

الگوریتم :

در بخش گذشته دیدیم که حل مسایل به کمک رایانه چه محدودیت هایی دارد. مجموعه دستورات عملهای قابل انجام توسط یک رایانه محدودند (منظور در هر زمان) و در عین حال دستورات ابتدایی و ساده اند. به عنوان مثال وقتی می خواهید یک عمل ضرب ساده را توسط رایانه های قدیمی انجام دهید، باید مرحله به مرحله روش را تفکیک کرده و هر مرحله را جداگانه بصورت دستور به رایانه اعمال کنید.

1. عدد c را صفر قرار بده

2. به تعداد b دفعه، c را با a جمع کن

یک نمونه از این تفکیک (که البته روش خوبی نیست) برای ضرب روش بالاست.

این نوع عملکرد را می توان به انجام عملیات یک کار پیچیده توسط یک کارگر ساده تشبیه کرد در اینگونه موارد نمی توان مفاهیم پیچیده را توضیح داد و سپس از فرد انتظار داشت که خود با توجه به طبیعت مساله کار را انجام دهد بلکه باید هر مرحله از کارهایی که باید او انجام دهد مجزا و بصورت دقیق (منظور بصورتی که فرد دقیقاً متوجه شود) و جزئی (تفکیک شده) برایش توضیح دهیم. به عنوان مثال دستگاهی را در نظر بگیرید که با گشودن و یا تنگ کردن دهانه مجرای عبور آب سد فشار پشت سد را تنظیم می کند؛ نمی توان به کاربر دستگاه گفت که حداکثر تنش سد a است و تیغه سد زاویه a را با افق دارد، کلید روبرویت بازکننده و مسدود کننده مجرا است و عقربه سمت راست فشار آب در ارتفاع 10 متری را نشان می دهد، مراقب باش تنش از حد مجاز بالاتر نرود!

می توان گفت مثلاً حواست باشد وقتی عقربه سمت راست که فشار آب است به b رسید، کلید روبرویت را باز کن و ... البته در مثالهای انسانی همواره انسان در طول زمان تجربه کسب می کند و می آموزد

ولی به مثال گفته شده از این دید نگاه نکنید.

رایانه برخلاف انسان با لذات قابلیت آموختن ندارد و در نتیجه تا زمانی که روشی برای شبیه سازی آموختن نیافته ایم برخورد با رایانه دقیقاً همان رویه برخورد با "کارگر ساده" است. بنابراین وقتی می خواهیم یک کار نسبتاً بزرگ توسط رایانه انجام شود آنرا به تعداد زیادی کار کوچکتر تقسیم می کنیم که هر بخش برای رایانه قابل اجرا باشد و سپس برای هر بخش دستور مربوطه را وارد می کنیم.

به مجموعه کارهای کوچک ذکر شده در بالا الگوریتم آن کار بزرگتر می گوئیم. البته باید در نظر داشت که الگوریتم نوشتن تنها برای رایانه که کارهای وسیع نمی تواند انجام دهد نیست. این کار (تفکیک کارهای بزرگ به بخشهای جزئی جهت تسهیل در فهم آن) یعنی الگوریتم نویسی علاوه بر کاربرد ذکر شده در کارهای بزرگی که خود ما می خواهیم انجام دهیم هم مفیدند. این عمل موجب نظم دادن به بخشهای کار بزرگی که می خواهیم انجام دهیم می شود. یک مثال کاربردی پیدا کردن اعداد اول 1 تا مثلاً 100 است. یک روش اینست که که بدون تفکیک کار این اعداد را از بقیه جدا کنیم! روش دیگر روش الگوریتمیک زیر است.

1. عدد 2 را در نظر بگیرید.

2. تمامی مضارب عدد در نظر گرفته شده را از مجموعه حذف کن.

3. آیا عدد خط نخورده ای پس از عدد در نظر گرفته شده وجود دارد؟

بله. آن عدد را در نظر بگیرید و به مرحله 2 برو.

خیر. اتمام .

نمونه بالا که کاملاً توسط رایانه هم قابل پیاده سازی است می تواند کار را برای انسان هم ساده کند و به سرعت بیشتری انجام شود.

در علم کامپیوتر دستورات جزئی که در واقع مثل آجرهای ساخت بنای مورد نظر هستند مشخص و محدودند این دستورات جزئی عبارتند از:

الف. انتساب (نسبت دادن عدد یا مقدار به یک نام)

ب. عملیات ریاضی ($\times, \div, +, -$) (پ) حلقه ها (انجام عملیات بصورت تکراری)

ت. دستورات شرطی

مثلاً در الگوریتم یافتن اعداد اول مرحله 1 انتساب مرحله 2 تکرار و شرط و عملیات ریاضی را با هم دارد مرحله 3 شرط است. با این مفهوم به مرور در روند آموزش بیشتر آشنا خواهیم شد.

حال اگر بخواهیم این الگوریتم ها را بصورت نموداری نمایش دهیم باز هم روش استاندارد وجود دارد. به این نوع نمودارها فلوچارت (*flow chart*) به معنای نمودار جریان گوییم. استاندارد فلوچارت چیزی جز شکلهای استاندارد نیست.

شکل بیضی \circ برای شروع و پایان مستطیل \square برای انتساب و عملیات ریاضی، لوزی \diamond برای شرط متوازی الاضلاع $\parallel\text{ogram}$ برای ورودی و خروجی (رابط کاربر) و شکل دو مستطیل برای مجموعه ای از

دستورالعملها که جداگانه نوشته شده (زیر روال) \square

