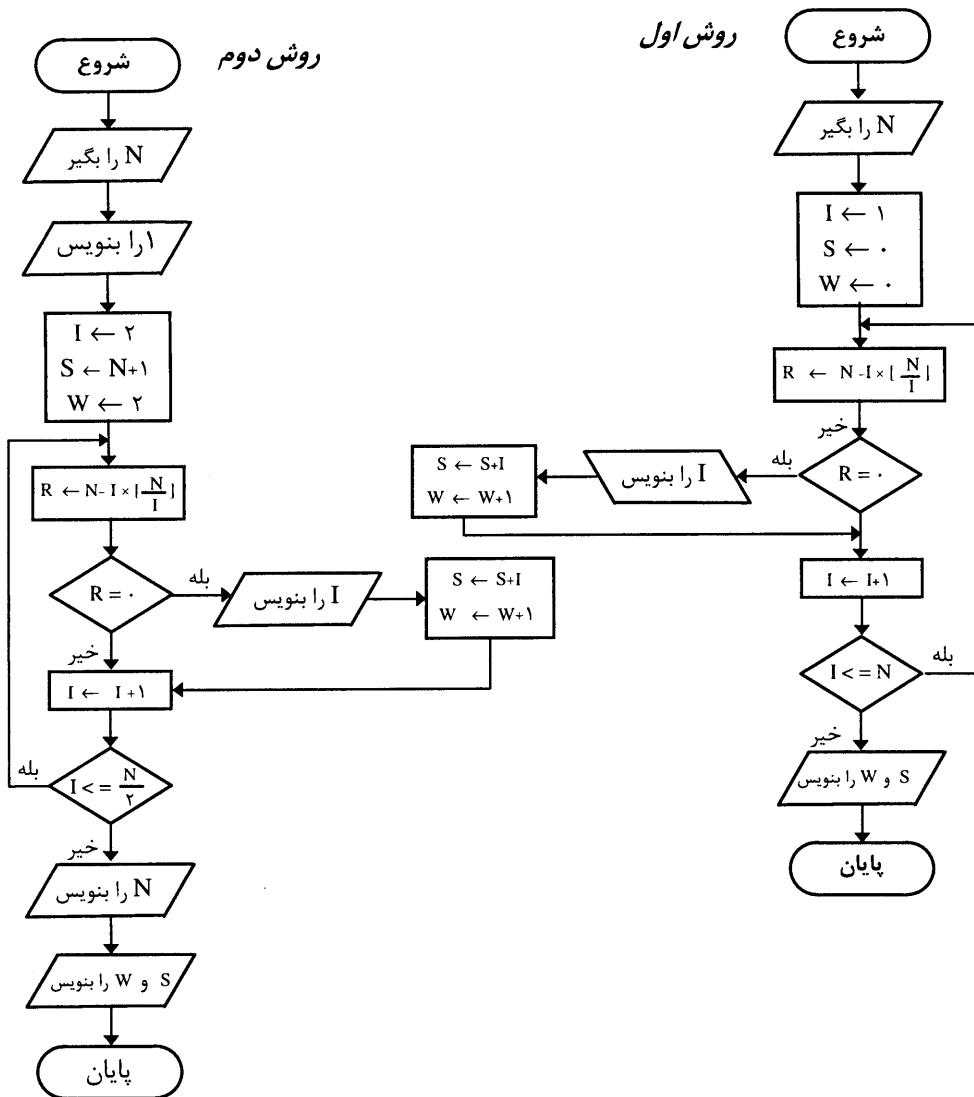
۷- پایان



توضیح : در اینجا از سه شرط استفاده شده ، اگر تعداد اعداد زیاد باشد تعداد شرطها به مراتب بیشتر می‌شود . به همین دلیل این روش، روش مناسبی برای محاسبه بزرگترین یا کوچکترین عدد از بین یک سری عدد نمی‌باشد .

روشهای کارآمدی وجود دارند که در آینده با آنها آشنا خواهید شد.

مثال ۲۰) فلوچارت مثال ۱۸ فصل اول را رسم نمایید.
توضیح: دو الگوریتم مختلف برای این مثال در فصل اول نوشته شد که فلوچارت هر یک را جداگانه رسم می‌کنیم.



هر دو روش جالب هستند، اما ، بهتر است بدانید که اگر برای هر کدام برنامه‌ای به یکی از زبانها نوشته شود و هر دو برای یک عدد کوچک آزمایش نمایید ، زمان اجرای آنها تقریباً برابر خواهد بود اما برای اعداد بزرگتر (مثلاً ۸ رقمی) اگر زمان اجرای دو برنامه را محاسبه کنیم ، برنامه نوشته شده به روش دوم نصف زمان اجرای برنامه به روش اول خواهد بود. برای سادگی برنامه هر دو روش را به زبان BASIC در زیر می‌آوریم .

برنامه روش اول :

```

10 T = TIMER
20 INPUT N
30 I = 1
40 S = 0
50 W = 0
60 R = N - I * INT ( N / I )
70 IF R = 0 THEN PRINT I, : S = S + I : W = W + 1
80 I = I + 1
90 IF I <= N THEN GOTO 60
100 PRINT
110 PRINT " SUM = " ; S
120 PRINT " NUMBER = " ; W
130 T = TIMER - T
140 PRINT " TIME = " ; T
150 END

```

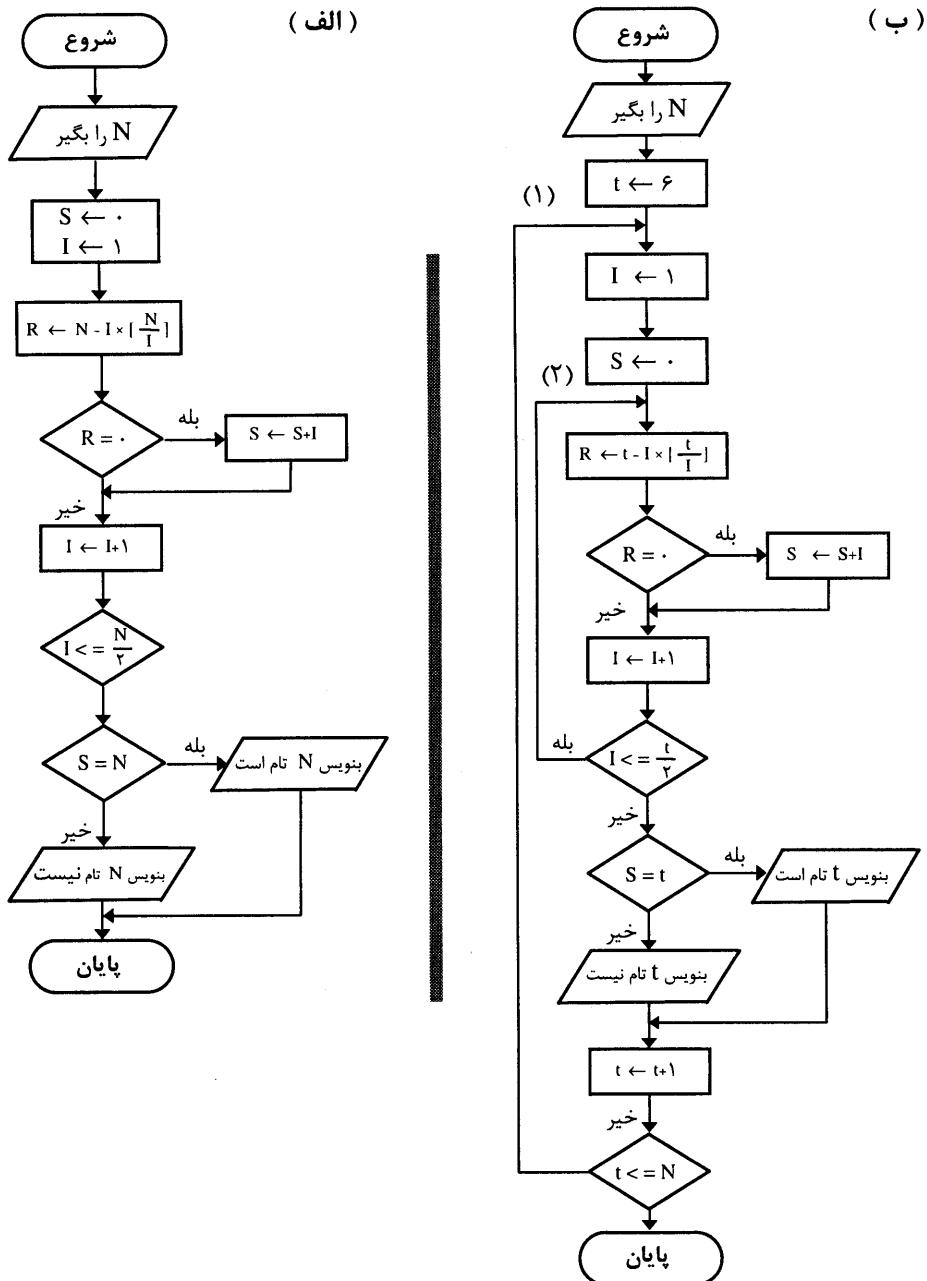
برنامه روش دوم :

```

10 T = TIMER
20 INPUT N
30 PRINT 1
40 I = 2
50 S = N + 1
60 W = 2
70 R = N - I * INT ( N / I )
80 IF R = 0 THEN PRINT I, : S = S + I : W = W + 1
90 I = I + 1
100 IF I <= N/2 THEN GOTO 70
110 PRINT N
120 PRINT
130 PRINT " SUM = " ; S
140 PRINT " NUMBER = " ; W
150 T = TIMER - T
160 PRINT " TIME = " ; T
170 END

```

مثال ۲۱) فلوچارت مثال ۱۹ فصل اول را رسم نمایید، سپس آنرا طوری تغییر دهید که اعداد تمام کوچکتر از عدد معین N ای را محاسبه و چاپ نماید.



در فلوچارت (ب) حلقهای ۱ و ۲، تام بودن تمام اهایی که از $\underline{6}$ تا N تغییر می‌کنند را محاسبه می‌کند (که به این حلقه‌ها، حلقه‌های تودرتو می‌گویند). ملاحظه می‌شود که این روش مانند فلوچارت (الف) می‌باشد، با این فرض که یک حلقه به آن اضافه شده است. الگوریتم آن نیز بصورت زیر می‌باشد :

۱ - شروع

۲ - N را بگیر

$t \leftarrow 6 - 3$

$S \leftarrow 0 - 4$

$I \leftarrow 1 - 5$

$R \leftarrow N - I \times \left[\frac{N}{I} \right] - 6$

۷ - اگر $S = R$ سپس $R = 0$

$I \leftarrow I + 1 - 8$

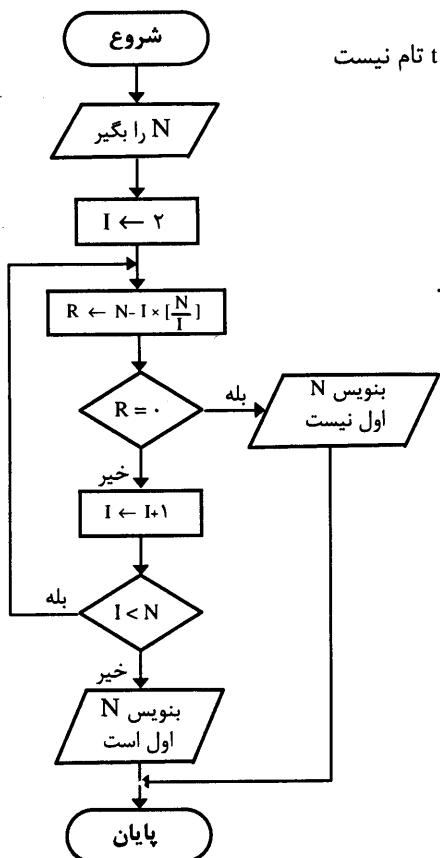
۹ - اگر $\frac{t}{2} = S$ سپس برو به ۶

۱۰ - اگر $S = t$ سپس بنویس t تام است و گرنه بنویس t تام نیست

$t \leftarrow t + 1 - 11$

۱۲ - اگر $N = t$ پس برو به ۴

۱۳ - پایان

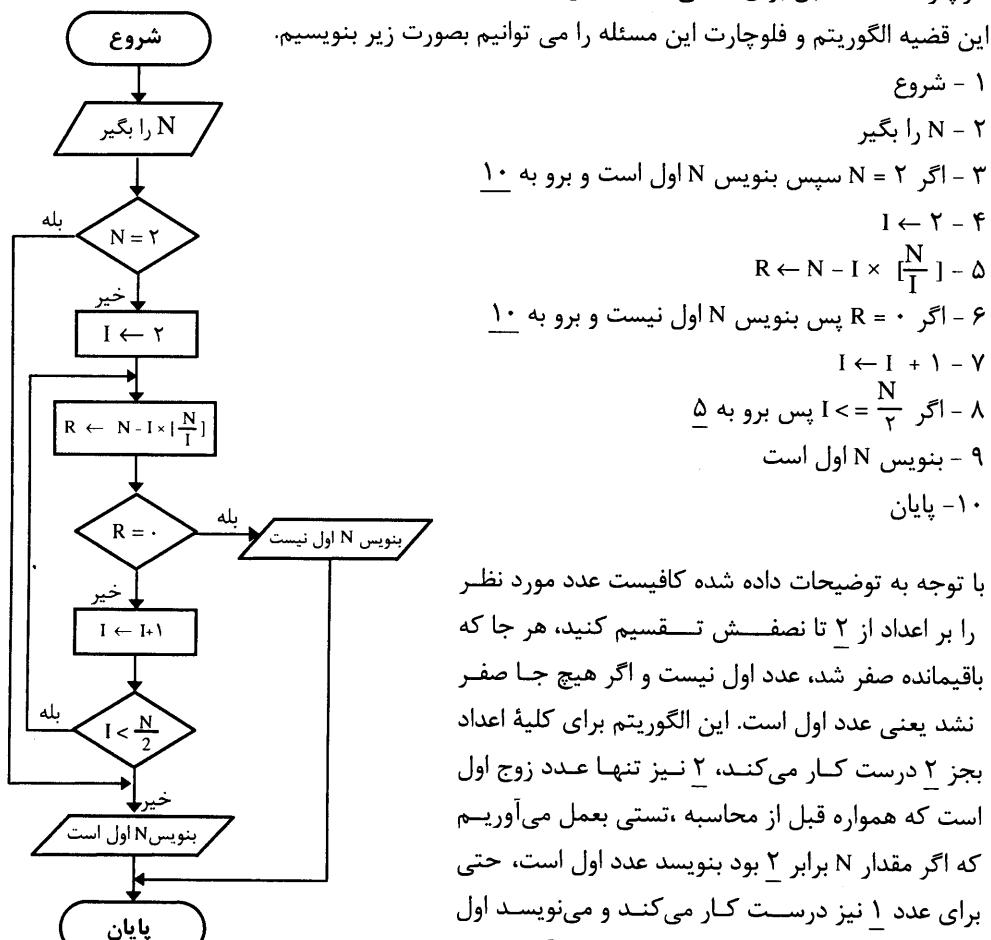


مثال ۲۲) فلوچارت مثال ۲۰ فصل اول را ترسیم کنید.

توضیح: در این مثال اگر N را ۲ وارد کنید، بیان می‌کند ۲ عدد اول نیست که نادرست می‌باشد پس در ابتدای فلوچارت باید آزمایش کنید که اگر عدد وارد شده ۲ یا ۱ بود، بنویسد عدد اول است، همچنین برای عدد وارد شده N از ۲ تا عدد کوچکتر از N ، $(N-1)$ ، امتحان می‌کنیم . در قضیه صفحه ۳۲ فصل اول ثابت کردیم که از نصف عدد تا خود عدد هیچ عددی وجود ندارد که عدد N بر آن تقسیم شود و باقیمانده صفر گردد پس بهتر است که تا نصف N امتحان کنیم ، که در نتیجه زمان اجرای آن خیلی کمتر خواهد شد .

حال با توجه به توضیحات داده شده الگوریتم و فلوچارت آنرا بصورت می‌نویسیم :
فلوچارت صفحه قبل برای حالتی است که از قضیه صفحه ۳۲ استفاده نشده است با توجه به مفروضات

این قضیه الگوریتم و فلوچارت این مسئله را می‌توانیم بصورت زیر بنویسیم.



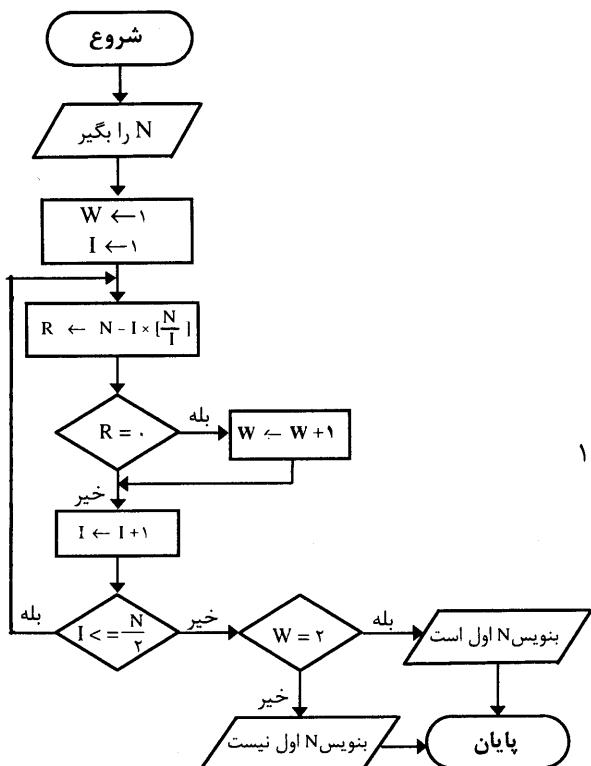
با توجه به توضیحات داده شده کافیست عدد مورد نظر را بر اعداد از $\underline{2}$ تا $\underline{\text{نصفش}} \text{ تقسیم} \text{ کنید}$ ، هر جا که باقیمانده صفر شد، عدد اول نیست و اگر هیچ جا صفر نشد یعنی عدد اول است. این الگوریتم برای کلیه اعداد بجز $\underline{2}$ درست کار می‌کند، $\underline{2}$ نیز تنها عدد زوج اول است که همواره قبل از محاسبه، تستی عمل می‌آوریم که اگر مقدار N برابر $\underline{2}$ بود بنویسد عدد اول است، حتی برای عدد $\underline{1}$ نیز درست کار می‌کند و می‌نویسد اول است. به عنوان تمرین خودتان بصورت دستی آن را یک بار اجرا نمائید.

حتی شرط قویتری وجود دارد که تحت عنوان قضیه‌ای آنرا در زیر می‌آوریم، این قضیه بیان می‌کند که برای اینکه ببینیم عدد اول است یا خیر کافیست عدد را به اعداد اول از $\underline{2}$ تا $\underline{\text{جذر آن}} \text{ تقسیم} \text{ کنیم}$ ، هر جا باقیمانده صفر شد عدد اول نیست و اگر در هیچ کجا صفر نشد عدد اول است، که با این شرط زمان اجرای آن باز هم کمتر می‌شود که برای مقایسه، برنامه‌های آنها را به زبان Basic نوشتایم، این برنامه‌ها را برای یک عدد بزرگ امتحان کنید تا زمان اجرای هر کدام را مشاهده نمائید.

قضیه: اگر $n > 1$ عدد صحیح غیر اول باشد، آنگاه عدد اولی مانند P وجود دارد که $P < \sqrt{n}$ و $P \mid n$ (علامت | به معنی بخش پذیر بودن است). اثبات آنرا در کتابهای نظریه اعداد می توانید مطالعه کنید. نتیجه: از قضیه فوق نتیجه می گیریم که برای تشخیص اینکه عدد n اول است یا نه، کافی است تحقیق کنیم که n بر اعداد اول P که $P < \sqrt{n}$ است بخش پذیر باشد، به این ترتیب دامنه اعدادی که باید عدد n را بر آنها تقسیم کند محدود تر می شود.

الگوریتم و فلوچارت مربوط به قضیه فوق همانند مثال قبلی است، با این تفاوت که بجای $\frac{N}{2}$ مقدار \sqrt{N} را قرار می دهیم (به عنوان تمرین الگوریتم و فلوچارت قبلی را تغییر داده و بنویسید).

روش ساده دیگری نیز برای محاسبه اول بودن یا نبودن عدد وجود دارد و عبارتست از اینکه مقسوم علیه های یک عدد را حساب کنید و اگر تعداد آنها ۲ عدد بود یعنی بجز ۱ و خود عدد، مقسوم علیه دیگری نداشت، عدد اول است و در غیر اینصورت عدد اول نیست. در ذیل الگوریتم و فلوچارت این روش را بیان می کنیم:



- ۱ - شروع
- ۲ - N را بگیر
- ۳ - W ← ۱
- ۴ - I ← ۱
- ۵ - R ← N - I × [N / I]
- ۶ - اگر W = ۰ سپس R = ۰
- ۷ - I ← I + ۱
- ۸ - اگر I ≤ N / 2 سپس برو به ۹
- ۹ - اگر W = ۲ سپس بنویس عدد اول است و برو به ۱۰
- ۱۰ - بنویس عدد اول نیست
- ۱۱ - پایان

تذکر : عموماً گفته می شود که عدد یک اول نیست. اما دلیلی برای آن ارایه نمی شود. توضیحی را که در این مورد لازم می باشد بیان کنیم اینست که اولاً عدد یک اول است ثانیاً علت به حساب نیاوردن آن در مجموعه اعداد اول آن است که اصل تجزیه را نقض می کند.

بنابر اصل تجزیه ، هر عدد طبیعی یا صحیح را می توان بصورت یکتاً بدون در نظر گرفتن ترتیب عاملها به حاصل ضرب عاملهای اول تجزیه کرد، به عنوان مثال :

$$12 = 2^2 \times 3$$

$$120 = 2^3 \times 3 \times 5$$

اگر عدد یک را هم جزء اعداد اول حساب کنیم، می توانیم بنویسیم :

$$12 = 2^2 \times 3 \times 1$$

بعبارت دیگر فرض یکتاً بودن تجزیه از بین می روید . بهمین دلیل عدد یک که عددی اول است را در این مجموعه در نظر نمی گیریم ، اما یک همواره عددی اول است. در اینجا برنامه های چهار روش فوق را می آوریم تا آنها را برای یک عدد تست نموده و زمانهای بدست آمده را با یکدیگر مقایسه کنید.

روش اول :

```

10 T = TIMER
20 INPUT N
30 I=2
40 R=N-I * INT (N/I)
50 IF R=0 THEN PRINT N ; "IS NOT PRIME" : GOTO 90
60 I=I+1
70 IF I<N THEN GOTO 40
80 PRINT N ; "IS PRIME" : GOTO 90
90 T = TIMER-T
100 PRINT "TIME =" ; T
110 END

```

روش دوم :

```

10 T = TIMER
20 INPUT N
30 IF N=2 THEN PRINT N : "IS PRIME" : GOTO 100
40 I=2
50 R = N-I * INT (N/I)
60 IF R=0 THEN PRINT N ; "IS NOT PRIME" : GOTO 100
70 I=I+1
80 IF I <= N/2 THEN GOTO 50
90 PRINT N ; "IS PRIME"
100 T=TIMER - T
110 PRINT "TIME =" ; T
120 END

```

روش سوم :

```

10 T = TIMER
20 INPUT N
30 IF N=2 THEN PRINT N ; "IS PRIME" : GOTO 100
40 I=2
50 R = N - I * INT (N/I)
60 IF R=0 THEN PRINT N ; "IS NOT PRIME" : GOTO 100
70 I=I+1
80 IF I <= SQR (N) THEN GOTO 50
90 PRINT N ; "IS PRIME"
100 T = TIMER-T
110 PRINT "TIME =" ; T
120 END

```

روش چهارم :

```

10 T = TIMER
20 INPUT N
30 W=1
40 I=1
50 R = N - I * INT (N/I)
60 IF R=0 THEN W=W+1
70 I=I+1
80 IF I <= N/2 THEN GOTO 50
90 IF W=2 THEN PRINT N ; "IS PRIME" ELSE PRINT N ; "IS NOT PRIME"
100 T = TIMER-T
110 PRINT "TIME =" ; T
120 END

```

مثال ۲۳) فلوچارتی رسم کنید که کلیه اعداد اول کوچکتر یا مساوی ۱۰۰۰ را یکی یکی تولید و چاپ نماید.

توضیح: این مثال همانند مثال ۲۰ است با این فرض که یک حلقة دیگر به آن اضافه می‌کنیم که برای کلیه اعداد کوچکتر یا مساوی هزار، اول بودن را تست نماید. عبارت دیگر اگر الگوریتم مثال ۲۰ را به عنوان یک بلوک در نظر بگیریم باید برای کلیه اعداد کوچکتر یا مساوی ۱۰۰۰ این بلوک اجرا گردد. الگوریتم اول بودن را در سه حالت با سه فرض مختلف نوشته‌یم برای سادگی روش آخر را انتخاب کرده و آنرا طوری تغییر می‌دهیم که برای اعداد کوچکتر یا مساوی ۱۰۰۰ نیز عمل نماید.

برای فهم بهتر ابتدا الگوریتم آن و سپس فلوچارت آن را رسم می‌کنیم. شما به عنوان تمرین می‌توانید برای حالتهای دیگر تغییرات ذکر شده را در نظر گرفته و الگوریتم و فلوچارت مربوط به هر یک را رسم نمایید.

۱ - شروع

$$N \leftarrow 1 - 2$$

$$W \leftarrow 1 - 3$$

$$I \leftarrow 1 - 4$$

$$R \leftarrow N - I \times \left[\frac{N}{I} \right] - 5$$

$$W \leftarrow W + I \text{ سپس } R = 6 - \text{اگر}$$

$$I \leftarrow I + 1 - 7$$

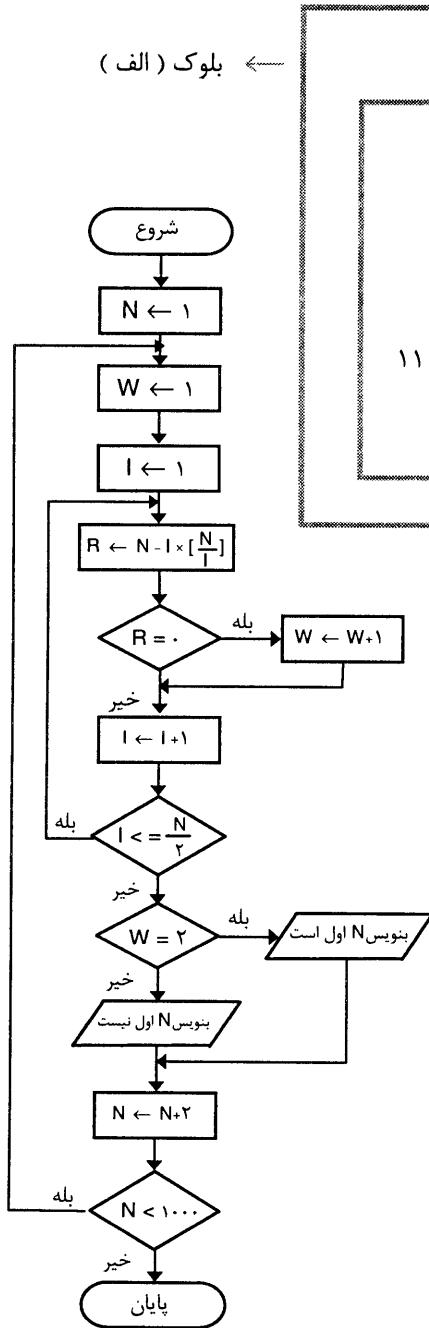
$$\text{اگر } I <= \frac{N}{2} \text{ سپس برو به ۸}$$

۹ - اگر $W = 2$ سپس بنویس عدد اول است و برو به ۱۱

$$N \leftarrow N + 2 - 10$$

۱۱ - اگر $N <= 1000$ سپس برو به ۳

۱۲ - پایان

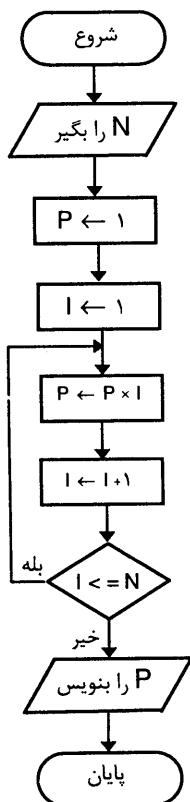


بلوک (الف) : بلوکی که برای اعداد از ۱ تا ۱۰۰۰ معین می‌کند هر کدام اول هستند یا نه .

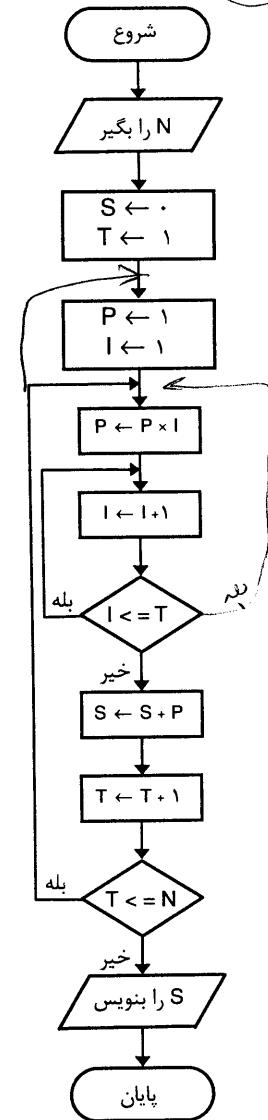
بلوک (ب) : بلوکی که برای یک عدد معین می‌کند که اول هست یا نه .

مثال ۲۴) فلوچارت مثال ۲۰ فصل اول رارسم کنید . (شکل الف)

مثال ۲۵) فلوچارت مثال ۲۱ فصل اول رارسم کنید . (شکل ب)



شکل الف

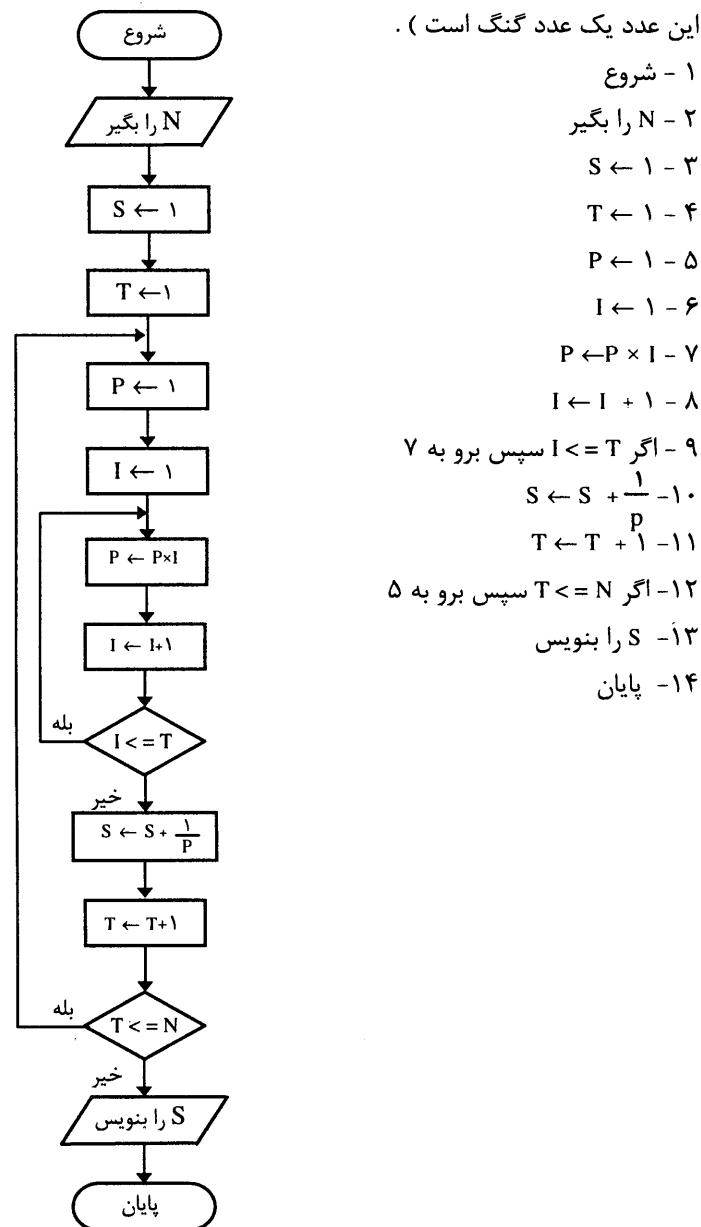


شکل ب

مثال ۲۶) فلوچارتی رسم کنید که عدد طبیعی N را سوال نماید و حاصل جمع زیر را محاسبه و چاپ کند.

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

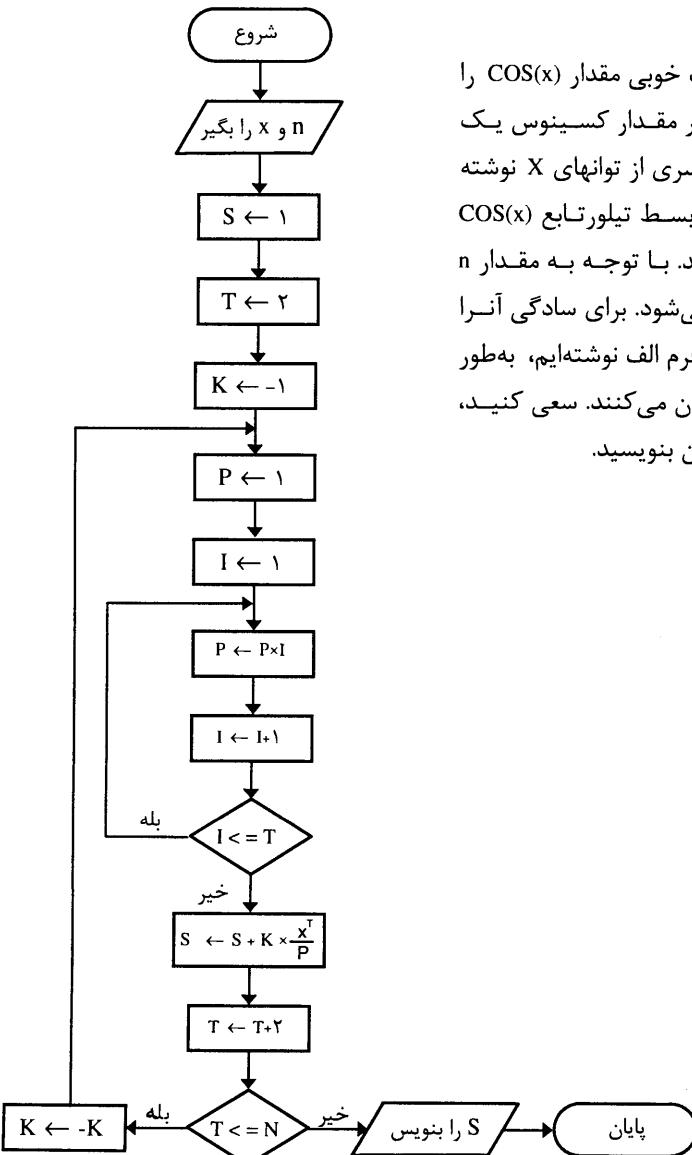
تذکر: مجموع بالا، عدد e را محاسبه می‌کند (e پایه لگاریتم نپرین است و مقدار آن برابر است با $2/7182818$ که با توجه به مقدار N می‌توان با تقریب خوبی این مقدار را محاسبه کرد. (توضیح آنکه این عدد یک عدد گنگ است).



مثال ۲۷) فلوچارتی رسم کنید که عدد طبیعی n و حقیقی x را دریافت و مقدار زیر را محاسبه نماید.

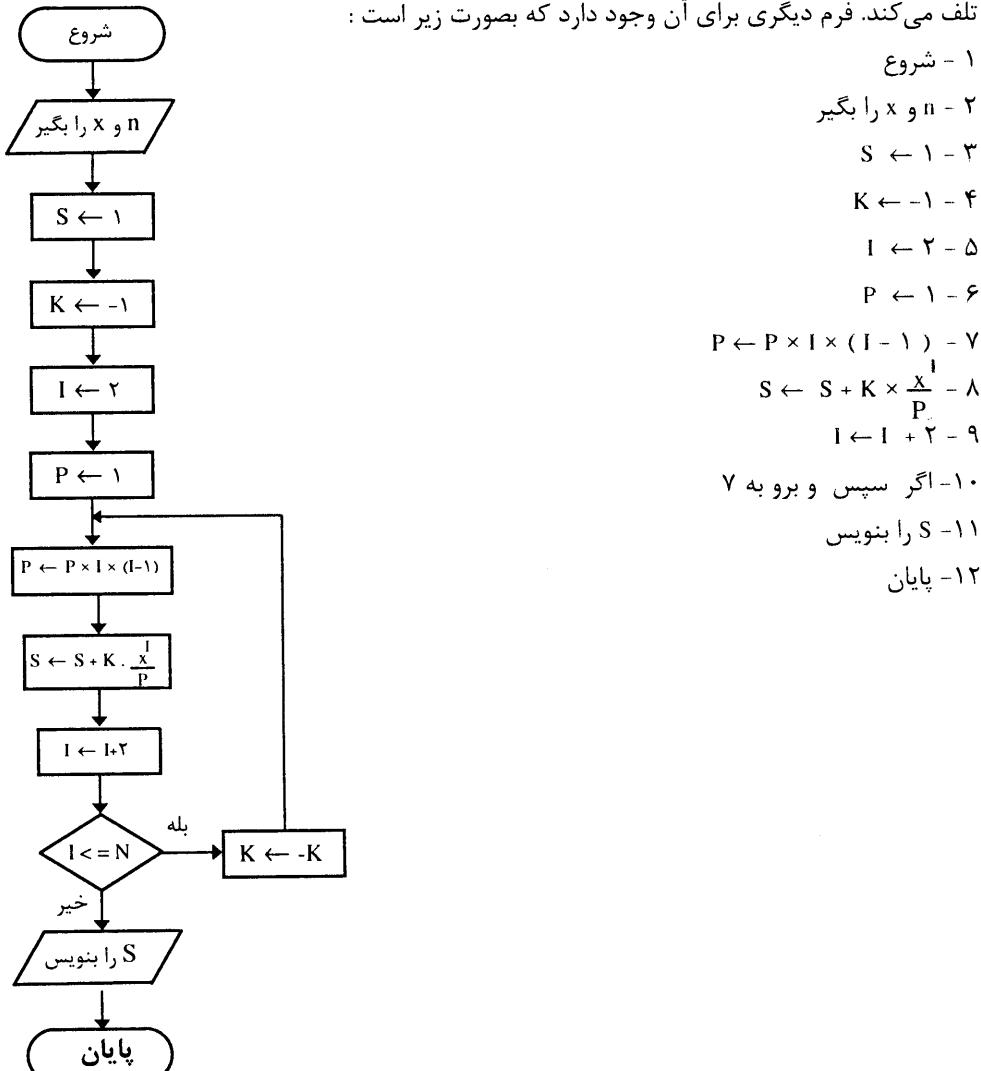
$$\text{الف} \quad S = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \cdots + \frac{x^{2n}}{(2n)!}$$

$$\text{ب) } \cos(x) = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} + \frac{x^6}{6!} + \cdots + (-1)^{n+1} \times \frac{x^{2n}}{(2n)!}$$



توضیح : این مقدار با تقریب خوبی مقدار $\cos(x)$ را بر می گرداند . عبارت دیگر مقدار کسینوس یک زاویه مانند X را بر حسب سری از توانهای X نوشته $\cos(x)$ یا بسط داده ایم که به آن بسط تیلورتابع $\cos(x)$ حول نقطه صفر می گویند. با توجه به مقدار n ضریب جمله n ام تعیین می شود. برای سادگی آنرا مثبت در نظر گرفته و به فرم الف نوشته ایم، به طور دقیق آن را در حالت ب بیان می کنند. سعی کنید الگوریتم آنرا به عنوان تمرین بنویسید.

نکته‌ای که در این مسئله حائز اهمیت است، آن است که هرگاه مقدار T به ترتیب برابر $2, 4, \dots$ و شود، هر بار مقدار P و I یک شده و از ابتدا فاکتوریل آنها محاسبه می‌شود که این کار زمان زیادی را تلف می‌کند. فرم دیگری برای آن وجود دارد که بصورت زیر است :



مشاهده کردید که این الگوریتم ، فقط با یک حلقه نوشته شده است و نکته‌ای که وجود دارد در خط ۷ الگوریتم است که به طور کامل آن را تشریح می‌کنیم. فرض کنید مقدار N برابر ۶ باشد، در ابتدا k منهای یک، I مساوی ۲ و P مساوی ۱ می‌باشد که بعد از آن مقدار P بصورت زیر محاسبه می‌شود :

$$P \leftarrow P \times I \times (I - 1)$$

$$2 \leftarrow 1 \times 2 \times 21!$$

بعد از این محاسبه ، مقدار $\underline{2}$ در P ذخیره شده و جمله $\frac{X^2}{2!}$ تولید و با S جمع می‌شود. سپس دو واحد به $\underline{2}$ اضافه شده و دوباره خط $\underline{7}$ اجرا می‌گردد که حاصل آن بصورت زیر است :

$$P \leftarrow P \times I \times (I - 1)$$

$$24 \leftarrow 2 \times 4 \times 3$$

مقدار قبلی P برابر $\underline{2}$ یعنی $2!$ بوده و بعد از محاسبه، مقدار جدید آن $\underline{24}$ (یعنی $4!$) خواهد بود. بدین ترتیب همواره به I که 2 تا 2 تا اضافه شده و بدون آنکه از ابتدا مقدار I و P یک شوند، مقدار فاکتوریل آن محاسبه می‌گردد. حال اگر دو واحد دیگر به I اضافه کنیم I مساوی $\underline{6}$ می‌شود که داریم :

$$P \leftarrow P \times I \times (I - 1)$$

$$720 \leftarrow 24 \times 6 \times 5$$

که مقدار P مساوی با $\underline{6!}$ می‌باشد، بدین ترتیب جملات ساده تر و در زمان کمتری محاسبه می‌شوند. از این دسته مسائل نمونه‌هایی را در فصل اول حل کردیم که می‌توانید به عنوان تمرین فلوچارت مربوط به هر کدام را رسم نمائید، همچنین در آخر همین فصل نمونه‌های متنوع دیگری را جزء مسائل فصل بیان کرده‌ایم.

تعريف دنباله : فرض کنید \mathbb{N} مجموعه اعداد طبیعی و \mathbb{R} مجموعه اعداد حقیقی باشد، یک دنباله تابعی است از \mathbb{N} به \mathbb{R} به طوریکه به هر عضو \mathbb{N} یک عضو \mathbb{R} را مرتبط می‌سازد که عموماً یک دنباله را بفرم

زیر نمایش می‌دهیم .

$$f: \mathbb{N} \rightarrow \mathbb{R} \quad \begin{cases} \text{اگر } 1 = n \text{ آنگاه } a_1 = f(1) \text{ جمله اول دنباله است} \\ \text{اگر } 2 = n \text{ آنگاه } a_2 = f(2) \text{ جمله دوم دنباله است} \\ \text{اگر } m = n \text{ آنگاه } a_m = f(m) \text{ جمله ام دنبال است} \end{cases}$$

به عبارت دیگر به ازاء هر n ای ، $f(n) = a_n$ جمله n ام دنباله است که گاه برای سادگی آن را بصورت \dots $a_1, a_2, a_3, \dots, a_n$ می‌نویسند .

$f: \mathbb{N} \longrightarrow \mathbb{N}$ مثال : دنباله زیر را در نظر بگیرید :

$$f(n) = a_n \quad \dots \text{ و } 1 \text{ و } n+1 \text{ و } \dots \text{ و } 4 \text{ و } 3 \text{ و } 2 \text{ و } 1$$

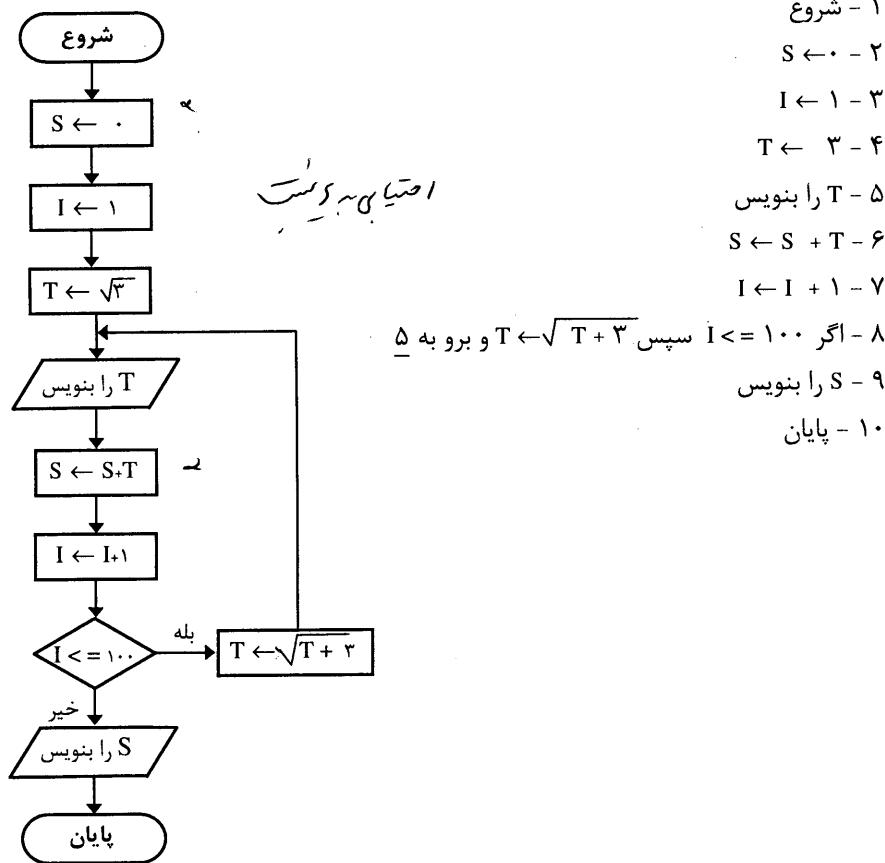
این دنباله ، همان دنباله اعداد طبیعی است.

برای تفهیم دقیق‌تر مطلب باستی بگوئیم که بین اعضای مجموعه اعداد طبیعی و جملات یک دنباله یک تناظر یک به یک برقرار است .

$$a_1 \text{ و } a_2 \text{ و } a_3 \text{ و } a_4 \text{ و } \dots \text{ و } a_n \quad \begin{matrix} \text{جمله} & \text{جمله} \\ \text{چهارم} & \text{سوم} \\ \text{ام} & \text{دوم} \\ \text{اول} & \text{ام} \end{matrix}$$

مثال ۲۸) فلوچارتی رسم کنید که ۱۰۰ جمله از دنباله زیر را تولید و چاپ نماید، همچنین مجموع آنها را نیز محاسبه و چاپ کند.

$$\sqrt{3}, \sqrt{3+\sqrt{3}}, \sqrt{3+\sqrt{3+\sqrt{3}}}, \dots$$

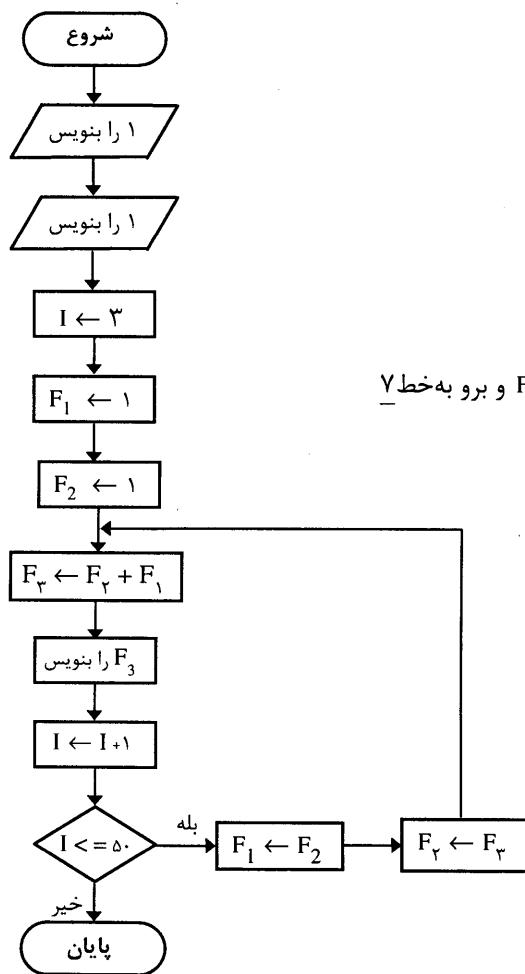


شکل دیگری از دنباله‌ها وجود دارند که به‌طور صریح نمی‌توان برای جملات آنها تابعی پیدا کرد و جملات بر حسب جملات قبلی یا بعدی دنباله، تعیین می‌شوند.

مثال ۲۹) فلوچارتی رسم کنید که ۵۰ جمله از دنباله زیر (دنباله فیبوناچی) را تولید و چاپ نماید.

$$1, 2, 3, 5, 8, \dots, a_n$$

جمله اول و دوم برابر یک است و از جمله سوم به بعد هر جمله، برابر مجموع دو جملهٔ قبل از خودش است پس $a_1 = 1$, $a_2 = 1$, $a_3 = a_1 + a_2 = 2$, \dots , $a_n = a_{n-1} + a_{n-2}$ و الی آخر. ($n > 2$)

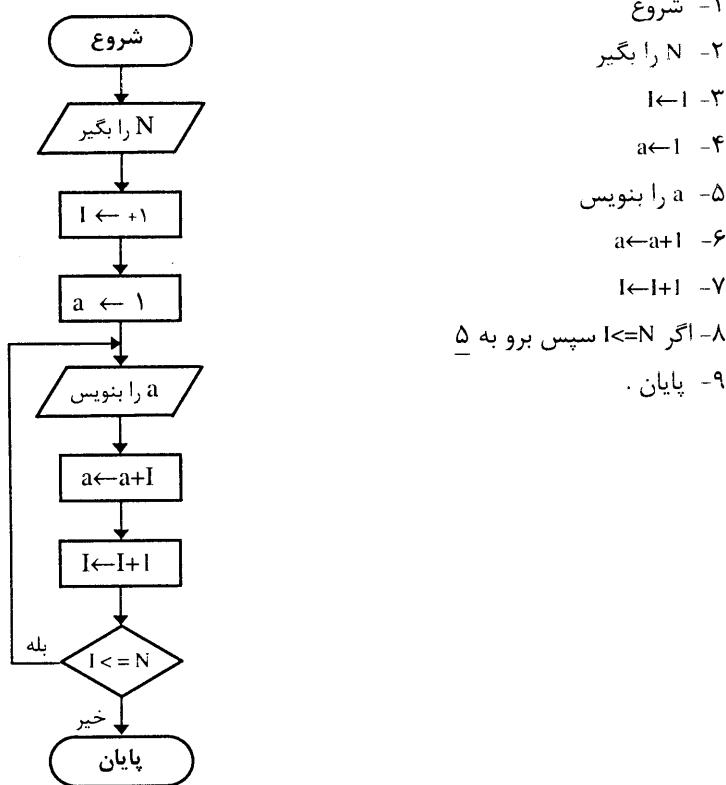


- ۱ - شروع
- ۲ - ۱ را بنویس
- ۳ - ۱ را بنویس
- $I \leftarrow ۳ - ۴$
- $F_1 \leftarrow ۱ - ۵$
- $F_2 \leftarrow ۱ - ۶$
- $F_3 : F_3 \leftarrow F_2 + F_1 - ۷$
- $I \leftarrow I + ۱ - ۸$
- ۹ - اگر $I <= ۵$ سپس $F_2 \leftarrow F_3$ و $F_3 \leftarrow F_1 + F_2$ و برو به خط ۷
- ۱۰ - پایان

مثال (۳۰) فلوچارتی رسم کنید که عدد N را سؤال نموده و N جمله اول دنباله زیر را محاسبه و چاپ نماید.

جملات دنباله فوق به ترتیب زیر بوجود آمده اند :

جمله اول را یک در نظر گرفته و جمله دوم مساوی با اندیس جمله اول به اضافه جمله اول و جمله سوم عبارتست از جمله دوم به اضافه اندیس جمله دوم و الی آخر (جمله n آم یعنی جمله $n-1$ آم به اضافه اندیس $n-1$ آم یعنی $a_{n-1} + a_n$) .



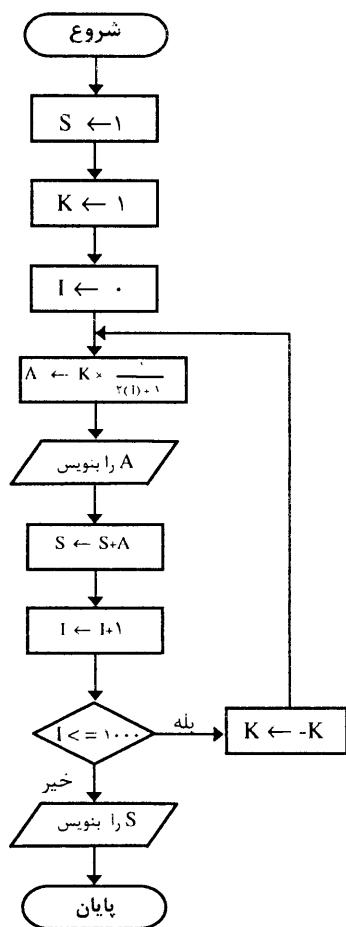
مثال ۲۸) فلوچارتی رسم کنید که جملات دنباله زیر را تا ۱۰۰۰ جمله محاسبه و همچنین مجموع آنها را نیز محاسبه و چاپ کند.

$$\dots + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \dots + \frac{1}{3}$$

توضیح آنکه اگر N بزرگ باشد مجموع جملات دنباله با تقریب خوبی به عدد $\frac{\pi}{4}$ نزدیک می‌شوند.

- ۱ - شروع
- ۲ - $S \leftarrow 0$
- ۳ - $K \leftarrow 1$
- ۴ - $I \leftarrow 0$

$$A \leftarrow K \times \frac{1}{2I+1} \quad \text{۵}$$



۶ - A را چاپ کن
 $S \leftarrow S + A - 7$
 $I \leftarrow I + 1 - 8$
 ۹ - اگر $I \leq 1$ پس $K \leftarrow -K$ و برو به ۵
 ۱۰ - S را بنویس
 ۱۱ - پایان

تعریف فایل (File) : محل ذخیره داده‌ها و اطلاعات را فایل می‌گوئیم.

در حل بعضی از مسائل اطلاعات ثابتی وجود دارد که بهتر است آنها را درون یک فایل قرار دهید و در موقع لزوم از درون فرآیند بخوانید.

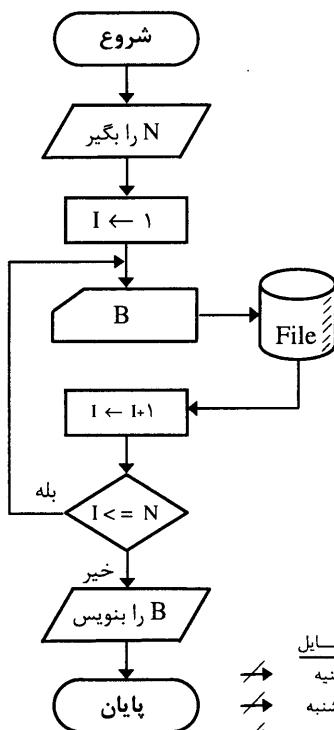
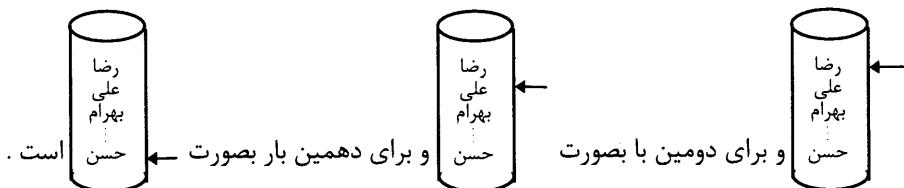
در فلوچارت نویسی برای نمایش فایل از نماد استفاده می‌کنیم (البته باید دقیقاً توضیح داده شود که اطلاعات درون فایل چگونه ذخیره شده‌اند) همچنین برای خواندن اطلاعات از درون فایل به عنوان ورودی مسئله، از نماد استفاده کنید.

فرض بر این است که هرگاه از نماد استفاده کنید یکی از اطلاعات درون فایل را بگیر و در خانه (متغیر) B قرار بده که الگوریتم ما این ارتباط را درک و توجیه می‌کند.

در مورد اطلاعات داخل فایل نیز فرض بر این است که یک نشانگر روی اطلاعات قرار دارد و هر بار که یکی از اطلاعات را می‌خوانید، این نشانگر یک واحد به سمت جلو می‌رود تا به انتهای فایل برسد. هرگاه برای اولین بار به فایلی رجوع کنید، نشانگر همواره روی اولین داده قرار خواهد داشت.

به عنوان مثال فرض کنید فایلی به نام Name محتوى نام ۱۰ نفر :

هرگاه برای اولین بار به فایل رجوع کنید و یکی از اطلاعات آن را بگیرید، رضا خوانده می‌شود و نشانگر روی داده بعدی قرار می‌گیرد :



مثال (۳۲) : فلوچارتی رسم کنید که عددی صحیح بین ۱ تا ۷ را دریافت و نام روز مربوط به آن عدد را چاپ نماید.

۱ - شروع

۲ - N را بگیر

۳ - I ← ۱

۴ - I امین داده فایل را بگیر و درون B قرار بده

۵ - I ← I + ۱

۶ - اگر I <= N سپس برو به ۴

۷ - B را بنویس

۸ - پایان

اطلاعات درون فایل بصورت

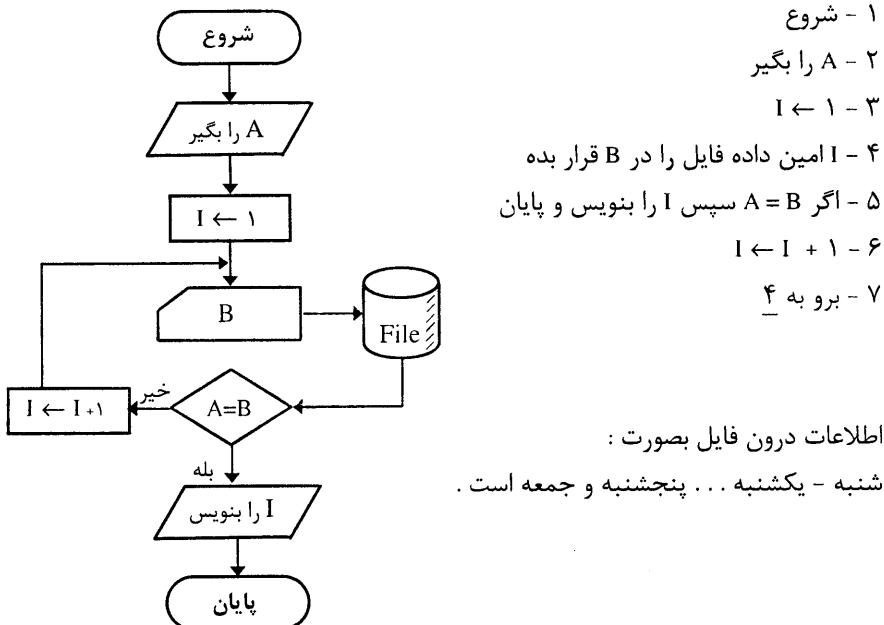
(شنبه - یکشنبه - دوشنبه)

... پنجشنبه و جمعه) است.

	فایل	N	I	B	چاپ	شنبه	شنبه
→	شنبه	۴	/	/	شنبه	شنبه	شنبه
→	یکشنبه		/	/	یکشنبه	یکشنبه	یکشنبه
→	دوشنبه		/	/	دوشنبه	دوشنبه	دوشنبه
→	سهشنبه		/	/	سهشنبه	سهشنبه	سهشنبه
→	چهارشنبه		۵				
→	پنجشنبه						
→	جمعه						

امتحان الگوریتم برای عدد ۴ :

مثال ۳۳) : فلوچارتی رسم کنید که نام یک روز هفته را به عنوان ورودی دریافت و معین کند چندمین روز هفته است .



	فایل	A	I	B	چاب
	شنبه	چهارشنبه	۱	شنبه	امین روز هفته است
↗	یکشنبه			یکشنبه	
↖	دوشنبه			دوشنبه	
↗	سهشنبه			سهشنبه	
→	چهارشنبه		۵	چهارشنبه	
↗	پنجشنبه				
↓	جمعه				

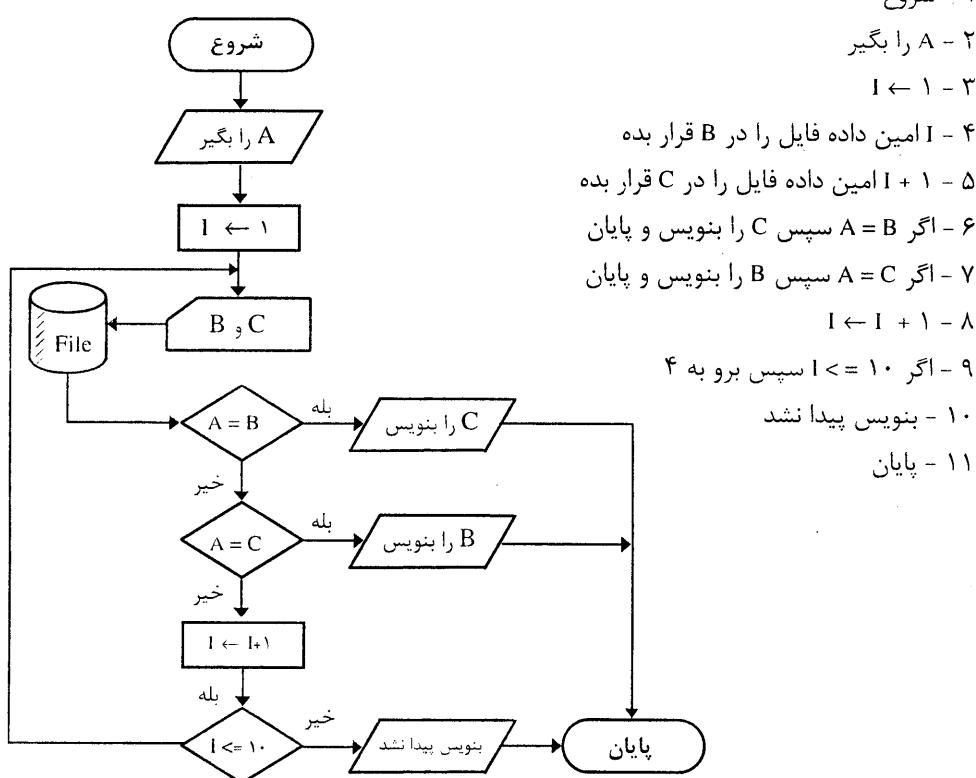
تست الگوریتم برای روز چهارشنبه :

مثال ۳۴) : فلوچارتی رسم کنید که نام کشورها بهمراه پایتحت آنها را بترتیب و پشت سر هم که درون یک فایل قرار دارد در نظر گرفته و نام یک کشور را از ورودی دریافت و نام پایتحت مربوط به آن را چاپ نماید . (بالعکس) .

فرض کنید اطلاعات درون فایل بصورت زیر باشد :

(ایران : تهران - عراق : بغداد - چین : پکن - ترکیه : آنکارا - ... - آمریکا : واشنگتن)

توجه داشته باشید که نام کشور و پایتحت باید پشت سر هم درون فایل نوشته شوند در غیر اینصورت این الگوریتم و فلوچارت درست عمل نخواهد کرد .



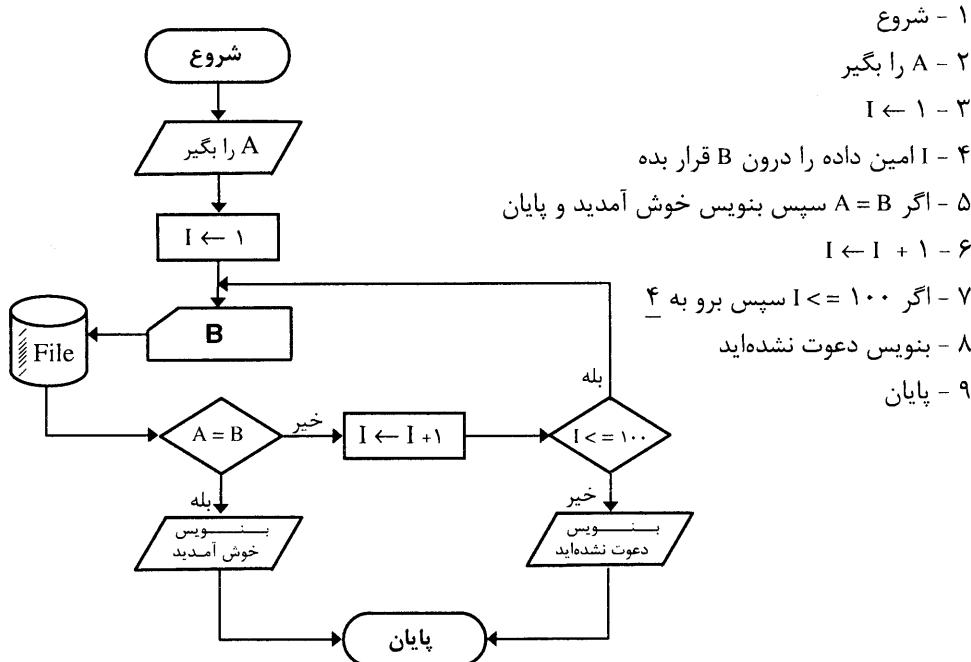
	فایل	A	I	B	C	جانب
→	ایران	عراق	/	ایران	تهران	بعداد
→	تهران		۲	عراق	بغداد	
→	عراق					
→	بغداد					
→	چین					
→	پکن					

تست الگوریتم برای عراق :

آمریکا
وائستگتن

مثال ۳۵) : فرض کنید ۱۰۰ نفر را به یک مهمانی دعوت کرده و برای هر کدام یک کارت دعوت فرستاده ایم. فلوچارتی رسم کنید که برای هر نفر که وارد مهمانی می شود اسم آن شخص پرسیده شود و اگر بین ۱۰۰ اسم بود پیغام خوش آمدید بنویسد و در غیر این صورت پیغام دهد که شما به این مهمانی دعوت نشده اید.

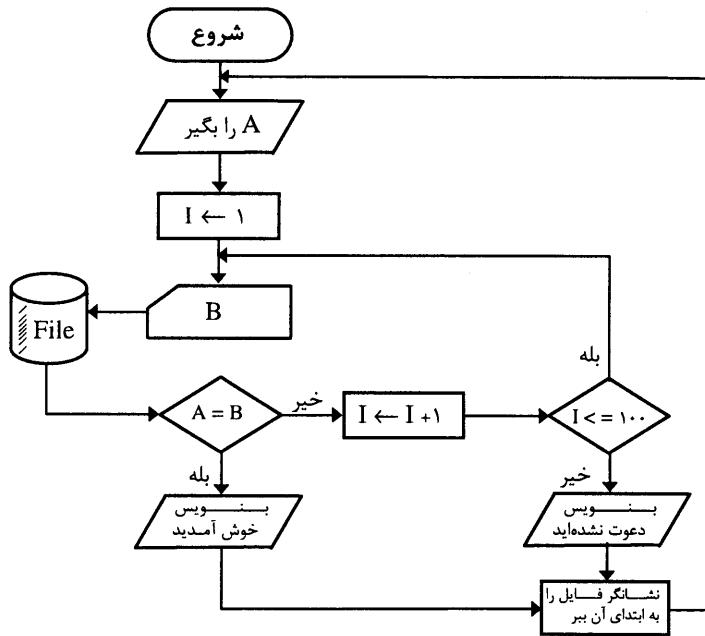
در این مسئله نیز از یک فایل استفاده می‌کنیم که اسم ۱۰۰ نفر را داخل آن نوشته‌ایم.



در کلیه مسائلی که اطلاعاتی را از درون فایلی فراخوانی می‌کنید، نکته مهم آن است که، اگر برای یک بار دیگر آنرا اجرا کنید (که عموماً کاربرد این قبیل مسائل اینتگونه است) باید توسط فرمانی، نشانگر فایل را به ابتدای آن برگردانید. برای روشن شدن مطلب، اگر مثال ۳۰ را یکبار دیگر اجرا کنید، جواب غلطی را مشاهده خواهید کرد.

فرض کنید نفر اول که وارد می‌شود، اسمش، سی امین داده در فایل باشد. حال، اگر برای نفر دوم آنرا دوباره اجرا کنیم مثلاً فرض کنیم که نفر بعدی که وارد می‌شود اسمش در فایل بیستمین نفر باشد، الگوریتم برای یافتن اسم نفر دوم در فایل از جایی که نشانگر فایل قرار دارد، شروع کرده و تا آخر فایل عمل جستجو را ادامه می‌دهد، سپس پیغام می‌دهد که شما دعوت نشده‌اید. در صورتیکه، می‌دانید اسم آن شخص در فایل بیستمین داده است.

برای رفع این مشکل دستوری به الگوریتم می‌دهیم که نشانگر فایل را به ابتدای آن بازگرداند و در فلوچارت هم، آن را داخل مستطیل می‌نویسیم، با این توضیحات مثال بالا را طوری تغییر می‌دهیم که برای هر چند مرتبه که اجرا می‌شود، درست کار کند :

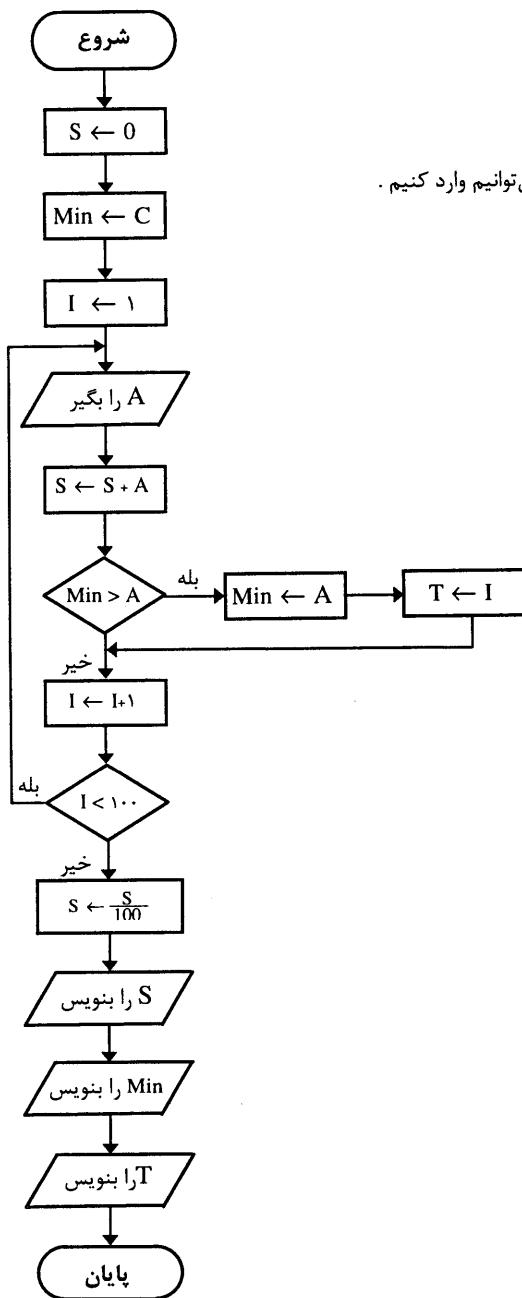


به عنوان تمرین سعی کنید الگوریتم این مثال و مثالهای قبلی را بنویسید.

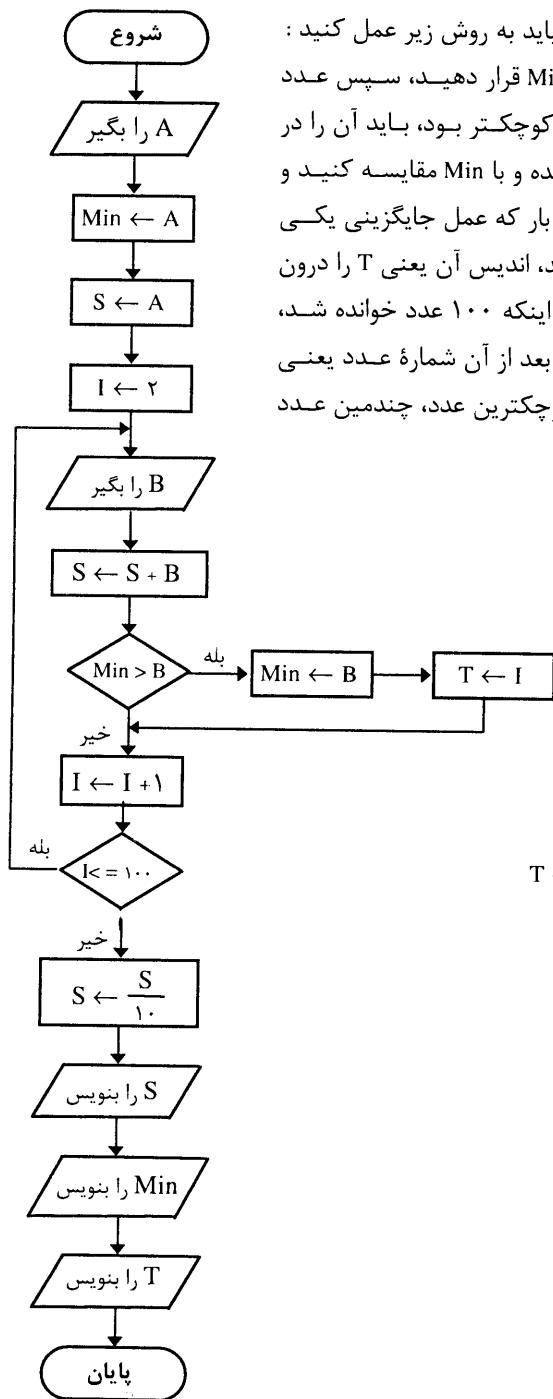
مثال ۳۶) : فلوچارتی رسم کنید که ۱۰۰ عدد را یکی دریافت و ابتدا میانگین آنها را محاسبه و چاپ کند ، سپس گوچکترین عدد از بین این ۱۰۰ عدد را محاسبه و معین کند چندمین عدد بوده است. (فرض که ۱۰۰ عدد مثبت و بزرگتر از صفر هستند).

- ۱ - شروع
- ۲ - $S \leftarrow 0$
- ۳ - $Min \leftarrow C$
- ۴ - $I \leftarrow 1$
- ۵ - را بگیر A
- ۶ - $S \leftarrow S + A$
- ۷ - اگر $Min > A$ پس $Min \leftarrow A$
- ۸ - $I \leftarrow I + 1$
- ۹ - اگر $I \leq 100$ سپس برو به ۵

-۱۰ $S \leftarrow \frac{S}{100}$
 -۱۱ S را بنویس
 -۱۲ Min را بنویس
 -۱۳ T را بنویس
 -۱۴ پایان



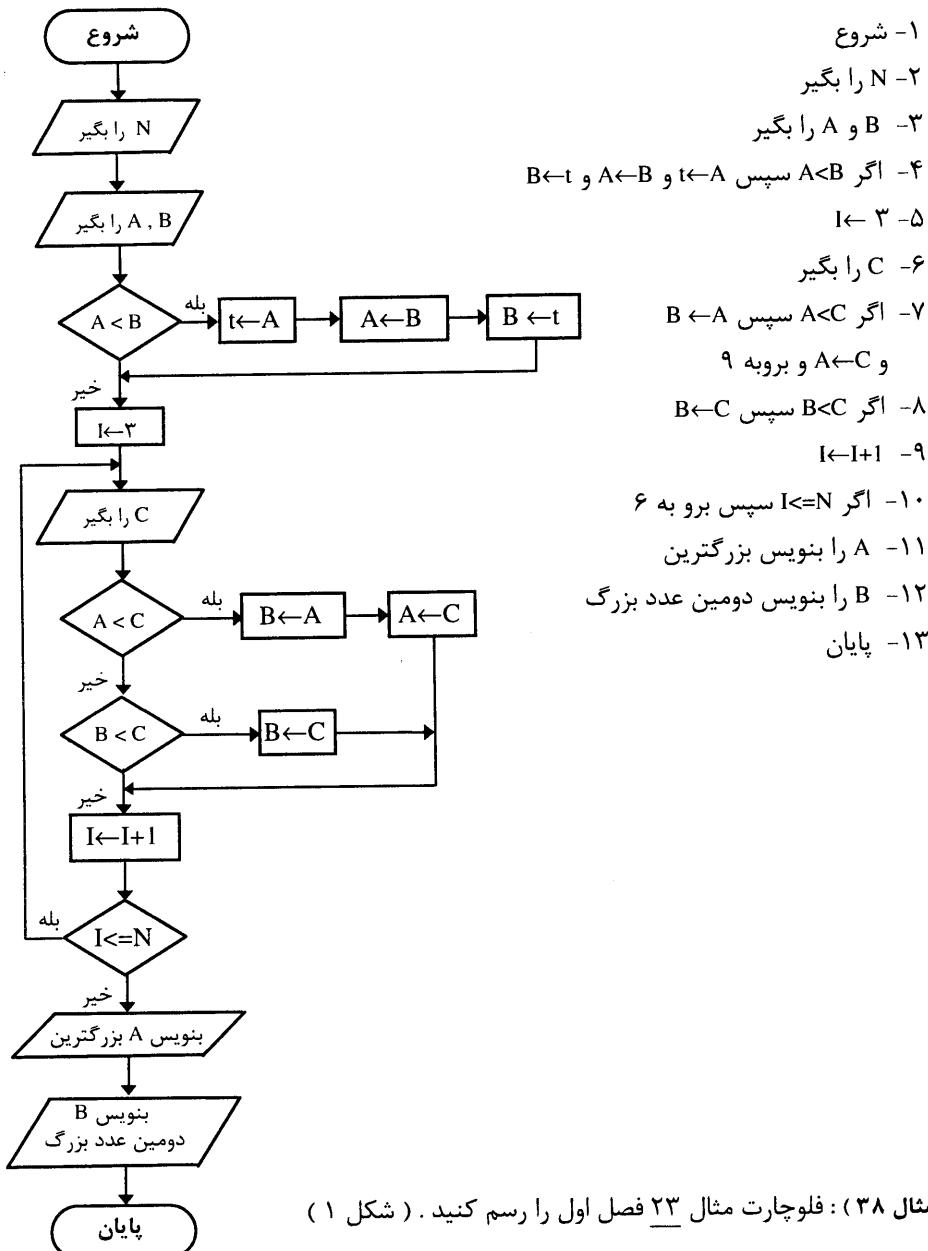
توضیح : C بزرگترین عددی است که در محدوده اعداد می‌توانیم وارد کنیم .



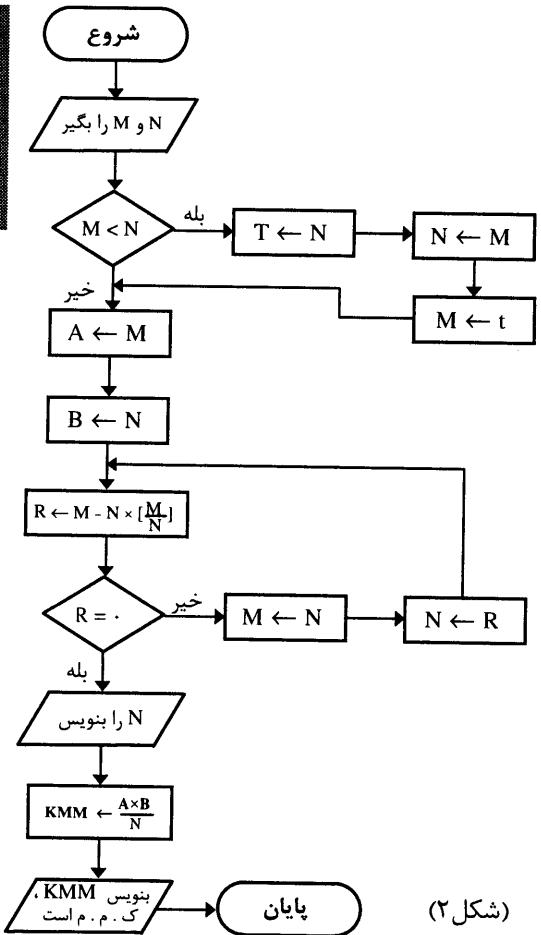
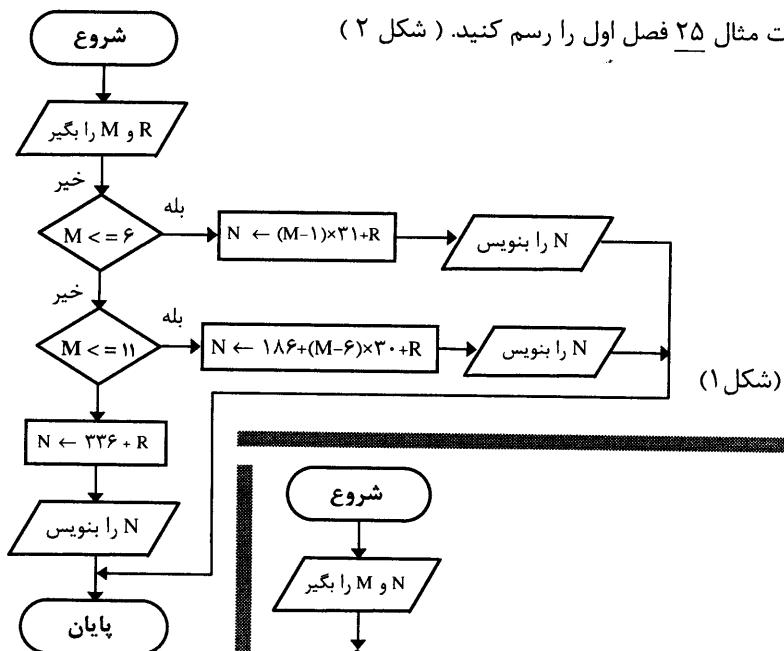
اگر فرض مثبت و مخالف بودن را بردارید، باید به روش زیر عمل کنید:
 ابتدا اولین عدد را گرفته و آن را در خانه Min قرار دهید، سپس عدد دوم را گرفته و با Min مقایسه کنید، اگر کوچکتر بود، باید آن را در Min قرار داده و اگر نبود عدد سوم را خوانده و با Min مقایسه کنید و این روند را تا صدمین عدد ادامه دهید. هر بار که عمل جایگزینی یکی از این اعداد را در خانه Min انجام می‌دهید، اندیس آن یعنی T را درون یک متغیر ذخیره کنید و در نهایت پس از اینکه ۱۰۰ عدد خوانده شد، ابتدا میانگین سپس محتويات خانه Min و بعد از آن شماره عدد یعنی اندیس آنرا چاپ کنید تا معلوم شود که کوچکترین عدد، چندمین عدد بوده است.

- ۱ - شروع
- ۲ - A را بگیر
- Min ← A - ۳
- S ← A - ۴
- I ← ۲ - ۵
- B را بگیر
- S ← S + B - ۷
- اگر Min > B پس T ← I و Min ← B
- I ← I + ۱ - ۹
- اگر I ≤ ۱۰۰ پس به ۶ برو
- $S \leftarrow \frac{S}{10}$ - ۱۱
- S را بنویس - ۱۲
- Min را بنویس - ۱۳
- T را بنویس - ۱۴
- پایان - ۱۵

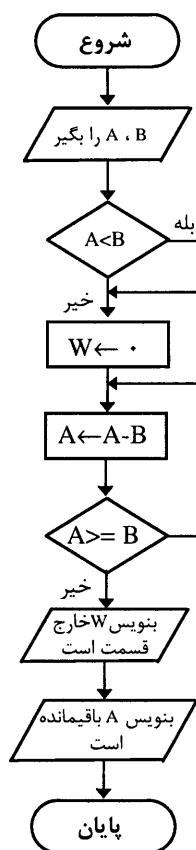
در فصل مربوط به مرتب سازی و جستجو به طور مفصل درباره این الگوها صحبت خواهیم کرد. مثال (۳۷) : فلوچارتی رسم کنید که N عدد را دریافت و بزرگترین عدد از بین N عدد و نیز بزرگترین عددی را که از همه اعداد به جز عدد محاسبه شده بزرگتر است، چاپ نماید . (منظور دومین عدد بزرگ است) .



مثال ۳۹) : فلوچارت مثال ۲۵ فصل اول را رسم کنید. (شکل ۲)



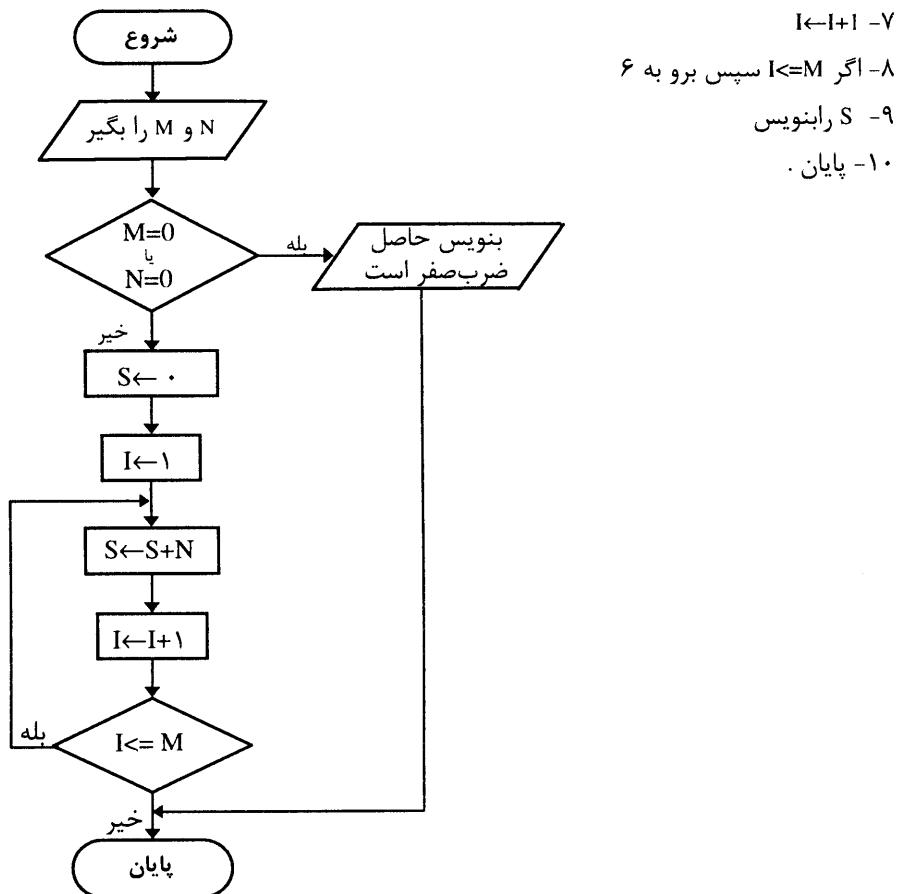
مثال (۴۰) : فلوچارتی رسم کنید که دو عدد صحیح و مثبت را دریافت و خارج قسمت و باقیمانده تقسیم عدد بزرگتر را به عدد کوچکتر محاسبه و چاپ نماید. (از طریق تفریق های متوالی).



- ۱- شروع
- ۲- A و B را بگیر
- ۳- اگر $A < B$ سپس $t \leftarrow A$ و $A \leftarrow B$ و $B \leftarrow t$
- ۴- $W \leftarrow 0$
- ۵- $A \leftarrow A - B$
- ۶- اگر $A \geq B$ سپس $W \leftarrow W + 1$ و برو به
- ۷- بنویس W خارج قسمت است.
- ۸- بنویس A باقیمانده است.

مثال (۴۱) : فلوچارتی رسم کنید که دو عدد صحیح و مثبت M و N را دریافت و حاصلضرب M در N را با استفاده از جمع های متوالی محاسبه و چاپ نماید.

- ۱- شروع
- ۲- M و N را بگیر
- ۳- اگر $M \neq 0$ سپس بنویس حاصل ضرب صفر است و برو به ۱۰
- ۴- $S \leftarrow 0$
- ۵- $I \leftarrow 1$
- ۶- $S \leftarrow S + N$

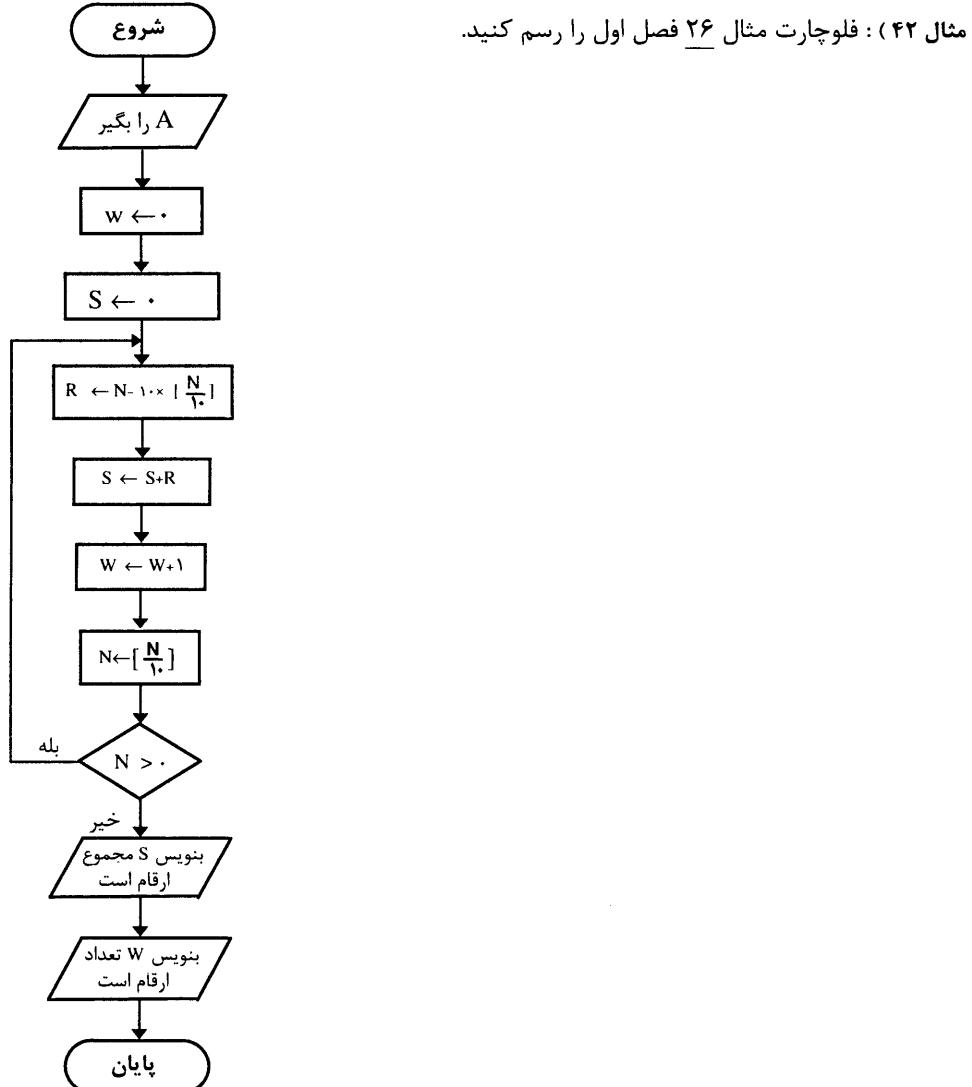


I ← I + 1 - ۷

- ۸ - اگر I ≤ M سپس برو به ۶

- ۹ - S را بنویس

- ۱۰ - پایان.



تذکر: اگر عددی از 10^I کوچکتر باشد، اولین عددی که در این رابطه صدق نماید، عدد I رقمی می‌شود (در مورد اثبات آن فکر کنید).

مثال (۴۳): فلوچارتی رسم کنید که با توجه به نکته بالا یک عدد را سوال و معین کند چند رقمی است.

۱- شروع

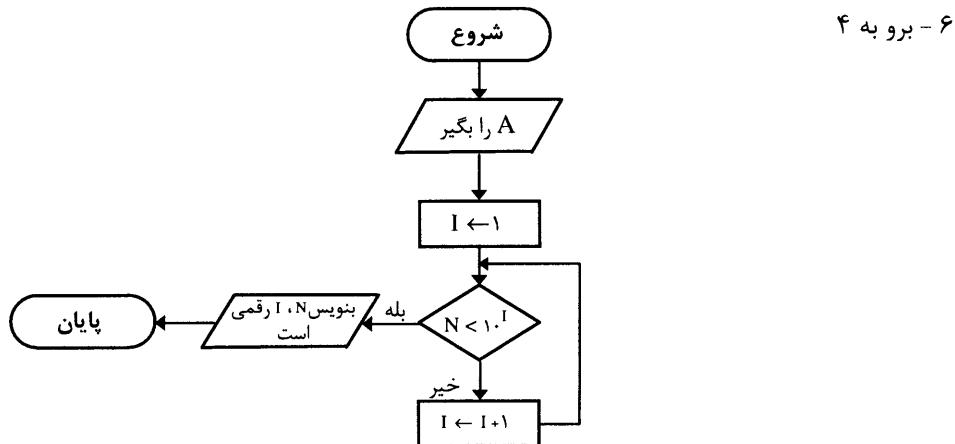
۲- را بگیر A

۳- $I \leftarrow 1$

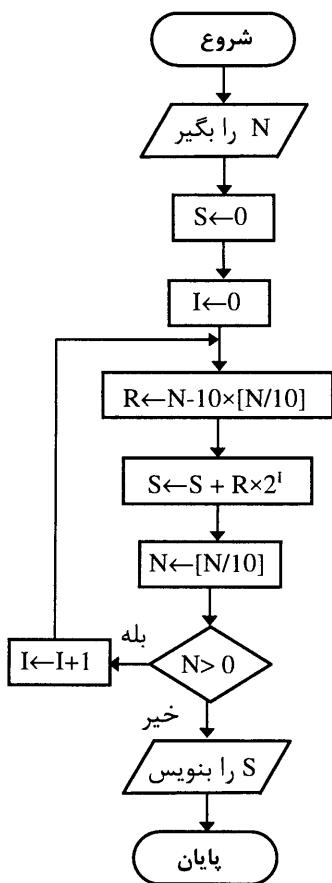
۴- اگر $I < 10^I$ پس بنویس I رقمی است و پایان

۵- $I \leftarrow I + 1$

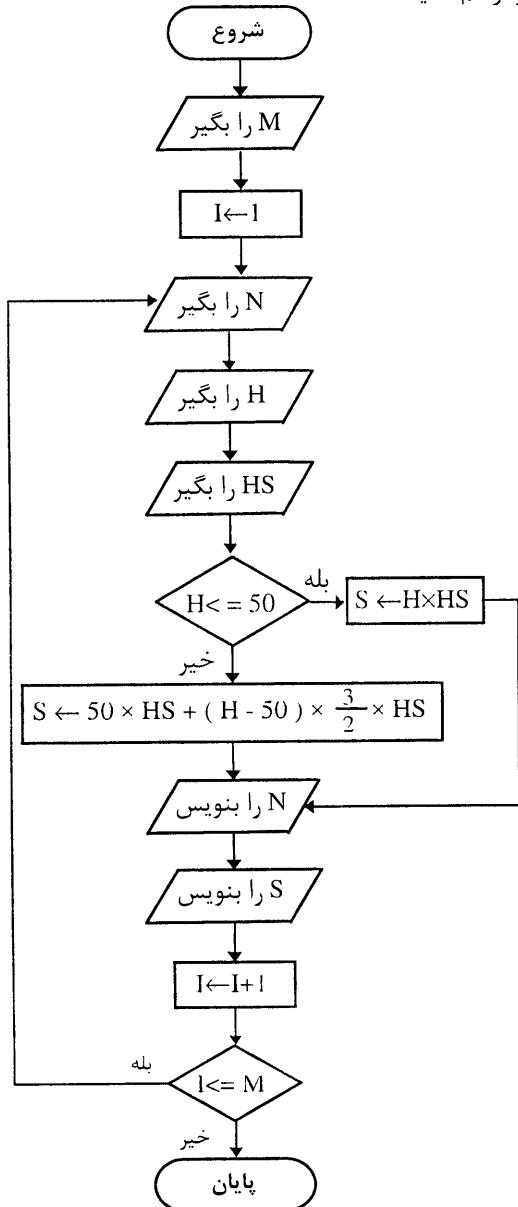
۶- برو به ۴



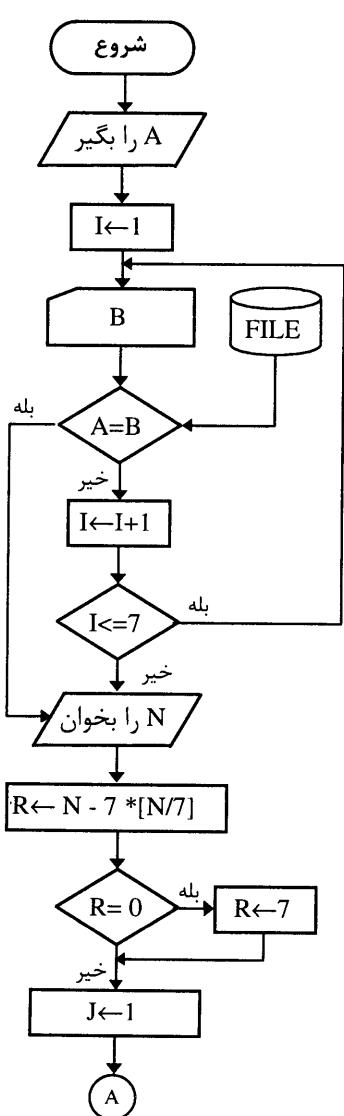
مثال ۴۴) فلوچارت مثال ۲۷ فصل اول را رسم کنید .



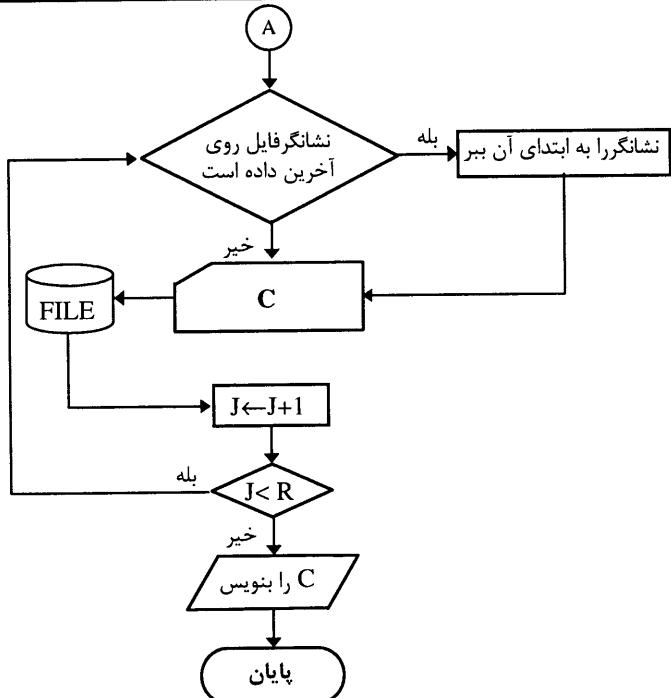
مثال ۴۵) فلوچارت مثال ۳۰ فصل اول را رسم کنید.



مثال ۴۶) فلوچارتی رسم کنید که نام روز N ام سال را با توجه به نام روز اول سال بنویسد.



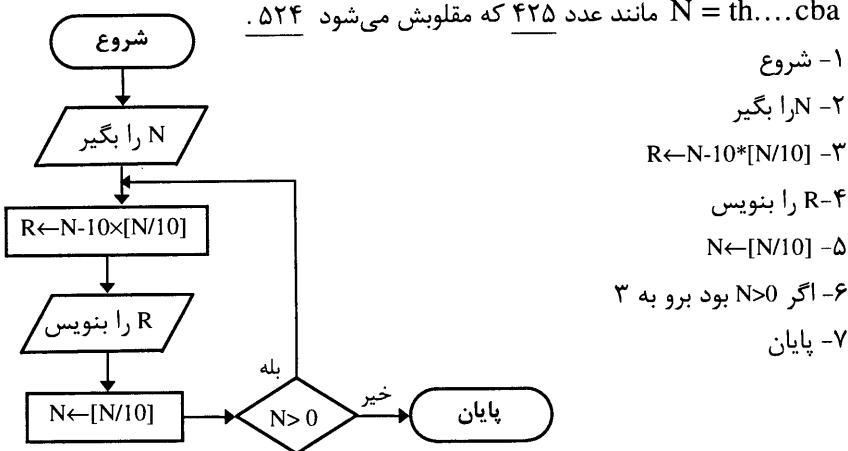
- ۱- شروع
- ۲- را بگیر A
- ۳- I ← 1
- ۴- امین داده فایل را بگیر و در B قرار بده .
- ۵- اگر A=B بود سپس ، برو به خط ۸ .
- ۶- I ← I+1
- ۷- اگر I <= 7 بود، سپس برو به ۴ .
- ۸- N را بگیر .
- ۹- R ← N-7×[N/7]
- ۱۰- اگر R=0 بود ، سپس R را مساوی ۷ قرار بده .
- ۱۱- j ← 1
- ۱۲- اگر نشانگرفایل روی آخرین داده بود ، سپس آنرا به ابتدای فایل ببر .
- ۱۳- زامین داده فایل را بگیر و در C قرار بده .
- ۱۴- J ← J+1
- ۱۵- اگر J < R بود ، سپس برو به ۱۲ .
- ۱۶- C را بنویس .
- ۱۷- پایان



مثال ۴۷) فلوچارتی رسم کنید که یک عدد صحیح را دریافت و مقلوب آنرا چاپ نماید.

توضیح: فرض میکنیم که عدد به صورت $\bar{N} = abc\dots ht$ باشد مقلوب آن برابر است با

$\bar{N} = th\dots cba$ مانند عدد ۴۲۵ که مقلوبش می‌شود ۵۲۴.



مثال ۴۸) با استفاده از روش دیگری، فلوچارتی رسم کنید که مقسوم‌علیه‌های یک عدد را یکی‌یکی محاسبه و چاپ نماید.

توضیح : فرض کنید که می خواهید مقسوم علیه های عدد ۱۸ را بدست آورید، از آنجاییکه $18 = 1 \times 18$ بخش - پذیر است، پس عدد ۱ یکی از مقسوم علیه های آنست. با بدست آوردن عدد یک، عدد $18/1 = 18$ حاصل می شود، که یکی دیگر از مقسوم علیه هاست. عدد ۲ نیز مقسوم علیه ۱۸ است و با بدست آمدن عدد ۲ عدد $18/2 = 9$ بدست می آید، که خود مقسوم علیه ۱۸ است بالاخره ۳ نیز مقسوم علیه ۱۸ است که با بدست آمدن آن، عدد $18/3 = 6$ بدست می آید که یکی دیگر از مقسوم علیه های ۱۸ می باشد و عملیات خاتمه می یابد. بطور کلی برای هر عدد a ، اگر اعداد b و c ای باشند به طوریکه $a = b \cdot c$ آنگاه هریک از b و c مقسوم علیه های عدد a هستند و در اصل با داشتن یکی ، می توان دیگری را نیز محاسبه کرد. بطور مثال، با داشتن a و بدست آمدن b میتوان $c = a/b$ را بدست آورد. پس اگر بخواهیم کلیه مقسوم علیه های عدد a را بدست آوریم، کافی است تمام b هایی را محاسبه کنیم که $a = b \cdot c$ و $b \leq c$ باشند.

قضیه : اگر $a = b \cdot c$ و $b \leq c$ آنگاه $b \leq \sqrt{a}$

اثبات :

طبق فرض داریم که :

طرفین را در b ضرب می کنیم.

$$b \leq c$$

$$b \cdot b \leq b \cdot c \Rightarrow$$

$$b^2 \leq b \cdot c \Rightarrow$$

$$b^2 \leq a \Rightarrow$$

$$b \leq \sqrt{a} \Rightarrow$$

به جای $b \cdot c$ مقدار آن a را قرار می دهیم.

یعنی برای بدست آوردن تمام اعدادی ما نند b و c که روابط $b \leq c$ و $a = b \cdot c$ برای آنها برقرار باشد کافیست که عدد b را تا جذر عدد a تغییر دهیم.

سبس در مورد محاسبه مقسوم علیه های عدد a نیز کافیست که اعداد b و c ای را پیدا کنیم که $c \leq b$ و $a = b \cdot c$ باشد و برای اینکار b را از یک تا جذر a تغییر می دهیم که با بدست آوردن b که یک مقسوم علیه است، c نیز محاسبه می شود.

۱ - شروع

۲ - N را بگیر

۳ - $M \leftarrow [\sqrt{N}]$

۴ - $I \leftarrow 1$

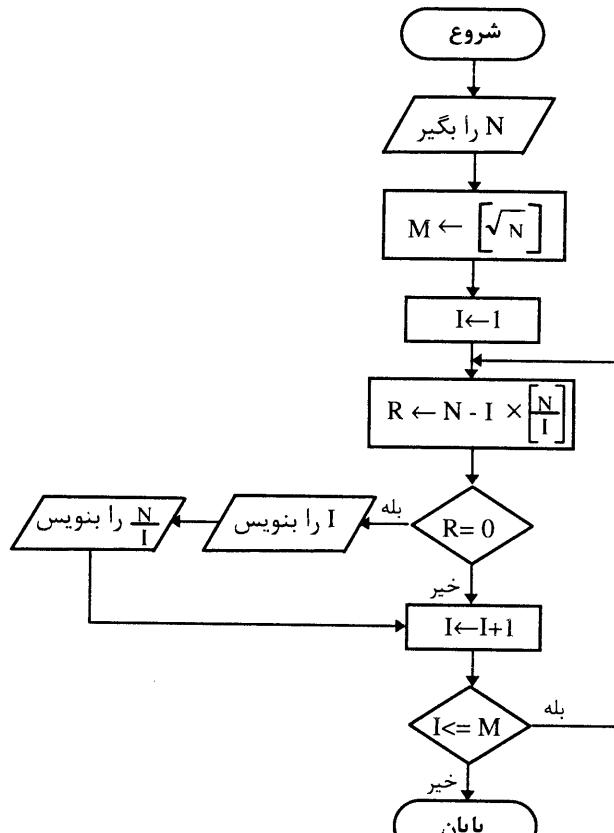
۵ - $R \leftarrow N - I \times [N/I]$

۶ - اگر $R=0$ پس I را بنویس و $\frac{N}{I}$ را بنویس

۷ - $I \leftarrow I+1$

۸ - اگر $I \leq M$ سپس برو به ۵

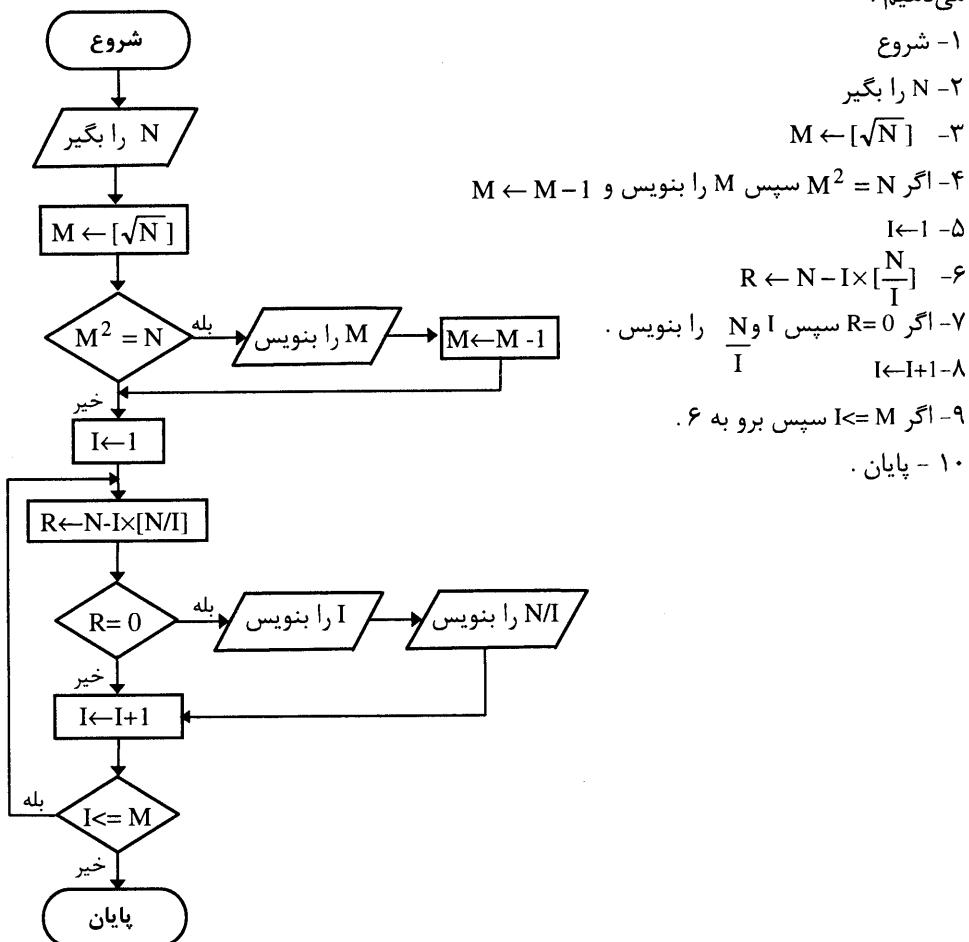
۹ - پایان



تست الگوریتم برای
اعداد ۳۶ و ۱۲

بار اول برای عدد ۳۶				
N	M	I	R	چاپ
۳۶	۶	۱	۰	۱۹۳۶
		۲	.	۲۹۱۸
		۳	.	۳۹۱۲
		۴	.	۴۹۹
		۵	۱	
		۶	.	۶۹۶
بار دوم برای عدد ۱۲				
N	M	I	R	چاپ
۱۲	۳	۱	.	۱۹۱۲
		۲	.	۲۹۶
		۳	.	۳۹۴

الگوریتم را برای دو عدد دلخواه امتحان کردید، نکته قابل توجه اینست که برای اعدادی که جذر کامل دارند، خود \sqrt{N} دو بار چاپ می شود که برای جلوگیری از این عمل، الگوریتم آنرا به صورت زیر تغییر می دهیم .



```

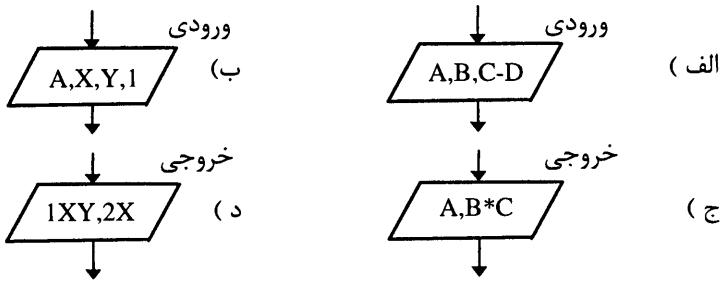
10 T=TIMER
20 INPUT N
30 M=INT(SQR(N))
40 IF M*M=N THEN PRINT M : M=M-1
50 I=1
60 R=N-1*INT(N /I)
70 IF R=0 THEN PRINT I,N /I
80 I=I+1
    
```

```
90 IF I<=M THEN GOTO 60  
100 T=TIMER - T : PRINT T  
110 END
```

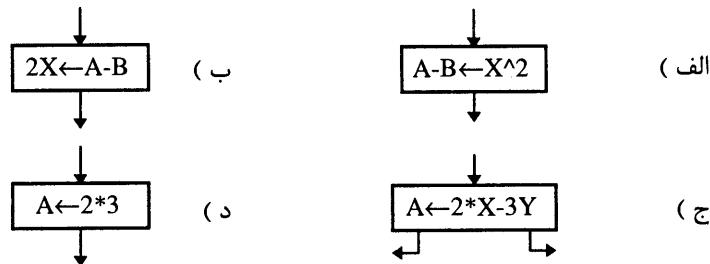
برای مقایسه، برنامه آنرا در بالا نوشته‌ایم و میتوانید برای اعداد مختلف با برنامه‌های نوشته شده در مثالهای قبلی، زمان را مقایسه کنید.

پرسش‌های تشریحی

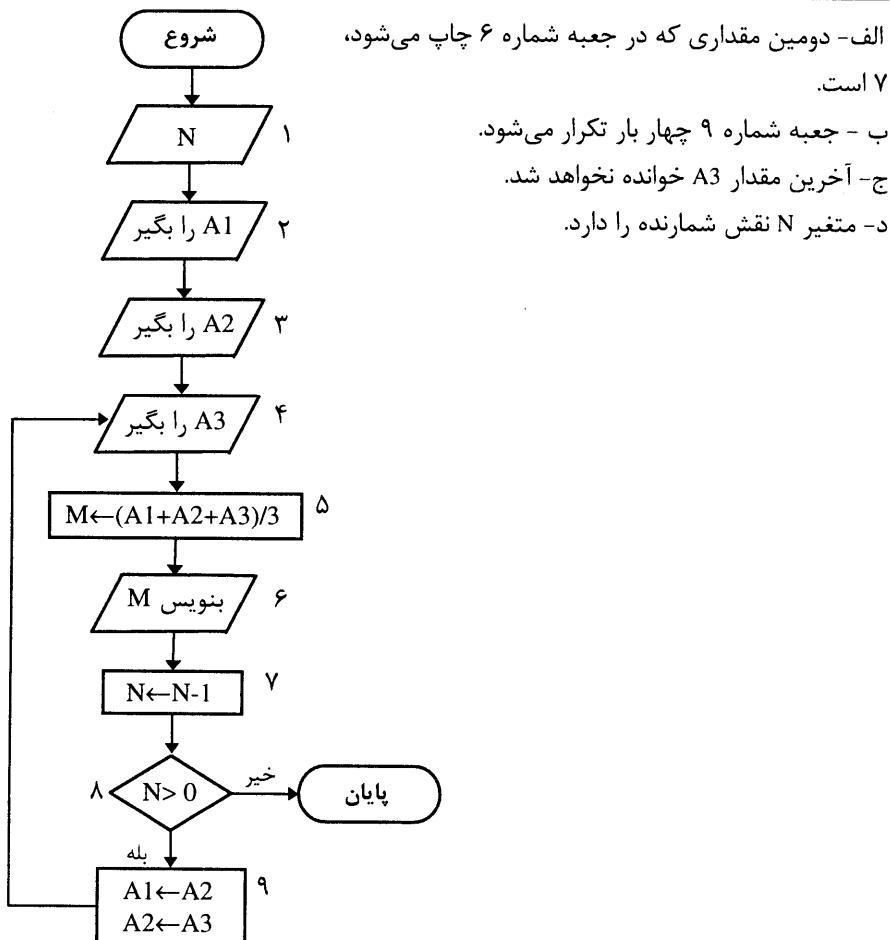
- ۱- فلوچارت را تعریف کرده و توضیح دهید که به چه علت از آن استفاده می‌کنیم.
- ۲- مولفه‌های تصویری یا علائمی را که در ترسیم فلوچارت بکار می‌روند را یکی یکی شرح دهید.
- ۳- حلقه را تعریف نموده و ساختار کلی آنرا در فلوچارت رسم نمائید.
- ۴- مولفه‌های تصویری شروع و پایان را توضیح داده و شرح دهید که هر فلوچارتی میتواند چند شروع و چند پایان داشته باشد.
- ۵- برای اینکه یک مسئله را بكمک کامپیوتر حل نمایید، سه مرحله اصلی را باید طی کنید، هر یک را توضیح دهید.
- ۶- اگر به هریک از علائمی که در ترسیم فلوچارت بکار می‌روند یک جعبه بگوئیم، سعی کنید برای مثالهای ۱ و ۲ و ۳ همین فصل، عمل هر یک از جعبه‌ها را به طور کامل و با توضیحات کافی و جامع تشریح نمایید.
- ۷- فرم کدامیک از علائم فلوچارت در هر مرحله غلط میباشد، هر یک را توضیح دهید.



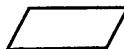
- ۸- فرم نوشتاری کدامیک از علائم فلوچارت در هر مرحله غلط میباشد، بطور کامل شرح دهید.



- ۹- نقش نشانگر فایل را بطور کامل شرح دهید.
 - ۱۰- علائمی را که برای نمایش فایل و خواندن داده‌ها از آن استفاده می‌شود. بطور کامل شرح داده و نحوه ارتباط آنها را با شکل نمایش دهید (برای سادگی می‌توانید یک مثال بیاورید).
 - ۱۱- با استفاده از فلوچارت مقابل و داده‌های ذیل، درستی یا نادرستی گزاره‌های زیر را مشخص کنید.
- داده‌ها : $N=6$ ، $A2=4$ ، $A1=3$ ، $A3=5,3,10,8,4,7,8$ و



پرسش‌های چهار گزینه‌ای



۱- در فلوچارت، شکل مقابل نشانه چیست ؟

الف) ورودی ب) خروجی ج) محاسبه و جایگزینی

د) موارد الف و ب

۲- کدامیک از عبارات زیر نادرست است ؟

الف) هر فلوچارت می‌تواند دارای چند شروع باشد.

ب) نقطه شروع و پایان فلوچارت را با نشان میدهیم.

ج) برای نمایش شرطها از استفاده می‌کنیم.

د) برای نشان دادن عملیات ورودی از استفاده می‌کنیم.

۳- کدامیک از عبارات زیر نادرست است ؟

الف) هر فلوچارت دارای یک شروع و یک پایان است.

ب) در علامت مستطیل فقط یک فلش خارج می‌شود اما می‌تواند چند فلش وارد شود.

ج) در لوزی می‌تواند یک فلش وارد اما حد اکثر سه فلش می‌تواند خارج شود.

د) در علامت مستطیل حداقل یک فلش می‌تواند خارج شود و حد اکثر سه تا .

۴- کدامیک از عبارات زیر درست می‌باشد؟

الف) در متوازی الاضلاع حداقل یک فلش وارد می‌شود و حداقل هم یک فلش خارج می‌شود.

ب) در متوازی الاضلاع فقط یک فلش وارد می‌شود و فقط یکی هم خارج می‌شود.

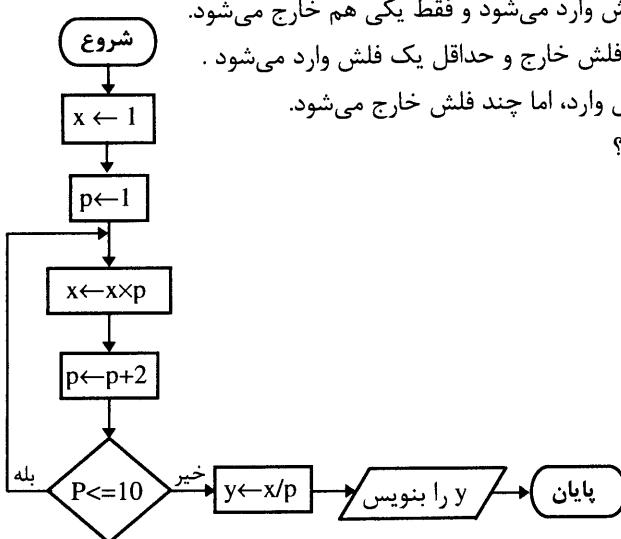
ج) در متوازی الاضلاع حداقل یک فلش خارج و حداقل یک فلش وارد می‌شود .

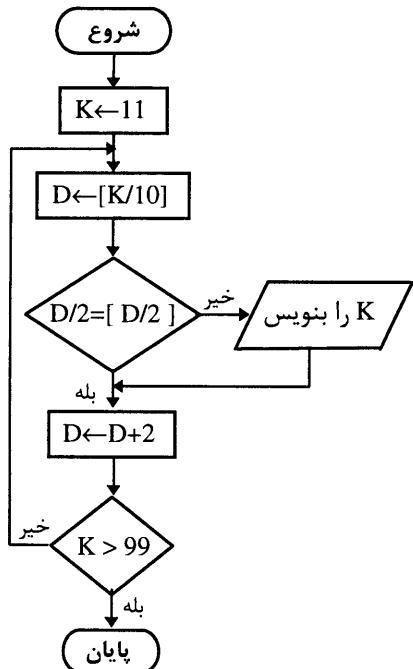
د) در متوازی الاضلاع فقط یک فلش وارد، اما چند فلش خارج می‌شود.

۵- خروجی فلوچارت مقابل چیست؟

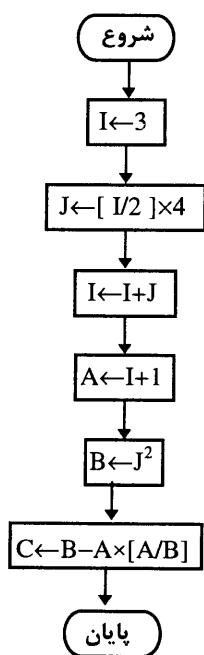
الف) ۹۹ ب) ۱۱

ج) ۹۰ د) ۱۰

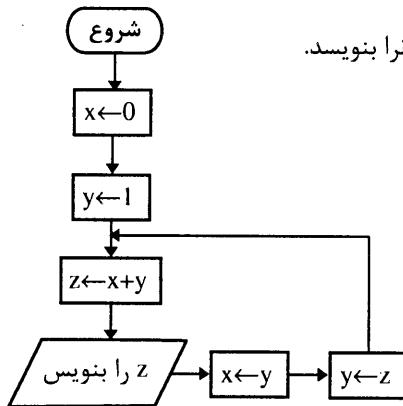




- ۶- نتیجه اجرای فلوچارت مقابل چیست؟
- الف) اعداد زوج دورقمی را چاپ می کند.
 - ب) همه اعداد ۱۱ تا ۹۹ را چاپ می کند.
 - ج) اعداد فرد دو رقمی را چاپ می کند.
 - د) هیچکدام.



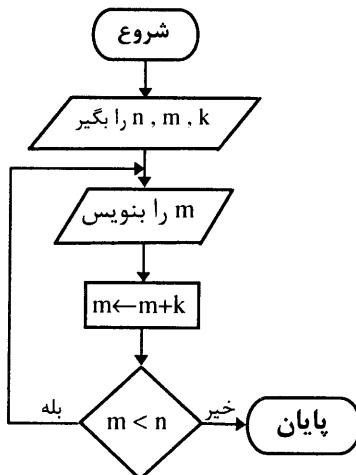
- ۷- در فلوچارت مقابل مقدارنهایی C چقدر است؟
- الف) ۶
 - ب) صفر
 - ج) ۳،۶
 - د) هیچکدام



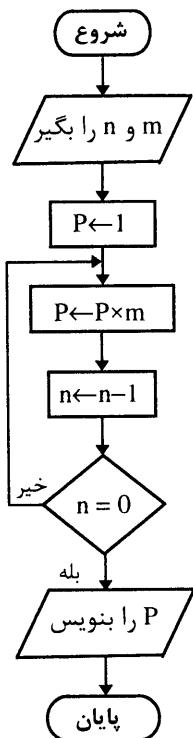
۸- در فلوچارت مقابل، پنج خروجی اولیه آنرا بنویسد.

- الف) ۰۱۰ و ۰۱۰
- ب) ۰۲۰ و ۰۳۰ و ۰۵۰
- ج) ۰۱۰ و ۰۲۰ و ۰۳۰
- د) ۰۱۰ و ۰۱۰ و ۰۰۰

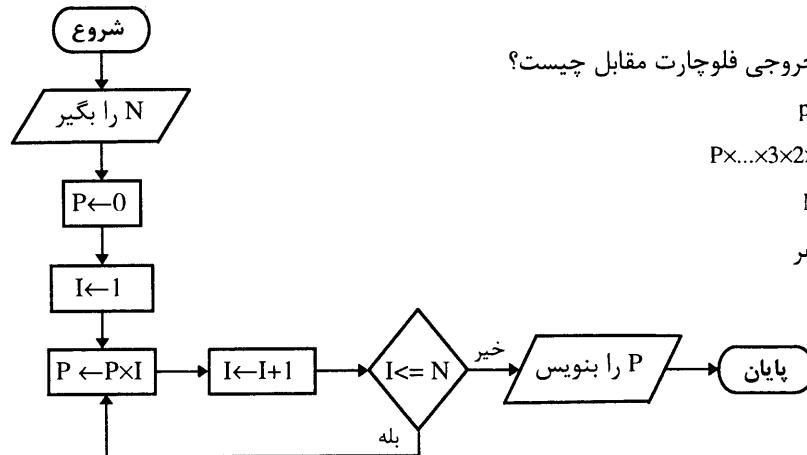
۹- فلوچارت زیر را در نظر گرفته و معین کنید که بجای n , m , k به ترتیب چه اعدادی باید قرار داد تا اعداد زوج بین ۲ تا ۲۰ را چاپ نماید.



- الف) ۱-۱-۰۰۰
- ب) ۰۰-۰-۲۲
- ج) ۰۰-۰-۲۱
- د) موارد ب و ج

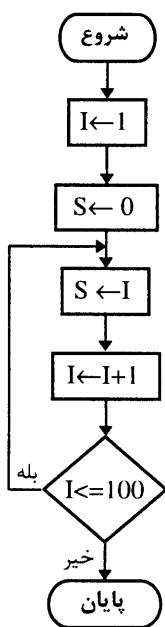


- ۱۰- کدام گزینه عمل فلوچارت زیر را توضیح می‌دهد؟
- الف) در p مقدار $m!$ ذخیره شده است.
 - ب) در p مقدار n به توان m ذخیره شده است.
 - ج) در p مقدار $n!$ ذخیره شده است.
 - د) در p مقدار m به توان n ذخیره شده است.



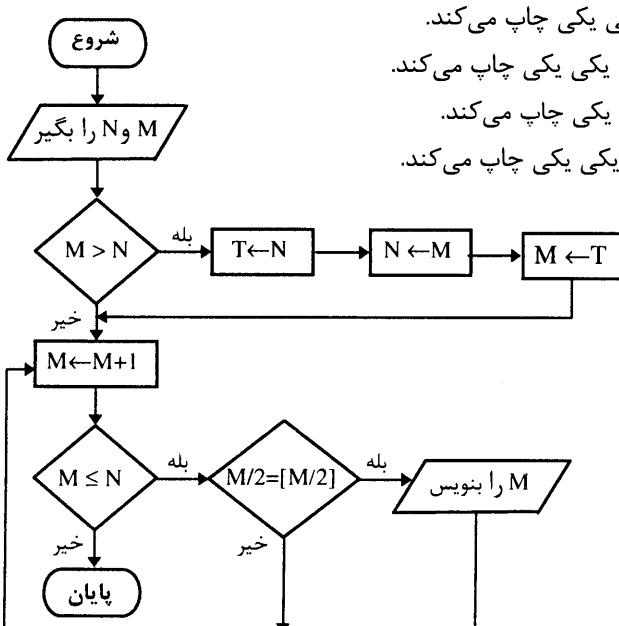
- ۱۱- خروجی فلوچارت مقابل چیست؟
- الف) $p!$
 - ب) $P \times \dots \times 3 \times 2 \times 1$
 - ج) $N!$
 - د) صفر

۱۲- کدامیک از گزینه ها عمل فلوچارت مقابل را تو ضیح می دهد.



- الف) مجموع اعداد متولی از ۱ تا ۱۰۰ را محاسبه می کند.
- ب) پس از انجام عملیات، مجموع اعداد از ۱ تا ۱۰۰ را در S قرار می دهد.
- ج) در نتیجه اجرای آن مقدار صفر در S ذخیره می گردد.
- د) بعد از اجرای آن فقط مقدار ۱۰۰ در S ذخیره می گردد.

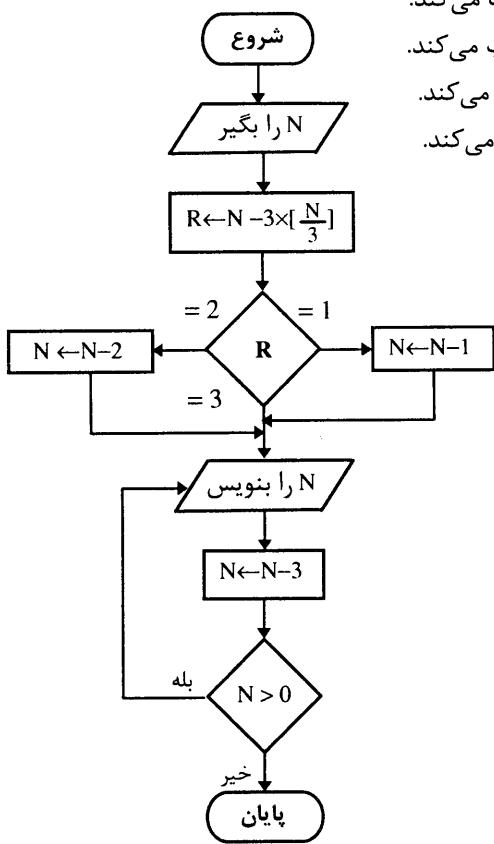
۱۳- کدامیک از گزینه های زیر در مورد فلوچارت زیر صحیح است؟



- الف) تمام اعداد از M تا N را یکی یکی چاپ می کند.
- ب) تمام اعداد زوج از M تا N را یکی یکی چاپ می کند.
- ج) تمام اعداد از N تا M را یکی یکی چاپ می کند.
- د) تمام اعداد زوج از N تا M را یکی یکی چاپ می کند.

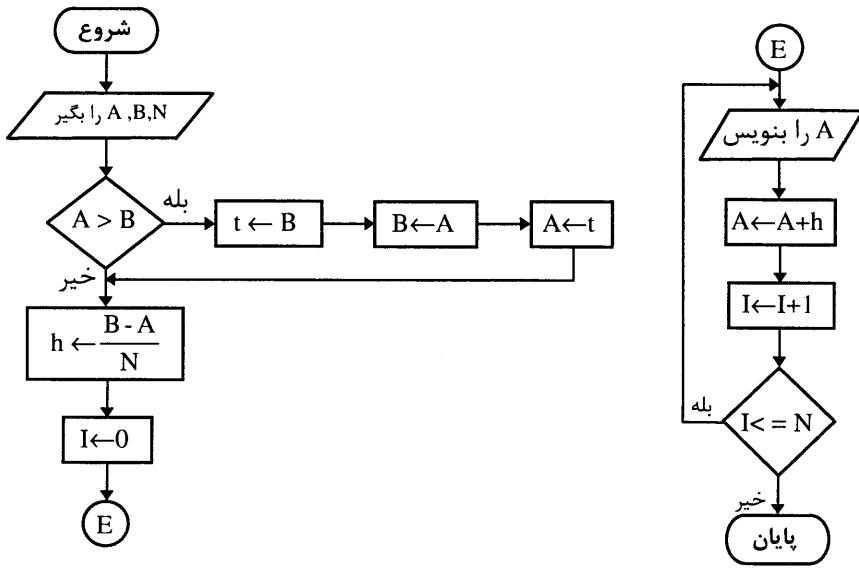
۱۴- کدام گزینه عمل فلوچارت مقابل را بطور کامل توضیح می دهد؟

- الف) اعداد مضرب ۳ از ۳ تا N را یکی یکی چاپ می کند.
- ب) اعداد مضرب ۳ از ۰ تا N را یکی یکی چاپ می کند.
- ج) اعداد مضرب ۳ از N تا ۳ را یکی یکی چاپ می کند.
- د) اعداد مضرب ۳ از N تا ۰ را یکی یکی چاپ می کند.



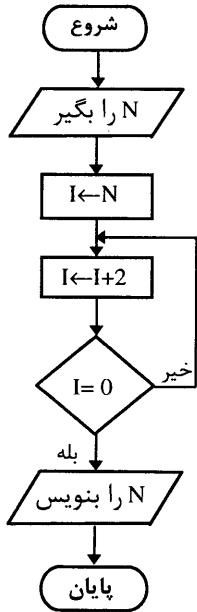
۱۵- کدام گزینه در مورد فلوچارت زیر نادرست است؟

- الف) تمام اعداد صحیح از A تا B را چاپ می کند.
- ب) تمام اعداد از A تا B را که به مقدار عددی h، دویدوی آنها اختلاف دارند را می نویسد.
- ج) بازه [A.B] را به $N+1$ قسمت مساوی تقسیم کرده و نقاط تقسیم را چاپ می کند.
- د) بازه [A.B] را به $N+1$ نقطه بطول یکسان تقسیم می کند.

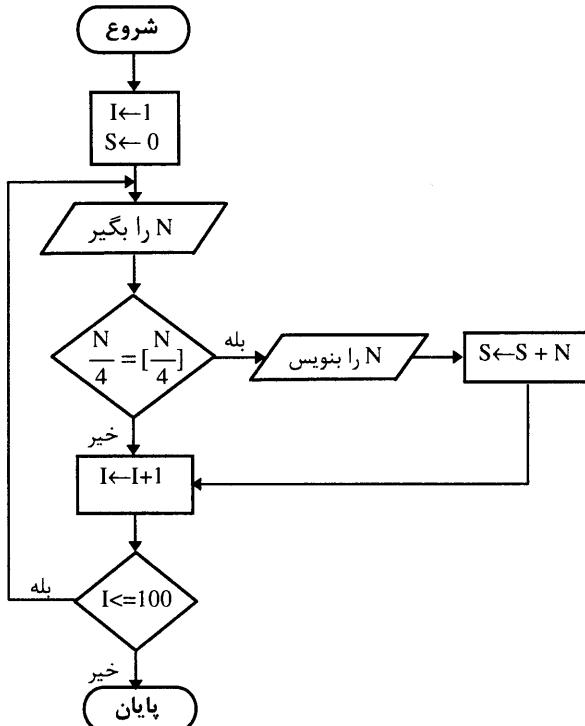


۱۶- خروجی فلوچارت مقابل چیست؟

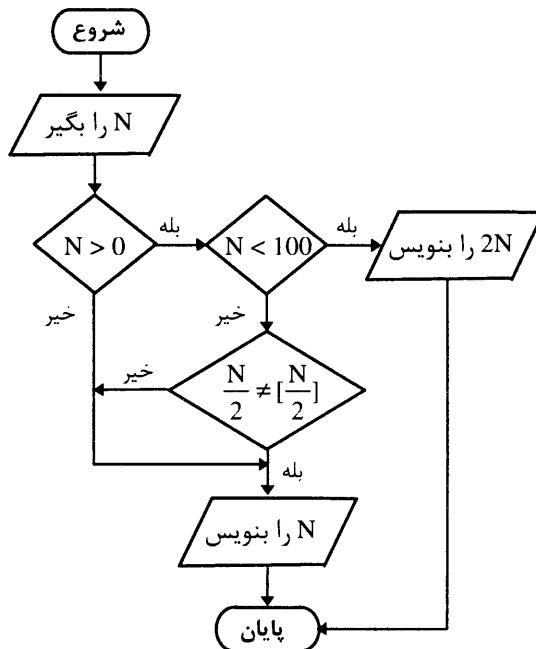
- الف) صفر را چاپ می کند.
- ب) مقدار N را چاپ می کند.
- ج) هیچ خروجی ندارد.
- د) مقدار I را چاپ می کند.



- ۱۷- کدام گزینه عمل فلوچارت زیر را توصیف می کند؟
- الف) صد عدد را یکی یکی گرفته و مجموع آنها را محاسبه می کند.
- ب) صد عدد را یکی یکی گرفته و اعداد مضرب چهار را یکی یکی چاپ می کند و نیز مجموع آنها را چاپ می کند.
- ج) صد عدد را یکی یکی گرفته و اعداد مضرب دو تا چهار را یکی یکی به همراه مجموع آنها چاپ می کند
- د) هیچکدام.



- ۱۸- در فلوچارت زیر کدام گزینه نادرست است؟
- الف) اگر $0 < N < 100$ و $N \neq 2N$ سپس 2N را چاپ می کند.
- ب) اگر $0 < N$ باشد مقدار N را چاپ می کند.
- ج) اگر $0 < N$ یا N فرد باشد N را چاپ می کند.
- د) اگر $100 < N$ باشد و فرد هم نباشد مقدار N را چاپ می کند.



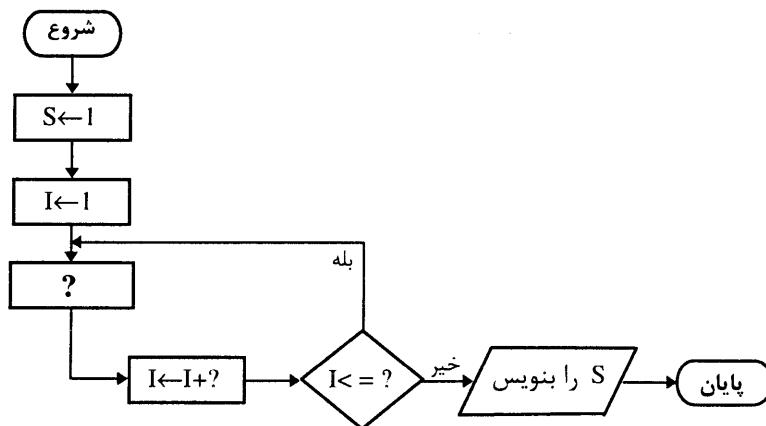
۱۹- در فلوچارت زیر بجای علامت سوال چه گزینه ای را باید قرار داد تا در S مقدار زیر ذخیره شود؟
 $S=1\times 3\times 5\times \dots \times 101$

ب) $100 \leftarrow S \times I$ و ۱ و

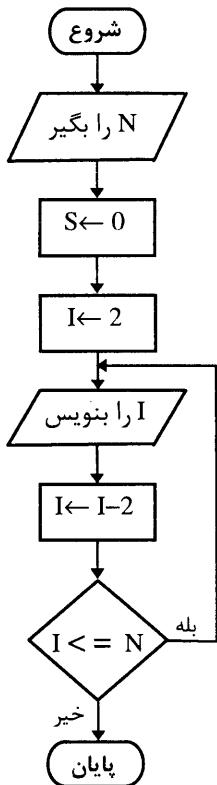
د) $S \leftarrow S \times I$ و ۲ و ۱۰۱

الف) $100 \leftarrow S \times I$ و ۱ و

ج) $I \leftarrow I+1$ و ۲ و ۱۰۱



۲۰- کدام گزینه درست است؟



- الف) فلوچارت هیچگاه پایان نمی‌پذیرد.
- ب) فلوچارت فقط مقدار دو را چاپ می‌کند.
- ج) فلوچارت مقادیر ۲ و ۰ و -۲ و ... را چاپ می‌کند.
- د) فلوچارت هیچ مقداری را چاپ نمی‌کند.

مسائل

۱- در هر یک از مجموعه های زیر عدد N طبیعی و x حقیقی است. فلوچارتی رسم کنید که هر یک از سری های زیر را محاسبه و مقدار نهایی را چاپ نماید (به ازاء N و X ای که دریافت می شود).

$$(الف) \quad S = \frac{4}{2} + \frac{5}{3} + \frac{6}{4} + \dots + \frac{N+3}{N+2}$$

$$(ب) \quad P = \frac{1}{3} + \frac{1}{3^2} + \frac{1}{3^3} + \dots + \frac{1}{3^N}$$

$$(ج) \quad T = 1 - x + x^2 - x^3 + \dots + (-1)^{2n-1} \cdot x^n$$

$$(د) \quad H = \frac{x^5}{5!} - \frac{x^{10}}{10!} + \frac{x^{15}}{15!} - \dots + (-1)^{5(2n-1)} \cdot \frac{x^n}{n!}$$

$$(ه) \quad E = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

$$(و) \quad F = 2 + 6 + 12 + \dots + (n-1)n$$

۲- فلوچارتی رسم کنید که کوچکترین مقدار I را طوری تعیین کند که رابطه زیر برقرار باشد:

$$2 + \frac{1}{2!} + \frac{1}{3!} + \dots + \frac{1}{I!} >= 2.71$$

۳- فلوچارتی رسم کنید که معین کند مجموع چند جمله از دنباله زیر بزرگتر از $5 / .$ است.

$$\frac{1}{3}, \frac{3}{5}, \frac{5}{7}, \frac{7}{9}, \dots$$

$$\frac{1}{3} - \frac{3}{5} + \frac{7}{5} - \frac{7}{9} + \dots > 0.5 \quad \text{يعني رابطه رو برو برقرار گردد}$$

۴- فلوچارتی رسم کنید که دو عدد صحیح M و N را دریافت و همچنین K ، و اعداد مضرب K بین M و N را یکی تولید و چاپ نماید. همچنین مجموع و تعداد آنها را محاسبه و چاپ کند (فقط بین M و N نه خود آنها).

۵- فلوجارتی رسم نماید که اعداد سه رقمی که خودشان با مقلوبشان برابرند، همچنین تعداد آنها را نیز محاسبه و چاپ نماید.

۶- فلوجارتی رسم کنید که دو عدد صحیح M و N را به عنوان ورودی دریافت و بزرگترین مقسوم علیه مشترک و کوچکترین مضرب مشترک آن دو عدد را با توجه به قضیه اراتستن تولید و چاپ نماید.

۷- فلوجارتی رسم کنید که عدد صحیح و مثبت N را دریافت و باقیمانده تقسیم آن را بر ۲ از طریق تفیریق های متوالی و همچنین خارج قسمت آنرا بدست آورده و چاپ نماید.

۸- فلوجارتی رسم کنید که اعداد سه رقمی را تولید نماید که در نوشتن آنها فقط از ارقام زوج استفاده شده باشد.

۹- فلوجارتی رسم کنید که داده های one two ten را که درون یک فایل قرار دارند در نظر گرفته و یک عدد طبیعی را دریافت و اعمال زیر را انجام دهد.

الف) اگر عدد بزرگتر از ۱۰ بود، عدد دیگری را دریافت نماید.

ب) معادل عدد را به انگلیسی چاپ نماید (اگر کوچکتر از ۱۰ بود).

ج) یک داده را سوال نماید اگر این داده یکی از داده های one two ten بود، معادل انگلیسی آنرا بنویسد و گرنه پیغام دهد که موجود نیست.

۱۰- فلوجارتی رسم کنید که یک عدد در مبنای ۲ را به عنوان ورودی دریافت و معین نماید چند رقمی است.

۱۱- فلوجارتی رسم کنید که عدد طبیعی N را ($2 < N$) سوال نموده و اعمال زیر را انجام دهد.

الف) عدد را یکی یکی سوال نموده و میانگین آنها، همچنین تعداد اعداد زوج و فرد را محاسبه و چاپ نماید.

ب) بزرگترین عدد از بین این N عدد را محاسبه و چاپ نماید.

۱۲- فلوجارتی رسم کنید که دو عدد N و X (که N عددی طبیعی و X عددی حقیقی است) را دریافت و حاصل زیر را محاسبه و چاپ کند.

$$S = \sqrt{1 + X\sqrt{1 + X\sqrt{1 + X\sqrt{1 + \dots + X\sqrt{1 + X}}}}$$

۱۳- فلوجارتی رسم کنید که عدد طبیعی N را به عنوان ورودی دریافت و مجموع و حاصلضرب و نیز تعداد ارقام آنرا محاسبه و چاپ نماید.

۱۴- فلوجارتی رسم کنید که اعداد سه رقمی را تولید و چاپ نماید که ارقام آن از اعداد ۰ و ۲ و ۴ و ۶ تشکیل شده باشد (هم با تکرار ارقام و هم بدون تکرار ارقام).

۱۵ - فلوچارتی رسم کنید که اعداد چهار رقمی را تولید کند که از ارقام ۷ و ۶ و ۹ و ۲ تشکیل شده باشد. (باتکرار ارقام) مسئله را در حالتی که صفر به این ارقام اضافه شود نیز حل نمایید.

۱۶ - فلوچارتی رسم کنید که دو عدد طبیعی M و N را به عنوان ورودی دریافت و حاصلضرب آنها را از طریق جمع بدست آورد.

۱۷ - فلوچارتی رسم کنید که مجدور یک عدد طبیعی مانند N را با توجه به قضیه زیر محاسبه و چاپ نماید.

قضیه : مجموع N عدد فرد متوالی برابر است با مجدور N ، مانند :

$$1+3+5+7+9+11+13 = 7^2 = 49$$

در این مثال دقت کنید که هریک از اعداد را میتوان به فرم زیر نوشت :

$$13 = 2 \times 7 - 1 = 13$$

$$7 = 2 \times 7 - 7 = 7$$

$$1 = 2 \times 7 - 13 = 1$$

$$11 = 2 \times 7 - 3 = 11$$

$$5 = 2 \times 7 - 9 = 5$$

$$9 = 2 \times 7 - 5 = 9$$

$$3 = 2 \times 7 - 11 = 3$$

بطور کلی اگر بخواهیم مجدور N را حساب کنیم باید اعداد فرد ۱ تا $2N-1$ را با یکدیگر جمع نمایید.

۱۸ - فلوچارتی رسم کنید که عدد طبیعی را عنوان ورودی دریافت و جذر آنرا به روش زیر محاسبه و چاپ نماید.

اگر تعداد اعداد فرد کوچکتر از N را جمع نماید و عدد N جذر کامل داشته باشد این تعداد دقیقاً برابر جذر کامل عدد N است، در غیر این صورت مقدار تقریبی جذر عدد N بدست می‌آید.

۱۹ - فلوچارتی رسم نماید که عدد طبیعی N را دریافت و N عدد رایکی یکی سوال نموده و آنهایی که به ۴ یا ۵ ختم میشوند را یکی یکی چاپ و نیز تعداد آنها را محاسبه و چاپ کند.

۲۰- فلوچارتی رسم نماید که یک عدد طبیعی را که معرف یک سال هجری است دریافت و معین کند آیا این سال کبیسه است یا خیر؟

(هر ۴ سال یک بار تعداد روزهای سال ۳۶۶ روزی شود، که به آن سال کبیسه گویند. همچنین اولین سال کبیسه سال دوم هجری شمسی بوده است) .

یادداشت‌های درس

فصل سوم

آرایه‌ها (لیستها)



اهداف کلی :

- ۱) فهم دقیق آرایه و انواع آن (یک بعدی و دو بعدی)
- ۲) آشنائی با اندیس و متغیر اندیس دار
- ۳) آشنا شدن با کاربرد آرایه ها در الگوریتم نویسی

هدفهای رفتاری :

دانشجو پس از مطالعه این فصل باید بتواند :

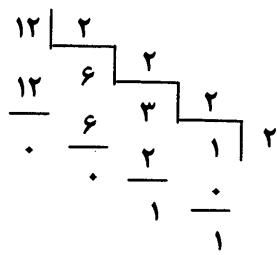
- ۱) خواص آرایه ها را بشناسد و از آنها در نوشتن الگوریتم ها استفاده کند .
- ۲) از متغیرهای اندیس دار در نوشتن الگوریتم ها استفاده نماید .

در فصل اول و دوم مفاهیم الگوریتم و فلوچارت را به طور مفصل توضیح دادیم و مسائل مختلفی را حل کردیم. ولی طیف گسترده‌ای از مسائل را یا نمی‌توانستیم حل کنیم و یا اگر هم حل می‌کردیم راه حل آنها بسیار حجیم و طولانی می‌شد. در این فصل مفاهیم جدیدی را معرفی نموده و هر یک را بطور کامل توضیح می‌دهیم، از قبیل آرایه‌های یک بعدی (بردار)، آرایه‌های دو بعدی (ماتریس) و متغیر اندیس دار که بسیاری از مسائل را با استفاده از این مفاهیم به طور ساده‌تری می‌توانید حل کنید.

برای سادگی مطلب ابتدا به ذکر مثالی می‌پردازیم:

مثال ۱) عدد طبیعی N مفروض است الگوریتمی بنویسید که معادل این عدد را در مبنای ۲ بدست آورد.

برای انجام این کار ابتدا باید عدد را به دو تقسیم نمائید و باقیمانده صحیح آن را یادداشت نموده و سپس خارج قسمت تقسیم را دوباره بر دو تقسیم کنید و این عمل را تازمانیکه آخرین خارج قسمت صفر نشده است ادامه دهید، سپس کلیه باقیمانده‌ها را از آخر به اول در کنار یکدیگر بنویسید.
برای مثال این عمل را برای عدد ۱۲ انجام دهید:



- ۱ - شروع
- ۲ - N را بگیر.
- ۳ - $R \leftarrow N - 2 \left[\frac{N}{2} \right]$
- ۴ - R را بنویس
- ۵ - $N \leftarrow \left[\frac{N}{2} \right]$
- ۶ - اگر $N > 0$ سپس برو به ۳.
- ۷ - پایان.

توجه کنید که به جای عدد (1100) عدد (0011) را بدست آوردید . پس نیاز دارید که تمامی این باقیمانده‌ها را در جایی ذخیره کرده و سپس از آخر به اول آنها را بنویسید که این عمل را با استفاده از آرایه‌ها به راحتی می‌توان انجام داد.

قبل از تعریف دقیق آرایه ابتدا مفهوم اندیس و متغیر اندیس‌دار را بیان می‌نمائیم.

تعریف ۱-۳: اندیس عبارت از علامت یا حرف یا عدد خاصی است که در کنار دو چیز همنام قرار می‌گیرد تا از یکدیگر به راحتی تمیز داده شوند.

به عنوان مثال فرض کنیم دو خانه را به نام X می‌شناسید ، برای اینکه به راحتی بتوانید این دو خانه را از یکدیگر تشخیص دهید در زیر اسم هر کدام عددی قرار دهید .

$(\boxed{} \quad \boxed{})$
 $X_1 \qquad X_2$

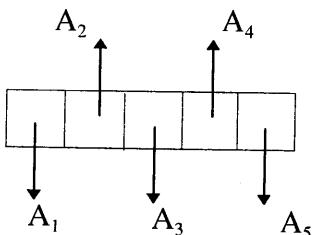
به این ترتیب خانه‌های X_1 و X_2 به راحتی از یکدیگر قابل تشخیص‌اند. در الگوریتم‌نویسی برای ذخیره کردن مقادیر و داده‌ها از خانه‌هایی به نام متغیر استفاده می‌کردیم، پس بجاست که متغیر اندیس‌دار را نیز تعریف کنیم.

تعریف ۲-۳: متغیر اندیس‌دار مجموعه‌ای از چند متغیر است که همگی آنها یک اسم دارند و وجه تمایز آنها اندیس‌شان است . به عنوان مثال متغیر اندیس‌دار A را بطول ۵ در نظر بگیرید . در این صورت



داریم :

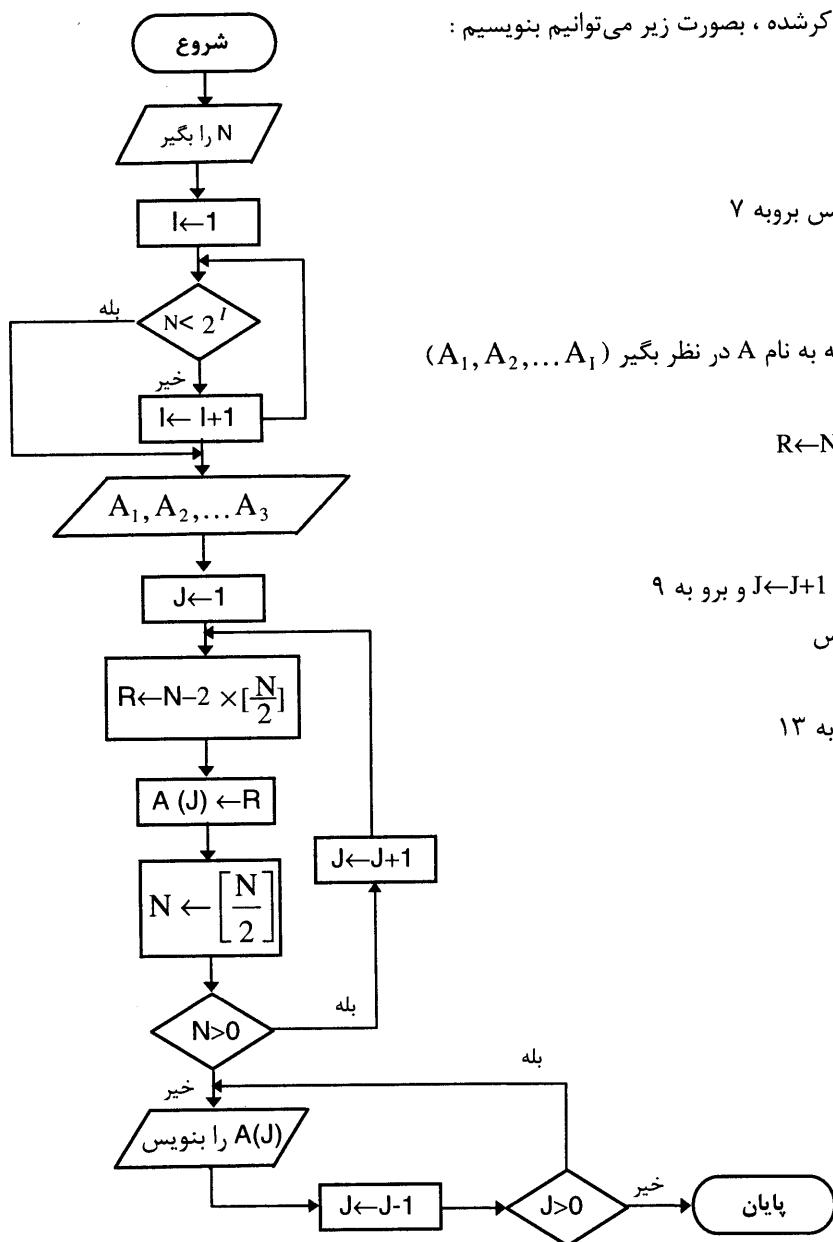
یعنی پنج خانه که نام همگی آنها A است ولی اولی را با اندیس ۱ (A_1) و دومی را با اندیس ۲ (A_2) می‌شناسیم و الی آخر. هرگاه بگوئیم A اندیس ۲ یعنی به سراغ مجموعه‌ای از خانه‌ها که اسمشان A است رفته و در بین آنها دومی را انتخاب می‌کنیم.



به این دسته از متغیرهای اندیس‌دار (با طول ثابت) بردار نیز گفته می‌شود و در الگوریتم‌نویسی به جای A_1 می‌توان نوشت ($A(1)$).

مثالاً به متغیر اندیس دار X به طول ۱۰ یک بردار ۱۰ عضوی یا بردار به طول ۱۰ گفته می‌شود. نکته قابل اهمیت این است که می‌توان اندیس خانه‌ها را متغیر گرفته و آنها را تغییر داد. حال مثال قبل را با

توجه به مفاهیم ذکر شده، بصورت زیر می‌توانیم بنویسیم:



۱- شروع

۲- N را بگیر
I ← ۱

۴- اگر $2^I > N$ پس برو به ۷

I ← I+1

۶- برو به ۴

۷- به تعداد I خانه به نام A در نظر بگیر (A_1, A_2, \dots, A_I)

J ← ۱

۹- $R \leftarrow N - 2 \times \left\lfloor \frac{N}{2} \right\rfloor$

۱۰- $A(J) \leftarrow R - 10$

۱۱- $N \leftarrow \left\lfloor \frac{N}{2} \right\rfloor$

۱۲- اگر $N > 0$ پس J ← J+1 و برو به ۹

۱۳- A(J) را بنویس

۱۴- J ← J-1

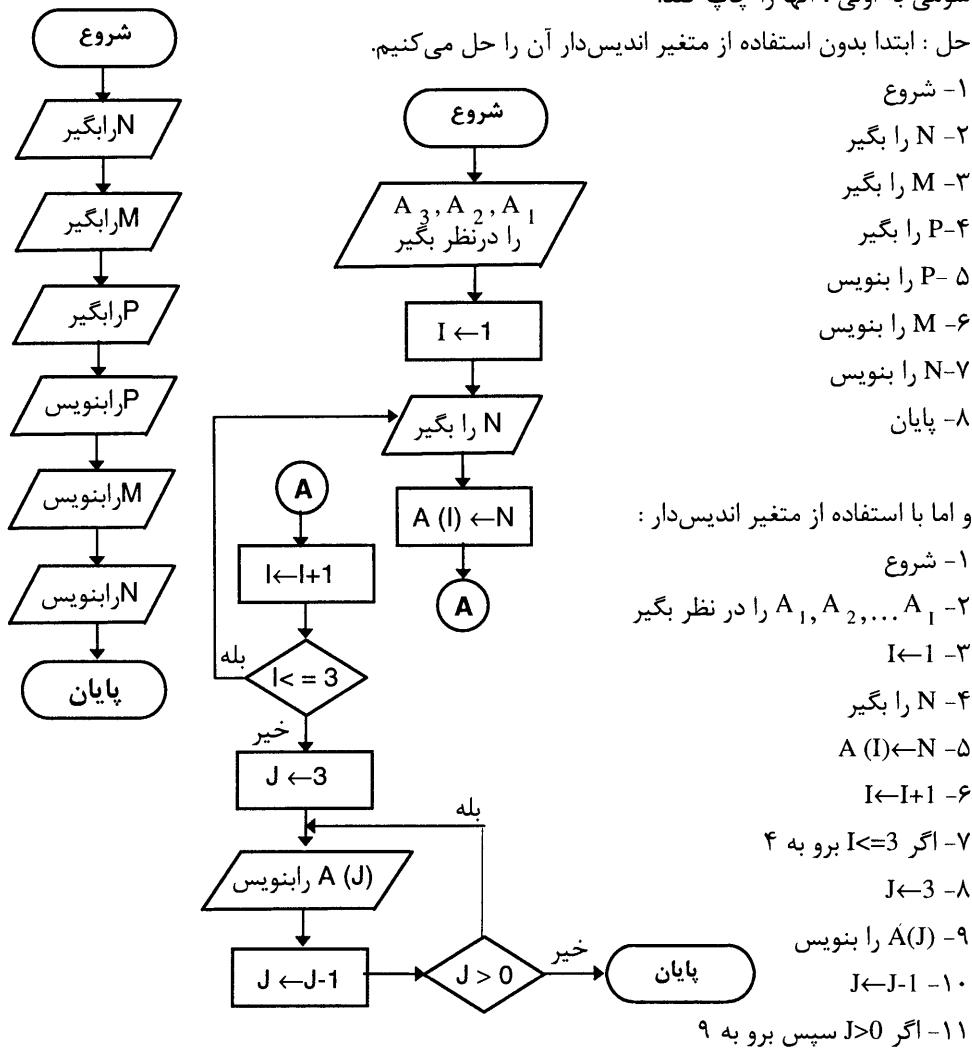
۱۵- اگر $J > 0$ برو به ۱۳

۱۶- پایان

فرض براین است که هر گاه بخواهید به تعداد I خانه در نظر بگیرید (یعنی یک بردار به طول n یا یک متغیر اندیس دار به طول n یا یک بردار n بعدی) آنها را داخل متوازی الاضلاع در فلوچارت قراردهید، چون جزو ورودی فلوچارت محسوب می شوند. ارائه مثال ۱ فقط جنبه کاربرد بردار و متغیر اندیس دار را داشت نه آموزش آنها ، پس اگر بطور دقیق آن را یادنگرفته اید، نگران نباشید. مطالب فعلی را دنبال کنید تا به نتیجه مطلوب دست یابید.

مثال ۲) فرض کنید می خواهید فلوچارتی رسم کنید که سه عدد را به ترتیب یکی یکی گرفته و از عدد سومی به اویلی ، آنها را چاپ کند.

حل : ابتدا بدون استفاده از متغیر اندیس دار آن را حل می کنیم.



۱۲- پایان .

ابتدا سه خانه به نامهای A_1, A_2, A_3 را رزرو می‌کنیم. متغیر I را از ۱ تا ۳ بعنوان اندیس آن تغییر داده و هر بار مقداری را سوال کرده و در یکی از خانه‌های A قرار می‌دهیم.

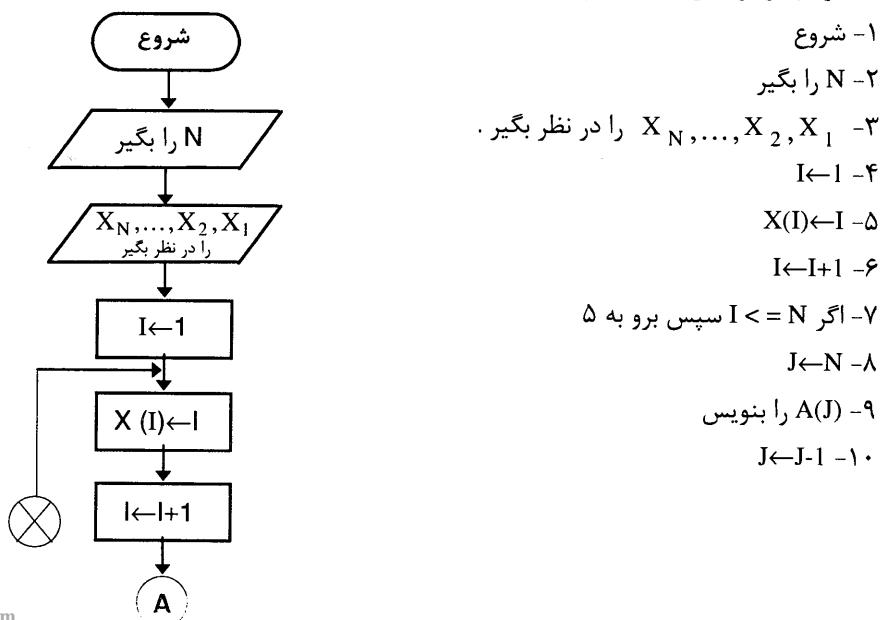
برای اولین بار چون مقدار I یک است، پس اولین مقدار در $A(I)$ یعنی $(1)A$ ذخیره شده و دومین مقدار در $(2)A$ که چون I مساوی ۲ است در $(2)A$ و الی آخر. سپس وقتی خانه‌ها را پر کردیم متغیری را به نام J مساوی ۳ یعنی اندیس آخرین خانه قرار داده و مقدار آن خانه را چاپ می‌کنیم و یکی یکی از J کم می‌کنیم تا به اولین خانه برسیم.

سوال مهم : فرض کنید به جای ۳ عدد بخواهید تعداد ۱۰۰ عدد را یکی یکی به ترتیب دریافت و از آخری به اولی بنویسید. به نظر شما کدامیک از راه حل‌های فوق قابل تعمیم به ۱۰۰ عدد است؟

جواب : اگر بگوئید راه حل اول شاید غلط نباشد ولی عاقلانه و الگوریتمیک نیست و ایرادی که دارد این است که اگر بخواهیم روی این ۱۰۰ عدد محاسبات دیگری را انجام دهیم باید دوباره ۱۰۰ عدد را وارد نمائید و یا ۱۰۰ خانه به نامهای A و B و C و در نظر بگیرید که باز اگر تعداد خانه‌ها زیاد باشد برای نامگذاری آنها با مشکل مواجه خواهید شد و مطمئناً از این روش خسته می‌شود.

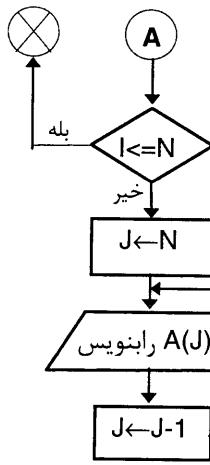
اما راه حل دوم قابل تعمیم به ۱۰۰ یا حتی بیشتر نیز می‌باشد. فقط کافی است به جای ۳ خانه تعداد ۱۰۰ خانه را در نظر بگیرید.

مثال ۳) فلوچارتی رسم کنید که عدد طبیعی N را دریافت و اعداد از ۱ تا N را در X خانه به نام X قرار داده و آنها را از آخر به اول چاپ نماید.



۱۱- اگر $J > 0$ سپس برو به ۹

۱۲- پایان .

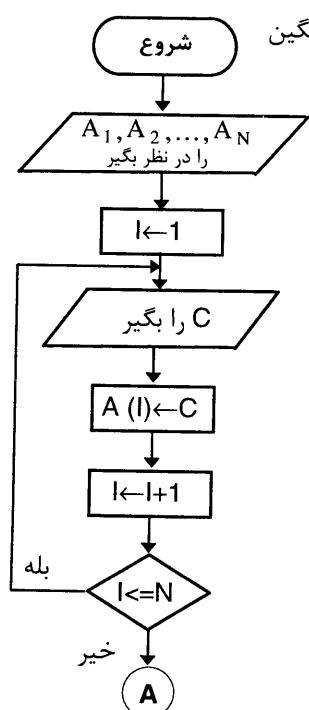


مشاهده می کنید که با تغییرات جزئی، این فلوچارت برای N عدد نیز پاسخگو می باشد.

ممکن است بگوئید که صورت مسئله را چنین نیزمی شود بیان کرد که اعداد از N تا ۱ را یکی یکی چاپ نماید، که در این صورت احتیاجی به این همه متغیر و این چیزها نبود و ساده تر می شد آن را حل نمود، از یک دیدگاه برای این مسئله شاید درست باشد ولی اگر تغییر جزئی در صورت مسئله بدھید دیگر حل آن به طریق بالا غیر ممکن است و حتما باید از متغیرهای اندیس دار استفاده کنید. این مطلب

را با ذکر مثالی بیشتر توضیح می دهیم .

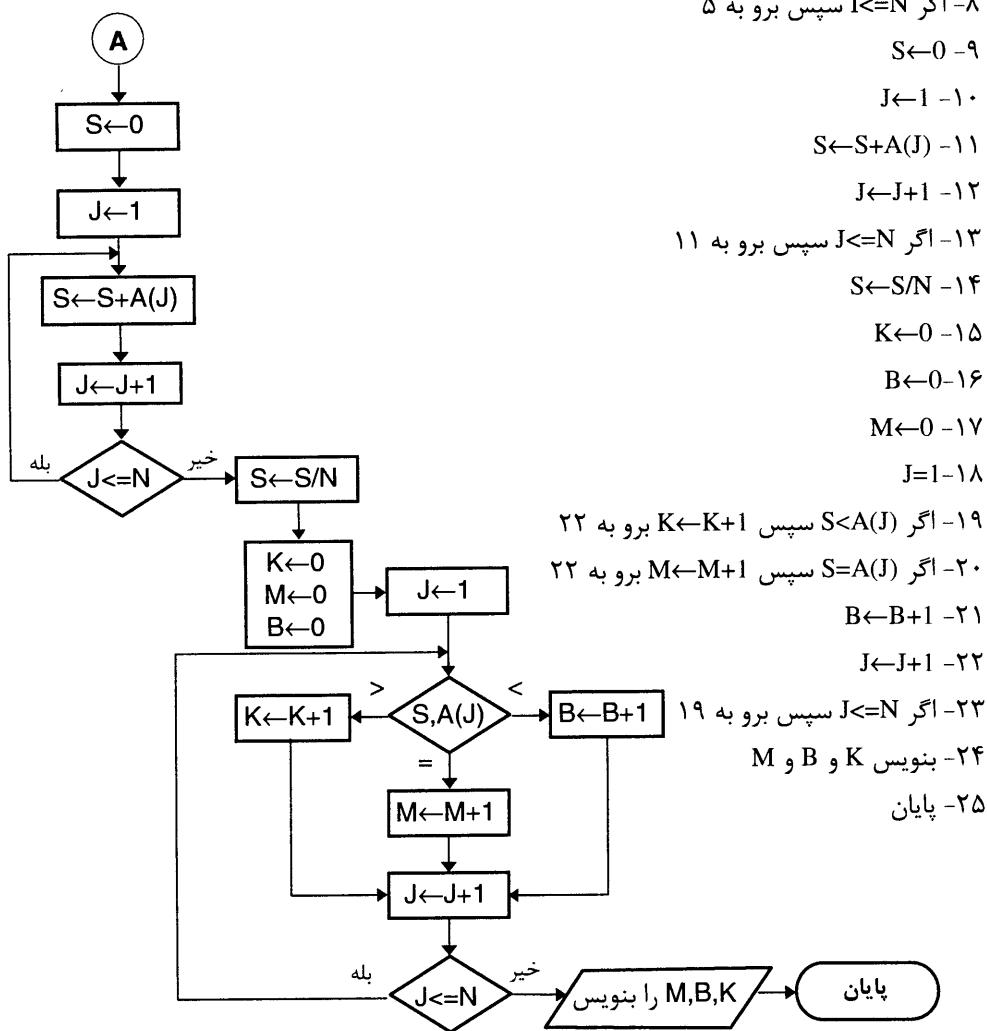
مثال (۴) فلوچارتی رسم نمائید که عدد طبیعی N را سوال و N خانه را در نظر گرفته و N عدد دلخواه را یکی یکی سوال نموده و داخل آنها قرار داده و ابتدا میانگین آنها را محاسبه نماید، سپس مشخص کند چند عدد از میانگین بزرگتر و چند عدد کوچکتر و چند عدد با میانگین برابر است .



حل این مثال بدون استفاده از متغیر اندیس دار بدین صورت است که ابتدا N عدد را سوال نموده، سپس میانگین آنها را محاسبه و برای قسمت بعد دوباره N عدد را وارد کرده و عملیات لازم را انجام دهید که کاری بیهوده است و عملادر الگوریتم نویسی از اینگونه روش ها استفاده نمی کنیم، چون اصولا روش حل یک مسئله نیست بلکه فقط یک راه حل ابتدائی می باشد.

۱- شروع

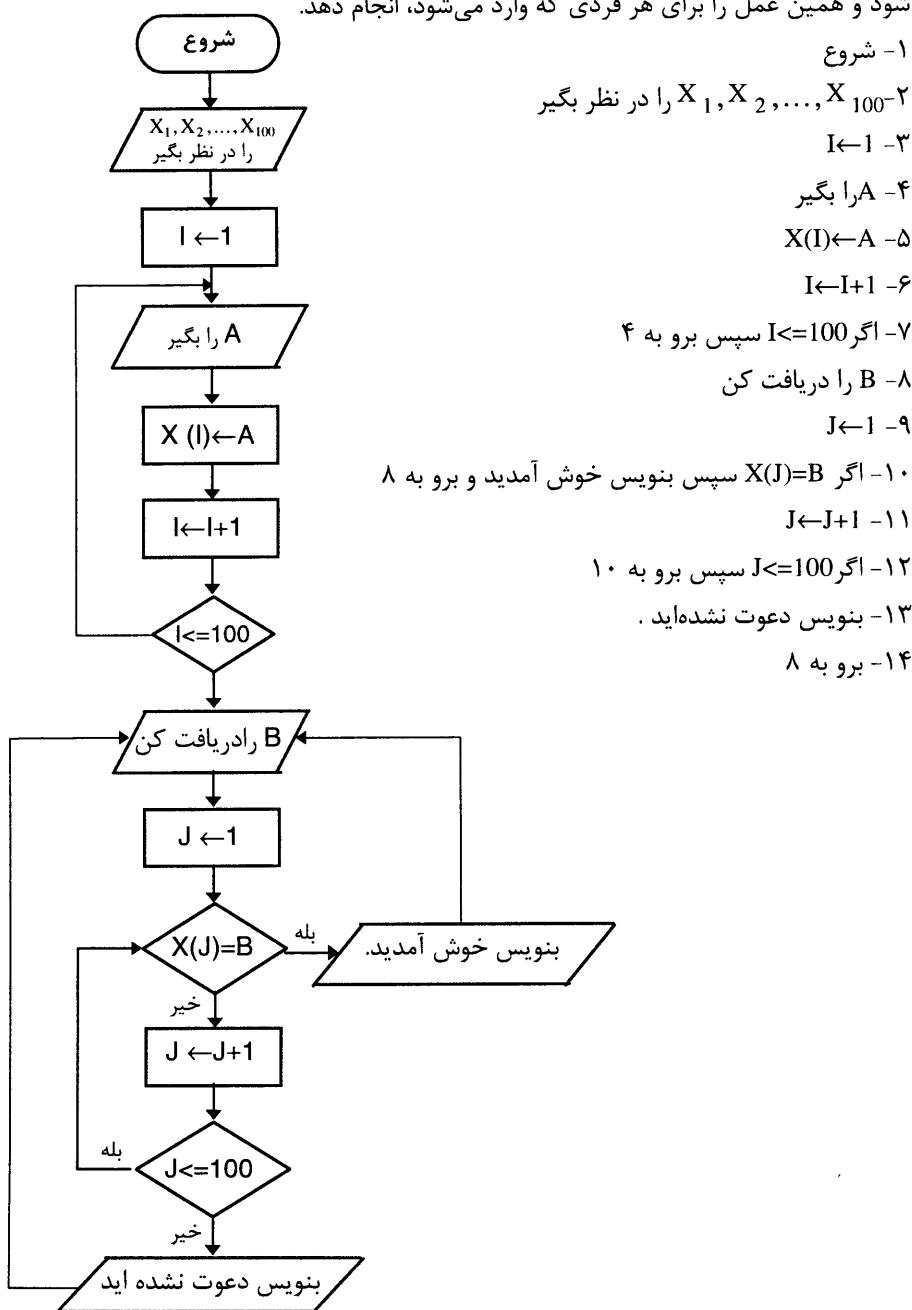
۲- N را بگیر۳- A_1, A_2, \dots, A_N را در نظر بگیر۴- $I \leftarrow 1$ ۵- C را بگیر۶- $A(I) \leftarrow C$ ۷- $I \leftarrow I + 1$



در متغیرهای اندیس دار یا بردارها می‌توان علاوه بر داده های عددی، داده های غیر عددی (حروفی نیز) ذخیره کرد. برای توضیح بیشتر مثال زیر را حل می کنیم.

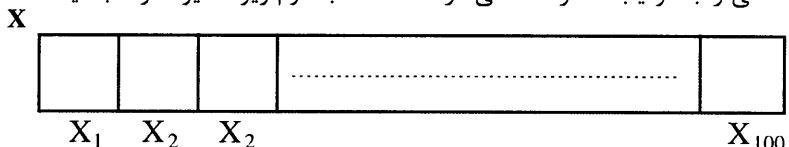
مثال ۵) فرض کنید اسامی ۱۰۰ نفر را که به یک میهمانی دعوت شده اند به کامپیوتر داده ایم، فلوچارتی رسم کنید که هر فردی وارد می شود اسم او را سوال نماید و اگر جزو اسامی دعوت شده بود

به او خوش آمد بگوید و در غیر این صورت بنویسد شما دعوت نشده‌اید و مانع از ورود وی به میهمانی شود و همین عمل را برای هر فردی که وارد می‌شود، انجام دهد.



اگر برای هر فردی صندلی خاصی با شماره خاصی در نظر گرفته باشد و اسمی را به ترتیب شماره صندلی‌ها یعنی از ۱ تا ۱۰۰ (از اسم اولین نفر تا سیمین نفر) وارد کرده باشید با تغییرات جزئی، می‌توان شماره صندلی افرادی که وارد می‌شوند را نیز تغییر نمود.

فرض کنیم اسمی را به ترتیب شماره صندلی در ۱۰۰ خانه به فرم زیر ذخیره کرده باشد.



حال اگر فردی وارد می‌شود، اسم او را سوال نموده و با تک تک اسمی داخل بردار مقایسه می‌کنیم، هر کجا که با یکی از آنها برابر شد کافی است اندیس آن خانه را در جایی ذخیره کنید که در اصل شماره صندلی شخص مورد نظرمی‌باشد. پس به جای خط ۱۲ در الگوریتم فوق، خط زیر را قرار دهید :

۱۲ - اگر $X(J)=B$ سپس $J \leftarrow T$ و بنویس "خوش آمدید" و J و برو به ۸
(به عنوان تمرین فلوچارت آن را با این تغییر رسم کنید).

حال به تشریح خواص بردارها یا متغیرهای اندیس‌دار می‌پردازیم. در حالت کلی دو بردار را که از یک نوع و دارای طول یکسان باشند، می‌توان با هم جمع کرده و یا از هم کم و یا حتی در یکدیگر ضرب نمود.

مثال ۶) فلوچارتی رسم کنید که دو بردار n بعدی (n را سوال نماید) A و B را در نظر گفته و مقادیر مختلفی را سوال نموده و در خانه‌های A و B ذخیره نماید و سپس حاصلجمع و حاصلضرب دو بردار B و A را به ترتیب در دو بردار C و D ذخیره نماید.

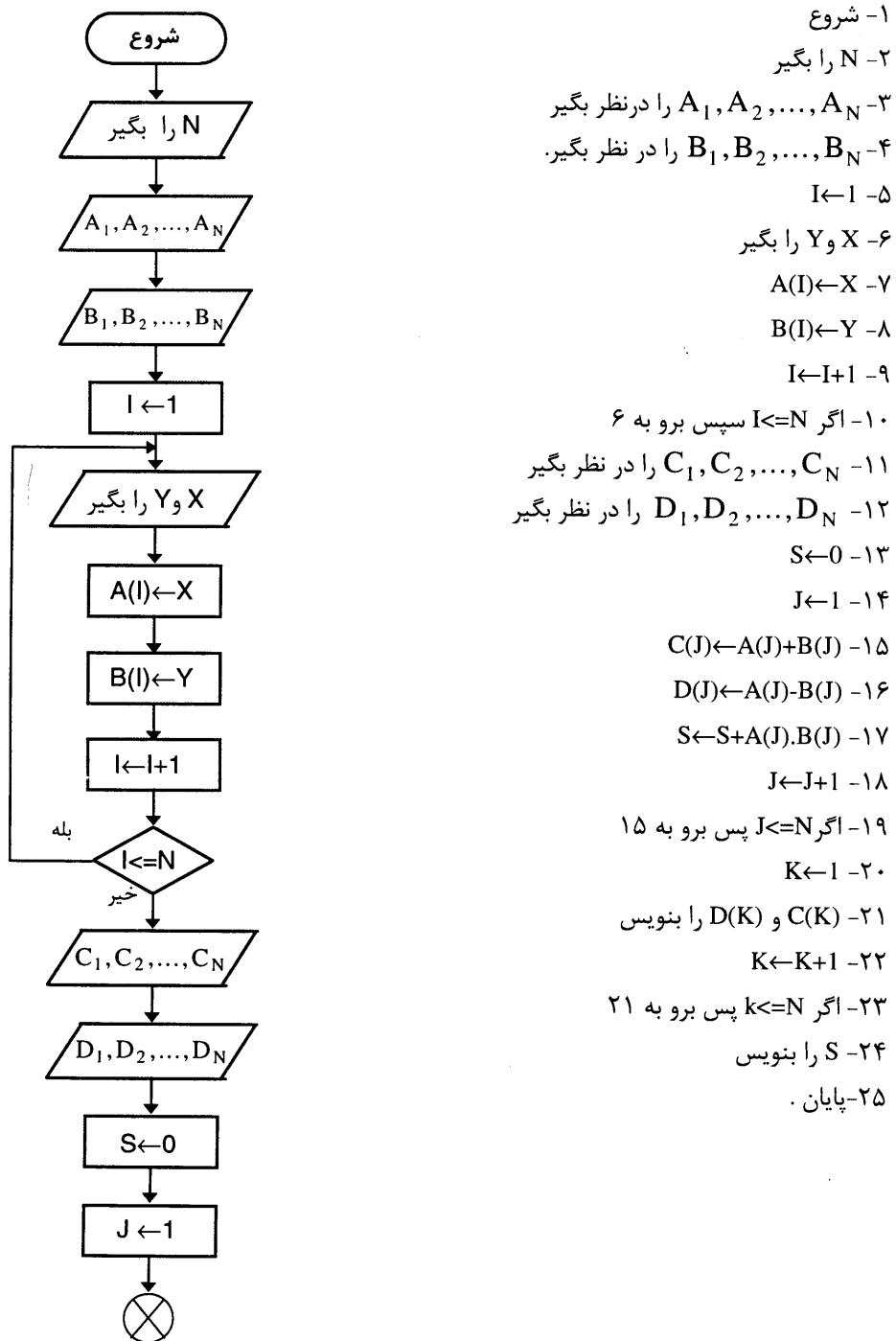
تذکر : فرض کنیم دو بردار A و B ، n بعدی هستند، در ریاضیات جمع ، تفیق و ضرب داخلی آنها را به فرم زیر نمایش می‌دهیم :

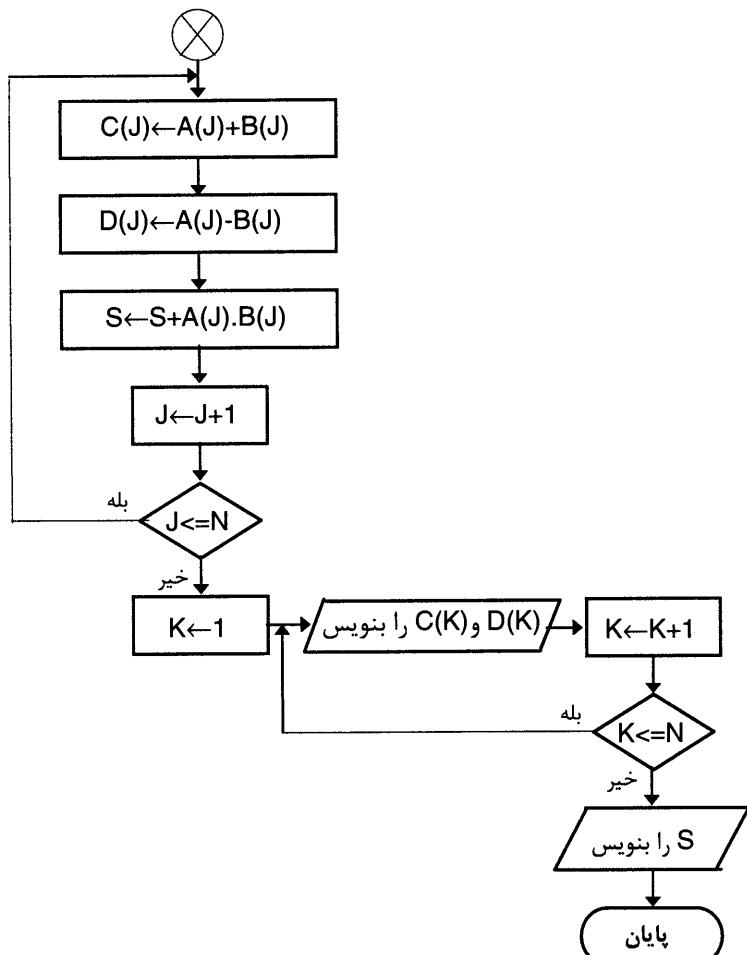
$$A + B = (A_1 + B_1, A_2 + B_2, \dots, A_n + B_n)$$

$$A - B = (A_1 - B_1, A_2 - B_2, \dots, A_n - B_n)$$

$$A \cdot B = (A_1 \cdot B_1 + A_2 \cdot B_2 + \dots + A_n \cdot B_n)$$

نکته‌ای که قابل توجه می‌باشد، این است که مجموع و تفاضل دو بردار، یک بردار است، اما حاصلضرب نقطه‌ای دو بردار یک عدد حقیقی است، که البته مفهوم ضرب دو بردار یک تعریف خاص ریاضی است و کاربردهای فراوانی دارد و بیان آن در اینجا صرفاً به عنوان یک تمرین است.





مثال ۷) الگوریتمی بنویسید که یک لیست N عنصری را در نظر گرفته و N عدد را در آن ذخیره نماید و عنصر اول را با عنصر آخر و عنصر دوم را با عنصر یکی مانده به آخر و به همین ترتیب تمام عناصر را جا به جا کند.

۱- شروع

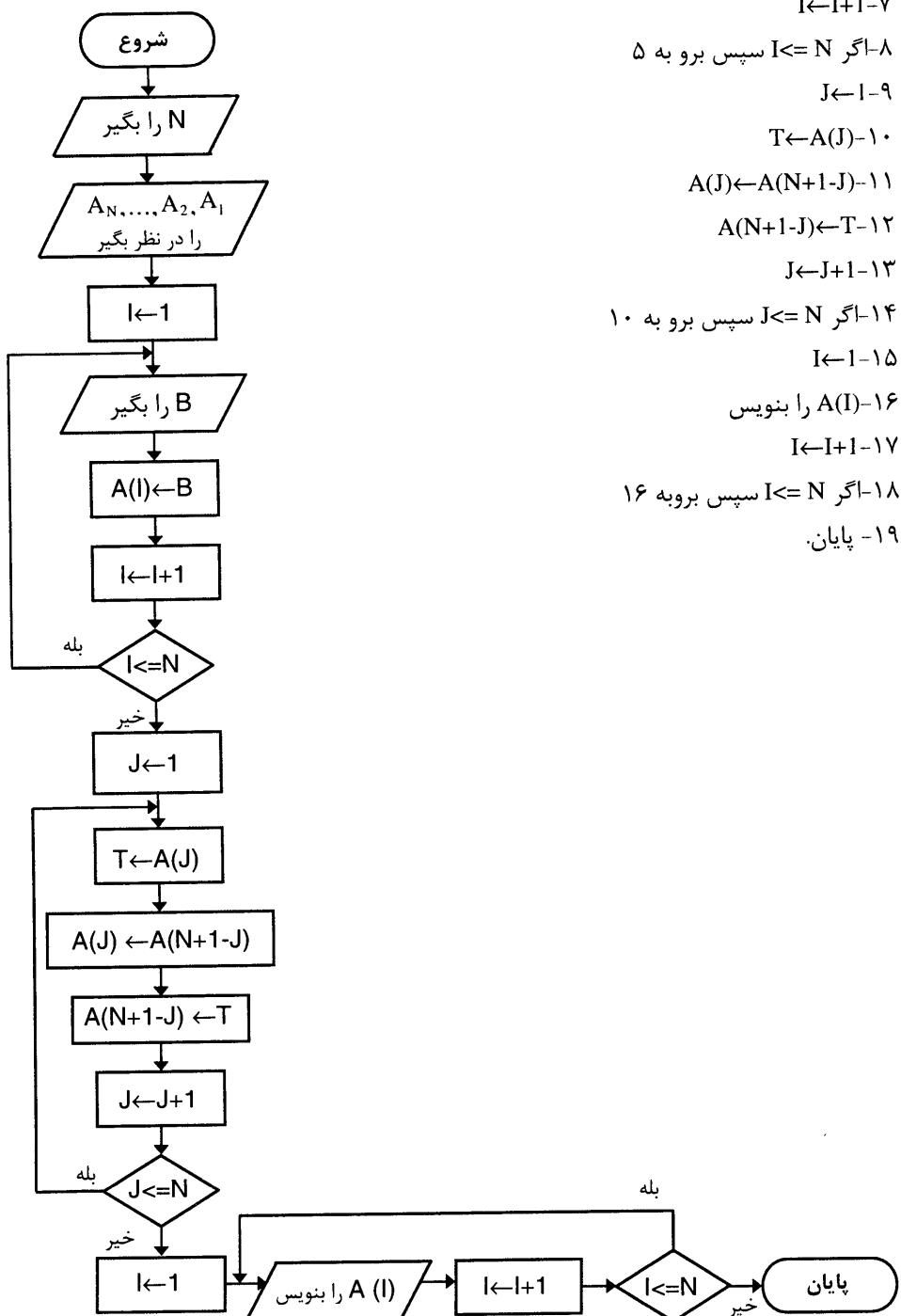
۲- N را بگیر.

۳- A_1, A_2, \dots, A_N را در نظر بگیر

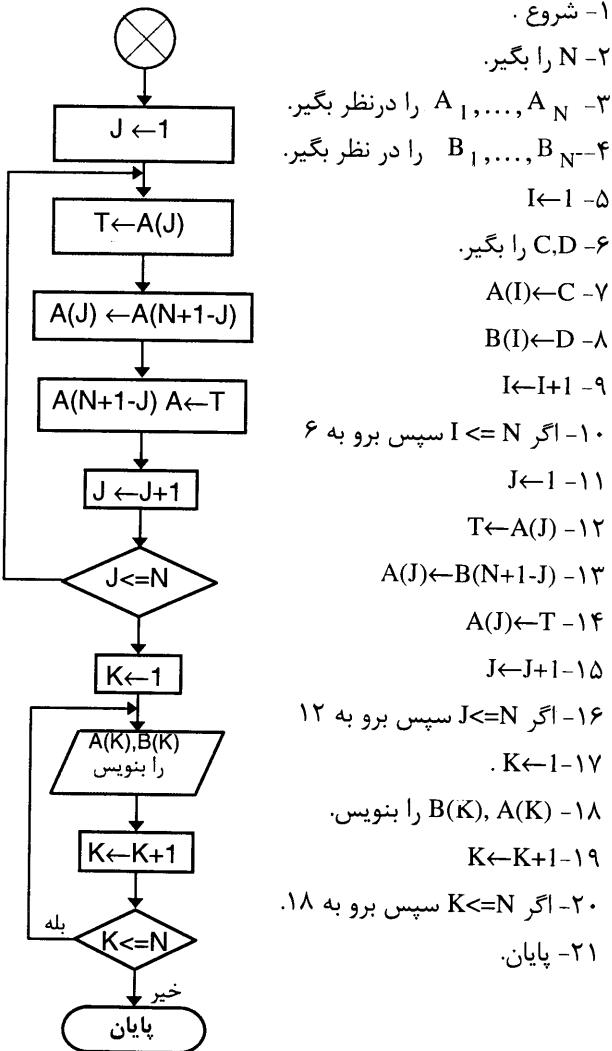
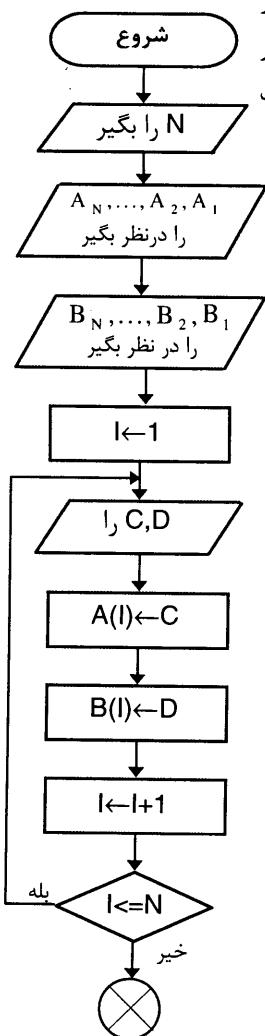
۴- $I \leftarrow 1$

۵- B را بگیر

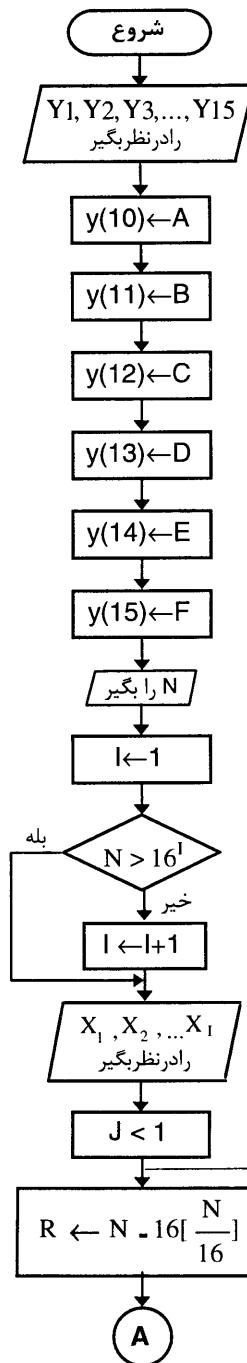
۶- $A(I) \leftarrow B$



مثال ۸، فلوچارتی رسم کنید که دو لیست به نامهای A, B با N خانه را در نظر گرفته و در هر کدام N عدد ذخیره کند، سپس عنصر اول را با عنصر آخر B و عنصر دوم A را با عنصر یکی مانده به آخر B و به همین ترتیب عناصر را جا به جا کرده و عناصر دو لیست را نمایش دهد.



مثال ۹) فلوچارتی رسم کنید که یک عدد طبیعی در مبنای ۱۰ را سوال نموده و معادل آن را در مبنای ۱۶ محاسبه و چاپ کند.



- ۱- شروع
۲- N را بگیر
۳- $I \leftarrow 1$
۴- اگر $N < 16^1$ سپس برو به ۷
۵- $I \leftarrow I + 1$
۶- برو به ۴
۷- X_1, X_2, \dots, X_I را در نظر بگیر
۸- $J \leftarrow 1$
۹- $R \leftarrow N - 16 \cdot [N/16]$
۱۰- اگر $R = 10$ پس $X(J) \leftarrow A$ و برو به ۱۱ و گرنه برو به ۱۱
۱۱- اگر $R = 11$ پس $X(J) \leftarrow B$ و برو به ۱۲ و گرنه برو به ۱۲
۱۲- اگر $R = 12$ پس $X(J) \leftarrow C$ و برو به ۱۳ و گرنه برو به ۱۳
۱۳- اگر $R = 13$ پس $X(J) \leftarrow D$ و برو به ۱۴ و گرنه برو به ۱۴
۱۴- اگر $R = 14$ پس $X(J) \leftarrow E$ و گرنه برو به ۱۵
۱۵- اگر $R = 15$ پس $X(J) \leftarrow F$ و گرنه برو به ۱۵
۱۶- $N \leftarrow [N/16]$
۱۷- اگر $N > 0$ پس $J \leftarrow J + 1$ و برو به ۹
۱۸- $T \leftarrow J - 1$
۱۹- $X(T)$ را بنویس
۲۰- $T \leftarrow T - 1$
۲۱- اگر $T > 0$ پس برو به ۱۹
۲۲- پایان .

ابتدا فصل را با نام لیست یا آرایه آغاز نمودیم اما صحبتی از آن به میان نیاوردیم به این علت که ابتدا با عناصری که آرایه ها را تشکیل می دهند آشنا شوید و مثالهایی را حل نمائید، سپس به طور اصولی آنها را تعریف نمائیم. اگر مثال ۲ این فصل را در نظر بگیرید، متوجه می شوید که فقط با مقادیر عددی سروکار داشتیم، حال می خواهیم مثالی را مطرح کنیم که داده ها در این مثال بیشتر جنبه اطلاعاتی داشته باشد.

مثال ۱۰) فرض کنید می خواهیم مشخصات دانشجویانی را که نمره آنها از میانگین نمرات کلاس بیشتر است بدست آوریم.

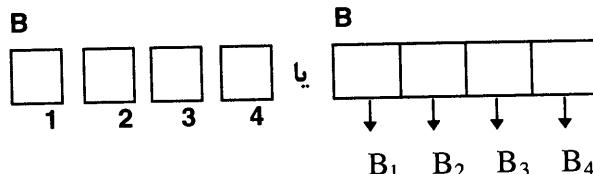
راه حلی که سریعاً به ذهن می رسد این است که مشخصات هر دانشجو را به همراه نمره اش یک بار خوانده و سپس میانگین نمرات کل کلاس را حساب نمائیم، سپس یک بار دیگر تک تک نمرات را خوانده و هر یک را با میانگین مقایسه نمائیم و هر کدام بیشتر بود، مشخصات آن دانشجورا چاپ کنیم. نکته مهم در این روش این است که ما عمل خواندن نمرات دانشجویان را دوبار انجام دادیم و در بعضی از مسائلی که با این روش حل می شوند نیاز به این است که چندین بار این عمل انجام شود که در بعضی از حالت ها امکان دستیابی مجدد به داده ها یا امکان پذیر نیست و یا خیلی وقت گیراست. همچنین اگر تعداد دانشجویان زیاد باشد (مثلا در این مثال ۱۰۰۰ نفر باشد)، مانیاز داریم که هزار متغیر تعریف کنیم. که حداقل در این مورد غیر ممکن است پس باید به دنبال یک راه حل اساسی باشیم.

راه حل کلی برای اینگونه مسائل به این صورت است که مکانی را در حافظه ماشین الگوریتم، برای هر داده ای در نظر گرفته و اطلاعات را در آنجا ذخیره کرده و هر بار در صورت نیاز به آنها مراجعه نمائیم. شاید فکر کنید که به طریق اول نیز، همین کار را انجام می دادیم، اما اگر کمی صبر کنید، به شما خواهیم گفت که این روش ذخیره سازی کمی با روشهایی که قبل از اگر گرفته اید فرق می کند. در قسمتهای قبلی (فصل اول) اگر متغیری را با نام A تعریف می کردیم، می توانستیم فقط یک داده را در آن ذخیره کنیم اما در این قسمت از آرایه استفاده می کنیم (که در اصل یک آرایه یعنی همان یک متغیر) که نه تنها یک عدد بلکه مجموعه ای از اعداد را می توان در خانه های آن ذخیره کرد و همگی را با یک نام می خوانیم و وجه تمایزشان فقط در شماره خانه هایشان می باشد. (یعنی همان اندیسشان) در زیر تعریف دقیقی از آرایه را می آوریم.

تعریف آرایه یا لیست :

آرایه ، مجموعه ای متناهی از متغیرها است که همگی دارای یک نام بوده و هر متغیر دارای یک اندیس خاص می باشد.

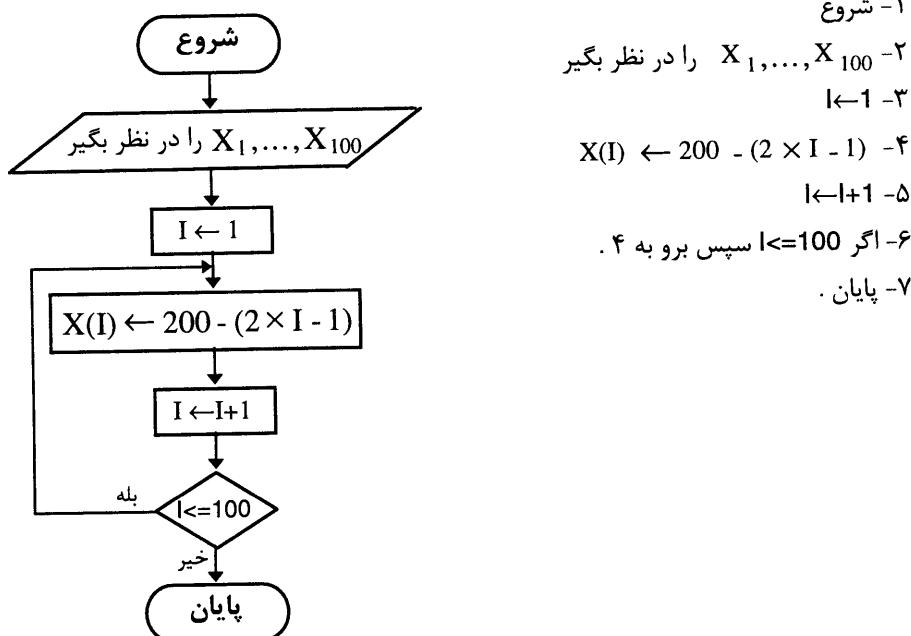
به عنوان مثال آرایه یا لیست B به تعداد یا طول ۴ را در نظر بگیرید :



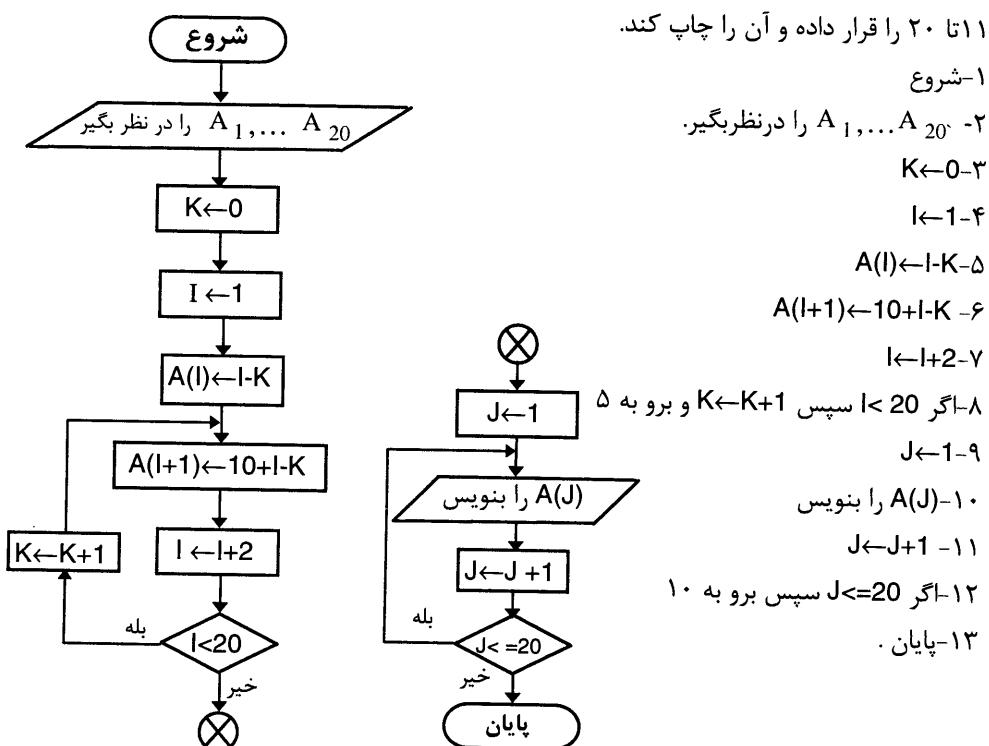
آرایه ها را می توان یک بعدی، دو بعدی و یا حتی چند بعدی نیز در نظر گرفت.

تذکر : منظور از آرایه های یک بعدی همان بردارهای با طول ثابت یا متغیرهای اندیس دار است که هر خانه با یک اندیس خاص مشخص می شود.

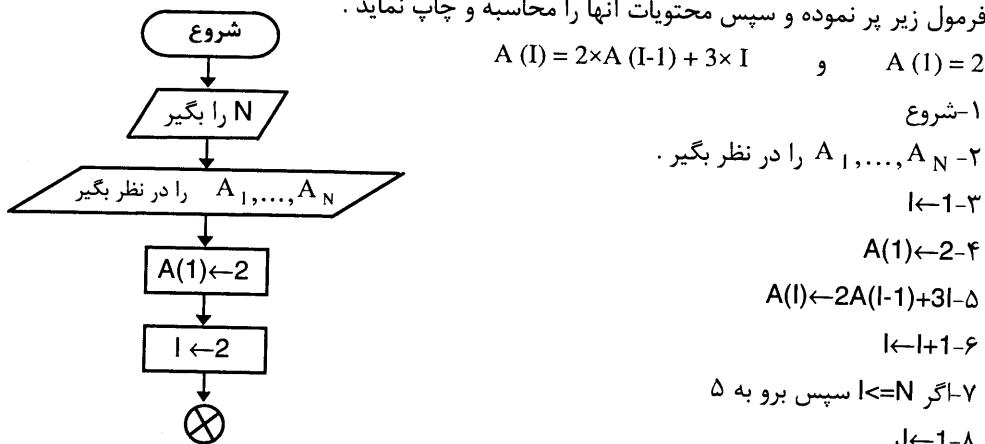
مثال (۱۱) الگوریتمی بنویسید که لیست X به طول ۱۰۰ را در نظر گرفته و اعداد فرد کوچکتر از ۲۰۰ را از آخر به ترتیب در این ۱۰۰ خانه ذخیره نماید.

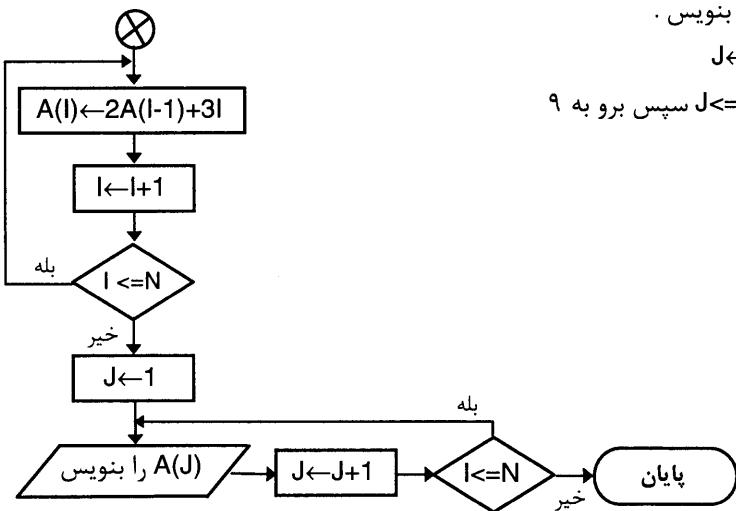


مثال ۱۲) فلوچارتی رسم کنید که یک لیست ۲۰ خانه‌ای در نظر گرفته و در خانه‌های اول و سوم و پنجم و نوزدهم آن اعداد از ۱ تا ۱۰ و در بقیه خانه‌های دوم و چهارم بیستم به ترتیب اعداد ۱۱ تا ۲۰ را قرار داده و آن را چاپ کند.

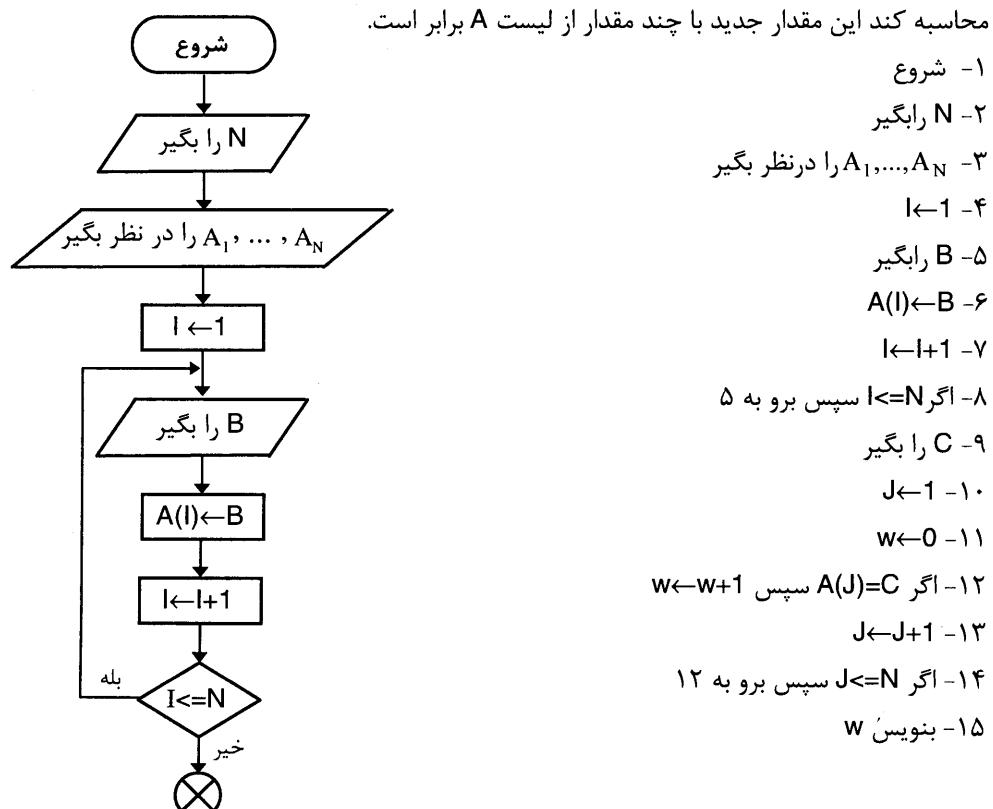


مثال ۱۳) الگوریتمی بنویسید که یک لیست N خانه‌ای را در نظر گرفته و خانه‌های آن را بر اساس فرمول زیر پر نموده و سپس محتويات آنها را محاسبه و چاپ نماید .





مثال ۱۴) الگوریتمی بنویسید که عدد طبیعی N را دریافت و سپس لیست (آرایه) یا بردار A به طول N را درنظر گرفته و N مقدار را سوال نموده و داخل آن ذخیره کند. سپس مقداری را سوال نموده و محاسبه کند این مقدار جدید با چند مقدار از لیست A برابر است.



۹- A(J) را بنویس .

۱۰- J ← J+1 .

۱۱- اگر J <= N سپس برو به ۹

۱۲- پایان .

۱- شروع

۲- رابگیر N

۳- A₁, ..., A_N را درنظر بگیر

۴- I ← 1

۵- رابگیر B

۶- Assign A(I) ← B

۷- I ← I+1

۸- اگر I <= N سپس برو به ۵

۹- رابگیر C

۱۰- J ← 1

۱۱- w ← 0

۱۲- اگر A(J)=C سپس برو به ۱۲

۱۳- J ← J+1

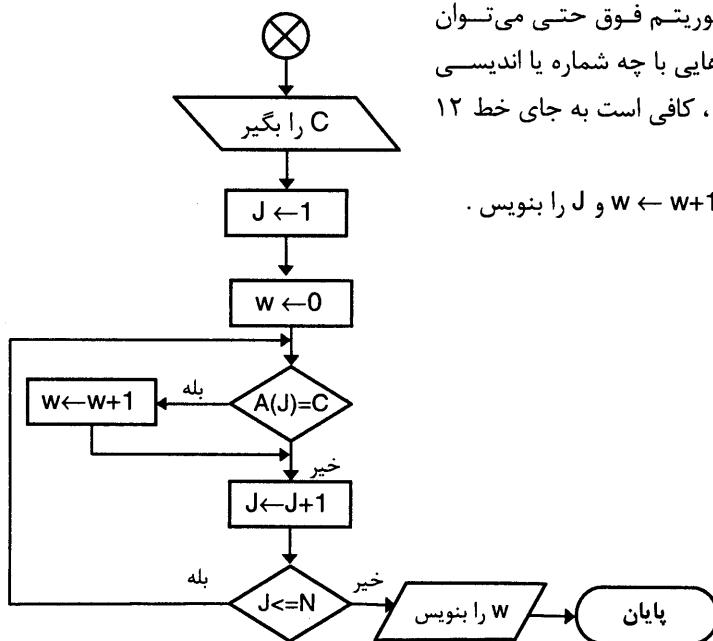
۱۴- اگر J <= N سپس برو به ۹

۱۵- بنویس w

۱۶- پایان

با تغییرات جزئی در الگوریتم فوق حتی می‌توان مشخص کرد که چه خانه‌هایی با چه شماره یا اندیسی با مقدار C برابر بوده است، کافی است به جای خط ۱۲ بنویسم:

۱۲- اگر $A(J)=C$ سپس $w \leftarrow w+1$ و J را بنویس.



مثال ۱۵) الگوریتمی بنویسید که یک آرایه یک بعدی به طول ۲۰ را در نظر گرفته و ۲۰ عدد را در آنها ذخیره نماید، سپس اعداد زوج و بعد اعداد مضرب ۳ را یکی چاپ نماید. فلوچارت آنرا به عنوان تمرین رسم کنید).

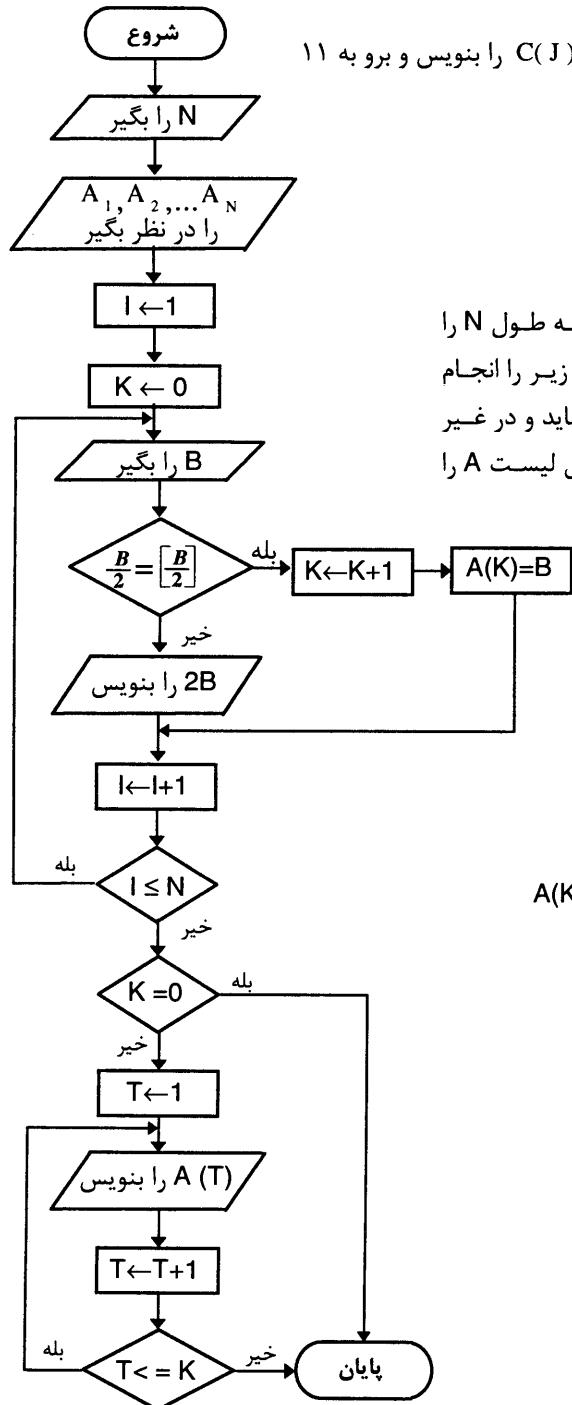
۱- شروع

۲- C_1, \dots, C_{20} را در نظر بگیر۳- $I \leftarrow 1$

۴- B را بگیر

۵- $C(I) \leftarrow B$ ۶- $I \leftarrow I+1$ ۷- اگر $I >= 20$ سپس برو به ۴۸- $J \leftarrow 1$

$$9- \text{اگر } \frac{C(J)}{2} = \left[\frac{C(J)}{2} \right] \text{ سپس } C(J) \text{ را بنویس و برو به ۱۱}$$



۱۰- اگر $\frac{C(J)}{3} = \left\lfloor \frac{C(J)}{3} \right\rfloor$

$J \leftarrow J + 1 - 11$

۱۲- اگر $J <= 20$ پس برو به ۹

۱۳- پایان

مثال ۱۶) الگوریتمی بنویسید که لیست A به طول N را در نظر گرفته و N عدد را خوانده و اعمال زیر را انجام دهد. اگر عدد زوج بود آن را ذخیره نماید و در غیر اینصورت دو برابر آن را چاپ نماید، سپس لیست A را چاپ نماید.

۱- شروع

۲- N را بگیر

۳- A₁, A₂, ..., A_N را در نظر بگیر

۴- I ← 1

۵- K ← 0

۶- را بخوان B

۷- اگر $A(K) = B$ و $K \leftarrow K + 1$ سپس $\frac{B}{2} = \left\lfloor \frac{B}{2} \right\rfloor$

۸- و برو به ۹

۹- ۲B را چاپ کن

۱۰- I ← I + 1

۱۱- اگر $I \leq N$ سپس برو به ۶

۱۲- اگر $K = 0$ سپس برو به ۱۶

۱۳- T ← 1

۱۴- A(T) را بنویس

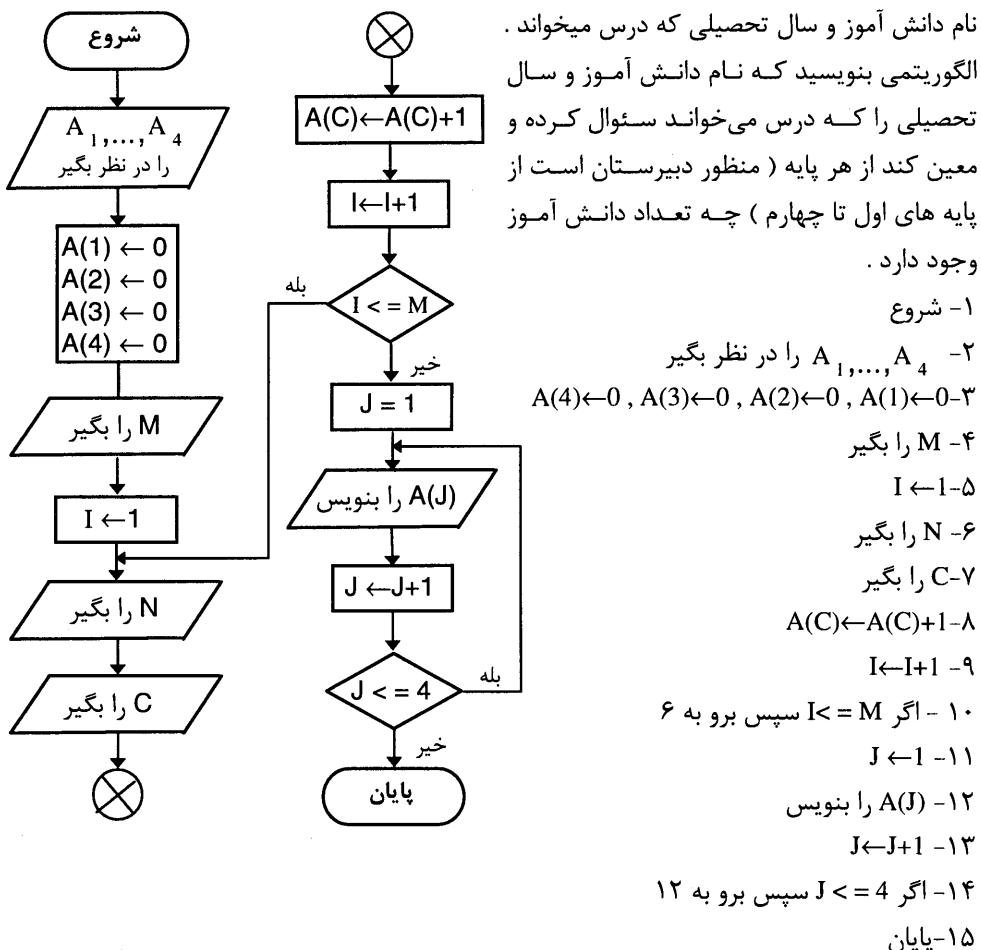
۱۵- T ← T + 1

۱۶- اگر $T \leq K$ سپس برو به ۱۳

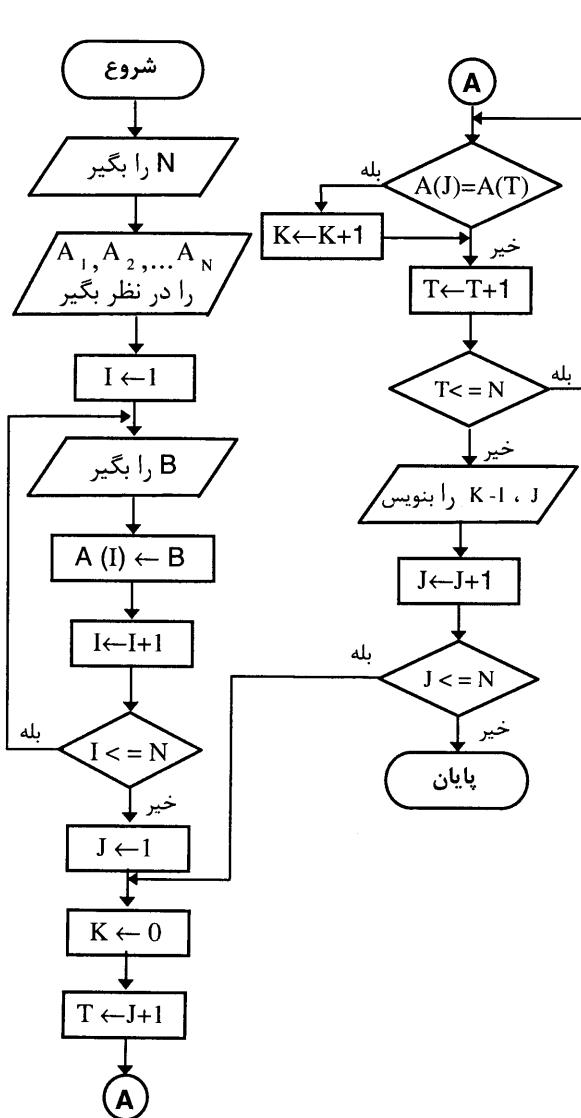
۱۷- پایان

روشی که در این مثال استفاده کردیم باروشهای قبلی اندکی تفاوت دارد . در روش‌های قبلی ابتدا N عدد را خوانده و آنها را در هرخانه ذخیره می‌کردیم ، سپس دوباره از اولین خانه شروع کرده و هر کدام که زوج بود ذخیره می‌شد و گرنه ۲ برابر آن خانه را چاپ می‌کردیم ، اما در این روش همزمان که عددی را می‌خوانیم اگر زوج باشد آنرا ذخیره می‌کنیم ، که برای اندیس خانه‌ها از متغیر K استفاده کردیم و متغیر A فقط برای خواندن اعداد بود. امکان دارد لیست ما 100×100 خانه داشته باشد یعنی از یک تا 100×100 تغییر کند و 100×100 عدد را بگیرد ولی تعداد خانه‌های پرشده 100×100 نباشد بلکه K باشد که از 1 شروع می‌شود. البته این برای حالتی است که لا اقل یک عدد زوج وارد کرده باشیم و یک خانه پر باشد . زیرا اگر عدد زوجی وارد نکنیم هیچ خانه‌ای پر نمی‌شود و الگوریتم خاتمه می‌پذیرد.

مثال ۱۷) اطلاعات زیر برای هر دانش آموز در نظر گرفته شده است :



در این مثال M تعداد دانش آموزان است و N نام دانش آموز و C پایه ایست که درس می خواند (عددی بین ۱ تا ۴ است). در اول الگوریتم مقادیر هریک از خانه های A_1 تا A_4 را که به ترتیب حرف پایه های اول تا چهارم هستند صفر در نظر گرفته شده است و بعد از پرسیده شدن نام و پایه هر دانش آموز با توجه به آنکه در چه پایه ای است به مقدار خانه مربوط به پایه کلی اضافه می شود.



مثال ۱۸) الگوریتمی بنویسید که لیست A را به طول N در نظر گرفته و N عدد را در آن ذخیره نماید، سپس عناصر تکراری در لیست A را در صورت وجود چاپ نماید.

-۱- شروع

-۲- N را بگیر

-۳- A_1, A_2, \dots, A_N را در نظر بگیر

-۴- $I \leftarrow 1$

-۵- B را بگیر

-۶- $A(I) \leftarrow B$

-۷- $I \leftarrow I + 1$

-۸- اگر $I \leq N$ سپس برو به ۵

-۹- $J \leftarrow 1$

-۱۰- $K \leftarrow 0$

-۱۱- $T \leftarrow J + 1$

-۱۲- اگر $A(J) = A(T)$ سپس $K \leftarrow K + 1$

-۱۳- $T \leftarrow T + 1$

-۱۴- اگر $T \leq N$ سپس برو به ۱۲

-۱۵- $K-1$ را بنویس و J را بنویس

-۱۶- $J \leftarrow J + 1$

-۱۷- اگر $J \leq N$ پس برو به ۱۰

-۱۸- پایان .

توضیح روند الگوریتم : ابتدا N عدد را به ترتیب در خانه‌های A_1, A_2, \dots, A_N ذخیره می‌کنیم ، سپس از اولین خانه شروع می‌کنیم تا آخرین خانه که برای انجام این عمل از شمارنده L استفاده می‌کنیم ، سپس یک شمارنده دیگر مانند K را از ابتدا صفر در نظر می‌گیریم و مقدار T را از مقدار یک شروع می‌کنیم و تا آخر لیست حرکت می‌کنیم . هر کجا که مقدار (J) با $(A(T))$ برابر بود یکی به شمارنده K اضافه می‌کنیم و (J) را با عنصر بعدی تست می‌کنیم و در نهایت که به آخر لیست برسیم مقدار J که اندیس خانه و K را که تعداد تکرار آن است چاپ می‌کنیم و این عمل را برای بقیه خانه‌ها انجام می‌دهیم .

A_1	A_2	A_3						A_{10}

به عنوان مثال : فرض کنیم ۱۰ خانه با نامهای A_1, \dots, A_{10} را در نظر گرفته‌ایم . و در درون آنها مقادیری را ذخیره کرده‌ایم، برای اینکه عناصر تکراری را پیدا کنیم از اولین خانه یعنی (1) $A(1)$ شروع کرده و آنرا با بقیه خانه‌های از (2) $A(2)$ تا (10) $A(10)$ تست می‌نماییم، با هر کدام که برابر بود اندیس ۱ (یعنی اولین خانه) و K یعنی تعداد خانه‌ایی که با (1) $A(1)$ برابر هستند را چاپ می‌کنیم، سپس خانه (2) $A(2)$ را در نظر گرفته و یک شمارنده را صفر در نظر گرفته و (2) $A(2)$ را با خانه‌های (1) $A(1)$ تا (10) $A(10)$ مقایسه می‌کنیم، هر کجا که برابر بود یکی به شمارنده اضافه می‌کنیم و در نهایت اندیس ۲ (یعنی دومین خانه) و K یعنی تعداد خانه‌هایی که با (2) $A(2)$ برابرند را چاپ می‌کنیم. توجه کنید که برای خانه (2) $A(2)$ از ابتدا یعنی (1) شروع می‌کنیم و آنرا با خانه‌های از (1) $A(1)$ تا (10) $A(10)$ مقایسه و هر کجا که برابر بودند به شمارنده K یکی اضافه می‌کنیم. اما چون خانه (2) $A(2)$ با (2) $A(2)$ برابر است لذا در نهایت مقدار K یکی اضافه‌تر است که برای چاپ مقدار $1-K$ را چاپ می‌کنیم. این عمل را برای بقیه خانه‌های (3) $A(3)$ تا (10) $A(10)$ تکرار می‌کنیم.

روند دیگری که برای الگوریتم وجود دارد آنست که هر خانه را با بقیه خانه‌ها تست نمائیم که اشکالاتی دارد. بعنوان مثال فرض می‌کنیم :

2	3	2	2
---	---	---	---

 را داریم با روندی که توضیح دادیم خروجی بصورت زیر خواهد بود :

خانه ۱ با ۲ خانه برابر است.

خانه ۲ با صفر خانه برابر است.

خانه ۳ با ۲ خانه برابر است.

خانه ۴ با ۲ خانه برابر است.

اما روند دیگر آن است که هر خانه را فقط با خانه‌های بعد از آن تست کنیم که خروجی به صورت زیر خواهد بود :

خانه ۱ با ۲ خانه برابر است.

خانه ۲ با صفر خانه برابر است.

خانه ۳ با ۱ خانه برابر است.

خانه ۴ با صفر خانه برابر است.

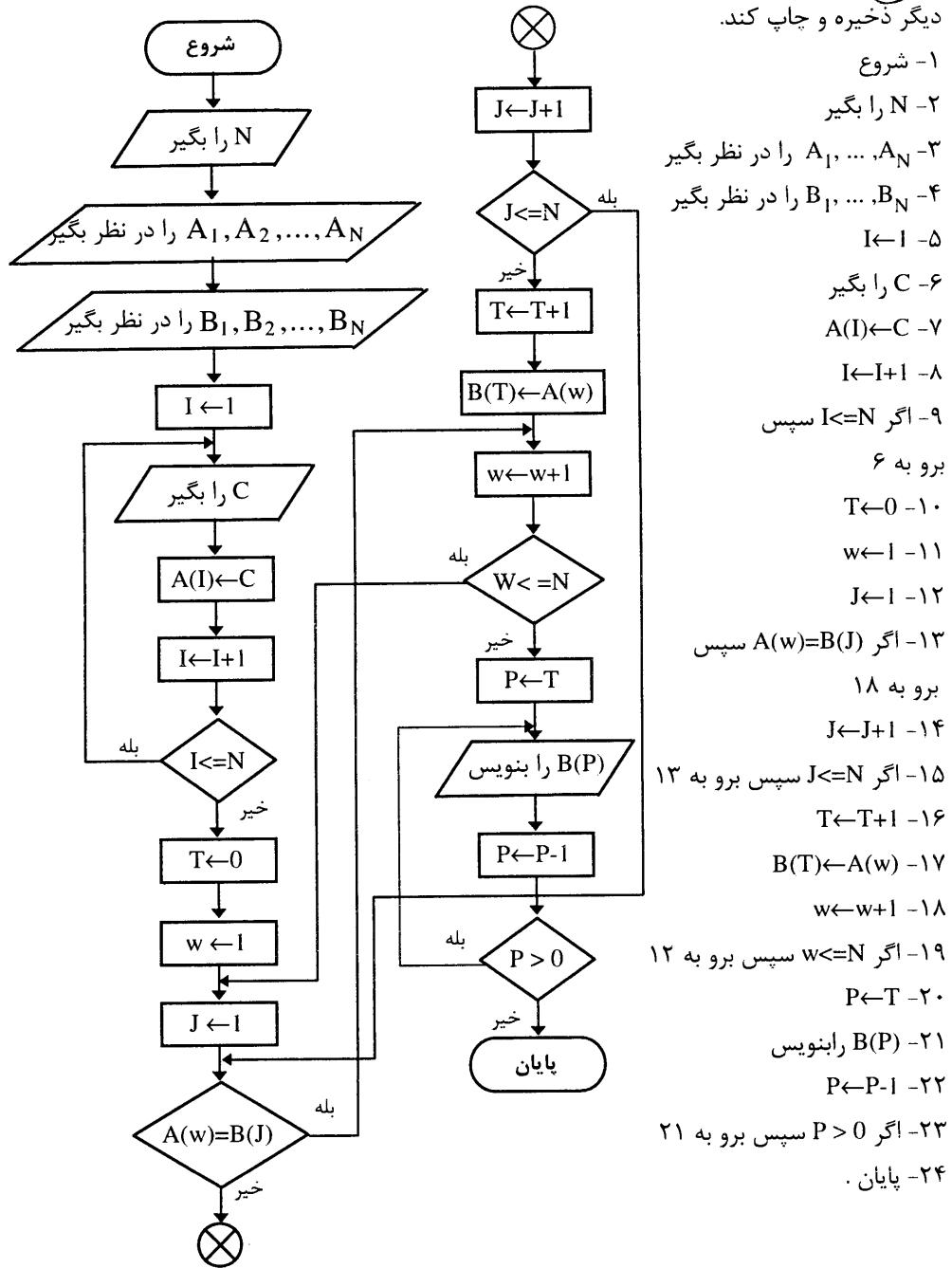
که خروجی صحیحی به ما نمی‌دهد.

برای اینکه از توضیحات اضافه کاسته باشیم الگوریتم را با خانه ۴ تست می‌نماییم.

N	I	B	A (I)	J	K	T	?
4	۱	2	A (1)=2				
	۲	3	A (2)=3				
	۳	4	A (3)=4				
	4	2	A 4)=2				
<hr/>							
				1	0	1	1,1
				1	2		
					3		
				2	4		
					5		
<hr/>							
				2	0	1	2,0
				1	2		
					3		
					4		
					5		
<hr/>							
				3	0	1	3,0
					2		
				1	3		
					4		
					5		
<hr/>							
				4	0	1	4,1
				1	2		
					3		
				2	4		
					5		
<hr/>							
				5			

(پایان اجراء)

مثال ۱۹) الگوریتم مثال قبل را طوری تغییردهید که عناصر تکراری را حذف نموده و آنها را در یک آرایه دیگر ذخیره و جاپ کند.



۱- شروع
۲- N را بگیر

۳- A_1, A_2, \dots, A_N را در نظر بگیر

۴- B_1, B_2, \dots, B_N را در نظر بگیر

۵- $I \leftarrow 1$

۶- C را بگیر

۷- $A(I) \leftarrow C$

۸- $I \leftarrow I + 1$

۹- اگر $I \leq N$ سپس

۱۰- برو به ۶

۱۱- $T \leftarrow 0$

۱۲- $w \leftarrow 1$

۱۳- $J \leftarrow 1$

۱۴- اگر $A(w) = B(J)$ سپس

۱۵- برو به ۱۸

۱۶- $J \leftarrow J + 1$

۱۷- اگر $J \leq N$ سپس برو به ۱۵

۱۸- $T \leftarrow T + 1$

۱۹- $B(T) \leftarrow A(w)$

۲۰- $w \leftarrow w + 1$

۲۱- اگر $w \leq N$ سپس برو به ۱۸

۲۲- $P \leftarrow P - 1$

۲۳- $B(P) \leftarrow B(P)$

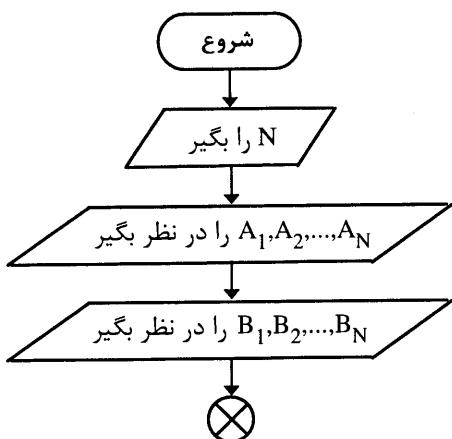
۲۴- $P \leftarrow P + 1$

۲۵- اگر $P > 0$ سپس برو به ۲۴

۲۶- پایان

این الگوریتم برای حالتی درست است که هیچیک از عناصر لیست A صفر نباشند . الگوریتم حالتی را که شامل صفر باشد، در مثال بعدی مطرح می‌کنیم . حال روند الگوریتم را توضیح می‌دهیم . در این روش هر یک از عناصر لیست A را با تمام عناصر آرایه B مقایسه می‌کنیم (که برای اولین بار مقدار هر یک از عناصر لیست B صفر است)، هر جا که برابر شدند، مقدار آن خانه را به لیست B اضافه می‌کنیم، برای اولین بار که تمام عناصر B صفر است ، فرض کنیم هیچ یک از عناصر A صفر نباشد، پس اولین مقدار A با هیچ یک از خانه های B برابر نمی‌شود پس آن مقدار را در اولین خانه B قرار می‌دهیم، سپس دومین عنصر A را با تک تک عناصر B مقایسه می‌کنیم، اگر برابر بود به سراغ سومین عنصر A رفته و اگر نبود آن را در دومین خانه B ذخیره می‌کنیم تا اینجا اندیس خانه هایی از A که باید در B قرار می‌گرفت برابر بودند، حال فرض کنیم سومین عنصر A با دومین عنصر برابر باشد یعنی تکراری باشد، پس در دومین خانه B قرار گرفته است و موقع مقایسه با آن برابر می‌شود و به سراغ عنصر بعدی می‌رویم یعنی چهارمین عنصر که فرض کنید تکراری نیست پس با هیچ یک از عناصر B برابر نمی‌شود، پس باید در یکی از خانه های B ذخیره شود . اما کدام خانه ؟ طبق روند توضیح داده شده، باید در خانه چهارم B قرار گیرد، اما خانه سوم چی ؟ (یعنی خالی باقی می‌ماند) . به همین دلیل یک مقدار مانند T را از اول صفر در نظرمی‌گیریم و هر بار که بخواهیم ، عنصری از A را در B ذخیره کنیم یک واحد به آن اضافه می‌کنیم ، که با این شمارنده وقتی چهارمین عنصر A تکراری نباشد، چون قبل ادو عنصر ذخیره شده است، پس مقدار T برابر 2 است و این بار نیز اگر بخواهیم، مقداری از A را در B ذخیره کنیم، یک واحد به T اضافه می‌کنیم یعنی ۳ ، سپس در سومین خانه آن را ذخیره می‌کنیم، با این روند هیچ جای خالی به وجود نمی‌آید .

کار شمارنده‌های w و J و بقیه متغیرها نیز واضح است، برای تمرین می‌توانید با یک لیست ۶ عنصری که ۳ عنصر تکراری دارد، آن را آزمایش کنید .



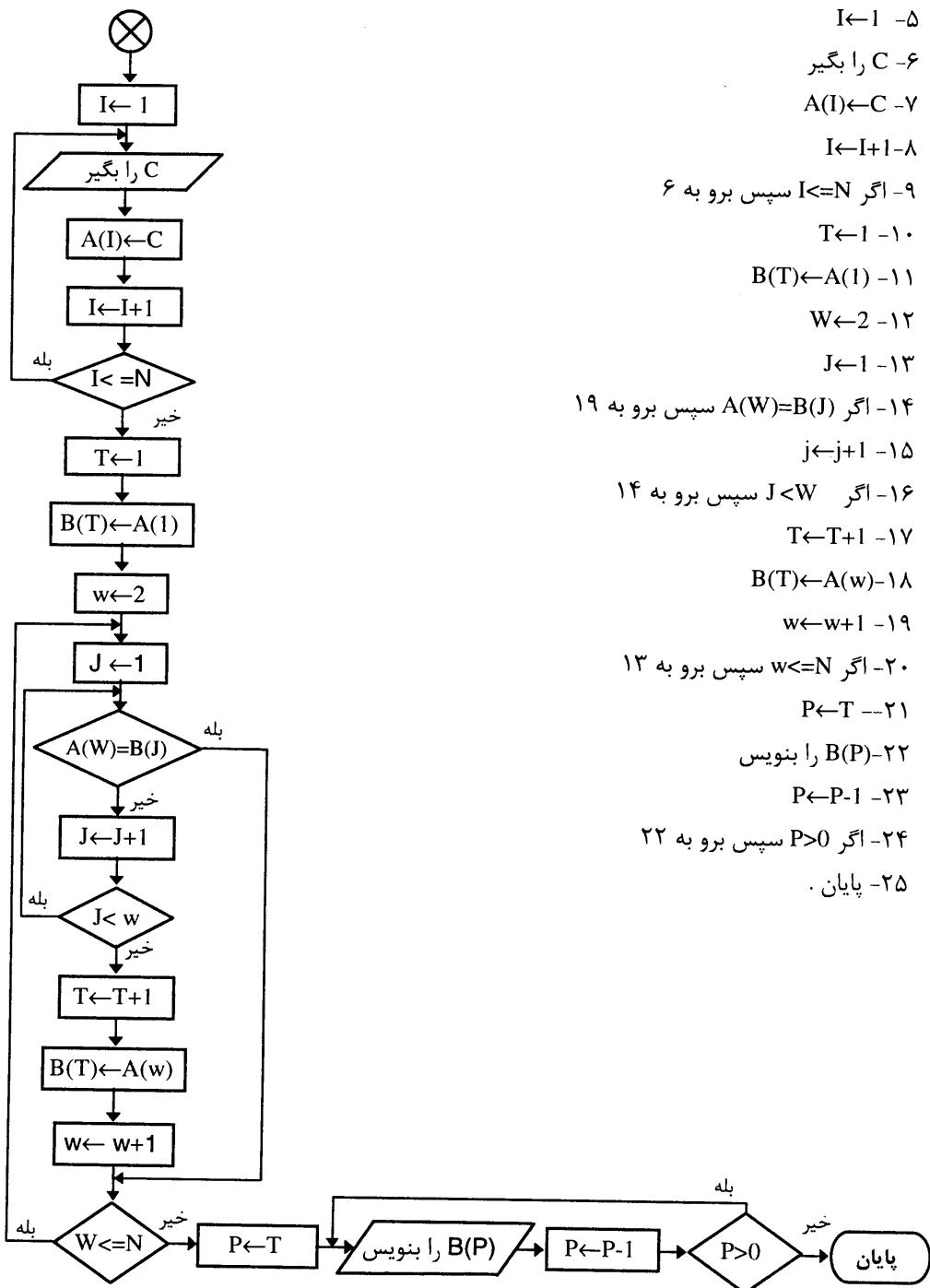
مثال ۲۰) الگوریتم مثال قبل را طوری تغییر دهید که اگر عناصر A صفر هم بودند ، عناصر تکراری را حذف و در یک آرایه دیگر ذخیره کند .

۱ - شروع

۲ - N را بگیر

۳ - A₁, A₂, ..., A_N را در نظر بگیر

۴ - B₁, B₂, ..., B_N را در نظر بگیر



مثال ۲۱) الگوریتمی بنویسد که یک آرایه N عنصری را در نظر گرفته و N عدد را در آن ذخیره کرده و بزرگترین عنصر را در آن مشخص و چاپ نماید، همچنین محل قرارگرفتن آن عنصر را نیز چاپ کند.

۱- شروع

۲- N را بگیر

۳- $X_1, X_2 \dots, X_N$ را در نظر بگیر

۴- $I \leftarrow 1$

۵- A را بگیر

۶- $X(I) \leftarrow A$

۷- $I \leftarrow I+1$

۸- اگر $I \leq N$ سپس برو به ۹

۹- $MAX \leftarrow X(1)$

۱۰- $J \leftarrow 2$

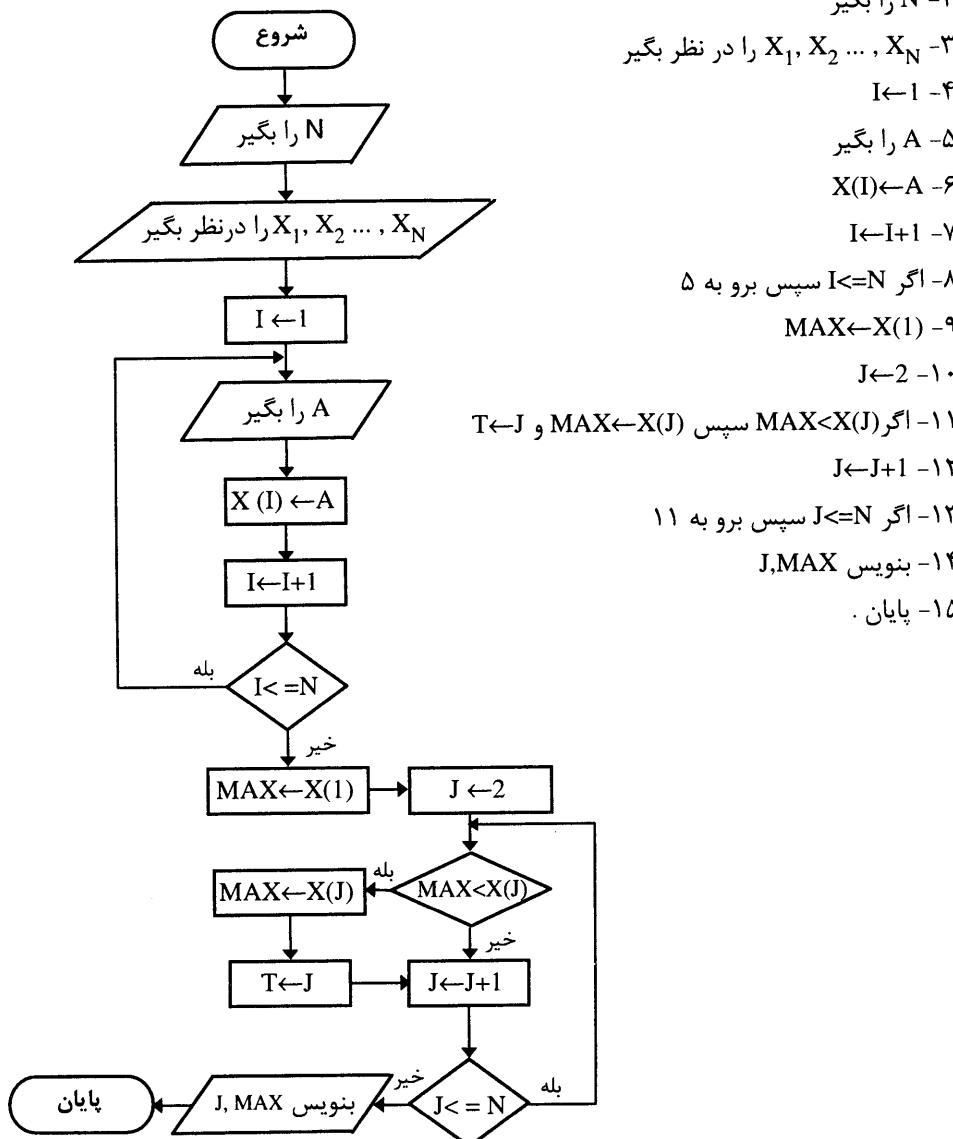
۱۱- اگر $MAX < X(J)$ سپس $MAX \leftarrow X(J)$ و $T \leftarrow J$

۱۲- $J \leftarrow J+1$

۱۳- اگر $J \leq N$ سپس برو به ۱۴

۱۴- بنویس J, MAX

۱۵- پایان .



مثال ۲۲) الگوریتمی بنویسید که یک لیست N عضوی را در نظر گرفته و N عدد را در آن ذخیره کرده و بزرگترین و کوچکترین عضو را مشخص و چاپ نماید.

۱- شروع.

۲- N را بگیر.

۳- A_N را در نظر بگیر.

$I \leftarrow I - 4$

۴- B را بگیر.

$A(I) \leftarrow B$

$I \leftarrow I + 1$

۵- اگر $I \leq N$ سپس برو به ۵

$MAX \leftarrow A(1)$

$MIN \leftarrow A(1)$

$J \leftarrow 2 - 11$

۶- اگر $A(J) > MAX$ سپس $A(J)$ را بنویس.

۷- اگر $A(J) < MIN$ سپس $A(J)$ را بنویس.

$J \leftarrow J + 1 - 14$

۸- اگر $J \leq N$ سپس برو به ۱۲.

۹- MAX را بنویس.

۱۰- MIN را بنویس.

۱۱- پایان

(فلوچارت الگوریتم فوق را به عنوان تمرین رسم کنید.)

مثال ۲۳) الگوریتمی بنویسید که یک آرایه N عنصری را در نظر گرفته و N عدد را در آن ذخیره نماید سپس یک عدد را دریافت و اگر این عدد با یکی از عناصر این آرایه برابر بود آن را حذف نماید.

۱- شروع

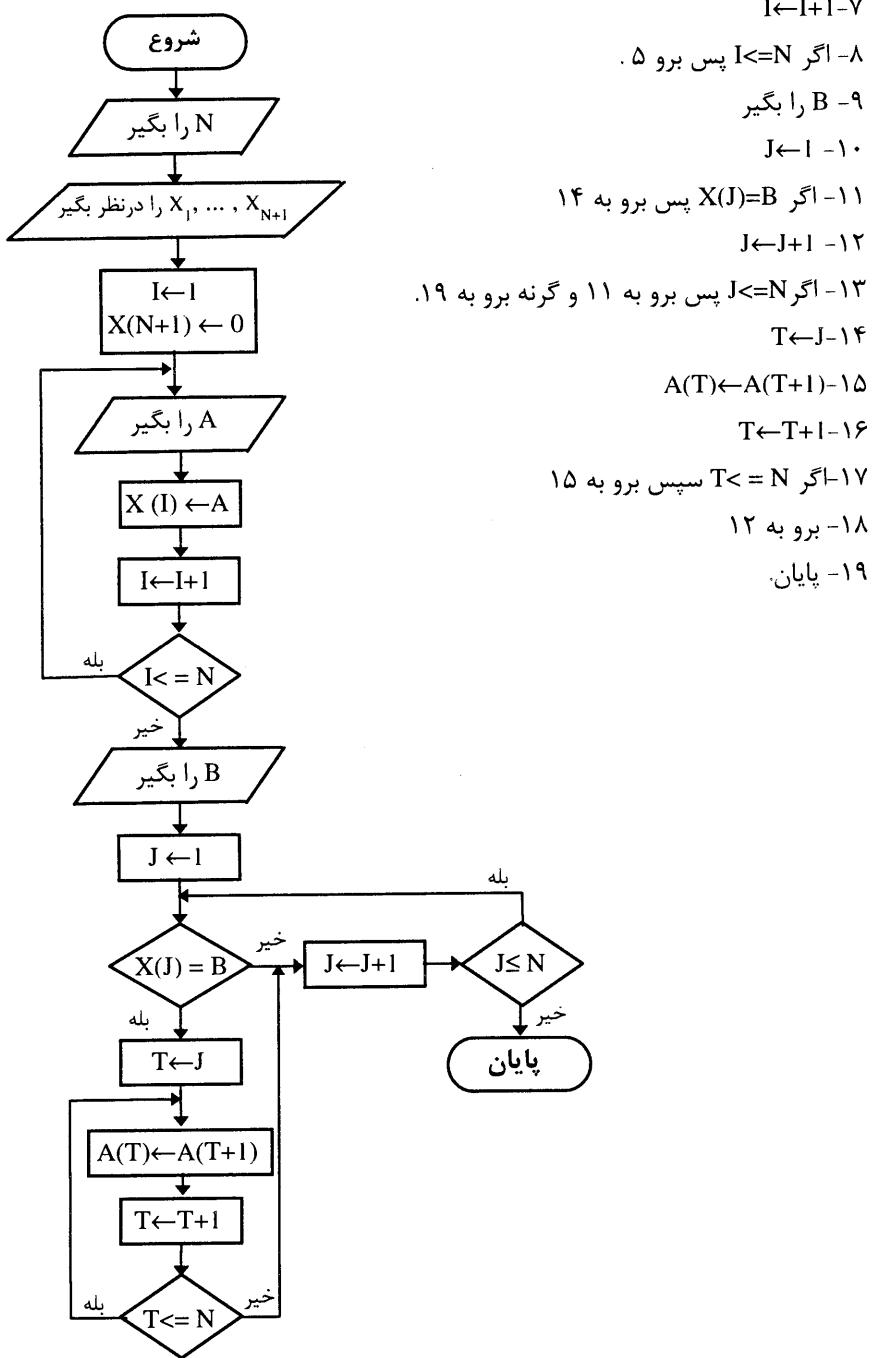
۲- N را بگیر

۳- X_1, \dots, X_{N+1} را در نظر بگیر

۴- $X(N+1) \leftarrow 0 : I \leftarrow 1$

۵- A را بگیر

۶- $X(I) \leftarrow A$

۸- اگر $I \leq N$ پس برو ۵ .

۹- B را بگیر

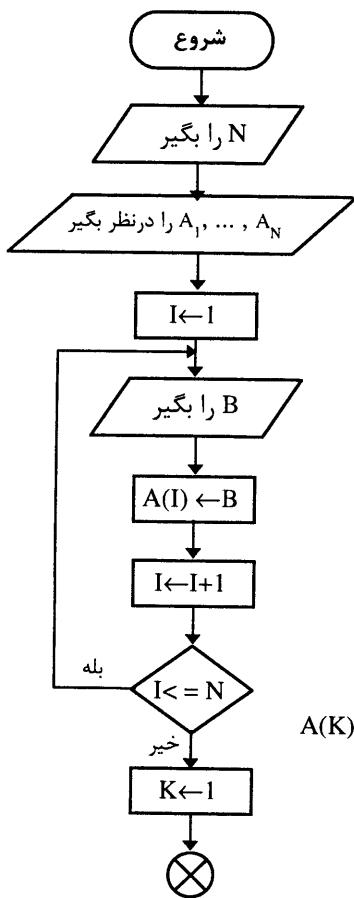
۱۰- $J \leftarrow J + 1$ ۱۱- اگر $X(J) = B$ پس برو به ۱۴۱۲- $J \leftarrow J + 1$ ۱۳- اگر $J \leq N$ پس برو به ۱۱ و گرنه برو به ۱۹ .۱۴- $T \leftarrow J - 1$ ۱۵- $A(T) \leftarrow A(T + 1)$ ۱۶- $T \leftarrow T + 1$ ۱۷- اگر $T \leq N$ سپس برو به ۱۵

۱۸- برو به ۱۲

۱۹- پایان .

توضیح: در این روش اگر عنصر وارد شده با یکی از عناصر لیست برابر باشد و بخواهیم آنرا حذف کنیم، یک جای خالی ایجاد می‌شود، به همین دلیل عناصر را یکی یکی به عقب انتقال می‌دهیم. مثلاً اگر لیست ۱۰۰ عنصری باشد و عدد وارد شده با عنصر پنجم برابر باشد، پنجمین عنصر حذف می‌شود و ششمی را در پنجمی قرار می‌دهیم و هفتمی را در ششمی و این عمل را تا آخرین عنصر لیست انجام می‌دهیم. با تغییرات جزئی در الگوریتم مثال قبل می‌توانیم کوچکترین (MIN) عدد را در یک رشته از اعداد پیدا کنیم. همچنین این الگوریتم برای زمانی است که عناصر تکراری در لیست پشت سر هم قرار نگرفته باشند، که در این صورت باید تغییرات جزئی در آن اعمال کنیم (به عنوان تمرین روی الگوریتم آن فکر کنید).

مثال (۲۴) الگوریتمی بنویسید که عدد N را سوال کرده و لیست N عنصری A را به همراه N عدد



در نظر گرفته و آنها را به ترتیب صعودی مرتب کند.

روش ساده‌ای را با استفاده از یک آرایه برای مرتب کردن یک رشته از اعداد در این مثال مطرح می‌کنیم. (روشهای کامل آنرا در فصل مربوط به جستجو و مرتب‌سازی مشاهده خواهید کرد)

۱- شروع

۲- N را بگیر

۳- A₁, ..., A_N را درنظر بگیر

۴- I ← I + 1

۵- B را بگیر

۶- A(I) ← B

۷- I ← I + 1

۸- اگر I <= N سپس برو به ۵

۹- K ← K + 1

۱۰- J ← K + 1

۱۱- اگر A(K) < T و A(J) < A(K) و T < A(J) سپس (A(K) > A(J))

۱۲- J ← J + 1

۱۳- اگر J <= N سپس برو به ۱۱

۱۴- K ← K + 1

۱۵- اگر K < N سپس برو به ۱۰

P←1-۱۶

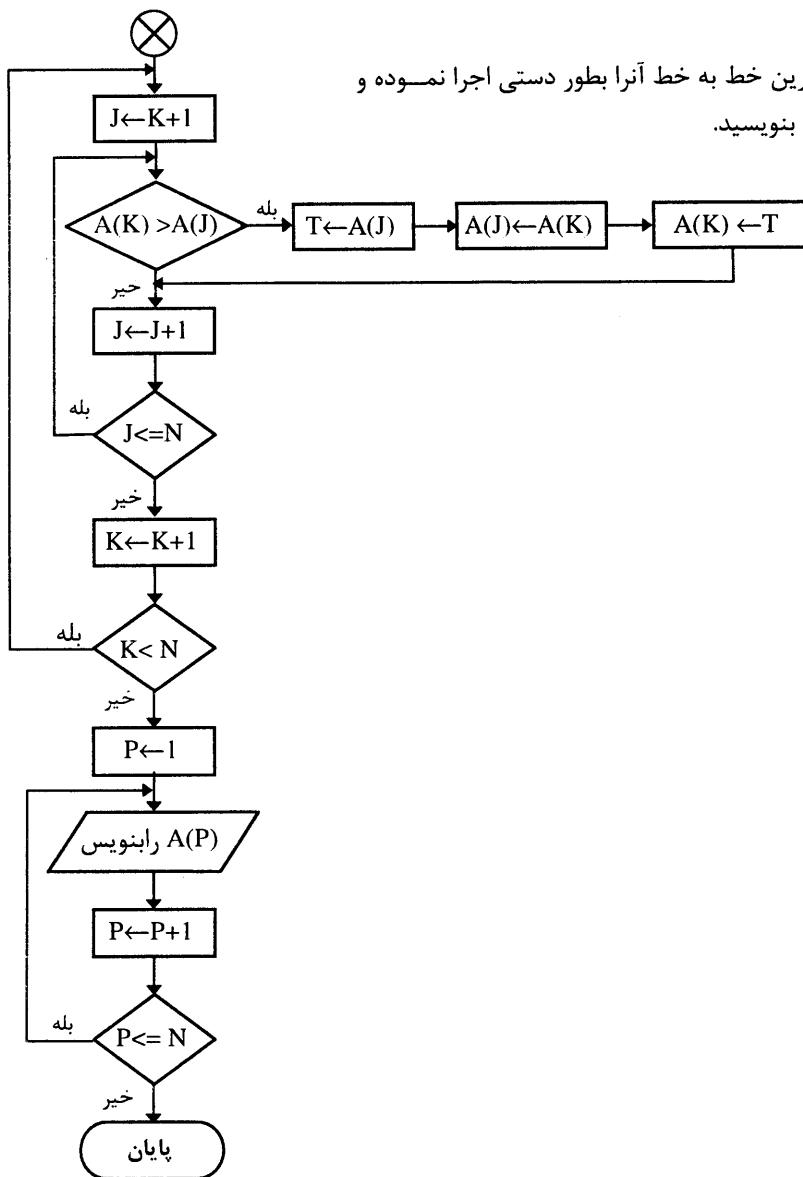
۱۷- A(P) را بنویس

P←P+1-۱۸

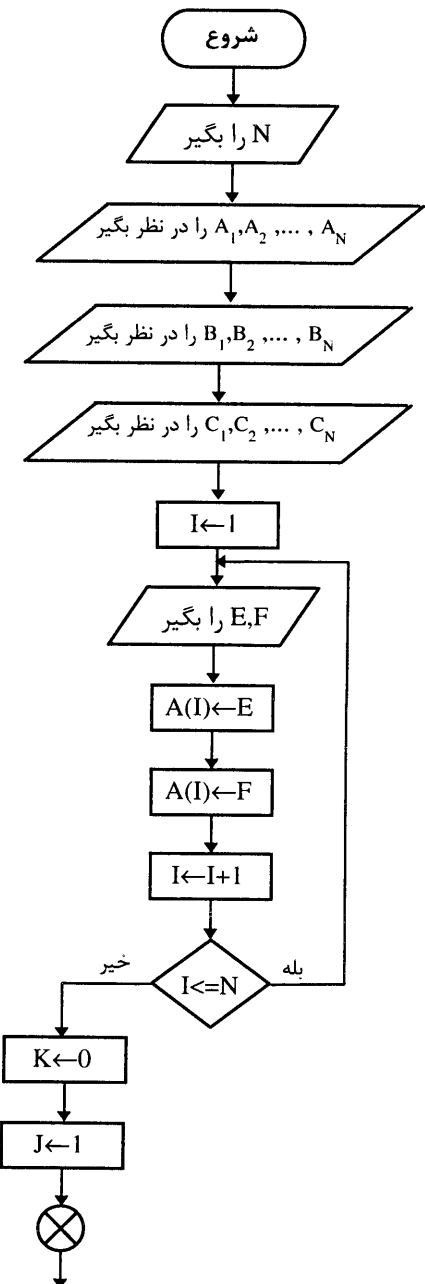
۱۹- اگر $p \leq N$ پس برو به

۲۰- پایان

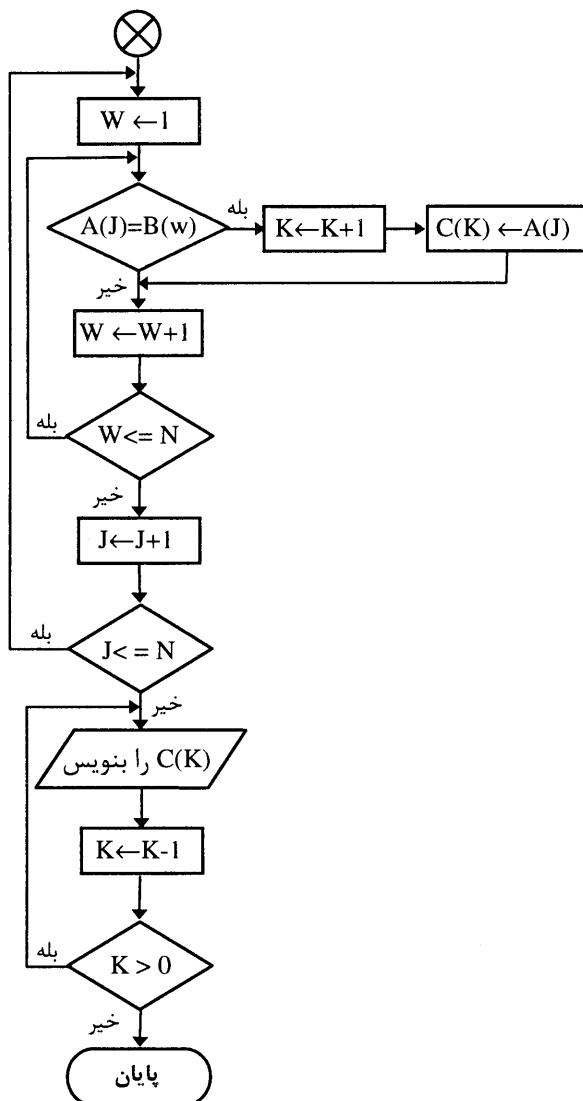
به عنوان تمرین خط به خط آنرا بطور دستی اجرا نموده و حاصل آن را بنویسید.



مثال ۲۵) الگوریتمی بنویسید که دو آرایه N عنصری با N عدد را درنظر گرفته و عضوهای مشترک دوآرایه را دریک آرایه ذخیره و عناصر آرایه حاصل را چاپ کند (فرض کردہایم دوآرایه فاقد عناصر تکراری هستند).

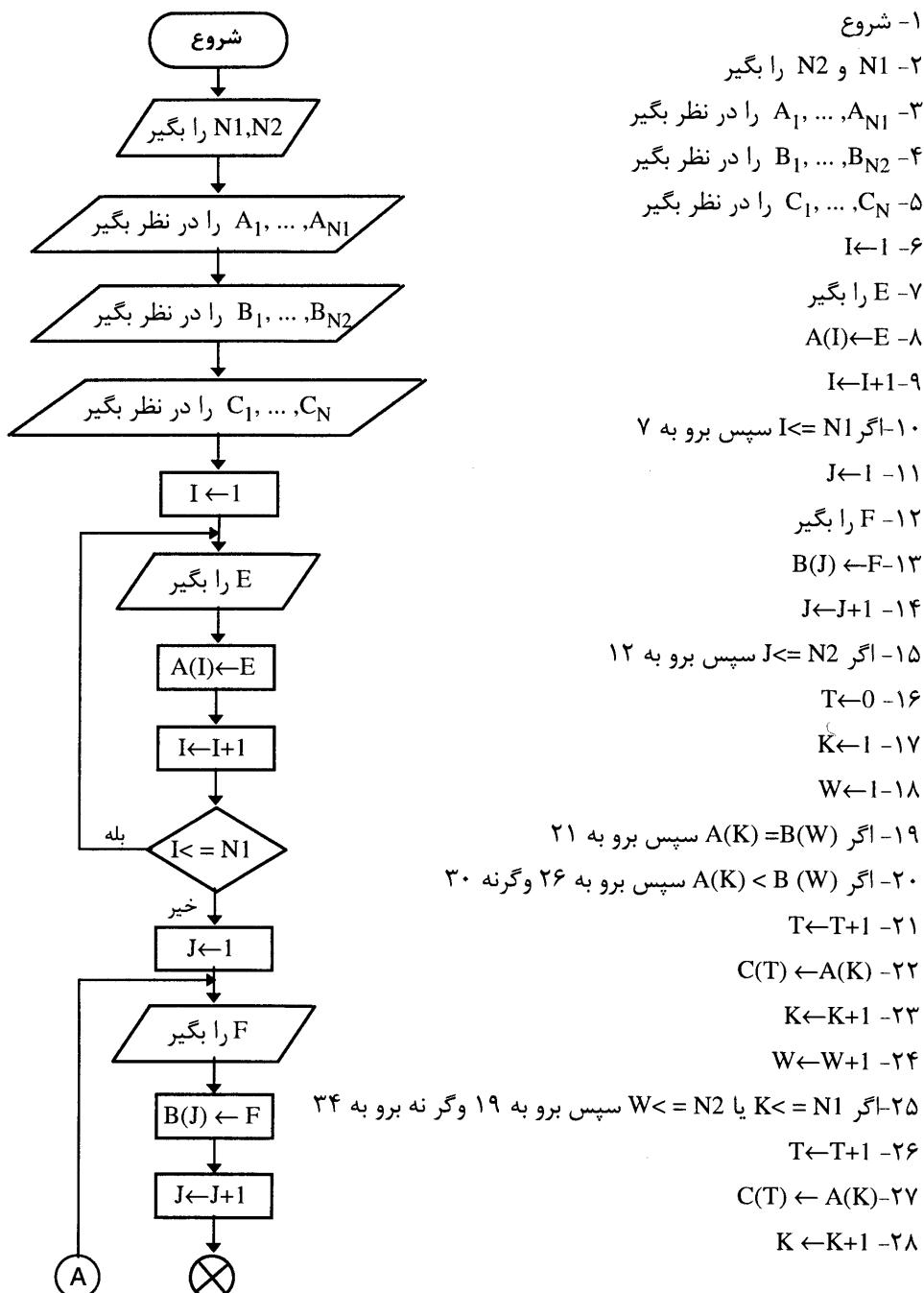


- ۱- شروع
- ۲- N را بگیر
- ۳- A_1, A_2, \dots, A_N را در نظر بگیر
- ۴- B_1, B_2, \dots, B_N را در نظر بگیر
- ۵- C_1, C_2, \dots, C_N را در نظر بگیر
- ۶- $I \leftarrow 1$
- ۷- $F \leftarrow E$ و R را بگیر
- ۸- $A(I) \leftarrow E$
- ۹- $B(I) \leftarrow F$
- ۱۰- $I \leftarrow I + 1$
- ۱۱- اگر $I \leq N$ سپس برو به ۷
- ۱۲- $k \leftarrow 0$
- ۱۳- $J \leftarrow 1$
- ۱۴- $w \leftarrow 1$
- ۱۵- اگر $A(J) = B(w)$ و $C(K) \leftarrow A(J)$ پس برو به ۱۵
- ۱۶- $w \leftarrow w + 1$
- ۱۷- اگر $w \leq N$ پس برو به ۱۵
- ۱۸- $J \leftarrow J + 1$
- ۱۹- اگر $J \leq N$ سپس برو به ۱۴
- ۲۰- $C(K) \leftarrow R$ را بنویس
- ۲۱- $K \leftarrow K - 1$
- ۲۲- اگر $K > 0$ سپس برو به ۲۰
- ۲۳- پایان .



مثال ۲۶) الگوریتمی بنویسید که آرایه A با N_1 عضو و آرایه B با N_2 عضو را در نظر گرفته و $N_1 + N_2$ عدد را در آرایه C ذخیره کند. سپس آرایه ای بنام C با $N_1 + N_2$ عضو را در نظر گرفته و عناصر A و B را در هم ادغام نموده و آنها را در آرایه C ذخیره و سپس حاصل را چاپ کند.

توضیح: اولاً دو آرایه A و B مرتب هستند و ثانیاً آرایه حاصل یعنی C نیز باید مرتب باشد (البته به ترتیب صعودی).



۲۹- اگر $K < N1$ سپس برو به ۱۹ و گر نه برو به ۳۴

$T \leftarrow T+1$ - ۳۰

$C(T) \leftarrow B(W)$ - ۳۱

$W \leftarrow W + 1$ - ۳۲

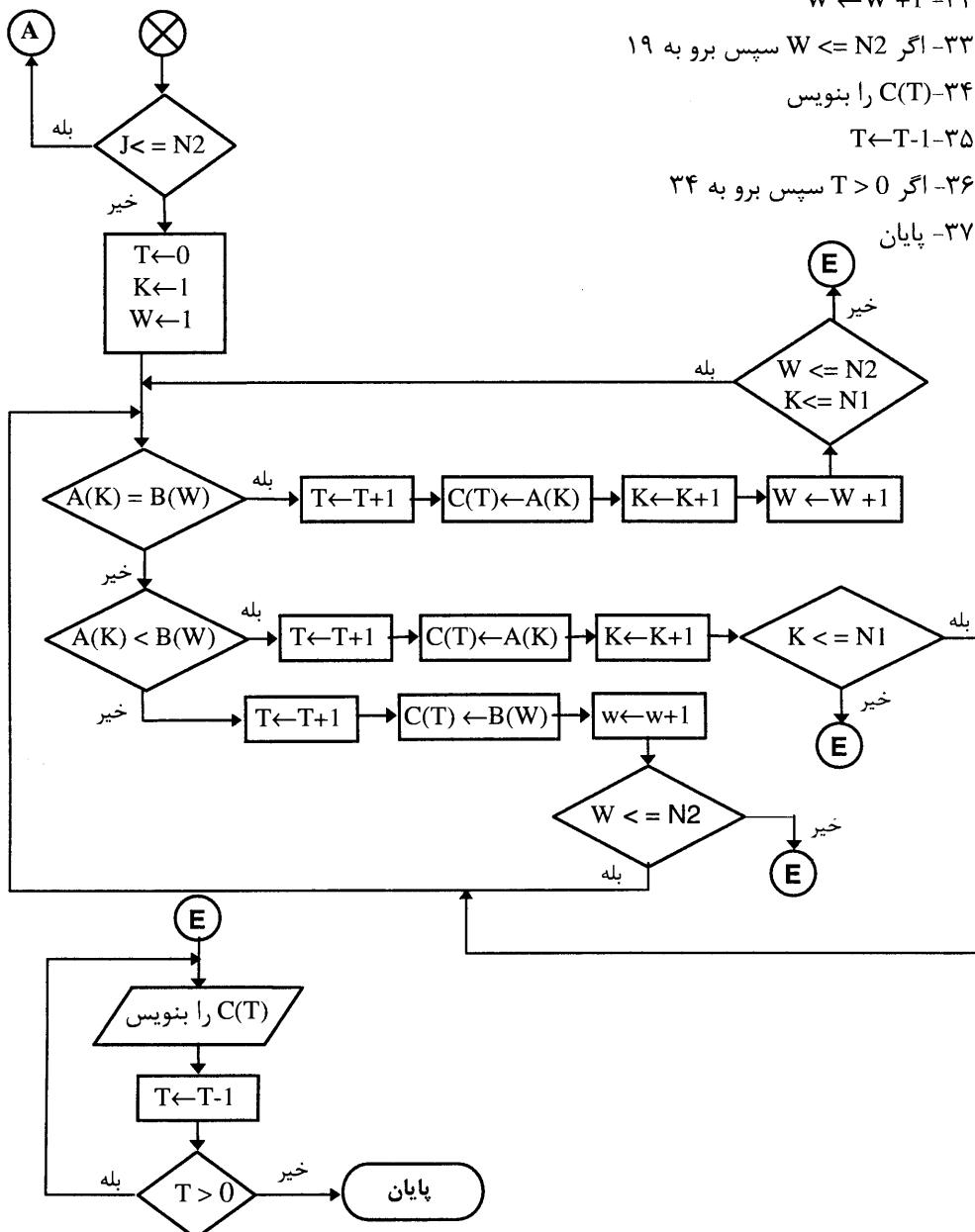
۳۳- اگر $W \leq N2$ سپس برو به ۱۹

را بنویس $C(T)$ - ۳۴

$T \leftarrow T-1$ - ۳۵

۳۶- اگر $T > 0$ سپس برو به ۳۴

پایان - ۳۷



روش مورد استفاده در این مساله عبارتست از اینکه در ابتدا عنصری از لیست A را با عنصری از لیست B مقایسه می‌کند. هر کدام که کوچکتر بودند آن را در لیست C ذخیره و به اندیس آن لیستی که عنصر کوچکتر به آن تعلق داشت، یک واحد اضافه می‌کند و این عمل را تا پایان یافتن یکی از دو لیست انجام می‌دهد و اگر مساوی بودند، یکی از دو عنصر را در C ذخیره نموده و به اندیس هر دو لیست یک واحد اضافه می‌کند و اگر به انتهای یکی از دو لیست رسیده بود، عمل مقایسه را خاتمه داده و لیست حاصل را چاپ می‌کند، در غیر این صورت دوباره عمل مقایسه را انجام می‌دهد. مثال ۲۷) الگوریتمی بنویسید که کلیه حالاتی را که سه حرف X, Y, Z می‌توانند کنار همیگر قرار گیرند را محاسبه و چاپ نماید.

۱- شروع

A₁, A₂, A₃ را در نظر بگیر

A(1) ← "X" -۳

A(2) ← "Y" -۴

A(3) ← "Z" -۵

I ← ۱ -۶

J ← ۱ -۷

-۸- اگر J=I سپس برو به ۱۳

K ← ۱ -۹

-۱۰- اگر I=K یا J=K سپس برو به ۱۲

-۱۱- بنویس A(I) و A(J) و A(K)

K ← K+1 -۱۲

-۱۳- اگر K<=3 سپس برو به ۱۰

J ← J+1 -۱۴

-۱۵- اگر 3<=J<=3 سپس برو به ۸

I ← I+1 -۱۶

-۱۷- اگر 3<=I<=3 سپس برو به ۷

-۱۸- پایان

مثال ۲۸) الگوریتمی بنویسید که دو لیست A و B را که ۱۰۰ عضوی هستند در نظر گرفته و در لیست A اسم ۱۰۰ دانش آموز و در لیست B معدل سال قبل دانش آموزان را قرار دهد، سپس یک اسم را گرفته و اگر در بین ۱۰۰ اسم بود معدل سال قبل وی را چاپ نماید و در غیر این صورت بنویسد موجود نیست و این عمل را برای ۱۰۰ داش آموز انجام دهد.

۱- شروع

۲- A_1, A_2, \dots, A_{100} را در نظر بگیر۳- B_1, B_2, \dots, B_{100} را در نظر بگیر۴- $I \leftarrow 1$

۵- NAME را بگیر و MOADEL

۶- $A(I) \leftarrow NAME$ ۷- $B(I) \leftarrow MOADEL$ ۸- $I \leftarrow I+1$ ۹- اگر $I \leq 100$ سپس برو به ۵

۱۰- C را بگیر

۱۱- $J \leftarrow 1$ ۱۲- $K \leftarrow 1$ ۱۳- اگر $A(K) = C$ سپس (K)B را بنویس و برو به ۱۷۱۴- $K \leftarrow K+1$ ۱۵- اگر $K \leq 100$ سپس برو به ۱۳

۱۶- بنویس پیدا نشد

۱۷- $J \leftarrow J+1$ ۱۸- اگر $J \leq N$ سپس برو به ۱۲

۱۹- پایان.

در مثال های قبلی، همواره از لیست های یک بعدی استفاده کردید یعنی همان متغیرهای اندیس دار یک بعدی که هر خانه ای دارای یک اندیس بود و بوسیله آن شناخته می شد، اما مسائلی هم وجود دارند که تنها بوسیله لیست های یک بعدی یا متغیرهای اندیس دار یک بعدی قابل حل نیستند. در این گونه مسائل از مفهوم دیگری به نام لیست دو بعدی یا ماتریس یا متغیر اندیس دار دو بعدی استفاده می شود که قبلاً از تعریف دقیق آن ابتدا به حل مثالی می پردازیم تا لزوم استفاده از چنین مفهومی برایتان روش شود.

مثال ۲۹) فرض کنید ۱۰ دانش آموز داریم که هر کدام در ۵ درس مختلف امتحان داده و نمره گرفته - اند، الگوریتمی بنویسید که معدل هر دانش آموز را جداگانه محاسبه و چاپ نماید.

روش اول :

یک لیست ۵ عنصری در نظر گرفته و نمرات اولین دانش آموز را در داخل آن قرار داده و

سپس معدل وی را حساب کرده و بعد آنرا چاپ کنید. سپس همین عمل را برای بقیه دانش آموزان تکرار کنید (بدون معرفی لیست جدید).

روش دوم:

۱۰ لیست ۵ عنصری مختلف معرفی کرده و نمرات هر دانش آموزی را در یکی از این لیست‌ها ذخیره و سپس معدل هریک را محاسبه و چاپ کنید.

سوال: فرض کنید می‌خواهیم بدانیم که بالاترین نمره درس از بین این ده دانش آموز متعلق به کدام یک بوده است. آیا با هریک از این روش‌ها می‌توانیم مساله را حل کنیم؟ مسلماً جواب منفی است. مگر اینکه یک جدول در نظر بگیرید که دارای پنج ستون و ده سطر باشد که در سطر اول نمرات درس اول تا پنجم دانش آموز اول را ذخیره کنید و الى آخر، که در این صورت جدول به شکل زیر خواهد بود:

					سطر ۱
					سطر ۲
					.
					.
					.
					.
					.
					.
					.
					.
					سطر ۱۰
↓	↓	ستون ۱	ستون ۵

فرض کنید اسمی ۵ درس عبارتند از: جبر، هندسه، فیزیک، شیمی و کامپیوتر. پس در سطر اول نمرات مربوط به درس جبر، هندسه، فیزیک، شیمی و کامپیوتر نفر اول ودر بقیه سطرها به

همین ترتیب نمرات افراد بعدی ذخیره شده‌اند. نکته مهم این است که در ستون اول نمرات جبر همه دانش آموزان وجود دارد، که به راحتی می‌توان ببروی آنها هر نوع عملی را انجام داد. پس مشاهده کردید که با استفاده از یک جدول می‌توان مساله را حل کرد (در این جدول هر خانه با دو عدد مشخص می‌شود که عدد اول نمایانگر سطر و عدد دوم نماینگر ستون آن خانه است).

تعریف ماتریس : یک جدول مستطیل شکل از اعداد را ماتریس می‌نامند. هر ماتریس دارای تعدادی سطر و تعدادی ستون است و همواره بصورت سطر در ستون خوانده می‌شود.

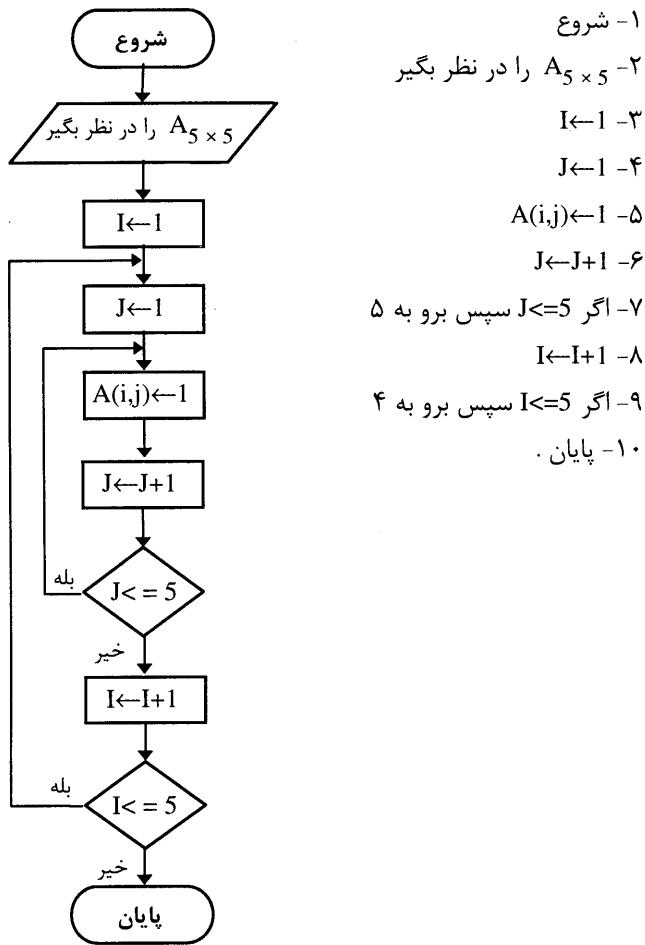
از برخورد سطر ها و ستونهای مختلف خانه‌های مختلفی بوجود می‌آید. پس هر خانه را به جای یک عدد با دو عدد می‌شناسیم که همان اندیس‌های آن خانه هستند.

یک ماتریس را همواره بصورت زیر نمایش می‌دهند. این ماتریس دارای n سطر و m ستون است که بصورت $[A]_{m \times n}$ نوشته و ماتریس m در n خوانده می‌شود.

$$A = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & \dots & \cdot \\ \cdot & \cdot & a_{ij} & \cdot \\ \cdot & \cdot & & \cdot \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}_{m \times n}$$

چنانچه مشاهده می‌کنید هر سطر ماتریس یا هر ستون آن در اصل یک لیست یک بعدی یا یک متغیر اندیس داریک بُعدی است. پس هر ماتریس متشکل از تعدادی متغیر اندیس‌دار یک بُعدی است. اگر ماتریس را به فرم ریاضی آن در نظر بگیریم (یعنی همواره جدولی از اعداد) می‌توانیم ببروی آنها برخی اعمال ریاضی از قبیل جمع، ضرب، تفریق، وارون و ... را انجام دهیم. در ماتریس بالا که با A نامگذاری شده است ، a_{ij} یعنی عنصری که محل آن در سطر i و ستون j است .

مثال (۳۲) الگوریتمی بنویسید که آرایه دو بعدی (لیست دو بعدی) A را با ۵ سطر و ۵ ستون در نظر گرفته و در تمام خانه‌های آن عدد ۱ را قرار دهد.

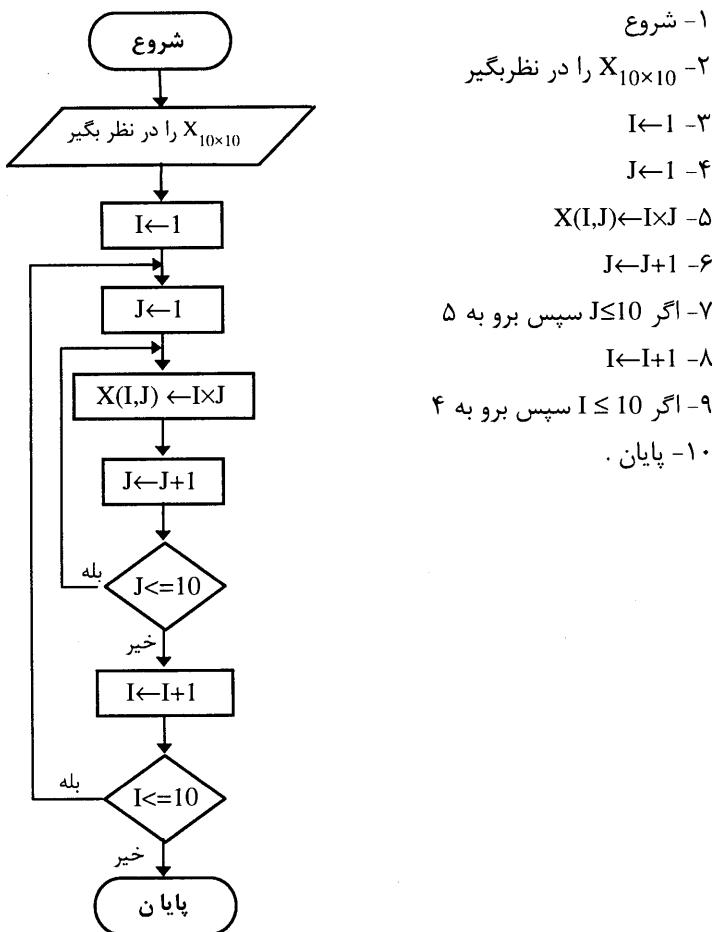


در الگوریتم نویسی برای نمایش اینکه ماتریس A ، m در n است از نماد $A_{m \times n}$ یا $A(M,N)$ استفاده می کنیم و در فلوچارت نیز آن را داخل متوازی الاضلاع قرار می دهیم. در این مثال دو حلقه داریم ، یکی I و دیگری J . حلقه J برای هر I از یک تا ۵ تغییر می کند و عناصر هر سطر را می سازد مثلا برای $I=2$ داریم :

$$J=1,2,3,4,5$$

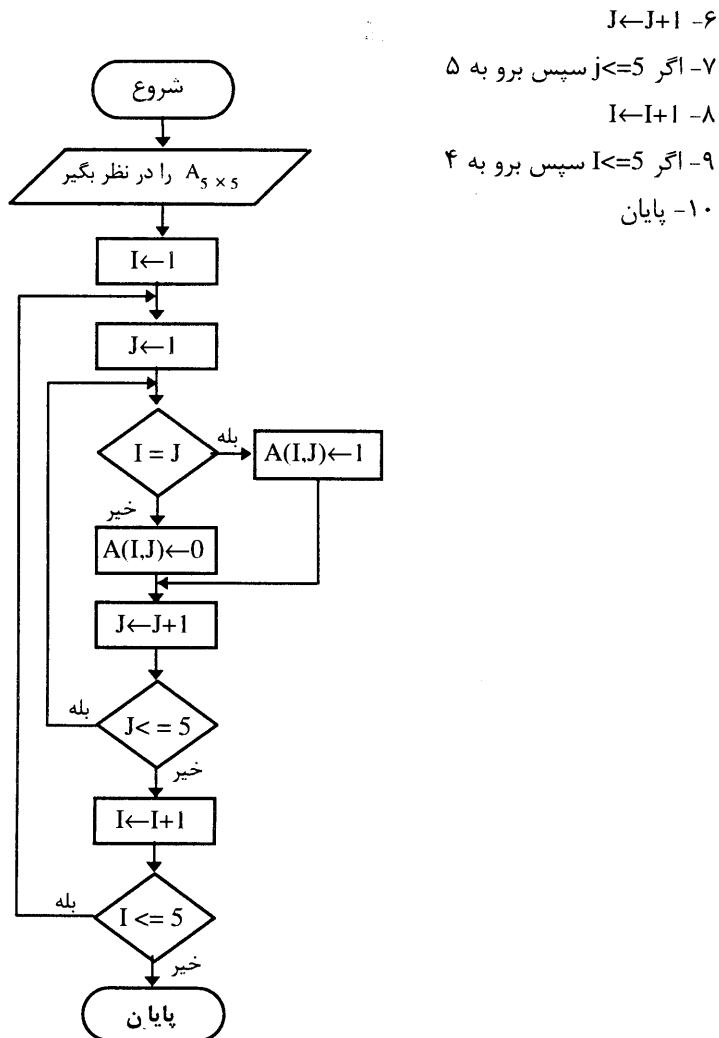
و عناصر آن به ترتیب $(A(2,1) , A(2,2) , A(2,3) , A(2,4) , A(2,5))$ مشخص می شوند که سطر دوم را بطور کامل می سازند. ماتریسهایی که تعداد سطراها و ستونهایشان برابر باشند دارای قطر هستند، عناصر روی قطر عناصری هستند که اندیس سطر و ستونشان برابر می باشند.

مثال ۳۳) الگوریتمی بنویسید که ماتریس (لیست) دو بعدی X را با ۱۰ سطر و ۱۰ ستون در نظر گرفته و جدول ضرب اعداد را از ۱ تا ۱۰ را در داخل خانه های آن ذخیره نماید.



مثال ۳۴) الگوریتمی بنویسید که ماتریس A را که دارای ۵ سطر و ۵ ستون است در نظر گرفته و عناصر روی قطر آن را یک و بقیه عناصر را صفر قرار دهد.

- ۱- شروع
- ۲- $A_{5 \times 5}$ را در نظر بگیر
- ۳- $I \leftarrow 1$
- ۴- $J \leftarrow 1$
- ۵- اگر $J = I$ سپس $A(I,J) \leftarrow 1$ و گرنه $A(I,J) \leftarrow 0$

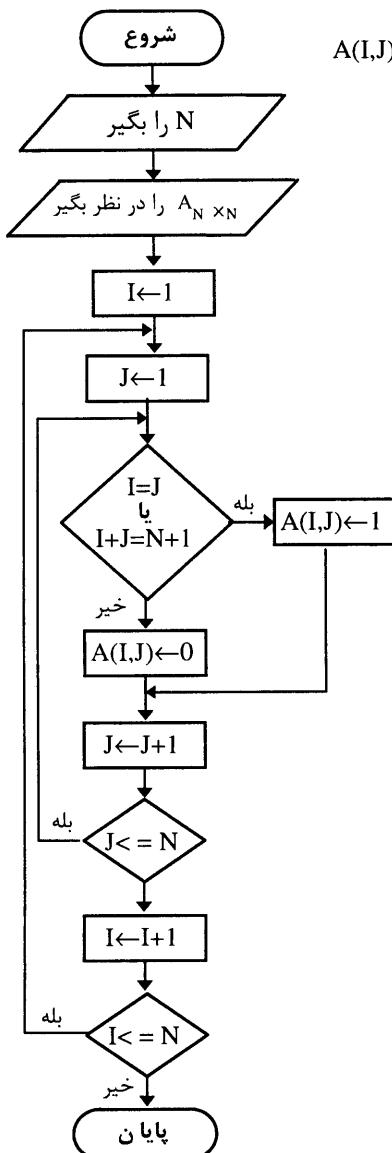


مثال ۳۵) فلوچارتی رسم کنید که عدد طبیعی N را سوال نماید و ماتریس مربع A را N در N در نظر گرفته و عناصر روی قطر اصلی و فرعی را یک و بقیه را صفر قرار دهد.

دریک ماتریس مربع ، عناصری که اندیس برابر دارند روی قطر اصلی هستند همچنین اگر اولین عنصر سطر آخر را توسط خطی به آخرین عنصر سطر اول وصل نمائیم، عناصری که روی این خط قرار می - گیرند، عناصر روی قطر فرعی نام دارند.

۱- شروع

۲- N را بگیر



- ۳ - $A_{N \times N}$ را در نظر بگیر
- ۴ - $I \leftarrow 1$
- ۵ - $J \leftarrow 1$
- ۶ - اگر $I = J$ یا $I + J = N + 1$ سپس $A(I, J) = 1$ و گرنه $A(I, J) = 0$
- ۷ - $J \leftarrow J + 1$
- ۸ - اگر $J \leq N$ سپس برو به ۹
- ۹ - $I \leftarrow I + 1$
- ۱۰ - اگر $I \leq N$ سپس برو به ۱۱
- ۱۱ - پایان

مثال ۳۶) الگوریتمی بنویسید که عدد طبیعی N را سوال نموده و ماتریس A را N در N در نظر گرفته و آن را بالامثلی نماید.

هرگاه عناصر روی قطر اصلی و بالای آن یک و بقیه عناصر صفر باشند، ماتریس مربع A را بالا مثلثی می‌گوییم. هرگاه عناصر روی قطر اصلی و زیر آن یک و بقیه عناصر صفر باشند، ماتریس را پائین مثلثی می‌گوئیم.

با اندک تغییراتی می‌توانیم الگوریتم این مسئله را برای ماتریس‌های مثلثی بنویسیم.

۱- شروع

۲- N را بگیر

۳- $A_{N \times N}$ را در نظر بگیر

۴- $I \leftarrow 1$

۵- $J \leftarrow I + 1$

۶- اگر $I = J$ یا $I > J$ سپس $A(I,J) \leftarrow 1$ و گرنه $A(I,J) \leftarrow 0$

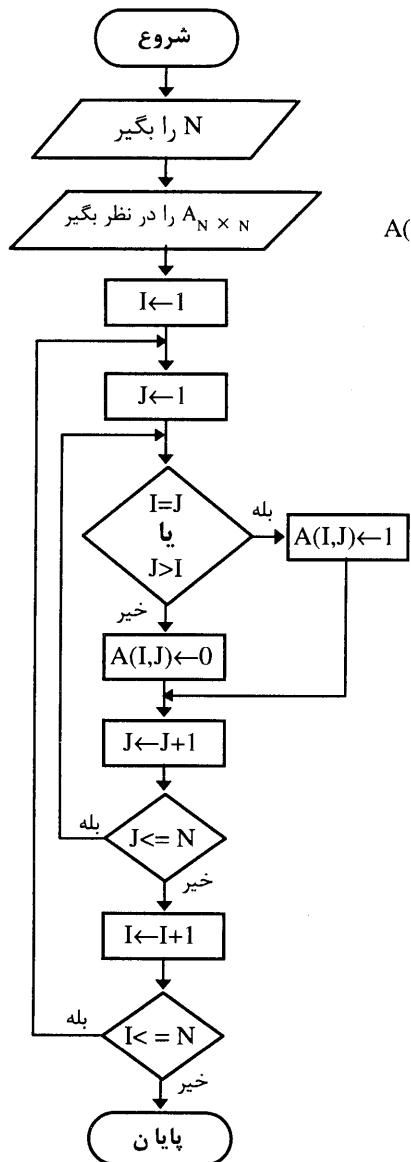
۷- $J \leftarrow J + 1$

۸- اگر $J \leq N$ سپس برو به ۶

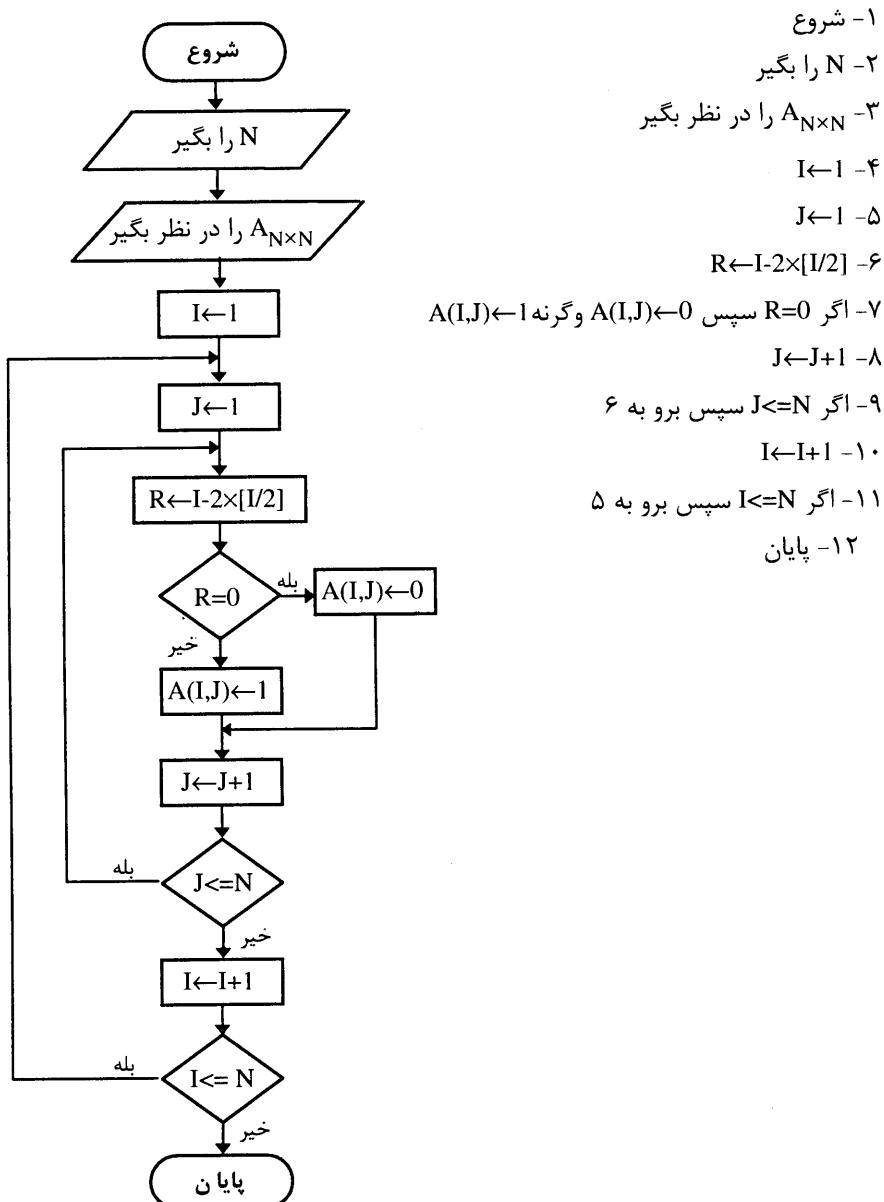
۹- $I \leftarrow I + 1$

۱۰- اگر $I \leq N$ سپس برو به ۵

۱۱- پایان.



مثال ۳۷) فلوچارتی رسم کنید که عدد طبیعی N را سوال نموده و ماتریس مربع A را N در N در نظر گرفته و سطرهای با اندیس فرد را ، ۱ و سطرهای با اندیس زوج را ، صفر قرار دهد .



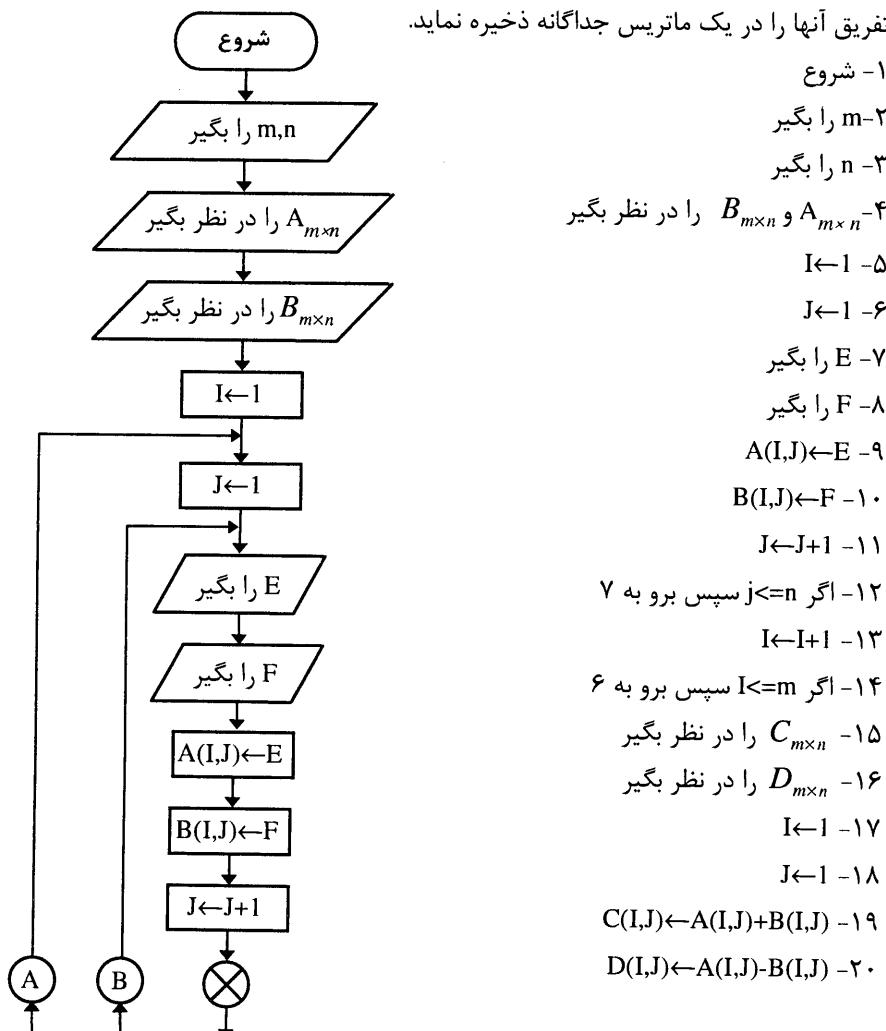
اگر دو ماتریس دارای سطر و ستون یکسان و برابر باشند می توانیم آنها را با هم جمع و یا از هم کنیم .
 مانند ماتریس های زیر :

$$A = \begin{bmatrix} 1 & 2 & 3 \\ -1 & 0 & 1 \end{bmatrix}_{2 \times 3} \quad B = \begin{bmatrix} 3 & 0 & 1 \\ 5 & 1 & 3 \end{bmatrix}_{2 \times 3}$$

$$A + B = \begin{bmatrix} 1 & 2 & 3 \\ -1 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 3 & 0 & 1 \\ 5 & 1 & 3 \end{bmatrix} = \begin{bmatrix} 1+3 & 2+0 & 3+1 \\ -1+5 & 0+1 & 1+3 \end{bmatrix} = \begin{bmatrix} 4 & 2 & 4 \\ 4 & 1 & 4 \end{bmatrix}_{2 \times 3}$$

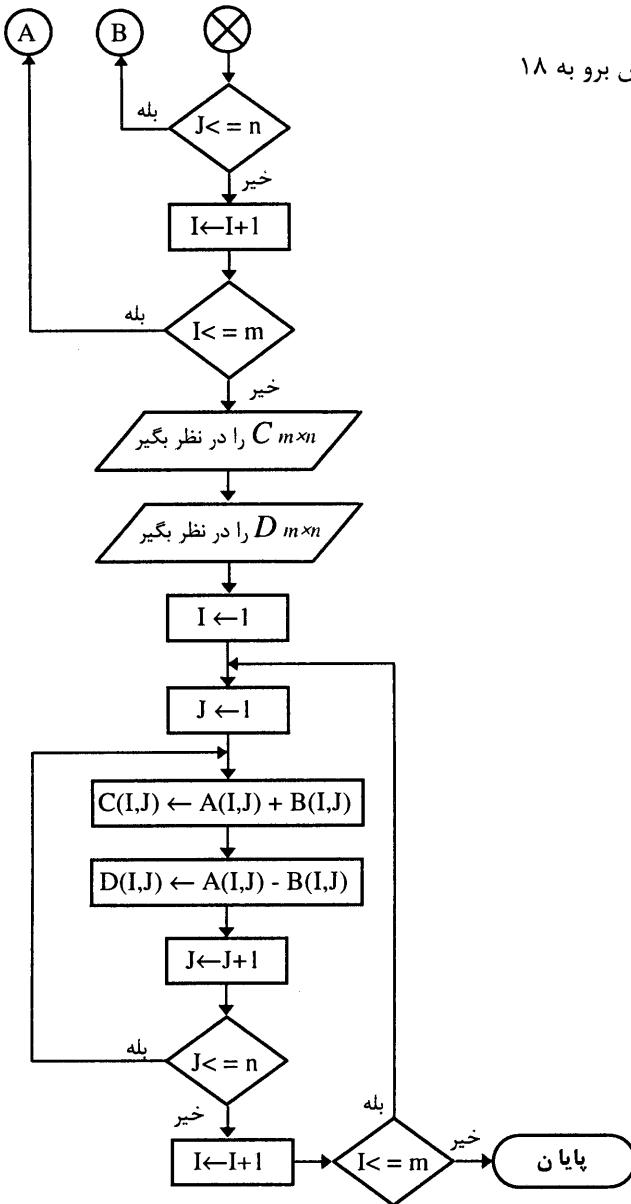
$$A - B = \begin{bmatrix} 1 & 2 & 3 \\ -1 & 0 & 1 \end{bmatrix} - \begin{bmatrix} 3 & 0 & 1 \\ 5 & 1 & 3 \end{bmatrix} = \begin{bmatrix} 1-3 & 2-0 & 3-1 \\ -1-5 & 0-1 & 1-3 \end{bmatrix} = \begin{bmatrix} -2 & 2 & 2 \\ -6 & -1 & -2 \end{bmatrix}_{2 \times 3}$$

مثال ۳۸) الگوریتمی بنویسید که دو ماتریس $A_{m \times n}$ و $B_{m \times n}$ را دریافت نموده و حاصل جمع و حاصل تفاضل آنها را در یک ماتریس جداگانه ذخیره نماید.

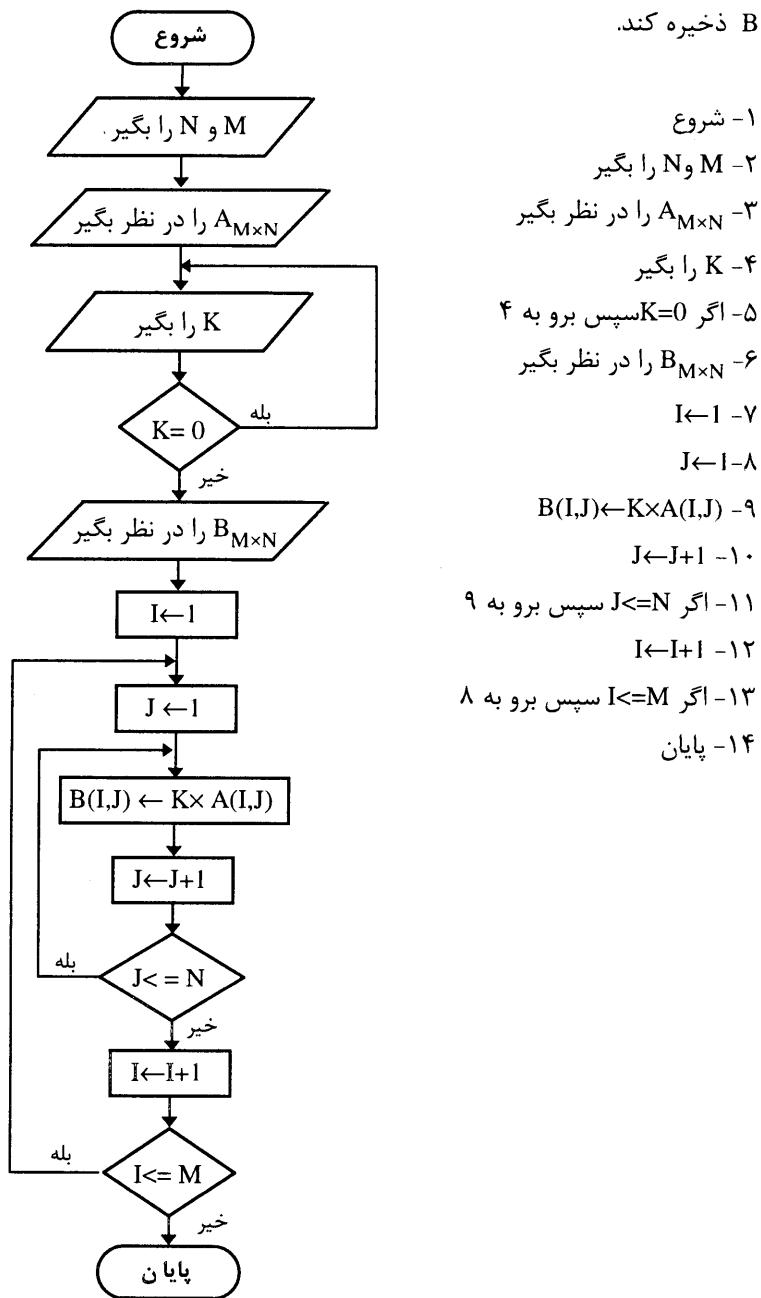


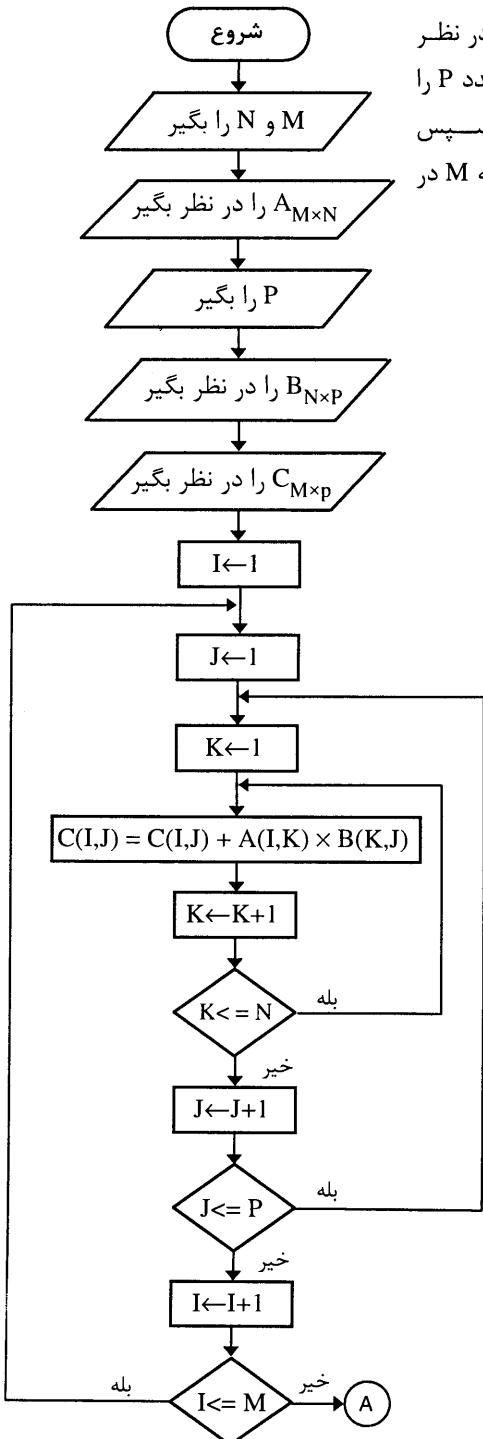
$J \leftarrow J + 1 - ۲۱$ ۱۹ - اگر $J \leq n$ سپس برو به ۲۲ $I \leftarrow I + 1 - ۲۳$ ۲۴ - اگر $I \leq m$ سپس برو به ۱۸

۲۵ - پایان.



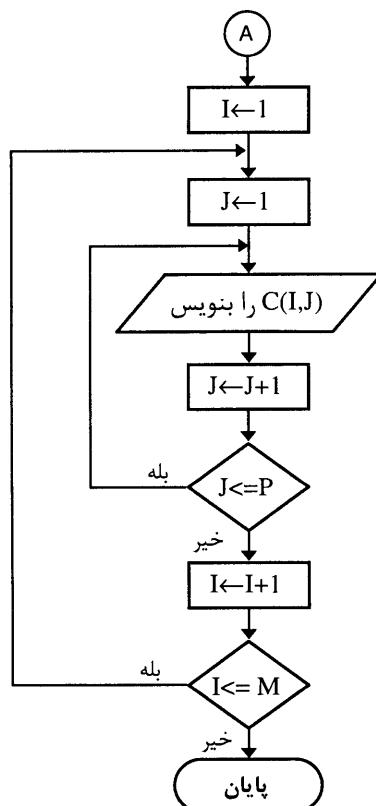
مثال ۳۹) فلوچارتی رسم کنید که ماتریس A را $M \times N$ در نظر گرفته (M و N را سوال کند) و عدد مخالف صفر K را سوال کرده و ماتریس حاصلضرب عدد K در A را محاسبه و در یک ماتریس دیگر بنام B ذخیره کند.





مثال ۴۰) الگوریتمی بنویسید که دو عدد M و N را در نظر گرفته و ماتریس A در N بسازد، همچنین عدد P را سوال نموده و ماتریس B را در P بسازد. سپس حاصلضرب ماتریس A در B را درون ماتریس C که M در P است ذخیره نموده و ماتریس C را چاپ کند.

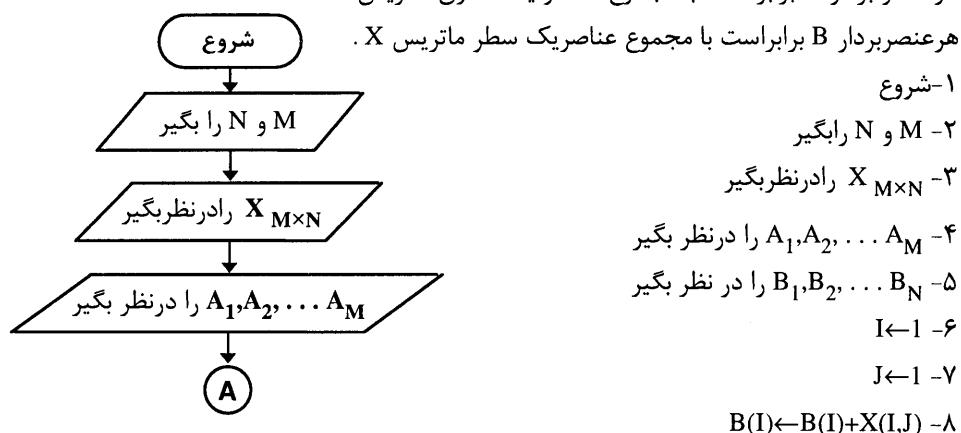
- ۱- شروع
- ۲- N و M را بگیر
- ۳- $A_{M \times N}$ را در نظر بگیر
- ۴- P را بگیر
- ۵- $B_{N \times p}$ را در نظر بگیر
- ۶- $C_{M \times p}$ را در نظر بگیر
- $I \leftarrow 1$
- $J \leftarrow 1$
- $K \leftarrow 1$
- $C(I,J) \leftarrow C(I,J) + A(I,K) \times B(K,J)$
- $K \leftarrow K+1$
- اگر $K \leq N$ سپس برو به ۱۰
- $J \leftarrow J+1$
- اگر $J \leq P$ سپس برو به ۹
- $I \leftarrow I+1$
- اگر $I \leq M$ سپس برو به ۸
- $I \leftarrow 1$
- $J \leftarrow 1$
- $C(I,J)$ را بنویس
- $J \leftarrow J+1$
- اگر $J \leq P$ سپس برو به ۱۹
- $I \leftarrow I+1$
- اگر $I \leq M$ سپس برو به ۱۸
- پایان



مثال ۴۱) الگوریتمی بنویسید که دو عدد M و N را سوال نموده و ماتریس X را به صورت M در N در نظر گرفته و بردارهای A و B به ترتیب N عنصری و M عنصری را به صورت زیر ساخته و نمایش دهد.

هر عنصر بردار A برابر است با مجموع عناصر یک ستون ماتریس X .

هر عنصر بردار B برابر است با مجموع عناصر یک سطر ماتریس X .



$$A(J) \leftarrow A(J) + X(I,J) \quad -9$$

$$J \leftarrow J+1 \quad -10$$

۱۱ - اگر $J \leq N$ سپس برو به ۸

$$I \leftarrow I+1 \quad -12$$

۱۳ - اگر $I \leq M$ سپس برو به ۷

$$I \leftarrow I-1 \quad -14$$

۱۵ - رابتویس $B(I)$

$$I \leftarrow I+1 \quad -16$$

۱۷ - اگر $I \leq M$ سپس برو به ۱۵

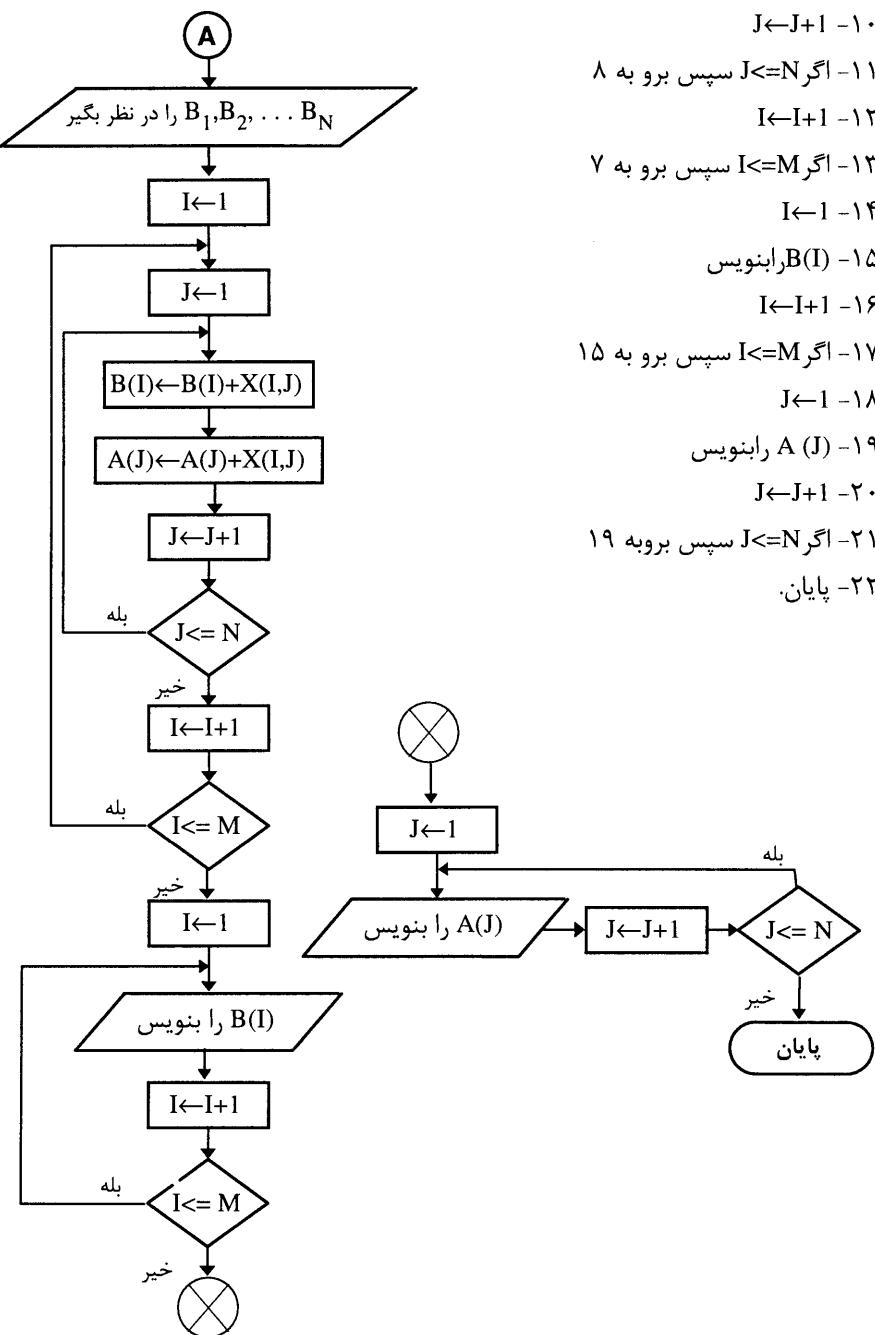
$$J \leftarrow 1 \quad -18$$

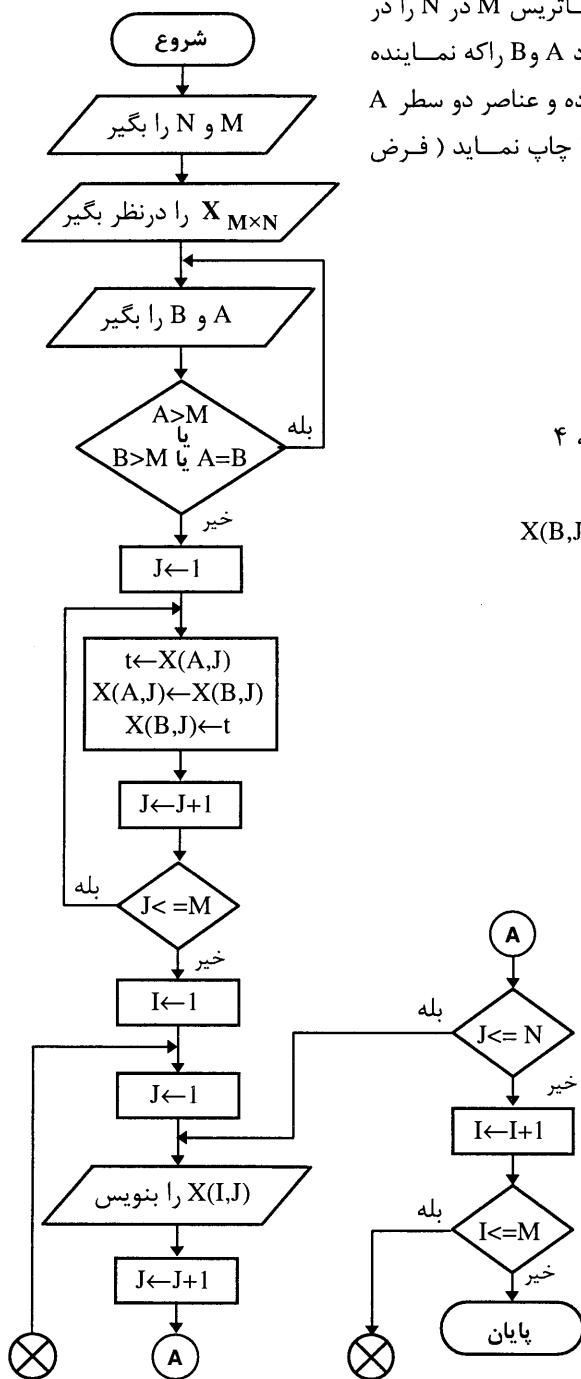
۱۹ - $A(J)$ را بنویس

$$J \leftarrow J+1 \quad -20$$

۲۱ - اگر $J \leq N$ سپس برو به ۹

۲۲ - پایان.





مثال ۴۲) فلوچارتی رسم کنید که یک ماتریس M در N را در نظر گرفته (M و N را سوال کند) و دو عدد A و B را که عناصر دو سطر A و B را جابجا کند. سپس عناصر ماتریس را چاپ نماید (فرض کنید که ماتریس دارای $M \times N$ عدد است).

۱- شروع

۲- M و N را بگیر

۳- $X_{M \times N}$ را در نظر بگیر

۴- A و B را بگیر

۵- اگر $A > M$ یا $B > M$ یا $A = B$ سپس برو به ۴

۶- $J \leftarrow 1$

۷- $X(B,J) \leftarrow t$ و $X(A,J) \leftarrow X(B,J)$ و $t \leftarrow X(A,J)$

۸- $J \leftarrow J + 1$

۹- اگر $J \leq M$ سپس برو به ۷

۱۰- $I \leftarrow 1$

۱۱- $J \leftarrow 1$

۱۲- $X(I,J)$ را بنویس

۱۳- $J \leftarrow J + 1$

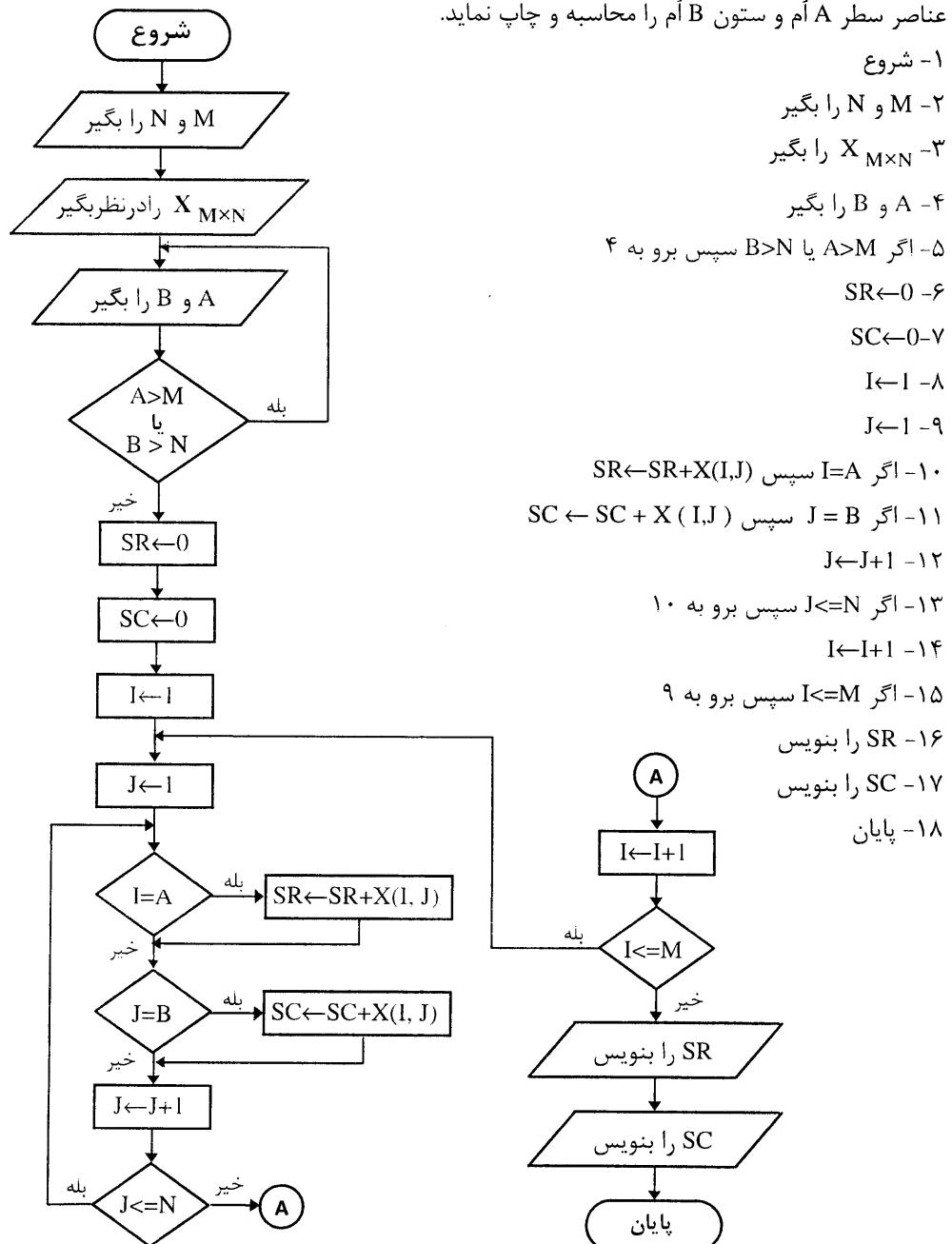
۱۴- اگر $J \leq N$ سپس برو به ۱۲

۱۵- $I \leftarrow I + 1$

۱۶- اگر $I \leq M$ سپس برو به ۱۱

۱۷- پایان

مثال ۴۳) فلوچارتی رسم کنید که یک ماتریس M در N را در نظر گرفته (M و N را سوال کند) و دو عدد A و B را که به ترتیب نماینده یک سطر و ستون دلخواه ماتریس هستند، سوال کرده و مجموع عناصر سطر A و ستون B ام را محاسبه و چاپ نماید.



فرض کنید ماتریس $A_{m \times n}$ موجود است اگر عنصر واقع در سطر I ام و ستون J ام را با عنصر واقع در سطر Z ام و ستون Y ام عوض کنید و این عمل را برای تمامی عناصر ماتریس A انجام دهید، آنگاه ماتریس حاصل را ترانهاده ماتریس A گفته و با $A_{n \times m}$ نمایش می‌دهند.
مثال (۴۴) الگوریتمی بنویسید که ماتریس A را درنظر گرفته و ترانهاده آن را محاسبه نماید.(فلوچارت آنرا به عنوان تمرین رسم کنید).

- شروع
- ۲ $A_{5 \times 5}$ را در نظر بگیر
- ۳ $I \leftarrow 1$
- ۴ $J \leftarrow 1$
- ۵ B را بگیر
- ۶ $A(I,J) \leftarrow B$
- ۷ $J \leftarrow J+1$
- ۸ اگر $J \leq 5$ سپس برو به ۶
- ۹ $I \leftarrow I+1$
- ۱۰ اگر $I \leq 5$ سپس برو به ۴
- ۱۱ $I \leftarrow I-1$
- ۱۲ $J \leftarrow 1$
- ۱۳ $A(I,J) \leftarrow A(J,I)$
- ۱۴ $J \leftarrow J+1$
- ۱۵ اگر $J \leq 5$ سپس برو به ۱۳
- ۱۶ $I \leftarrow I+1$
- ۱۷ اگر $I \leq 5$ سپس برو به ۱۲
- ۱۸ پایان .

بحث مربوط به لیستهای دو بعدی را با ذکر این نکته که لیستهای چند بعدی نیز وجوددارند، به پایان می‌رسانیم و در بعضی از مسائل که نیاز به آنها احساس می‌شود استفاده می‌شوند، لازم به توضیح است که به ندرت از لیستهای سه بعدی و یا بالاتر استفاده می‌کنیم.

پرسش‌های تشریحی

- ۱- تعریف جامع و دقیقی از اندیس ارائه نمایید .
- ۲- مفهوم متغیر اندیسدار را به طور کامل توضیح دهید .
- ۳- بردار را تعریف نمایید و دلایل استفاده از آنرا توضیح دهید .
- ۴- آرایه یا لیست را بطور کامل توضیح دهید .
- ۵- ماتریس را تعریف کنید .
- ۶- لیستهای دو بعدی را تشریح نمایید و دلایل استفاده از آنها را بنویسید .
- ۷- نمادهایی که برای نشان دادن انواع لیستها در الگوریتم و فلوچارت استفاده می شود توضیح دهید .
- ۸- مثال ۱ این فصل را به طور کامل توضیح دهید .
- ۹- الگوریتمهای مثالهای ۱۷ و ۱۸ و ۲۰ را بطور دستی اجرا نمایید .
- ۱۰- توضیح دهید تحت چه شرائطی میتوانیم دو ماتریس را با هم جمع یا از هم کم کنیم ؟
- ۱۱- با استفاده از فایل مسائلی را فصل دوم حل نمودیم ، توضیح دهید که آیا میتوانیم با استفاده از لیستها آنها حل کنیم ؟
- ۱۲- با استفاده از مفهوم لیستها روشی برای ذخیره کردن اعداد بزرگ ارائه دهید .
- ۱۳- در آرایه های دو بعدی توضیح دهید که چگونه خانه ها با مقادیر مختلف پر می شوند (به دو روش)
- ۱۴- فلوچارت مثالهای ۲۱ و ۲۲ را رسم نمایید . و آنها را بطور دستی اجرا نمایید .
- ۱۵- ایده ای از لیستها (آرایه ها) را سه بعدی ارائه نمایید .

مسائل

۱- الگوریتمی بنویسید که آرایه N عنصری X را در نظر بگیرد و عددی مانند B را سوال نماید و معین کند چند عضو X با عدد B برابر است . همچنین شماره خانه‌هایی از X را که مقدارشان با عدد B برابر است محاسبه و چاپ نماید .

۲- الگوریتمی بنویسید که N را سوال نماید و آرایه N عنصری A را در نظر بگیرد و N عدد راسوال کرده و آنها را در آرایه A ذخیره نماید و کارهای زیر را انجام دهد .
الف - میانگین اعداد را محاسبه کند .

ب- بزرگترین و کوچکترین عدد را از بین اعداد آرایه A محاسبه کند .

ج- معین کند چند عدد کوچکتر از میانگین وجود دارد ، سپس آنها را چاپ کند .

د- عدد دیگری مانند Z را سوال نماید و تعداد اعداد بین Z و میانگین را محاسبه نماید ، و اعداد را یکی یکی چاپ نماید .

۳- الگوریتمی بنویسید که تمام کلمات ۴ حرفی را که از حروف IRAN میتوان ساخت محاسبه نماید .

۴- الگوریتمی بنویسید که تمام اعداد ۳ رقمی را که باستفاده از ارقام ۱ و ۲ و ۴ و ۷ میتوان ساخت محاسبه نماید .

۵- الگوریتمی بنویسید که یک چند جمله‌ای درجه N را درنظر بگیرد و مقدار آن را دریک نقطه خاصی مانند b محاسبه و چاپ نماید .

توضیح :

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$$

$$P(b) = ?$$

۶- الگوریتمی بنویسید که دو چند جمله‌ای P و q را که از درجه های N و M هستند در نظر بگیردو مجموع و تفاضل آنها را محاسبه و چاپ نماید .

۷- الگوریتمی بنویسید که دو عدد ۲۰ رقمی در مبنای ۴ را دریافت نماید و حاصل جمع آنها را محاسبه و چاپ نماید .

۸- الگوریتمی بنویسید که دنباله اعداد فیبوناچی را با استفاده از لیستها تا ۱۰۰ جمله محاسبه و چاپ نماید .

۹- الگوریتمی بنویسید که ارقام عدد حاصل از تقسیم عدد ۱ را بر عدد ۳ تا ۲۰۰ رقم محاسبه و چاپ نماید .

۱۰- الگوریتمی بنویسید که ماتریس $A_{N \times N}$ را در نظر بگیرد و عناصر بالای قطر را همگی صفر و روی قطر را ۱ و زیر قطر را ۱ قرار دهد.

۱۱- ماتریسی متقارن است که عناصر واقع در دو طرف قطر آن برابر باشند. الگوریتمی بنویسید که ماتریس $A_{N \times N}$ را در نظر بگیرد و معین کند که ماتریس متقارن است یا نه.

۱۲- فرض کنیم $A_{M \times N}$ و $B_{N \times P}$ باشد. در این صورت ضرب ماتریس A در B تعریف شده است بعارت دیگر باید تعداد ستونهای A با تعداد سطرهای B برابر باشد. که ماتریس حاصل ضرب همواره $M \times P$ است.

مثال :

$$A = \begin{bmatrix} 1 & 2 & 0 \\ 1 & -1 & 1 \end{bmatrix}_{2 \times 3}$$

$$B = \begin{bmatrix} 2 & 1 \\ 3 & -1 \\ 0 & 1 \end{bmatrix}_{3 \times 2}$$

بین ترتیب ضرب $B \times A$ تعریف شده است اما $A \times B$ نه. و برای انجام عمل ضرب هریک از سطرهای ماتریس A را در ستونهای ماتریس B ضرب می کنیم و متناظر با عنصر خاص قرار می دهیم. (سطر در ستون) و این عمل را برای کلیه سطرها و ستونها انجام میدهیم.

(ضرب سطر در ستون همان ضرب تک تک عناصر و سپس جمع آنهاست)

الگوریتمی بنویسید که دو ماتریس A و B را در نظر بگیرد و معین کند آیا می توانیم آنها را درهم ضرب کنیم یا نه در صورت امکان حاصل ضرب آنها را محاسبه کند.

۱۳- الگوریتمی بنویسید که ماتریس مربع، $A_{N \times N}$ را در نظر بگیرد و خانه های آرایه را به صورت زیر پر نماید.

$$A(1,1)=1$$

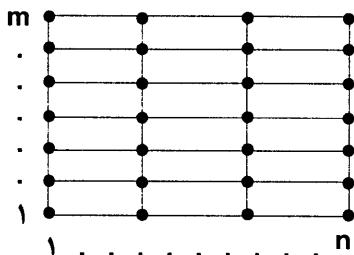
$$A(I,J)=I+J$$

۱۴- الگوریتمی بنویسید که عدد N را سوال نماید و ماتریس $A_{N \times N}$ را در نظر بگیرد و اعداد از ۱ تا N^2 را به ترتیب در خانه های آن ذخیره نماید.

۱۵- الگوریتمی بنویسید که دستگاه سه معادله سه مجهولی زیرا در نظر بگیرد و به روش ماتریسی آنرا حل نماید.

$$\begin{cases} a_{11}x + a_{12}y + a_{13}z = b_1 \\ a_{21}x + a_{22}y + a_{23}z = b_2 \\ a_{31}x + a_{32}y + a_{33}z = b_3 \end{cases}$$

- سعی کنید این مسئله را برای دستگاههای با N معادله و N مجهول حل نماید.
- ۱۶- الگوریتمی بنویسید که کلیه زیر مجموعه های یک مجموعه n عضوی را محاسبه و چاپ نماید. (منظور تمامی حالات است).
- ۱۷- فرض کنید جدولی مستطیل شکل داریم که می خواهیم از خانه (i, j) به خانه (l, k) حرکت کنیم. تعداد راه های ممکن را محاسبه و چاپ نماید (حرکت ها شامل بالا و راست هستند).



- توجه : مسائل ۱۶ و ۱۷ را با استفاده از مبنای ۲ حل نماید.
- ۱۸- الگوریتمی بنویسید که معادل یک عدد در مبنای ۲ را مستقیماً به مبنای ۱۶ ببرد (بدون استفاده از مبنای ۱۰)
- ۱۹- مسئله ۱۵ را برای مبنای ۸ نیز حل کنید.
- ۲۰- الگوریتمی بنویسید که با استفاده از آرایه ها بزرگترین مقسوم علیه مشترک و کوچکترین مضرب مشترک دو عدد را محاسبه کند.
- ۲۱- با استفاده از فرمول زیر روشی برای تولید ارقام عدد π تا ۵۰ رقم را ارائه کنید.
- $$\pi = 4 \left(1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots \right)$$
- ۲۲- با استفاده از فرمول زیر روندی را برای تولید عدد e (پایه لگاریتم نپر) ارائه نماید.
- $$e = \left(1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots \right)$$

فصل چهارم

مرتب کردن
و جستجوی اطلاعات

اهداف کلی

- ۱- آشنایی با مفهوم جستجو کردن داده‌ها
- ۲- آشنایی با مفهوم مرتب کردن داده‌ها و دلایل استفاده از آنها

هدفهای رفتاری

دانشجو پس از مطالعه این فصل باید بتواند :

- ۱- هریک از روش‌های مرتب سازی را بیان و آنها را با هم مقایسه کند.
- ۲- در برخورد با یک مساله واقعی بتواند از بین این روشها بهترین را با توجه به نوع داده‌ها و مساله انتخاب و اجراء نماید.
- ۳- الگوریتم هر یک از روش‌های مرتب سازی را برای داده‌های مختلفی بطور دستی تست نماید.

مرتب کردن و جستجوی اطلاعات از عملیات اساسی و اصلی علم کامپیوتر می‌باشند. مرتب کردن عبارت است از عمل تجدید آرایش داده‌ها با یک ترتیب مشخص، مثلاً برای داده‌های عددی، ترتیب صعودی یا نزولی اعداد یا برای داده‌های کاراکتری ترتیب الفبائی آنها. عمل پیدا کردن مکان یک عنصر داده شده در بین مجموعه‌ای از عناصر را جستجو کردن می‌نامند.

برای مرتب کردن و جستجوی اطلاعات الگوریتم‌های متعددی وجود دارد. الگوریتمی که برای هر مسئله خاص انتخاب می‌شود، بستگی به خصوصیات داده‌ها و خصوصیات عملیاتی دارد که ممکن است روی داده‌ها انجام شود.

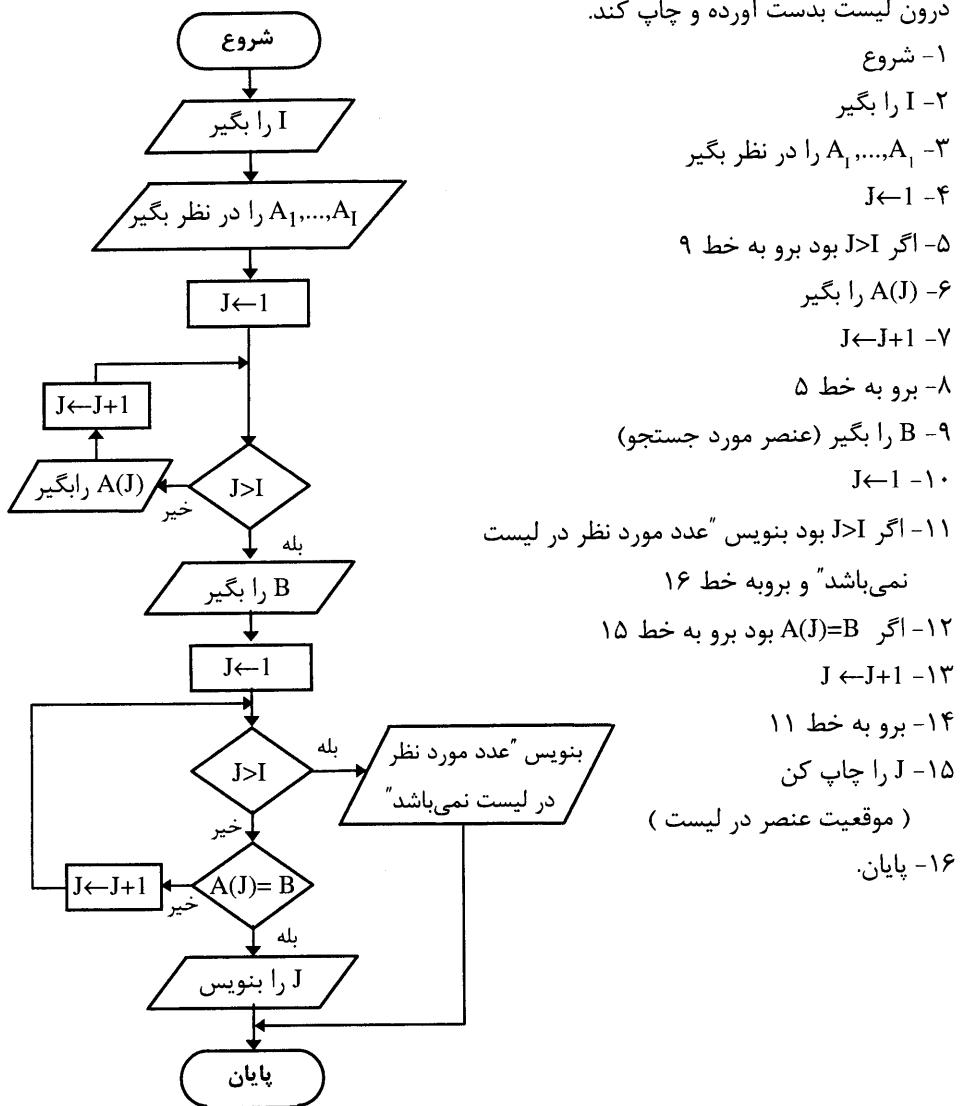
روشهای جستجو :

- ۱- روش جستجوی خطی یا ترتیبی
- ۲- روش جستجوی دودوئی

جستجوی ترتیبی :

اگر عنصری در مجموعه داده‌ها موجود باشد، جستجو را موفق و در غیر این صورت ناموفق می‌نامند. فرض کنید که داده‌ها در یک لیست یک بعدی با n عنصر قرار دارند. در این روش برای پیدا کردن یک مقدار در لیست، آن را با تمام عناصر (n عنصر) مقایسه می‌کنیم تا محل آن در لیست معلوم شود. در صورتیکه مقدار مورد جستجو در لیست نباشد.

مثال ۱) الگوریتمی بنویسید که تعداد محدودی عدد را از ورودی گرفته و در یک لیست یک بعدی ذخیره نموده و سپس عدد دیگری را از ورودی خوانده و موقعیت آنرا در درون لیست بدست آورده و چاپ کند.

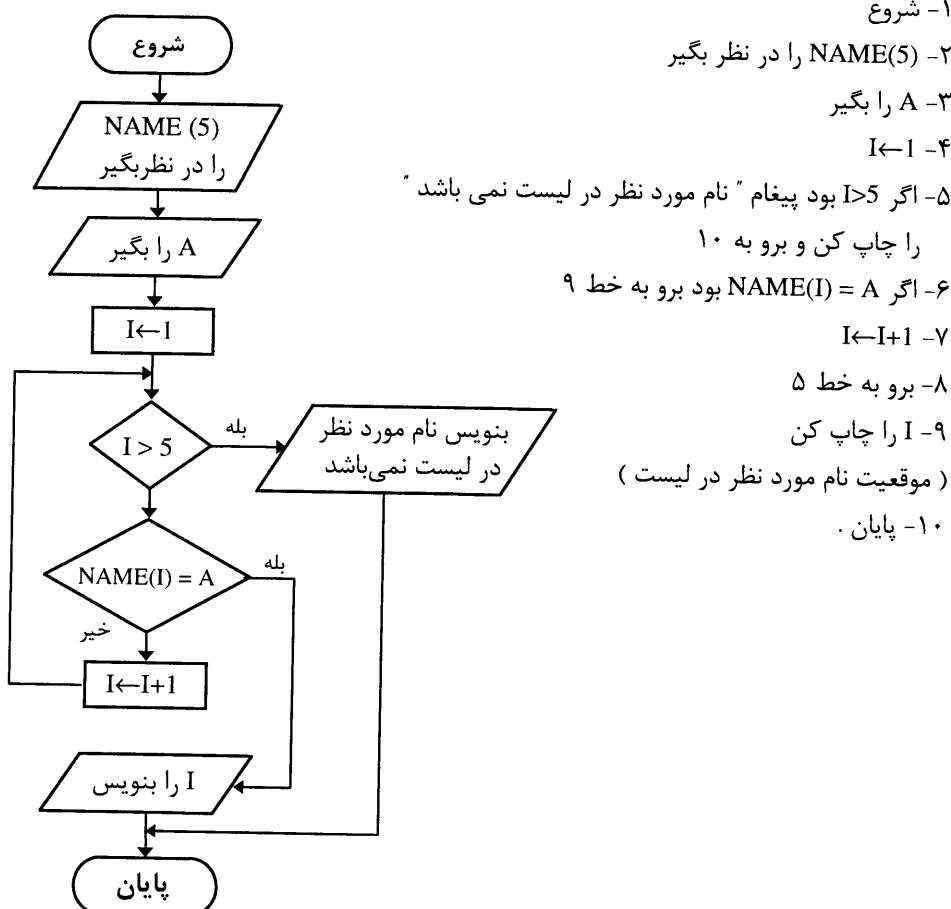


- ۱- شروع
- ۲- I را بگیر
- ۳- A_1, \dots, A_I را در نظر بگیر
- ۴- $J \leftarrow 1$
- ۵- اگر $J > I$ بود برو به خط ۹
- ۶- A(J) را بگیر
- ۷- $J \leftarrow J + 1$
- ۸- برو به خط ۵
- ۹- B را بگیر (عنصر مورد جستجو)
- ۱۰- $J \leftarrow 1$
- ۱۱- اگر $J > I$ بود بنویس "عدد مورد نظر در لیست نمیباشد" و برو به خط ۱۶
- ۱۲- اگر A(J)=B بود برو به خط ۱۵
- ۱۳- $J \leftarrow J + 1$
- ۱۴- برو به خط ۱۱
- ۱۵- J را چاپ کن
- (موقعیت عنصر در لیست)
- ۱۶- پایان.

مثال ۲) یک لیست یک بعدی بنام NAME را در نظر بگیرید که دارای ۵ عنصر بترتیب زیر می باشد :
NAME (5) :

ALIREZA	BAHRAM	AFSHIN	AMIR	SARA
---------	--------	--------	------	------

الگوریتمی بنویسید که یک نام را از ورودی گرفته و در لیست فوق مکان آن را مشخص نماید.

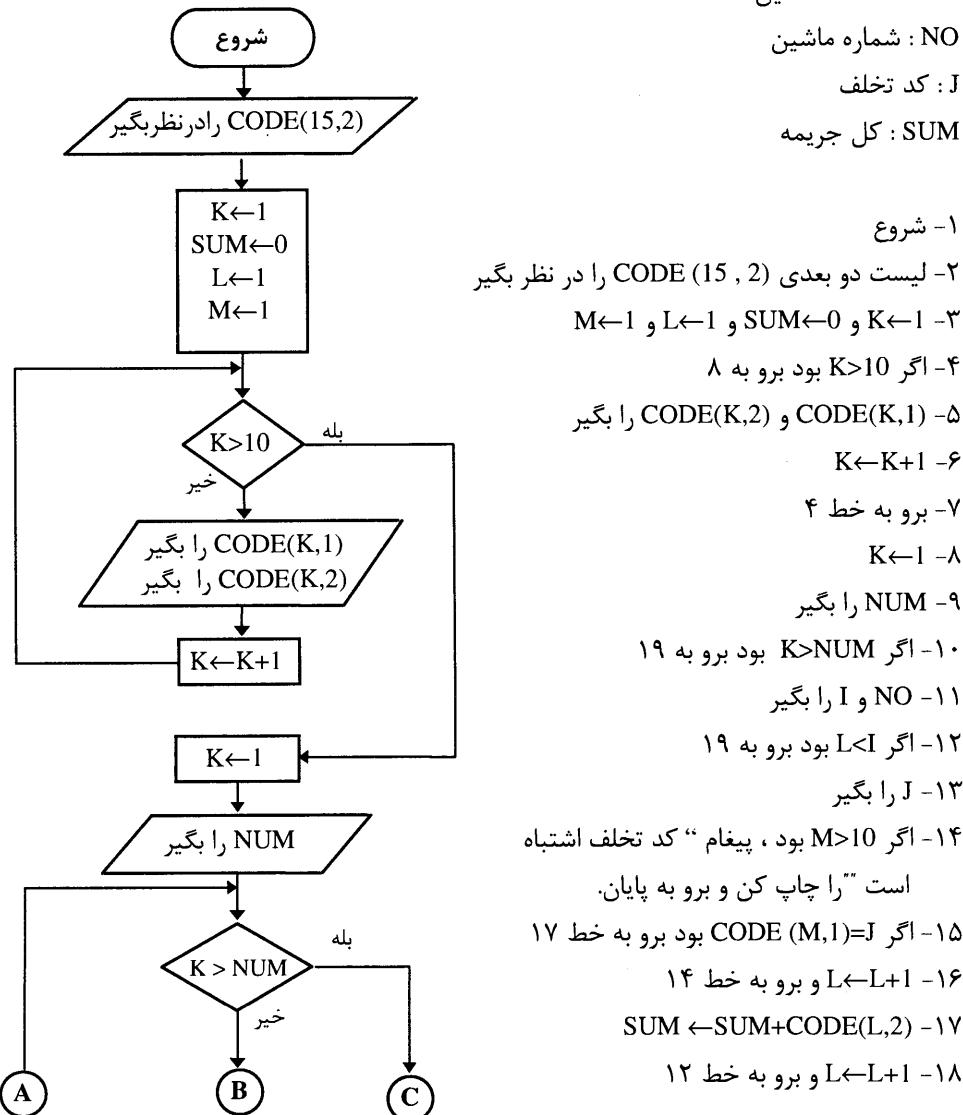


مثال ۳) فرض کنید در اداره صدور عدم خلافی اتومبیلها ، ۱۰ نوع تخلف در نظر گرفته شده است و هر تخلفی دارای یک کد خاص از یک تا ۱۰ بوده و هر تخلفی یک جرمیه خاصی دارد.

الگوریتمی بنویسید که شماره ماشین، تعداد تخلفات و کد تخلف را از ورودی دریافت نموده و مبلغ جرمیه را برای هر ماشین محاسبه و نمایش دهد.

CODE : یک لیست دو بعدی با ۱۰ سطر و ۲ ستون می باشد. ستون ۱ کدها و ستون ۲ مبلغ جرمیه .

NUM : تعداد ماشین ها

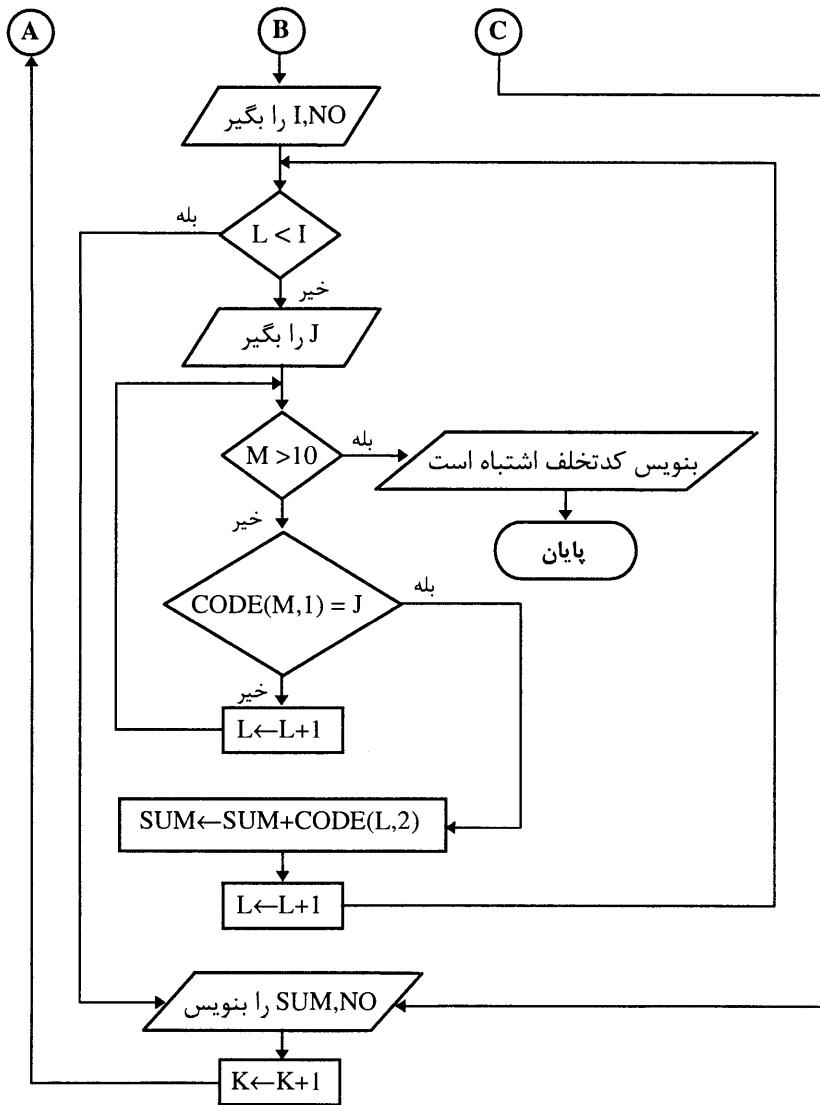


۱۹- NO و SUM را چاپ کن

۲۰- $K \leftarrow K+1$

۲۱- برو به خط ۱۰

۲۲- پایان.



جستجوی دودویی :

فرض کنید داده‌ها به صورت یک آرایه می‌باشند که در آن داده‌های عددی یا حرفی بصورت مرتب شده ذخیره شده‌اند. آنگاه الگوریتم جستجوی بسیار کارائی به نام جستجوی دودویی وجود دارد که می‌توان از آن برای پیدا کردن موقعیت و مکان یک عنصر داده شده از اطلاعات در داده‌ها استفاده کرد.

در صورتی که لیست مورد جستجو مرتب باشد، می‌توان با تعداد مقایسه کمتری، داده مورد نظر را جستجو و پیدا نمود. در این روش داده‌های مورد نظر با عنصر وسطی لیست مقایسه گردیده و سپس بسته به صعودی یا نزولی بودن لیست، قسمت پائین یا بالای لیست، جستجو و با عنصر آن قسمت مقایسه می‌شود. این عمل تازمانی ادامه می‌یابد که آخرین عنصر، (حد بالائی) کمتر از اولین عنصر (حد پائینی) لیست جدید باشد. در صورتیکه حد بالایی کمتر از حد پائینی باشد، جستجو ناموفق بوده است. این روش را با شرح یک مثال آشنا در زندگی روزمره توضیح می‌دهیم.

فرض کنید بخواهید مکان یک اسم را در راهنمای تلفن پیدا کرده یا در فرهنگ لغت، یک لغت را جستجو کنید. واضح است که یک جستجوی خطی را انجام نمی‌دهید و به جای آن، کتابچه راهنمای را از وسط باز کرده و دنبال آن نیمه از راهنمای گردید که حدس می‌زنید، اسم مورد نظر در آن نیمه قرار دارد. آنگاه نیمه اخیر را از وسط نصف کرده و در یک چهارم از راهنمای گردید که حدس می‌زنید اسم مورد نظر در آن یک چهارم قرار دارد و بدنبال آن یک چهارم اخیر را از وسط نصف کرده و کار را همین طورتا آخر ادامه می‌دهید. با توجه به این که خیلی سریع تعداد مکانهای ممکن در راهنمای کاهش می‌یابند، در نهایت مکان اسم و اسم مورد نظر را پیدا می‌کنید.

مثال ۴) فرض کنید داده‌ها در لیستی که بصورت صعودی مرتب شده و دارای N عضو می‌باشد، قراردارند. برای جستجوی یک عنصر (ITEM)، داریم :

۱- شروع

۲- N را بگیر

۳- $\text{END} \leftarrow \text{N}$ و $\text{BEG} \leftarrow 1$

۴- $\text{MID} \leftarrow [(\text{BEG}+\text{END})/2]$

۵- اگر ITEM بود ، $\text{END} \leftarrow \text{MID}-1$ و برو به ۳

۶- اگر ITEM بود ، $\text{BEG} \leftarrow \text{MID}+1$ و برو به ۳

۷- اگر $\text{BEG} > \text{END}$ بود ، چاپ کن " عدد در لیست وجود ندارد " .

۸- چاپ کن MID را . (موقعیت عنصر در لیست)

۹- پایان .

مثال ۵) الگوریتمی بنویسید که N عدد نزولی را خوانده و در یک لیست قرار دهد و سپس عددی را خوانده و محل آن را در لیست مشخص کند.

- ۱- شروع
- ۲- N را بگیر
- ۳- لیست A_1, A_2, \dots, A_N را در نظر بگیر
- ۴- $M \leftarrow 1$
- ۵- اگر $M > N$ بود برو به خط ۸
- ۶- $A(M)$ را بگیر
- ۷- $M \leftarrow M+1$
- ۸- B را بگیر
- ۹- $K \leftarrow 1$ (حد پائینی)
- ۱۰- $J \leftarrow N$ (حد بالائی)
- ۱۱- $I \leftarrow [(K+J)/2]$
- ۱۲- اگر $A(I)=B$ بود ، چاپ کن I را و برو به ۱۶
- ۱۳- اگر $A(I)>B$ بود آنگاه $J \leftarrow I-1$ و گرنم $K \leftarrow I+1$
- ۱۴- اگر $K>J$ بود چاپ کن " عدد در لیست وجود ندارد " و برو به ۱۶
- ۱۵- برو به خط ۱۱
- ۱۶- پایان .

کنترل اجرای برنامه مثال ۱ به ازای $N=13$ و لیست $(13)(13)$ به ترتیب زیر می باشد :

ITEM = 40

11	22	30	33	40	44	55	60	66	77	80	88	99
----	----	----	----	----	----	----	----	----	----	----	----	----

BEG	END	MID	SORT(MID)	SORT(MID),ITEM
1	13	7	55	55>40
1	7-1=6	3	30	30>40
1+3=4	6	5	40	40=40

کنترل اجرای برنامه مثال ۲ به ازای $N=4$

$B=25$

50	40	28	20
----	----	----	----

N	I	J	K	B
4				25
	2	4	1	
			$2+1=3$	

روشهای مرتب سازی

مرتب کردن عبارتست از عمل تجدید آرایش داده‌ها با یک ترتیب مشخص ، مثلا برای داده‌های عددی به ترتیب صعودی یا نزولی ، یا برای داده‌های کاراکتری به ترتیب الفبایی آنها . برای مرتب سازی نیز مثل جستجو، روش‌ها و الگوریتم‌های متعددی وجود دارد که بستگی به خصوصیات داده‌ها و خصوصیات عملیاتی دارد که روی داده‌ها انجام خواهند شد.

اختلاف عمده روش‌های مختلف در سرعت مرتب سازی و محیطی است که اطلاعات مرتب شده و در آنجا ذخیره می‌شوند، می‌باشد.

روشهای مورد بحث در این فصل عبارتند از : روش حبابی ، روش درجی ، روش SHELL ، روش ... HEEP SORT

روش مرتب سازی حبابی (BUBBLE SORT)

توسط این روش می‌توان، داده‌ها را با دو الگوریتم متفاوت مرتب نمود.
الگوریتم اول :

تمام داده‌ها در این روش با هم دیگر مقایسه می‌شوند، یعنی داده اول ابتدا با داده دوم مقایسه می‌شود، در صورتی که داده اول بیشتر از داده دوم باشد، محل آنها با هم عوض می‌شود. سپس داده اول با داده سوم مقایسه شده، و در صورتیکه بیشتر از داده سوم باشد، جای آن دو با هم تعویض می‌شود و این عمل مقایسه تا آخرین داده گرفته و سپس کمترین عنصر در مکان اول قرار می‌گیرد. سپس داده دوم را با بقیه داده‌ها بعدی مقایسه می‌کنند و در نهایت یک لیست مرتب شده صعودی بدست آید .

مثال ۶) الگوریتمی بنویسید که تعدادی عدد را از ورودی گرفته و به شکل صعودی مرتب کند. سپس لیست مرتب شده را چاپ کند.

- ۱- شروع
- ۲- I را بگیر
- ۳- A_1, \dots, A_n را در نظر بگیر
- $J \leftarrow 1$
- ۵- $A(J)$ را بگیر
- $J \leftarrow J+1$
- ۷- اگر $I < J$ بود برو به خط ۵
- $J \leftarrow 1$
- $K \leftarrow J+1$
- ۱۰- اگر $K > I$ برویه ۱۳
- ۱۱- اگر $A(K) < B$ و $A(J) < A(K)$ بود ، $A(J) > A(K)$ بود ، $B \leftarrow A(J)$ و $A(J) \leftarrow A(K)$
- ۱۲- $K \leftarrow K+1$ و برو به خط ۱۰
- $J \leftarrow J+1$
- ۱۴- اگر $J < I$ بود برو به خط ۹
- $M \leftarrow 0$
- $M \leftarrow M+1$
- ۱۷- $A(M)$ را بنویس
- ۱۸- اگر $M < I$ بود ، برویه ۱۶
- ۱۹- پایان .

به عنوان تمرین برای چند داده مختلف الگوریتم را تست نمایید.
مثال ۷) الگوریتمی بنویسید که عدد را از ورودی خوانده و بصورت نزولی مرتب نماید.

- ۱- شروع
- ۲- A_1, \dots, A_{20} را در نظر بگیر
- $I \leftarrow 0$
- $I \leftarrow I+1$
- ۵- $A(I)$ را بگیر
- ۶- اگر $I < 20$ بود برویه ۴
- $I \leftarrow 0$
- $I \leftarrow I+1$
- $J \leftarrow I+1$

- ۰- اگر $A(J) < A(I)$ بود، $X \leftarrow A(I)$ و $A(I) \leftarrow A(J)$
- ۱- اگر $I < J$ بود برو به ۹
- ۲- اگر $I > J$ بود برو به ۸
- ۳- $K \leftarrow I$
- ۴- $A(K) \leftarrow A(I)$
- ۵- $K \leftarrow K+1$
- ۶- اگر $K < 20$ بود برو به ۱۴
- ۷- پایان

الگوریتم دوم: بکار گیری سویچ در روش مرتب سازی حبابی

با استفاده از یک سویچ در الگوریتم مرتب سازی می توان کنترل نمود که آیا مرتب سازی داده ها تمام شده است یا نه ؟ اگر مثال اول را در نظر بگیرید برای مرتب نمودن ۵ عدد، عمل حلقه مرتب سازی دوبار تکرار شده است و در $J=3$ ، داده ها مرتب شده اند، برای جلوگیری از چنین حالتی از یک متغیر بنام سویچ استفاده می کنیم.

مثال ۸) الگوریتمی بنویسید که نام خانوادگی ۱۰ دانش آموز را از ورودی گرفته و بصورت صعودی مرتب نماید.

- ۱- شروع
- ۲- $NAME(10)$ را در نظر بگیر : $I \leftarrow 0$
- ۳- $I \leftarrow I+1$
- ۴- $NAME(I)$ را بگیر
- ۵- اگر $I < 10$ بود برو به ۳
- ۶- $SW \leftarrow I : J \leftarrow 0 : I \leftarrow 0$
- ۷- $I \leftarrow I+1$
- ۸- $J \leftarrow J+1$
- ۹- اگر $A(I) > A(J)$ بود ، $A(I) \leftarrow A(J)$ و $A(J) \leftarrow B$: $B \leftarrow A(I)$
- ۱۰- اگر $J < I$ بود برو به ۸
- ۱۱- اگر $SW = 0$ بود برو به ۷
- ۱۲- $M \leftarrow I$
- ۱۳- چاپ کن $NAME(M)$ را

M \leftarrow M+1 - ۱۴۱۵ - اگر $M < 20$ برو به

۱۶ - پایان

در این الگوریتم از متغیری بنام SW به عنوان سویچ استفاده شده است. ابتدا مقدار ۱ را در آن قرار داده و هر وقت عمل جابجایی انجام گیرد، SW برابر صفر می‌گردد. شرط حلقه نیز کنترل می‌کند که آیا مقدار $SW = 0$ شده است یا نه؟ در صورتیکه SW مساوی صفر نباشد، جابجایی صورت نگرفته است و لیست مرتب می‌باشد.

روش مرتب سازی درج یا جایگزینی

در این روش داده‌ها یکی از ورودی خوانده شده و با بقیه عناصر لیست مقایسه می‌شوند در صورتی که عنصر قبلی از داده‌های خوانده شده بیشتر باکتر باشد، با عناصر دیگر لیست مقایسه شده و در محل مناسبی قرار خواهد گرفت. فرض کنید که می‌خواهید ۵ عدد ۹، ۱۳، ۱۵، ۲۵ و ۱۱ را از ورودی خوانده و در یک لیست مرتب شده صعودی قرار دهید.

برای اینکار ابتدا ۹ را خوانده و در اولین عنصر لیست قرار می‌دهیم، سپس عدد ۱۵ را خوانده و با ۹ مقایسه می‌کنیم و الى آخر

مرحله اول : ۹ را به عنوان A(1) در نظر می‌گیریم : $A(1)=9$

مرحله دوم : ۱۵ را خوانده و چون $15 > 9$ است، لیست می‌شود : ۹, ۱۵ و داریم : $A(2)=15$

مرحله سوم : ۱۳ را خوانده و چون $13 > 9$ می‌باشد، ۱۵ و ۱۳ و ۹ و داریم : $A(2)=13, A(3)=15$

مرحله چهارم : ۲۵ را خوانده و چون $25 > 15$ می‌باشد، ۹, ۱۳, ۱۵, ۲۵ و داریم :

مرحله پنجم : ۱۱ را خوانده و چون $25 > 11$ و $15 > 11$ و $13 > 11$ و $9 > 11$ می‌باشد، داریم :

$A(2)=11, A(3)=13, A(4)=15, A(5)=25$

یعنی لیست مرتب شده عبارت خواهد بود از :

9	11	13	15	25
---	----	----	----	----

مثال ۹) الگوریتمی بنویسید که ۲۰ عدد را از ورودی خوانده و به روش درجی مرتب نماید.

۱- شروع

۲- لیست A_1, \dots, A_{20} را در نظر بگیر

۳- $N \leftarrow 1$

۴- اگر $N > 20$ بود برو به

۵- B را بگیر

۶- اگر $N \neq 1$ بود برو به ۸ و گرنه $B \leftarrow A(I)$

۷- $N \leftarrow N+1$ و برو به خط ۴

$M \leftarrow N-1 - 8$ ۹- اگر $A(M) < B$ بود برو به ۱۲ $A(M+1) \leftarrow A(M) - 10$ ۱۱- $M \leftarrow M-1$ و برو به خط ۹۱۲- $A(M+1) \leftarrow B$ و برو به ۷

۱۳- پایان .

روش مرتب سازی SHELL

این روش دارای سرعت بیشتری از دو روش قبلی می‌باشد. زیرا در این روش ابتدا عناصر در محل تقریبی خود قرار گرفته و سپس با بررسی دقیق‌تر در محل واقعی خود قرار می‌گیرند. لازم به ذکر است که در این روش به راحتی می‌توان تعیین نمود که چه عناصری از لیست و با چه فاصله‌ای باید مقایسه شوند. یعنی توسط این روش می‌توان تعیین نمود که به عنوان مثال عناصری که دارای فاصله ۴ می‌باشند، با هم مقایسه شوند و بدین ترتیب با تعداد مقایسه کمتر می‌توان لیست را مرتب نمود.

فرض کنید ۱۰ عدد بصورت زیر در اختیار داریم برای مرتب نمودن صعودی آنها توسط روش SHELL به ترتیب زیر عمل می‌کنیم :

اولین فاصله اختیاری است ولی بعد از آن، فاصله بر دو تقسیم صحیح خواهد شد.

۳ , ۵ , ۱۰ , ۱۵ , ۱۶ , ۱۷ , ۲۰ , ۹۹ , ۹ , ۱۱

ابتدا فاصله را ۴ انتخاب کنید، در نتیجه عنصر اول با پنجم و عنصر دوم با هفتم و ... مقایسه می‌شوند.

۳<17 جابجایی انجام نمی‌شود .

5<20 جابجایی انجام نمی‌شود .

10<99 جابجایی انجام نمی‌شود .

15>9 جابجایی انجام می‌شود . 3,5,10,9,16,17,20,99,15,11

16>11 جابجایی انجام می‌شود . 3,5,10,9,11,17,20,99,15,16

فاصله ۲ می‌شود.

3<9 جابجایی انجام نمی‌شود .

9<20 جابجایی انجام نمی‌شود .

20>16 جابجایی انجام می‌شود . 3,5,10,9,11,17,16,99,15,20

5<11 جابجایی انجام نمی‌شود .

3,5,9,10,11,15,16,17,20,99

مثال ۱۰) الگوریتمی بنویسید که N عدد را خوانده و در لیستی قرار داده و سپس آنها را به روش SHELL مرتب نماید.

۱- شروع

۲- N را بگیر

۳- A_1, A_2, \dots, A_N را در نظر بگیر

۴- $K \leftarrow 1$

۵- اگر $K > N$ بود برویه خط ۹

۶- $A(K)$ را بگیر

۷- $K \leftarrow K+1$

۸- $G \leftarrow \lceil \frac{N}{2} \rceil$

۹- اگر $G > 0$ بود برویه پایان

۱۰- $K \leftarrow G+1$

۱۱- اگر $K > N$ بود برویه خط ۱۸

۱۲- $L \leftarrow K-G$

۱۳- اگر $L < 0$ بود برویه خط ۱۷

۱۴- $M \leftarrow L+G$

۱۵- اگر $A(M) \leftarrow B$ و $A(L) \leftarrow A(M)$ و $B \leftarrow A(L)$ بود ، $A(L) > A(M)$ بود

۱۶- و برویه خط ۱۳ $L \leftarrow L-G$

۱۷- و برویه خط ۱۱ $K \leftarrow K+1$

۱۸- و برویه خط ۹ $G \leftarrow \lceil G/2 \rceil$

۱۹- پایان

پرسش‌های تشریحی

- ۱- علت استفاده از مرتب کردن داده ها را مفصلأً توضیح دهید.
- ۲- هریک از روش‌های مرتب سازی را با ذکر یک مثال بیان کرده و آنها را با هم مقایسه کنید.

مسائل

- ۱- فلوچارت هر یک از مسائل حل شده فصل را به طور کامل رسم نمائید.
- ۲- تعمیم هریک از مثالهای حل شده را ابتدا تشریح نمائید، سپس فلوچارت تعمیم یافته آنها را به طور کامل رسم نمائید. (منظور از تعمیم، یعنی اینکه چگونه می‌توان با تغییراتی در الگوریتم آن را برای داده‌های مختلفی به کار برد).

یادداشت‌های درس

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

.....♦♦

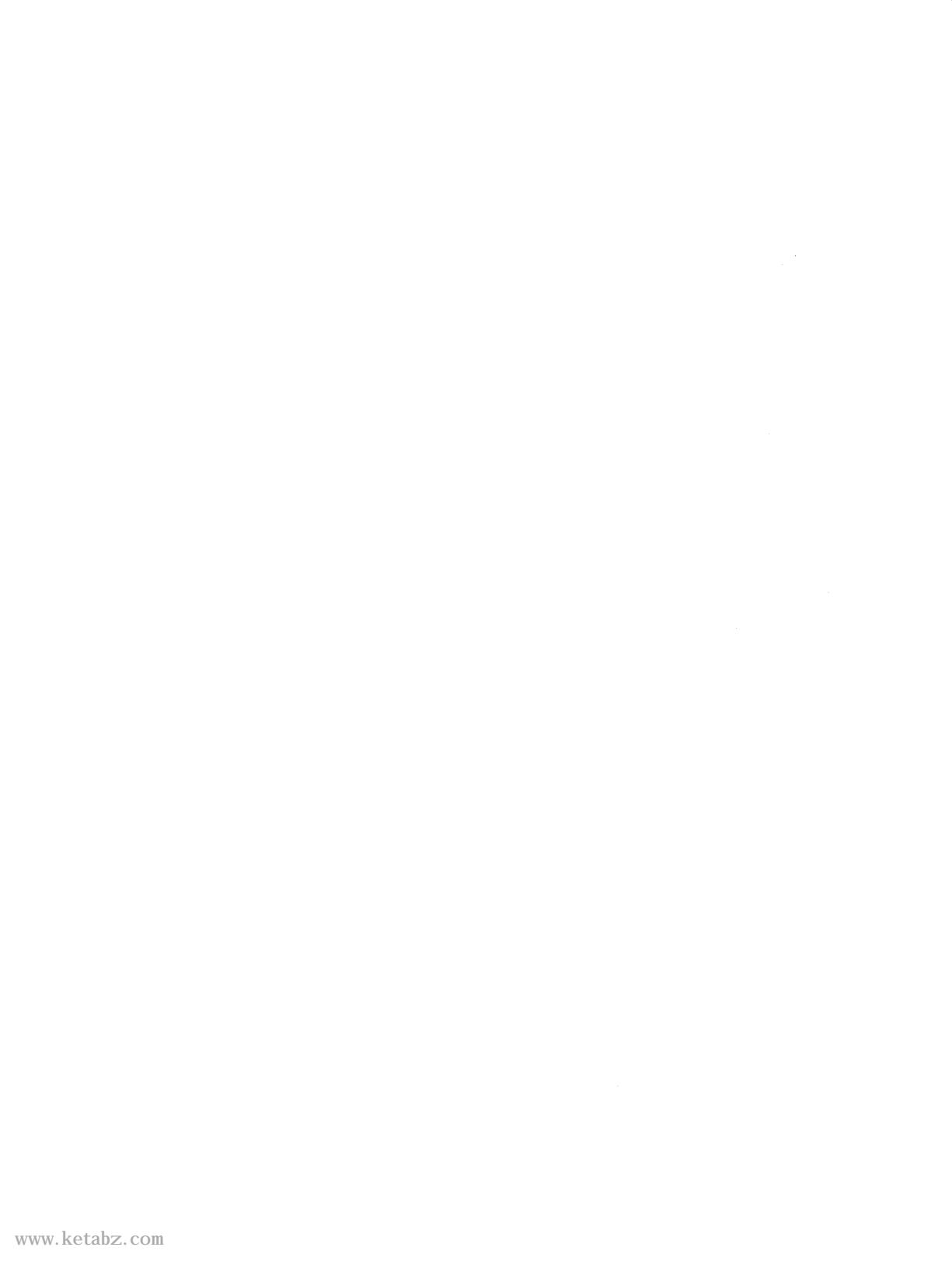
.....♦♦

.....♦♦

.....♦♦

فصل پنجم

ذير الگوريتم



اهداف کلی :

- ۱- آشنائی با مفهوم MODULAR و قسمت بندی یک الگوریتم.
- ۲- آشنائی با زیرالگوریتم و چگونگی بکارگیری آنها در الگوریتم‌ها.

هدفهای رفتاری :

دانشجو پس از مطالعه این فصل باید بتواند :

- ۱- زیر الگوریتم را تعریف نماید.
- ۲- معایب و همچنین فواید استفاده کردن از زیر الگوریتم‌ها را تشریح نماید.
- ۳- با استفاده از مفهوم MODULAR بتواند الگوریتم‌های مختلفی را که شاید حل آنها مشکل یا طولانی است، ساده‌تر حل نماید.
- ۴- توضیحات کافی برای مفهوم شناسه و نیز اسم، برای یک زیر الگوریتم ارایه نماید.
- ۵- بتواند از این به بعد الگوریتم‌های خود را با توجه به زیرالگوریتم‌ها در زمان کمتر، با سرعت بیشتر و به کارگیری نیروی فکری بیشتری حل نماید.

زیر الگوریتمها (SUB ALGORITHM)

بیشتر اوقات با مسائلی سروکار داریم که در حل آنها از قسمت‌های تکراری استفاده شده است، یعنی یک بخش از الگوریتم چندین بار در قسمتهای مختلف الگوریتم تکرار شده است. از سوی دیگر ما همواره در نوشتن الگوریتم‌ها به دنبال بهینه ترین (OPTIMAL) روش هستیم. لذا باید روشی را ابداع کنیم که از دوباره نوشتن یک بخش از الگوریتم در قسمتهای دیگر اجتناب کنیم.

همچنین چون هر یک از الگوریتم‌هایی که می‌نویسیم با یک زبان کامپیوتی برنامه‌ای برای آن نوشته می‌شود، لذا دوباره نوشتن الگوریتم‌ها دارای معایبی است که بعد از معرفی روشی برای اجتناب از این مشکل، آنها را بیان می‌کنیم. ساده ترین روش برای انجام این عمل این است که قسمت تکراری را در یک جا نوشته و یک اسم برای آن در نظر بگیرید، و هر بار در داخل الگوریتم به آن نیاز داشتید با صدازدن اسم آن، از آن قسمت استفاده کنید و دوباره به ادامه الگوریتم اصلی برگردید. ولی آیا فقط داشتن یک اسم کافی است؟

در عمل با نوشتن الگوریتم‌های مختلف مشاهده می‌کنید، با وجود اینکه بعضی از قسمت‌های الگوریتم عیناً تکرار شده است، ولی متغیرهای موجود در آن قسمت‌ها مقادیر دیگری گرفته‌اند، پس بهتر است همراه اسم آن قسمت، متغیرهای آنها را هم صدا زده و به آنها مقدار بدھیم تا عملیات الگوریتم ادامه یابد. برای انجام این عمل از زیرالگوریتم (SUB ALGORITHM) استفاده می‌کنیم.

تعريف زیرالگوریتم : زیرالگوریتم قسمتی از یک الگوریتم است که برای حل بخش خاصی از آن نوشته می‌شود.

دلایل استفاده از زیرالگوریتم‌ها بطور خلاصه عبارتست از :

۱- سبب کم شدن حجم الگوریتم اصلی می‌گردد.

۲- سبب می‌شود که بتوانید با تغییراتی که در آنها حاصل می‌کنید، الگوریتم اصلی را تغییر یا تعمیم دهید که در صورت تکراری بودن این عمل بسیار مشکل خواهد بود.

۳- سبب می‌شود با استفاده از این مفهوم بتوانید یک الگوریتم مشکل را به چند زیرالگوریتم ساده تقسیم نموده و آن را ساده‌تر و سریعتر حل کنید (MODULAR کردن الگوریتم).

۴- همچنین سبب می‌شود نه تنها نوشتن یک الگوریتم که به زیرالگوریتم‌هایی تقسیم شده است ساده‌تر شود، بلکه می‌توانید نوشتن هر زیرالگوریتمی را به یک شخص واگذار نمایید که این عمل خود سبب صرفه‌جویی در وقت می‌گردد.

مثال ۱) الگوریتمی بنویسید که دو عدد طبیعی m و n را دریافت و C''' را محاسبه نماید.

$$\text{داریم: } C_n^m = \frac{m!}{n!(m-n)!} \quad (\text{همواره } m > n).$$

قبل از حل مثال نیاز به توضیحات ریاضی داریم.

تعریف فاکتوریل یک عدد: فرض کنید عدد N یک عدد صحیح مثبت باشد در این صورت فاکتوریل آن را با $N!$ نمایش داده و به طریق زیر آن را محاسبه می‌کنند.

$$\left\{ \begin{array}{l} \text{IF } N=0 \rightarrow N! = 1 \\ \text{IF } N \neq 0 \rightarrow N! = 1 \times 2 \times 3 \times \dots \times (N-1) \times N \\ 5! = 5 \times 4 \times 3 \times 2 \times 1 = 120 \\ 2! = 2 \times 1 = 2 \\ 1! = 1 \\ 0! = 1 \end{array} \right.$$

۱- شروع
۲- را بگیر n و m
۳- اگر $m \leq n$ بود سپس

بلوک الف

$I \leftarrow 1 - 4$
 $P \leftarrow 1 - 5$
 $P \leftarrow P \times I - 6$
 $I \leftarrow I + 1 - 7$
اگر $I \leq m$ بود سپس برو به ۸

$A \leftarrow P - 9$

بلوک ب

$I \leftarrow 1 - 10$
 $P \leftarrow 1 - 11$
 $P \leftarrow P \times I - 12$
 $I \leftarrow I + 1 - 13$
اگر $I \leq (m-n)$ سپس برو به ۱۴

$B \leftarrow P - 15$

بلوک ج

$I \leftarrow 1 - 16$
 $P \leftarrow 1 - 17$
 $P \leftarrow P \times I - 18$
 $I \leftarrow I + 1 - 19$
اگر $I \leq n$ سپس برو به ۲۰

$C \leftarrow P - 21$

$T \leftarrow A / (B \times P) - 22$

بنویس T - ۲۳

پایان - ۲۴

اگر در الگوریتم فوق دقت کنید، مشاهده می‌کنید که سه بلوک الف و ب و ج داریم که دقیقاً مانند هم هستند، با این تفاوت که در بلوک الف، I از یک تا m تغییر کرده و در P ضرب شده و مقدار m! در P ذخیره می‌شود. در بلوک ب، I از یک تا m-n تغییر کرده و در P ضرب شده و مقدار! (m-n)! در P ذخیره می‌شود. در بلوک ج، I از یک تا n تغییر کرده و در P ضرب شده و در P مقدار! n! ذخیره می‌شود. می‌توانستید الگوریتم عمومی برای محاسبه فاکتوریل یک عدد بنویسید که دو عدد را گرفته و فاکتوریل هر دو تای آنها و تفاضل آنها را محاسبه نماید و این عمل با صدا زدن اسم آن الگوریتم که از این به بعد زیرالگوریتم نامیده می‌شود، انجام دهد.

حال که می‌خواهیم از آن استفاده کنیم، باید مشخصات یک زیرالگوریتم را بدانیم. این مشخصات عبارتند از :

- ۱- حتماً باید دارای یک نام باشد. این عمل سبب می‌شود در الگوریتم اصلی هر گاه به آن نیاز داشتید برایتی با صدازدن نام آن، آن را به اجرا در آورده و یا اگر دو زیرالگوریتم داشته باشید، از هم تمیز داده شوند.
- ۲- حتماً متغیرهایی که قرار است در زیرالگوریتم تغییر کنند را به همراه نام آن بیاورید که به آن شناسه می‌گویند .

حال مثال ۱ را با استفاده از یک زیرالگوریتم می‌توانیم بصورت زیر بنویسید :

الگوریتم اصلی :

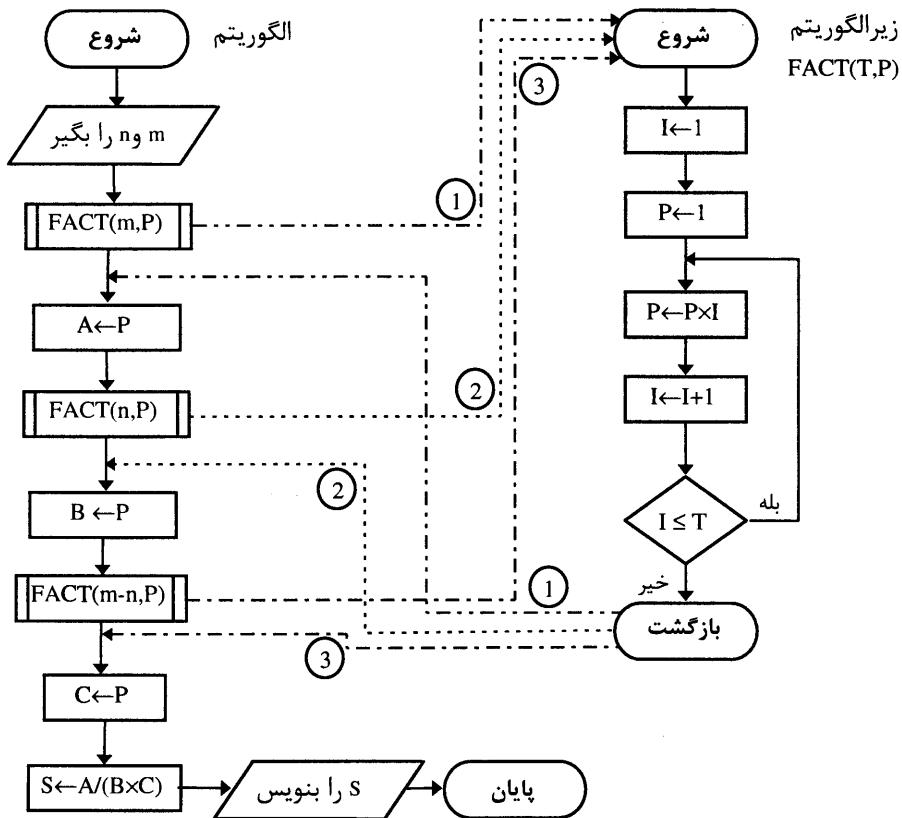
- ۱- شروع
- ۲- m و n را بگیر (فرض m>n)
- ۳- زیرالگوریتم FACT(m , P) را اجرا کن
- ۴- A←P
- ۵- زیرالگوریتم FACT(n , P) را اجرا کن
- ۶- B←P
- ۷- زیرالگوریتم FACT(m-n , P) را اجرا کن
- ۸- C←P
- ۹- S←A/(B×C)
- ۱۰- S را بنویس
- ۱۱- پایان .

ZIRALGORITYM : نام FACT(T,P)

- ۱۰- شروع
- ۱۱- I←1
- P←1
- P←P×I
- I←I+1
- ۶۰- اگر I<=T سپس برو به ۴۰
- ۷۰- برگرد به الگوریتم اصلی
- توضیح : مقداری است که زیرالگوریتم به عنوان ورودی گرفته و P مقداریست که به عنوان خروجی باز می‌گرداند .

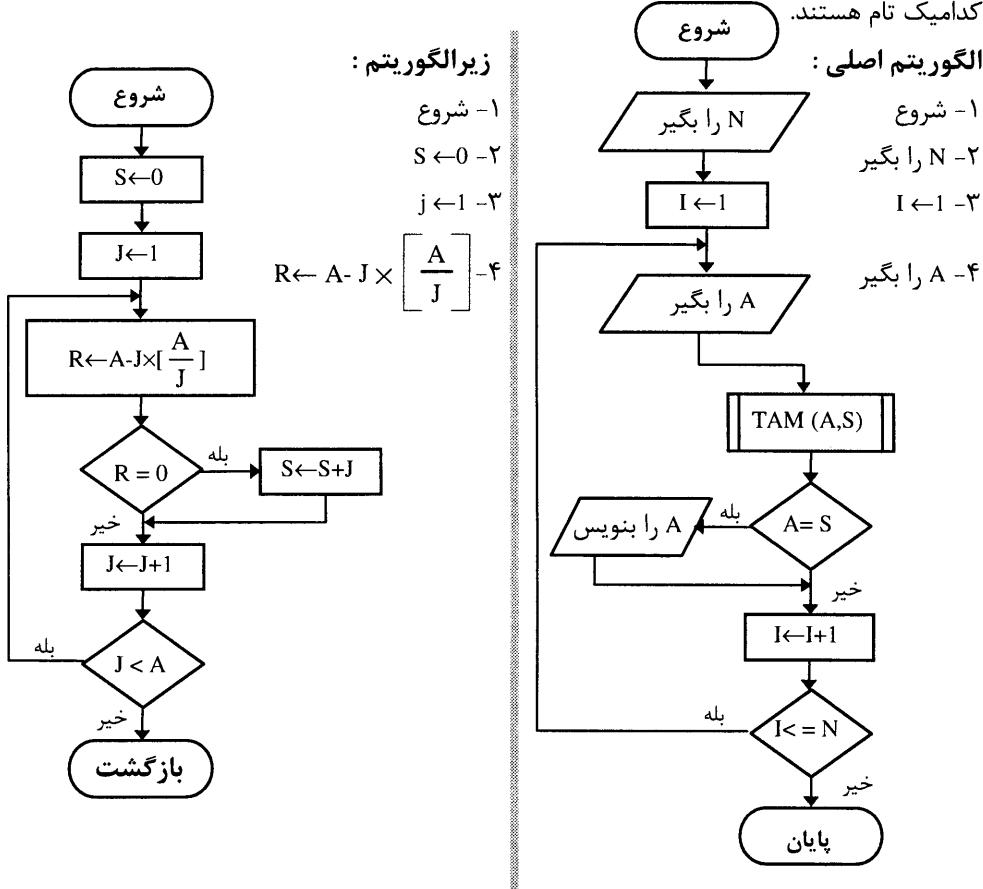
هربار که زیرالگوریتم FACT را صدا می‌زنید، شناسه آنرا تغییرمی‌دهید، در نتیجه برای هر کدام فاکتوریل شان محاسبه شده و بعد از آنکه در زیرالگوریتم به "دستور برگرد به الگوریتم اصلی" رسید، اجرای آن مستقیماً به اولین دستور بعد از دستور ارجاع به زیرالگوریتم برمی‌گردد.

تذکر : در فلوچارت به جای دستور $\text{IF } \dots \text{ THEN } \dots$ یا $\text{IF } \dots \text{ THEN } \dots \text{ ELSE } \dots$ می توانید از علامت استفاده کنید و نام زیرالگوریتم را به همراه شناسه های آن، داخل این شکل بنویسید و همچنین در زیرالگوریتم به جای "برگرد به الگوریتم اصلی " از استفاده کرده و داخل آن دستور بازگشت را بنویسید :



این فصل از کتاب صرف‌آجنبه کاربرد مفهوم زیرالگوریتم را دارد ، به همین دلیل این فصل را فقط با چند مثال به پایان می‌بریم تا با نحوه بکارگیری زیرالگوریتمها آشنا شویم، همچنین مانند مفهوم حلقه‌های تودرتو دریابید که می‌شود در داخل یک زیرالگوریتم، یک زیرالگوریتم دیگر را صدا زد و همین روال را تکرار کنید که با مثالهایی که حل شده است مطلب کاملاً روشن خواهد شد .

مثال ۲) الگوریتمی بنویسید که عدد طبیعی N را سوال نموده، N عدد را یکی یکی دریافت و معین کند کدامیک تام هستند.



-۵ اگر $R=0$ سپس $R=0$

-۶ $J \leftarrow J + 1$

-۷ اگر $J < A$ سپس برو به ۴

-۸ بازگشت .

-۵ TAM(A, S) را اجرا کن

-۶ اگر $S=A$ سپس A را بنویس

-۷ $I \leftarrow I + 1$

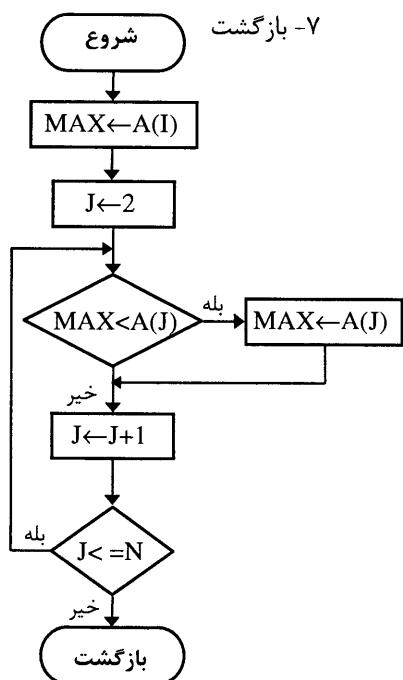
-۸ اگر $I \leq N$ سپس برو به ۴

-۹ پایان .

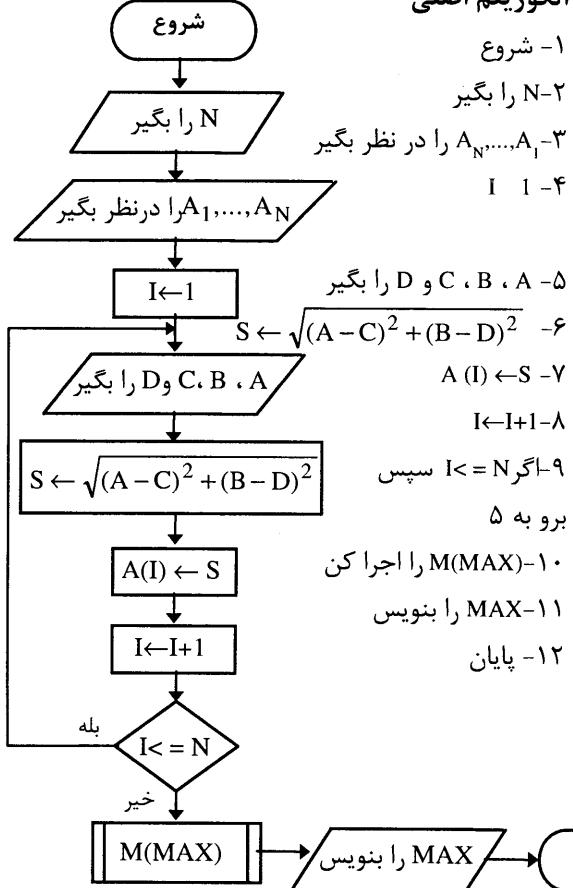
مثال ۳) الگوریتمی بنویسید که N دسته دوتایی زوج مرتب را دریافت و فاصله دو به دوی نقاط زوج ها را محاسبه کرده و درون یک بردار N بعدی با N عنصر ذخیره نماید سپس بزرگترین فاصله را محاسبه و چاپ کند.

زیرالگوریتم M(MAX)

۱- شروع
 ۲- $MAX \leftarrow A(1)$
 ۳- $J \leftarrow 2$
 ۴- اگر $MAX < A(J)$
 $MAX \leftarrow A(J)$
 سپس
 $J \leftarrow J+1$
 ۵- اگر $J \leq N$ سپس برو به ۶
 ۶- بازگشت



الگوریتم اصلی



مثال ۴) الگوریتمی بنویسید که مقدار حقیقی X و طبیعی N را دریافت و مقدار زیر را محاسبه و چاپ کند.

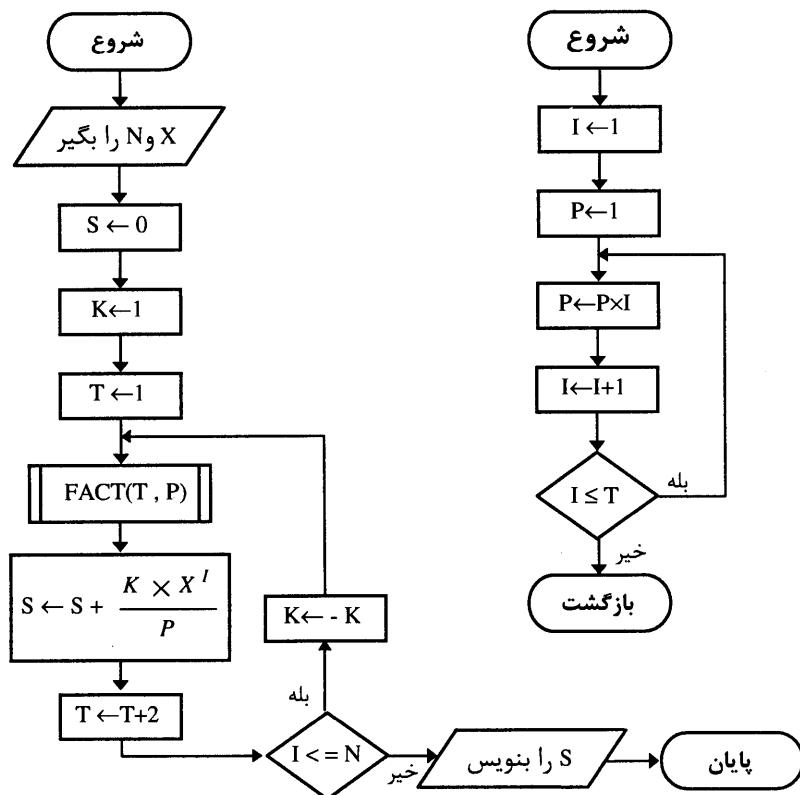
$$\sin(X) = X - \frac{X^3}{3!} + \frac{X^5}{5!} - \dots$$

الگوریتم اصلی :

- شروع
- ۱ را بگیر X و N
- ۲ $S \leftarrow 0$
- ۳ $K \leftarrow 1$
- ۴ $T \leftarrow 1$
- ۵ اجرا کن $FACT(T, P)$
- ۶ $S \leftarrow S + K \times \frac{X^I}{P}$
- ۷ $T \leftarrow T + 2$
- ۸ اگر $I \leq N$ سپس برو به ۵
- ۹ S را بنویس
- ۱۰ پایان

زیرالگوریتم $FACT(T, P)$:

- شروع
- ۱ $I \leftarrow 1$
- ۲ $P \leftarrow 1$
- ۳ $P \leftarrow P \times I$
- ۴ $I \leftarrow I + 1$
- ۵ اگر $I \leq T$ سپس برو به ۴
- ۶ بازگشت



مثال ۵) الگوریتمی بنویسید که عدد طبیعی ($N > 2$) را سوال نموده و ضرایب N سطر از مثلث خیام - پاسکال را محاسبه و چاپ نماید.

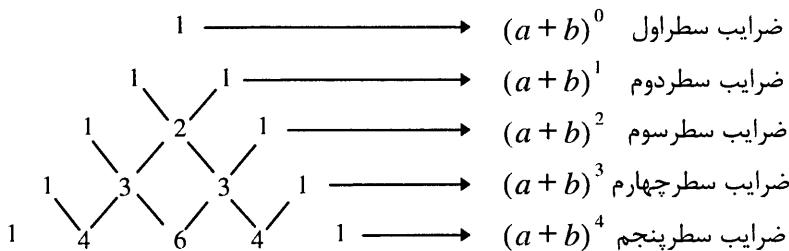
توضیح : فرض کنید می خواهید عبارات زیر را محاسبه کنید.

$$(a+b)^0 = 1$$

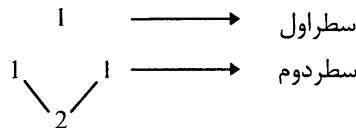
$$(a+b)^1 = a + b = 1a + 1b$$

$$(a+b)^2 = a^2 + 2ab + b^2 = 1a^2 + 2ab + 1b^2$$

اگر بخواهیم این روند را ادامه دهیم، جدول مثلث شکل زیر را خواهید داشت.



این جدول به صورت زیر درست شده است که در سطر اول ۱ و سطر دوم آن ۱ و ۱ است و سطر سوم به طریق زیر است :



ابتدا یک قرار می دهیم ، سپس از سطر بالا دو عدد بالایی را جمع می کنیم که ۲ می شود و در آخر دوباره یک قرار می دهیم.

طبق فرمول بسط ضرایب دو جمله‌ای نیوتن محاسبه می شود که فرمول کلی " $(a+b)^n$ " بصورت زیر است .

$$(a+b)^n = c_0^n a^n + c_1^n a^{n-1} b + c_2^n a^{n-2} b^2 + \dots + c_{n-1}^n a b^{n-1} + c_n^n b^n = \sum_{i=0}^n c_i^n a^{n-i} b^i$$

برای حل مسئله یک زیرالگوریتم می نویسیم که بتواند مقدار c_i^n را محاسبه کند و اسم آنرا x می گذاریم.

همچنین یک زیرالگوریتم دیگر که بتواند مقدار فاکتوریل یک عدد را محاسبه کند که اسم آن را FACT می گذاریم.

زیرالگوریتم FACT(T,P)

۱۱۰۰ - شروع

۱۲۰۰ - $I \leftarrow I - 1$

$P \leftarrow I - 1300$

$P \leftarrow P \times I - 1400$

$I \leftarrow I + 1 - 1500$

۱۶۰۰ - اگر $I \leq T$ سپس برو به ۱۴۰۰

۱۷۰۰ - بازگشت .

حال زیرالگوریتم X را می نویسیم.

۱۰۰ - شروع

FACT(m,p) را اجرا کن - ۲۰۰

$A \leftarrow P - 300$

FACT(n,p) را اجرا کن - ۴۰۰

$B \leftarrow P - 500$

FACT(m-n,p) را اجرا کن - ۶۰۰

$C \leftarrow p - 700$

$S \leftarrow A/(B \times C) - 800$

۹۰۰ - بازگشت .

حال الگوریتم اصلی را بصورت زیر می نویسیم .

۱ - شروع

۲ - N را بگیر

۳ - ۱ را بنویس

۴ - ۱۰ را بنویس

$J \leftarrow 2 - 5$

$K \leftarrow 0 - 6$

X(J,K,J-K,FACT(T,P),S) را اجرا کن - ۷

S را بنویس - ۸

$K \leftarrow K + 1 - 9$

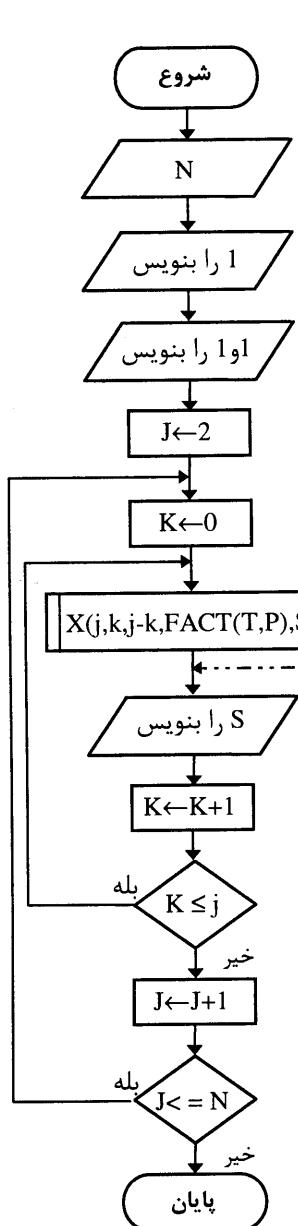
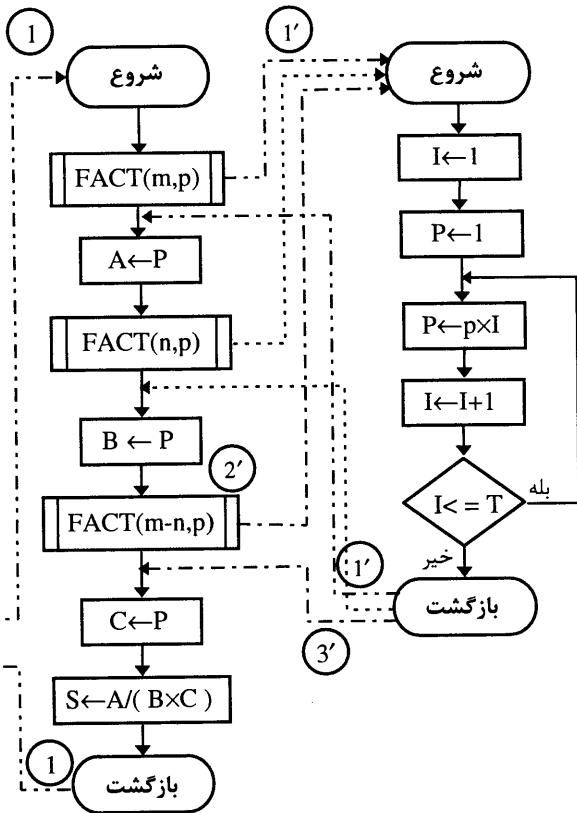
۱۰ - اگر $K \leq J$ سپس برو به ۷

$J \leftarrow J + 1 - 11$

۱۲ - اگر $N \leq j$ سپس برو به ۶

۱۳ - پایان .

حال به ترسیم فلوچارت حاصله می پردازیم.

زیرالگوریتم $X(m,n,m-n,FACT(T,P),S)$ زیرالگوریتم $FACT(T,P)$ 

توضیح: در مثال بالا از زیرالگوریتمی استفاده کردیم که در داخل آن زیرالگوریتم دیگری صدا زده می‌شود.

مثال ۶) الگوریتمی بنویسید که N عدد را یکی یکی سوال نموده و معین کند کدامیک اول هستند. سپس اعداد اولی را که دو رقمی هستند محاسبه و چاپ نماید.

الگوریتم اصلی

- ۱ شروع
- ۲ N را بگیر
- ۳ I←۱
- ۴ A را بگیر
- ۵ Prime(A,T) را اجرا کن
- ۶ digit(A,K) را اجرا کن
- ۷ اگر T=۱ و K=۲ سپس A را بنویس
- ۸ I←I+۱
- ۹ اگر I<=N سپس برو به ۴
- ۱۰ پایان.

زیرالگوریتم (Prime (A,T))

- ۱ شروع
- ۲ J←۲
- ۳ اگر $\frac{A}{J} = \left[\begin{matrix} A \\ J \end{matrix} \right]$ سپس I←۱ و برو به ۷
- ۴ J←J+۱
- ۵ اگر J<A سپس برو به ۳
- ۶ T←۰
- ۷ بازگشت.

مثال ۷) الگوریتمی بنویسید که یک ماتریس ۲ در ۲ را در نظر گرفته و وارون (Inverse) آن را محاسبه و چاپ نماید.

زیرالگوریتم (Inverse (A,B,C,D))

الگوریتم اصلی:

- ۱ شروع
- ۲ $A_{2 \times 2}$ را در نظر بگیر
- ۳ Inverse (A,B,C,D) را اجرا کن
- ۴ I←۱

D←t : A←D : t←A -۳

$$B \leftarrow \frac{-B}{det} -۴$$

$C \leftarrow \frac{-C}{\det} - 5$ $J \leftarrow 1 - 5$
 $A \leftarrow \frac{A}{\det} - 6$ $A(I,J)$ را بنویس
 $D \leftarrow \frac{D}{\det} - 7$ $J \leftarrow J+1 - 7$
 - بازگشت . $J \leq 2$ سپس برو به ۶ - ۸
 $I \leftarrow I+1 - 9$
 ۱۰- اگر $I \leq 2$ سپس برو به ۵
 ۱۱- پایان .

فلوچارت آنرا به عنوان تمرین رسم کنید.

توضیح : اگر $X = \begin{bmatrix} A & B \\ C & D \end{bmatrix}_{2 \times 2}$ باشد ، آنگاه اولاً دترمینان X بصورت زیر محاسبه می شود :

$\det(X) = A \times D - B \times C$ و وارون X ماتریسی 2×2 است ، بصورت زیر :

$$X^{-1} = \frac{1}{\det(X)} \begin{bmatrix} D & -B \\ -C & A \end{bmatrix}$$

مثال ۸) با استفاده از زیرالگوریتم ، الگوریتمی بنویسید که ۱۰۰ عدد رایکی یکی خوانده و اعدادی را چاپ نماید که خودشان به مجموع ارقامشان بخش پذیرند .
در اینجا زیرالگوریتمی مینویسیم که تعداد ارقام یک عدد دلخواه را محاسبه نماید .

زیرالگوریتم SUMDIGIT(N,S)

۱۰- شروع
 $S \leftarrow 0 - ۲۰$
 $R \leftarrow N - 10 \times [\frac{N}{10}] - ۳۰$
 $S \leftarrow S + R - ۴۰$
 $N \leftarrow [\frac{N}{10}] - ۵۰$
 ۳۰- اگر $N > 0$ سپس برو به ۷۰
 ۷۰- بازگشت .

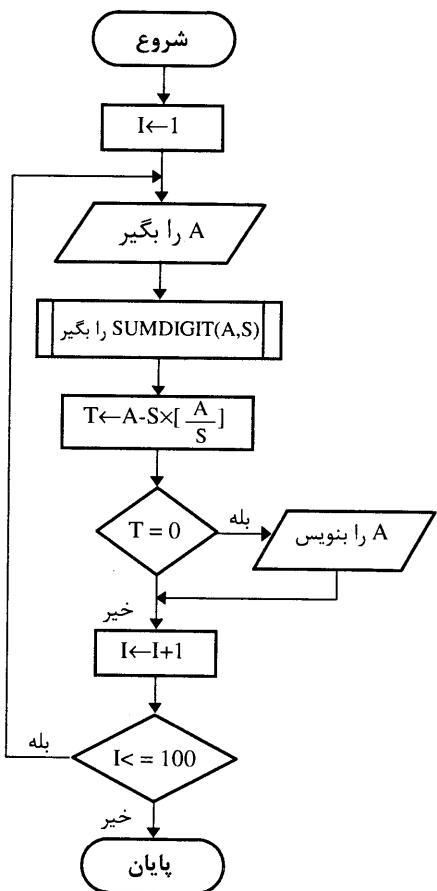
الگوریتم اصلی

۱- شروع
 $I \leftarrow 1 - ۲$
 ۳- A را بگیر
 ۴- $SUMDIGIT(A,S)$ را اجرا کن

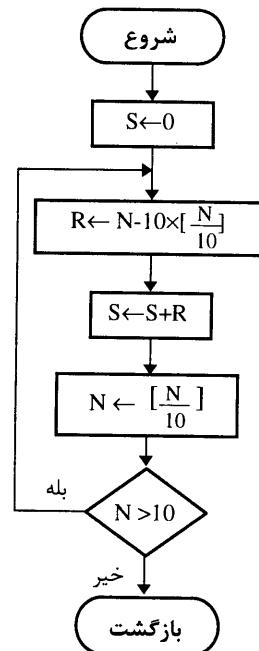
- ۵- $T \leftarrow A - S \times \left[\frac{A}{S} \right]$
- ۶- اگر $T=0$ سپس A را بنویس
- ۷- $I \leftarrow I+1$
- ۸- اگر $I \leq 100$ سپس برویه ۳
- ۹- پایان .

در نتیجه فلوچارت آن به صورت زیر خواهد بود .

الگوریتم اصلی



زیرالگوریتم



مثال ۹) فرض کنید دو بردار n مولفه ای X و Y در فضای n بعدی بصورت زیر باشند.

$$X = (x_1, x_2, \dots, x_n)$$

$$Y = (y_1, y_2, \dots, y_n)$$

منظور از طول بردار n مولفه ای X یعنی عددی که به صورت زیر محاسبه می شود.

$$d = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}$$

الگوریتمی بنویسید که دو بردار n مولفه ای X و Y را درنظر گرفته و حاصل زیر را محاسبه کند.

$$K = \frac{(\text{طول بردار } X) \times (\text{طول بردار } Y)}{(\text{حاصلضرب دوبردار } X \text{ و } Y)^{1/2}}$$

منظور از حاصلضرب داخلی دو بردار یعنی عدد حقیقی بصورت زیر :

$$(X_1, X_2, \dots, X_n) \cdot (Y_1, Y_2, \dots, Y_n) = X_1Y_1 + X_2Y_2 + \dots + X_nY_n$$

ابتدا زیر الگوریتم S را بصورت زیر می نویسیم :

X و Y دو بردار هستند و N طول بردار است که ورودی $S(X, Y, N, P)$ زیرالگوریتم هستند و P مقداری است که محاسبه کرده و به عنوان خروجی باز می گرداند (که همه آنها شناسه هستند).

زیر الگوریتم $S(X, Y, N, P)$

- شروع ۱۰۰

$P \leftarrow 0$ ۲۰۰

$J \leftarrow 1$ ۳۰۰

$P \leftarrow P + X(J) \times Y(J)$ ۴۰۰

$J \leftarrow J + 1$ ۵۰۰

۶۰۰ - اگر $N = J$ سپس برو به ۴۰۰

۷۰۰ - بازگشت .

الگوریتم اصلی

- شروع ۱

۲ - N را بگیر

۳ - X_1, X_2, \dots, X_N را در نظر بگیر

۴ - Y_1, Y_2, \dots, Y_N را در نظر بگیر

۵ - $I \leftarrow I - 1$

۶- را بگیر A,B

X(I)←A -۷

Y(I)←B -۸

I←I+1 -۹

۱۰- اگر I<=N سپس برو به ۶

۱۱- زیر الگوریتم S(X,X,N,P1) را اجرا کن

۱۲- زیر الگوریتم S(Y,Y,N,P2) را اجرا کن

۱۳- زیر الگوریتم S(X,Y,N,P3) را اجرا کن

۱۴- $K \leftarrow (P1 * P2) / \sqrt{P3}$

۱۵- K را بنویس

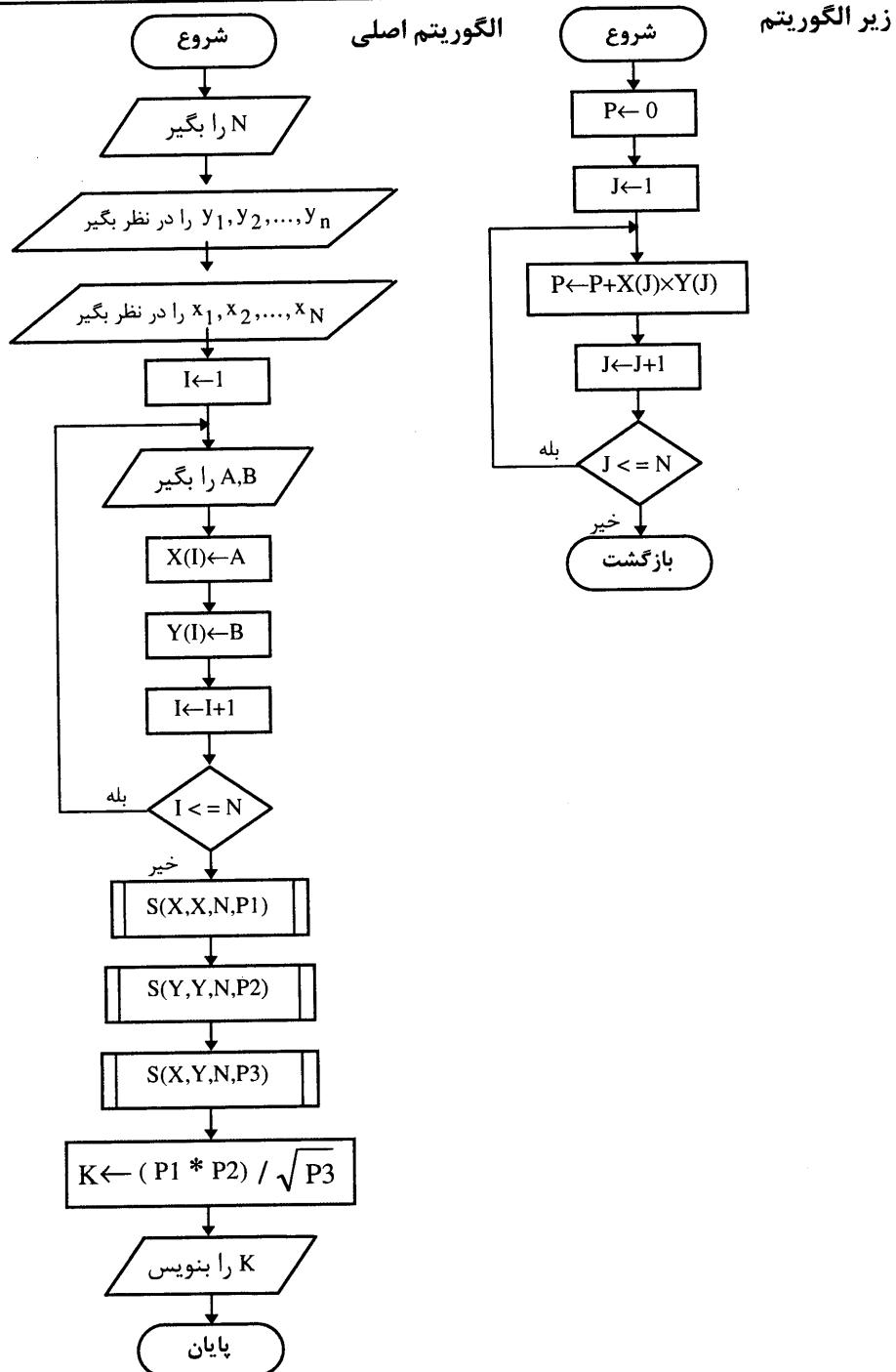
۱۶- پایان .

خط ۱۱ یعنی : زیر الگوریتم S را برای دو بردار X و Y بطول N محاسبه کن و خروجی آن را که در P محاسبه می شود، در P1 ذخیره کن.

همچنان خط ۱۲ یعنی : زیر الگوریتم S را برای دو بردار X و Y بطول N محاسبه کن و خروجی آن را که باز در P محاسبه می شود در P2 ذخیره کن.

خط ۱۳ نیز می گوید زیر الگوریتم S را برای دو بردار X و Y بطول N محاسبه کن و خروجی را نیز در P3 ذخیره کن.

به همین دلیل در بالا زیر الگوریتم را با شناسه های X و Y و N و P تعریف کردیم که X و Y دو بردار بودند که توانستیم هر دو را X یا هر دو را Y در نظر بگیریم.



این فصل را با این امید که مفهوم زیرالگوریتم و نیاز به آن را درک کرده و بتوانید الگوریتم‌های مشکل و طولانی را توسط آنها ساده‌تر بنویسید، به پایان می‌بریم.
توصیه ما به شما این است که سعی کنید بیشتر الگوریتم‌های مشکل و طولانی را که در فصل‌های قبلی بررسی کردیم، توسط زیرالگوریتم‌ها نیز حل نمایید.

پرسش‌های تشریحی

- ۱- مفهوم زیرالگوریتم را با ذکر یک مثال تشریح نمایید.
- ۲- دلایل استفاده از زیرالگوریتم‌هارا فهرست‌وار بنویسید.
- ۳- شناسه را تعریف نمایید.
- ۴- مشخصات یک زیرالگوریتم را بنویسید.
- ۵- علاوه‌تری را که در فلوچارت برای زیرالگوریتم‌ها استفاده می‌شوند، به طور کامل توضیح دهید.

مسائل

- ۱- الگوریتمی بنویسید که اعداد اول کوچکتر از عدد طبیعی N را محاسبه و چاپ نماید.
 - ۲- الگوریتمی بنویسید که N عدد طبیعی را یکی یکی سوال نموده و هر کدام که تام بودند، مجموع ارقامشان را محاسبه و چاپ نماید.
- توجه : عددی تام است که مجموع مقسوم علیه‌های کوچکتر از آن عدد با خود عدد برابر باشد.
- ۳- الگوریتمی بنویسید که یک آرایه یک بعدی را نظر گرفته و عناصر تکراری را در آن حذف نماید.
 - ۴- الگوریتمی بنویسید که دو عدد M و a را که به ترتیب عدد و مبنای عدد می‌باشند و همچنین عدد b را سوال نموده و معادل عدد M را در مبنای b محاسبه و چاپ نماید.