

جزوه پودمان الگوریتم (کتاب کار و فناوری پایه نهم)

الگوریتم چیست؟

قبل از آنکه به بحث الگوریتم و الگوریتم نویسی وارد شویم، ابتدا باید به تعریف مسئله و روش حل مسئله بپردازیم.

مسئله را می توان وضعیتی تعریف کرد که می خواهیم از آن گذر کنیم و به وضعیت مطلوب (جدید) برسیم.

بنابراین :

وضعیت فعلی ← مسئله و مشکل

راههای گذشته از این وضعیت فعلی ← راههای حل مسئله

پیدا کردن راه مناسب، عاقلانه و با صرفه از میان تعداد زیادی راه ← حل مسئله

مسئله و حل مسئله همواره با انسان همراه بوده است زمانی که انسان غار نشین بود از خود می پرسید:

چگونه خود را گرم کنم؟ چگونه شکار کنم؟ چگونه آتش روشن کنم؟

بنابراین حل مسئله مهارتی است که بقای فرد و بقای نسل در گروی آن است.

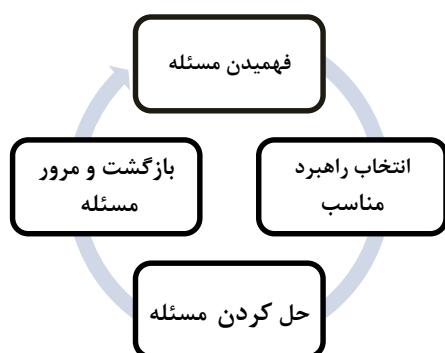
برای حل مسئله باید برای آن برنامه ریزی کرد این کار اعتماد به نفسمان را بالا برده و باعث می شود که در مراحل حل مشکل و مسئله سردر گم نشده و برای اشکالات احتمالی راه حلهای منطقی پیدا کنیم.

تعریف حل مسئله :

به فرایندی که فرد سعی می کند با کمک آن برای مسائل روزمره زندگی خودش، راه حلهای موثری پیدا کند "حل مسئله" می گوئیم.

حل مسئله یک مهارت است پس مانند تمام مهارتهای دیگر مثل شنا کردن، رانندگی، خوشنویسی، آشپزی و... می توان آن را یاد گرفت.

نکته: یادگیری مهارت حل مسئله، میتواند به حل خلاقانه یک مسئله ساده بیانجامد و یا در سطوح بالاتر منجر به ابداع و اختراع شود.



مراحل حل یک مسئله :

گاهی مسائل ساده اند و به راحتی می شود راه حل آنها را تشخیص داد مثل :

محاسبه مساحت یک دایره یا محاسبه مجموع یا حاصلضرب اعداد، شیر آبی که چکه می کند، لامپی که روشن نمی شود. ولی گاهی مسئله را باید به مسائل ریز تر تقسیم کرد تا با حل ریز مسئله ها و کنارهم قرار دادن آنها به صورت گام به گام مسئله کلی را حل کرد.

به روش حل کردن مسائل به صورت گام به گام الگوریتم می گویند.

تعریف الگوریتم :

به هر دستورالعملی که مراحل انجام دادن کاری را با زبان دقیق و ساده و قابل فهم و با جزئیات کافی بیان کند به طوری که ترتیب مراحل و شرط خاتمه عملیات در آن مشخص شده باشد "الگوریتم" می گوئیم.

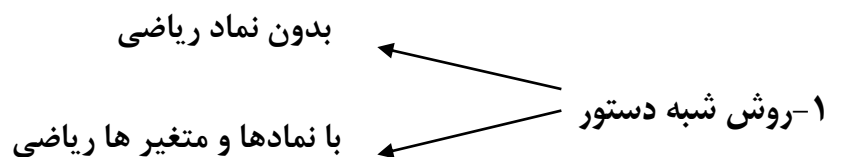
زبان دقیق و ساده و قابل فهم : این ویژگی الگوریتم سبب می شود تا در انجام دستورالعمل ها همواره یک برداشت یکسان حاصل شود در غیر اینصورت برداشت های متفاوت سبب خواهد شد تا دستورالعمل ها نتایج متفاوتی را به وجود آورند.

استفاده از جزئیات کافی : این ویژگی سبب می شود تا دستورالعملها به طور کامل اجرا شود.

ترتیب انجام دستورالعمل ها : یکی از ویژگی های مهم الگوریتم می باشد که اگر به درستی انجام نشود نتیجه کار مشخص نخواهد شد. ترتیب انجام عملیات از بالا به پایین انجام می شود که در صورت نیاز می شود ترتیب اجرای دستورالعمل ها را تغییر داد.

شروع و پایان الگوریتم : در یک الگوریتم باید نقطه شروع دستورالعمل ها و پایان الگوریتم نیز مشخص شود. یک الگوریتم می تواند بیش از یک نقطه پایان داشته باشد.

روش های بیان الگوریتم ها:



۲- روند نما (فلوچارت): بیان تصویری الگوریتم با استفاده از اشکال هندسی استاندارد و تعریف شده که ترتیب اجرای مراحل با خطوط جهت دار (فلش) مشخص می شود روندنما یا فلوچارت گویند.

شکل های هندسی روندنما

شرط	دریافت ورودی، نمایش خروجی	انجام محاسبات، پردازش، مقدار دهی	شروع و پایان عملیات

واژه الگوریتم از کجا آمده است ؟



الگوریتم از نام ریاضیدان و منجم بزرگ ایرانی خوارزمی گرفته شده است. چون خوارزمی در مسائل ریاضی روش مرحله به مرحله و دقیقی بکار می برد که در نهایت به جواب منجر می شد به همین خاطر هر روشی که حل مسائل رو به صورت مرحله به مرحله و دقیق و با جزئیات کافی بیان کند و در پایان به جواب برسد به افتخار او روش الگوریتمی نامیدند. تلفظ لاتینی الخوارزمی به صورت Algorithm یا Algorithmic از آن ناشی شده است.



آیا الگوریتم مختص کامپیوتر یا ریاضی است؟

الگوریتم مختص به عملیتهای ریاضی یا برنامه های پیچیده رایانه ای نیست.

- ✓ اگر شما برای رفتن به مدرسه از یک برنامه خاص و تکراری پیروی می کنید .
- ✓ اگر برای بهبودی بیماری ، روش خاص درمانی را به صورت مشخص و برنامه ریزی شده به کار می برید.
- ✓ اگر ساخت کار دستی را به مراحل جزئی تر تقسیم کرده و...

در همه اینها شما از روش گام به گام یا الگوریتم استفاده کرده اید.

مثال : نمرات ماهانه مریم ۱۸، ۱۹، ۱۶، ۷۵ است. معدل مریم را حساب کنید.

روش شبه دستور با نماد و متغیر های ریاضی

روش شبه دستور بدون نماد ریاضی

۱-شروع

۲- نمره ها را در متغیر های A، B، C قرار می دهیم.

$$S \leftarrow A+B+C$$

$$M \leftarrow S \div 4$$

۵- M معدل مریم است.

۶- پایان

۱-شروع

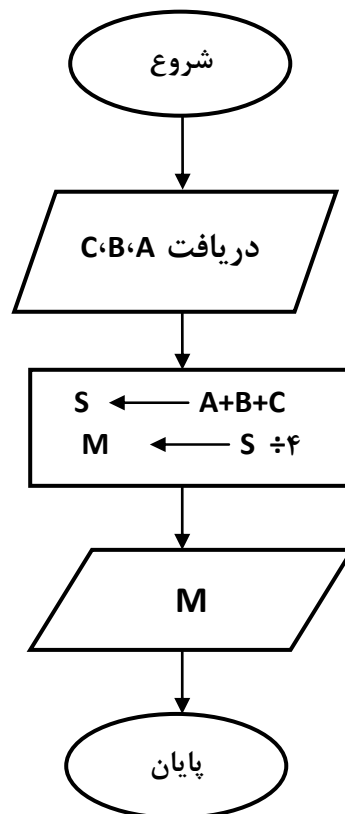
۲-نمرات ۱۸، ۱۹، ۱۶، ۷۵ را باهم جمع می کنیم.

۳- حاصل جمع را به ۴ تقسیم می کنیم.

۴- حاصل تقسیم معدل مریم است.

۵- پایان

روش روند نما (فلوچارت)



نکته :

برای قرار دادن حاصل محاسبه در یک متغیر از نماد \leftarrow استفاده می شود تا با علامت مساوی که برای مقایسه بکار می رود اشتباه گرفته نشود.

برای دریافت چند ورودی به جای چند متوازی الاضلاع می توان از یک نماد متوازی الاضلاع برای دریافت چند ورودی استفاده کرد و همچنین به جای استفاده از چند مستطیل برای چند عملیات می توان از یک نماد مستطیل برای چند عملیات استفاده کرد.

روند نمای حلقوی:

ممکن است در الگوریتم نویسی مواردی پیش بیاید که لازم باشد مراحل از عملیات چندبار تکرار شود، با استفاده از یک شرط می توان به جای چندبار نوشتن این مراحل مسیر روندنما (فلوچارت) به گونه ای طراحی کرد که بتوان آنها را به تعداد موردنیاز تکرار کرد. به تکرار این مراحل " حلقه " می گویند.

دستور تکرار حلقه از ۴ بخش اصلی تشکیل می شود:

۱- شمارنده : متغیری عددی که تعداد دفعات تکرار را کنترل می کند و مقدار آن در هر بار اجرای حلقه افزایش یا کاهش می یابد.

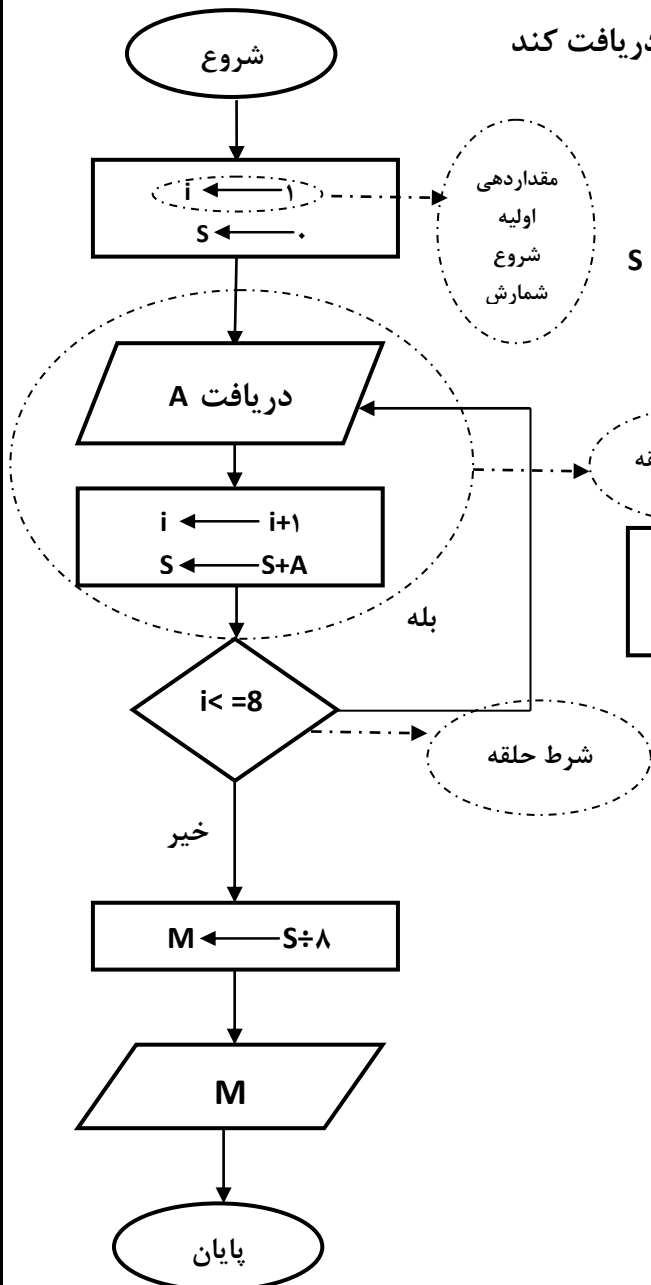
۲- مقدار اولیه : مقداری که قبل از شروع حلقه برای شمارنده تعیین می شود.

۳- شرط حلقه : شمارنده را کنترل کرده پایان تکرار (خروج از حلقه) را مشخص می کند.

۴- دستورات حلقه : دستورات مورد نظر ما که اجرا می شوند.

هرگاه قرار است مراحل از یک الگوریتم چندبار تکرار شود حلقه وجود دارد حتماً باید شرط حلقه به طور صحیح انتخاب شود و گرنه تعداد تکرارها کمتر یا بیشتر از تعداد مورد نیاز می شود یا در مواردی ممکن است با نادرست انتخاب کردن شرط پایان مراحل حلقه بی نهایت بار تکرار شود.

مثال : روند نمای الگوریتمی را رسم کنید که ۸ نمره از ورودی دریافت کند و میانگین آن ها را نمایش دهد؟



در این جا به جای استفاده از ۸ متغیر برای دریافت نمره ها از یک متغیر به نام A استفاده می کنیم. از متغیر i برای شمارش تکرار مراحل ، از متغیر S برای نگهداشتن حاصل جمع مقدار A استفاده می کنیم. می دانیم صفر با هر عددی جمع شود همان عدد می شود پس مقدار اولیه S به عنوان حاصل جمع را صفر قرار می دهیم.

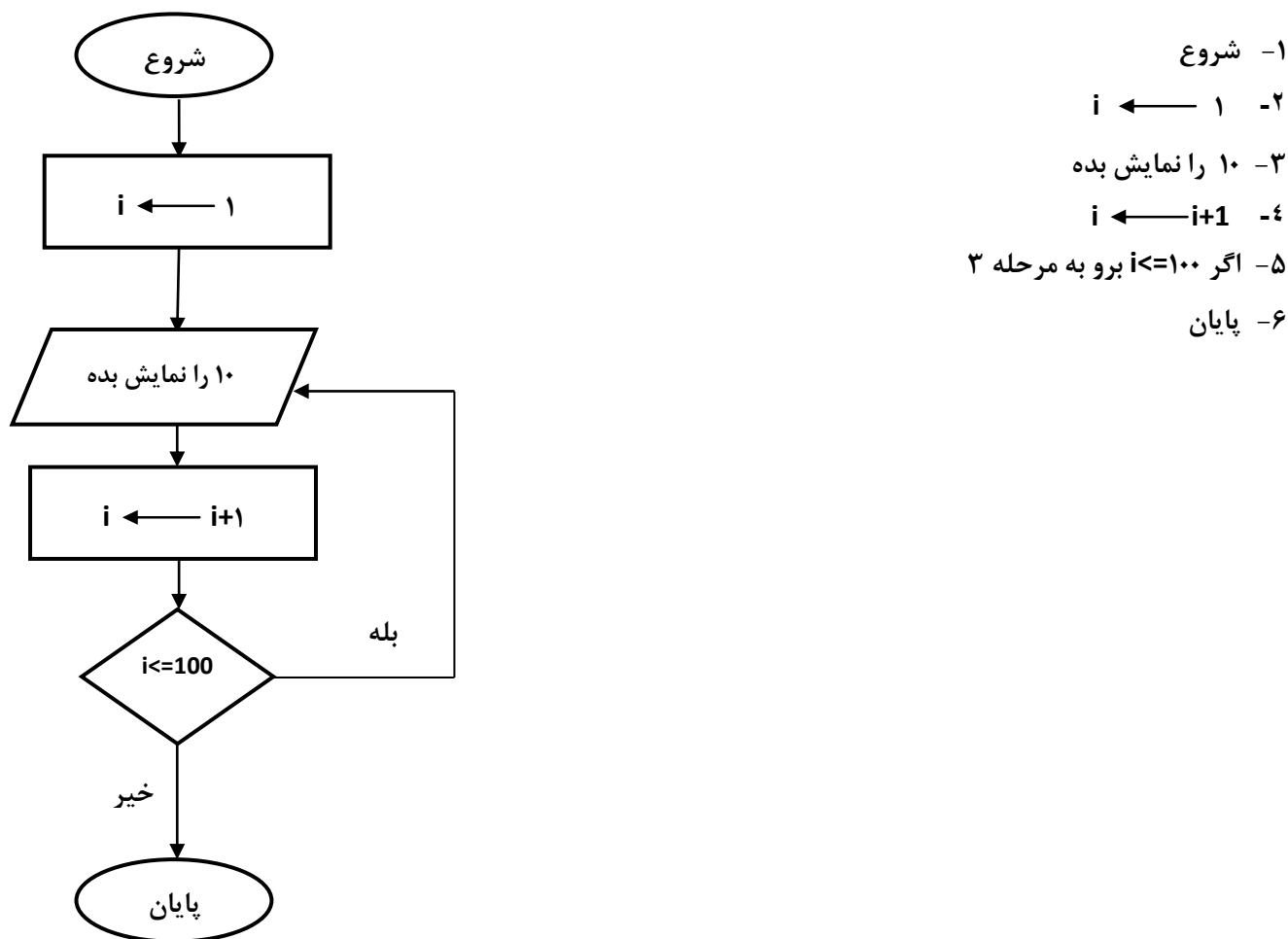
نکته:

هر بار که مقدار جدید در یک متغیر قرار گیرد جایگزین مقدار قبلی می شود.

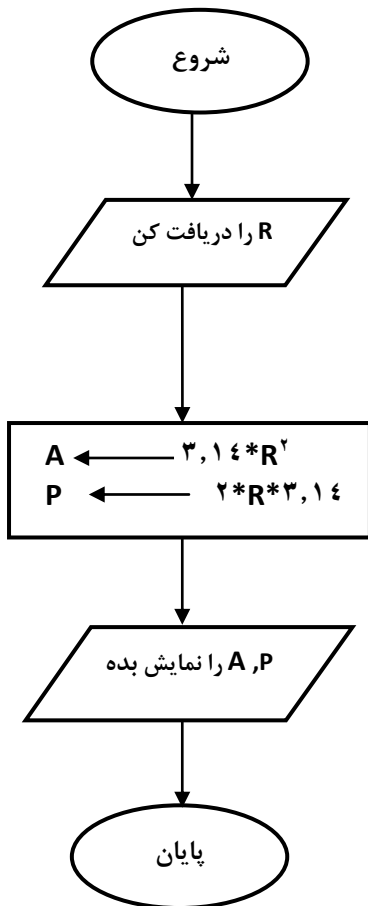
- ۱- شروع
- ۲- $i \leftarrow 1, S \leftarrow 0$
- ۳- A را دریافت کن
- ۴- $i \leftarrow i+1, S \leftarrow S+A$
- ۵- اگر $i < 8$ برو به مرحله ۳
- ۶- $M \leftarrow S \div 8$
- ۷- M را نمایش بده
- ۸- پایان

توضیح مثال با لا : پس از شروع الگوریتم متغیر i را برای شمارش تعداد دفعات تکرار تعریف می کنیم و مقدار اولیه را یک قرار می دهیم و متغیر S را برای نگهداشتن حاصل جمع مقدارهای A تعریف می کنیم و مقدار S را نیز صفر قرار می دهیم چون صفر عضو خنثی است و با هر عددی جمع شود حاصل همان عدد می شود. در مرحله بعد A را دریافت می کنیم یعنی اولین نمره دریافت می شود و در متغیر A قرار می گیرد. در مرحله بعد یک واحد به شمارنده A اضافه می شود , و مجدداً در متغیر i قرار می دهیم و مقدار متغیر A با متغیر S جمع و حاصل در متغیر S قرار می گیرد و بعد شرط حلقه را بررسی می کنیم اگر شمارنده i کوچکتر یا مساوی 8 بود مجدداً دستور 3 (دریافت A) تکرار می کنیم و مجدداً مراحل را طی می کنیم تا به مرحله بررسی شرط برسیم اگر شرط برقرار بود مجدداً دستورات اجرا می شود به مرحله 3 باز می گردیم این فرایند آنقدر تکرار می شود تا شرط حلقه از حالت برقرار بودن خارج شود یعنی شمارنده به عدد 9 برسد در این حالت دستور شرط رفتن به مرحله 3 انجام نمی شود الگوریتم از اجرای حلقه خارج می شود و جمع نمرات دریافتی که در متغیر S قرار گرفته بر تعداد نمرات که عدد 8 است تقسیم می شود و حاصل را در متغیر M قرار می دهیم و در آخر M را نمایش می دهیم .

مثال : روند نمای الگوریتمی را رسم کنید که عدد 10 را 100 بار نمایش دهد.



مثال : روند نمای الگوریتمی را رسم کنید که شعاع دایره را دریافت کند و محیط و مساحت آن را نمایش دهد.



مثال : روند نمای الگوریتمی را رسم ۲۰ عدد را دریافت کند هر عدد را دریافت کرد دو برابر آن را نمایش دهد.

