
مبانی کامپیوتر و برنامه‌سازی با رویکرد الگوریتم و فلوجارت، پیاده‌سازی شده با پایتون

تألیف:

دکتر رمضان عباس نژادورزی
مهندس علی جهانیان بهنمیری
مهندس نسترن شعبی



فن‌آوری نوین

سرشناسه	: عباس نژاد ورزی، رمضان، ۱۳۴۸ -
عنوان و نام پدیدآور	: مبانی کامپیوتر و برنامه‌سازی با رویکرد الگوریتم و فلوجارت، پیاده‌سازی شده با پایتون / تالیف رمضان عباس نژاد ورزی، علی جهانیان بهنمیری، نسترن شعبی.
مشخصات نشر	: بابل: فناوری نوین، ۱۳۹۹.
مشخصات ظاهری	: ۲۴۶ ص.
شابک	: ۷۳۰۰۰۰ ریال: ۳-۲۹-۷۳۹۳-۶۲۲-۹۷۸
وضعیت فهرست نویسی	: فیبا
یادداشت	: کتابنامه.
موضوع	: پایتون (زبان برنامه‌نویسی کامپیوتر)
موضوع	: Python (Computer program language)
موضوع	: برنامه‌نویسی
موضوع	: Computer programming
شناسه افزوده	: جهانیان بهنمیری، علی، ۱۳۶۷ -
شناسه افزوده	: شعبی، نسترن، ۱۳۷۳ -
رده بندی کنگره	: ۷۶/۷۳QA
رده بندی دیویی	: ۰۰۵/۱۳۳
شماره کتابشناسی ملی	: ۷۵۴۲۲۱۶
وضعیت رکورد	: فیبا

@fanavarienovinpub

تلفن: ۰۱۱-۳۲۲۵۶۶۸۷

بابل، کد پستی ۷۳۴۴۸-۴۷۱۶۷

فن آوری نوین

مبانی کامپیوتر و برنامه‌سازی با رویکرد الگوریتم و فلوجارت، پیاده‌سازی شده با پایتون

تألیف: رمضان عباس نژاد ورزی، علی جهانیان بهنمیری، نسترن شعبی.

نوبت چاپ: چاپ اول

سال چاپ: زمستان ۹۹

شمارگان: ۲۰۰

قیمت: ۷۳۰۰۰ تومان

نام چاپخانه و صحافی: دفتر فنی سورنا

شابک: ۳-۲۹-۷۳۹۳-۶۲۲-۹۷۸

نشانی ناشر: بابل، چهارراه نواب، کاظم بیگی، جنب مسجد منصور کاظم بیگی، طبقه اول

طراح جلد: کانون آگهی و تبلیغات آبان (احمد فرجی)

پخش و فروش کتاب‌های چاپی: تهران، تلفن ۰۲۱-۶۶۴۰۰۱۴۴-۶۶۴۰۰۲۲۰

تهران، خ اردیبهشت، نش وحید نظری، پلاک ۱۴۲ تلفکس: ۶۶۴۰۰۱۴۴-۶۶۴۰۰۲۲۰

فهرست مطالب

فصل اول: آشنایی با الگوریتم و برنامه‌نویسی پایتون	۱۷
فصل دوم: فلوجارت، ساختار تصمیم و تکرار	۵۳
فصل سوم: الگوریتم‌های فرعی و توابع	۱۲۳
فصل چهارم: آرایه‌ها	۱۵۳
فصل پنجم: پردازش رشته‌ها	۲۲۶
منابع:	۲۴۶

الگوریتم‌ها و برنامه‌های حل شده در کتاب در یک نگاه

۱. الگوریتم آماده‌سازی نیمرو
۲. الگوریتم و برنامه نمایش "Hello"
۳. الگوریتم و برنامه دریافت نام خودتان و نمایش عبارت "Hello" به همراه نام خودتان
۴. الگوریتم و برنامه خواندن یک عدد و نمایش آن
۵. الگوریتم و برنامه خواندن دو عدد و نمایش حاصل جمع آن‌ها
۶. الگوریتم و برنامه خواندن a و نمایش حاصل $a^3 / (a^2 + a + 1)$
۷. الگوریتم و برنامه خواندن سه عدد و نمایش میانگین آن‌ها
۸. الگوریتم و برنامه خواندن قاعده و ارتفاع متوازی‌الاضلاع و نمایش مساحت آن
۹. الگوریتم و برنامه خواندن شعاع و ارتفاع استوانه‌ای و نمایش حجم و مساحت کل استوانه آن
۱۰. الگوریتم و برنامه خواندن شعاع کره و نمایش مساحت و حجم آن
۱۱. الگوریتم و برنامه خواندن تعداد اضلاع و طول یک ضلع چندضلعی و نمایش مساحت آن
۱۲. الگوریتم و برنامه خواندن وزن آب و محاسبه تعداد مولکول‌های آن
۱۳. الگوریتم و برنامه خواندن حقوق کارمند و نمایش بیمه، مالیات و دریافتی او
۱۴. الگوریتم و برنامه خواندن قیمت کالا در سال قبل، فعلی و نرخ تورم و نمایش قیمت کالا در سال بعد
۱۵. الگوریتم و برنامه خواندن تعداد متخصصین یک شرکت با فرض افزایش ۱۳٫۵ درصد به حقوق آن‌ها، پیش‌بینی هزینه اضافه‌شده در سال بعد
۱۶. الگوریتم و برنامه خواندن یک عدد دورقمی و نمایش مجموع ارقام و مغلوب آن
۱۷. الگوریتم و برنامه خواندن وزن کالا برحسب کیلوگرم و تبدیل آن برحسب گرم
۱۸. الگوریتم و برنامه خواندن حقوق کارمند و نمایش پاداش ۱۵ درصدی حقوق.
۱۹. الگوریتم و برنامه خواندن ارتفاع و قاعده مثلث و نمایش مساحت آن
۲۰. الگوریتم و برنامه خواندن دو عدد و نمایش حاصل جمع، ضرب، تقسیم، تفریق و باقی‌مانده تقسیم صحیح آن‌ها
۲۱. الگوریتم و برنامه خواندن مقدار x و نمایش حاصل عبارت $\frac{1}{x^2+x+1}$
۲۲. الگوریتم و برنامه خواندن مقاومت و جریان یک مدار الکتریکی و نمایش ولتاژ آن
۲۳. الگوریتم و برنامه محاسبه افزایش هزینه تحریر یک شرکت با توجه به خواندن نرخ تورم
۲۴. الگوریتم و برنامه دریافت سرعت اولیه دو چرخه و مدت زمانی که به سرعت k می‌رسد و نمایش شتاب دو چرخه
۲۵. الگوریتم و برنامه خواندن سن به سال و نمایش تعداد ماه، تعداد چند روز و تعداد ثانیه
۲۶. الگوریتم و برنامه خواندن عدد پنج‌رقمی و نمایش بافاصله ارقام آن
۲۷. الگوریتم و برنامه خواندن میزان بنزین مصرفی اتومبیل برحسب لیتر و مسافت طی شده آن به مایل و نمایش مصرف به گالن
۲۸. الگوریتم و برنامه خواندن m و n و نمایش حاصل عبارات $m^2 - n^2$ ، $m^2 + n^2$ ، $m^2 * n$ و $n * m^2$

۲۹. الگوریتم و برنامه خواندن a و b و نمایش حاصل a^2+b^2 و a^3+b^3
۳۰. الگوریتم و برنامه خواندن تعداد کالا و قیمت هر کالا و نمایش مبلغ فروش
۳۱. الگوریتم و برنامه خواندن سن تا به سال و نمایش دقایقی که دقیقه زندگی کرده‌اید
۳۲. الگوریتم و برنامه خواندن مختصات دو نقطه و نمایش فاصله بین آنها
۳۳. الگوریتم و برنامه خواندن x و y و نمایش حاصل عبارت $z = x^3 + 2x^2 + 3y - 5$
۳۴. الگوریتم و برنامه خواندن دو عدد و تعویض محتوی آنها بدون استفاده از متغیر کمکی
۳۵. الگوریتم و برنامه خواندن تعداد صفحات یک کتاب و ظرفیت یک فلش به گیگابایت و نمایش این که این فلش چند تا از این کتاب‌ها را می‌تواند در خودش ذخیره کند
۳۶. الگوریتم و برنامه خواندن مایل و فوت و تبدیل آن به متر و کیلومتر
۳۷. الگوریتم و برنامه خواندن نرخ حقوق به ازای هر ساعت و تعداد ساعت کار کرد یک کارمند و نمایش میزان حقوق او
۳۸. الگوریتم و برنامه میزان موزاییک‌های مورد نیاز برای فرش کردن یک زمین نیاز
۳۹. الگوریتم و برنامه نمایش نقطه تقاطع دو خط $1y = ax + b$ و $1y = 3x + c$
۴۰. الگوریتم و برنامه خواندن یک عدد و n و نمایش مقدار n امین بیت عدد خوانده شده
۴۱. الگوریتم و برنامه خواندن یک عدد و n و یک کردن n امین بیت عدد
۴۲. الگوریتم و برنامه خواندن یک عدد و n و صفر نمودن n امین بیت عدد خوانده شده
۴۳. الگوریتم و برنامه خواندن یک عدد و n و معکوس نمودن n امین بیت عدد خوانده شده
۴۴. الگوریتم و برنامه خواندن دو عدد و تعویض محتوی آنها با استفاده از عملگر یا انحصاری
۴۵. الگوریتم و برنامه خواندن قاعده کوچک، قاعده بزرگ و ارتفاع یک دوزنقه و نمایش مساحت دوزنقه
۴۶. الگوریتم و برنامه خواندن یک عدد و معکوس تمام بیت‌های عدد خوانده شده
۴۷. الگوریتم و برنامه خواندن زاویه به درجه و تبدیل آن به رادیان
۴۸. الگوریتم و برنامه خواندن زاویه به رادیان و تبدیل به درجه
۴۹. الگوریتم و برنامه خواندن درجه حرارت بر حسب سانتی‌گراد و تبدیل آن به فارنهایت
۵۰. الگوریتم و برنامه خواندن شعاع دایره و اختلاف مساحت دایره و مربع محصورکننده آن
۵۱. الگوریتم و برنامه حل معمای گرگ، گوسفند و یک بسته کلم
۵۲. فلوچارت و برنامه محاسبه میانگین سه عدد
۵۳. الگوریتم، فلوچارت و برنامه خواندن اضلاع مثلث و محاسبه مساحت آن
۵۴. الگوریتم، برنامه و فلوچارت خواندن طول و عرض مستطیل و محاسبه محیط و مساحت آن
۵۵. الگوریتم، فلوچارت و برنامه خواندن مقدار x و محاسبه مقدار یک تابع y
۵۶. الگوریتم، فلوچارت و برنامه خواندن سه عدد و تعیین می‌کند که آیا سه عدد تشکیل مثلث می‌دهند یا خیر؟
۵۷. الگوریتم، فلوچارت و برنامه خواندن سه ضلع مثلث و تشخیص می‌دهد آیا این سه ضلع تشکیل مثلث متساوی‌الساقین را می‌دهند یا خیر؟

۵۸. الگوریتم و برنامه نمایش اعداد ۱ تا ۵
۵۹. الگوریتم و فلوجارت نمایش اعداد طبیعی کم تر از ۱۰۰
۶۰. الگوریتم و فلوجارت خواندن یک عدد مثبت و نمایش اعداد ۱ تا عدد خوانده شده
۶۱. الگوریتم و فلوجارت خواندن یک عدد مثبت و نمایش اعداد فرد ۱ تا عدد خوانده شده
۶۲. الگوریتم، فلوجارت و برنامه خواندن اعداد صحیح و مثبت m و n و نمایش کلیه اعداد صحیح بین آن-ها
۶۳. الگوریتم، فلوجارت و برنامه خواندن اعداد صحیح و مثبت m و n و نمایش کلیه اعداد زوج بین آن‌ها
۶۴. الگوریتم، فلوجارت و برنامه خواندن اعداد صحیح و مثبت m و n و نمایش ب.م.م آن‌ها
۶۵. الگوریتم، فلوجارت و برنامه خواندن یک عدد مثبت و نمایش اعداد فرد ۱ تا عدد خوانده شده و مجموع آن‌ها
۶۶. الگوریتم، فلوجارت و برنامه خواندن n و نمایش مجموع سری $sum = 1 - 2 + 3 - \dots \pm n$
۶۷. الگوریتم، فلوجارت و برنامه خواندن x و n و نمایش مجموع سری $sum = x - x^2 + x^3 - x^4 + \dots \pm x^n$
۶۸. الگوریتم، فلوجارت و برنامه خواندن n , a و نمایش a^n
۶۹. الگوریتم، فلوجارت و برنامه خواندن عدد صحیح و مثبت n و نمایش حاصل عبارت زیر:
- $$1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \dots \pm \frac{1}{n}$$
۷۰. الگوریتم، فلوجارت و برنامه خواندن عدد صحیح و مثبت n و نمایش حاصل عبارت $n!$
۷۱. الگوریتم، فلوجارت و برنامه خواندن عدد صحیح و مثبت n و نمایش حاصل از عبارت $1! + 2! + 3! + \dots + n!$
۷۲. الگوریتم، فلوجارت و برنامه خواندن عدد صحیح و مثبت n و نمایش حاصل عبارت زیر:
- $$\frac{1}{1!} - \frac{1}{2!} + \frac{1}{3!} - \frac{1}{4!} + \dots \pm \frac{1}{n!}$$
۷۳. الگوریتم، فلوجارت و برنامه خواندن x و n و نمایش مجموع سری زیر:
- $$sum = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \dots \pm \frac{x^n}{n!}$$
۷۴. الگوریتم، فلوجارت و برنامه خواندن x و n و نمایش مجموع سری زیر:
- $$\cos x \approx 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \dots \pm \frac{x^{2n}}{(2n)!}$$
۷۵. الگوریتم، فلوجارت و برنامه خواندن عدد صحیح و مثبت n و عدد x نمایش حاصل عبارت زیر:
- $$e^x \approx 1 + \frac{x^1}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$$
۷۶. الگوریتم، فلوجارت و برنامه خواندن عدد صحیح و مثبت n و عدد x نمایش حاصل عبارت زیر:
- $$sum = x - \frac{x^2}{2!} + \frac{x^3}{3!} - \dots \pm \frac{x^n}{n!}$$

۷۷. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n و اضلاع n مثلث، سپس محاسبه مجموع مساحت‌های آن‌ها
۷۸. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n و معدل n دانشجو، سپس محاسبه میانگین معدل آن‌ها
۷۹. الگوریتم، فلوچارت و برنامه خواندن تعداد دانشجویان یک کلاس و سپس، نمرات آن دانشجویان را خوانده، در پایان نمایش و بیش‌ترین و کم‌ترین نمره کلاس
۸۰. الگوریتم، فلوچارت و برنامه خواندن n سپس، خواندن n عدد دیگر نمایش تعداد اعداد مثبت و تعداد اعداد منفی بین آن‌ها
۸۱. الگوریتم، فلوچارت و برنامه خواندن اعداد x و n ، سپس خواندن n عدد دیگر و نمایش تعداد دفعات تکرار عدد x ، در آن n عدد
۸۲. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n ، نمایش مقسوم‌علیه‌های آن عدد
۸۳. الگوریتم، فلوچارت و برنامه نمایش جدول ضرب 10×10
۸۴. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n ، سپس، تعیین می‌کند n اول است یا خیر؟
۸۵. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n ، سپس، نمایش کلیه‌ی اعداد اول کوچک‌تر از n
۸۶. الگوریتم، فلوچارت و برنامه خواندن تعداد دانشجویان یک کلاس و نمرات آن دانشجویان، نمایش نمرات اول و دوم کلاس
۸۷. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n (بزرگ‌تر یا مساوی ۲) و نمایش n جمله‌ی اول سری فیبوناچی
۸۸. الگوریتم، فلوچارت و برنامه خواندن صحیح و مثبت n و نمایش تعداد ارقام n
۸۹. الگوریتم، فلوچارت و برنامه خواندن عدد صحیح و مثبت n و نمایش مجموع ارقام n
۹۰. الگوریتم، فلوچارت و برنامه خواندن صحیح و مثبت n و نمایش وارون n
۹۱. الگوریتم، فلوچارت و برنامه‌ای که n را خوانده خروجی زیر را نمایش می‌دهد:

```

1
2 2
3 3 3
4 4 4 4
5 5 5 5 5
6 6 6 6 6 6

```

۹۲. الگوریتم، فلوچارت و برنامه‌ای که خروجی زیر را نمایش می‌دهد:

۷ ۶ ۵ ۴ ۳ ۲ ۱
 ۶ ۵ ۴ ۳ ۲ ۱
 ۵ ۴ ۳ ۲ ۱
 ۴ ۳ ۲ ۱
 ۳ ۲
 ۲
 ۱

۹۳. الگوریتم، فلوچارت و برنامه خواندن عدد n و نمایش حاصل ضرب ارقام غیر صفر آن

۹۴. الگوریتم، فلوچارت و برنامه‌ای که حاصل ضرب ۱۰۰ جمله عبارت زیر را نمایش می‌دهد:

$$s = \frac{1}{2} \times \frac{3}{4} \times \frac{5}{4} \times \dots$$

۹۵. الگوریتم، فلوچارت و برنامه‌ای که دو عدد صحیح x و y را خوانده حاصل ضرب آن‌ها با عمل جمع حساب کرده، نمایش می‌دهد. به‌عنوان مثال، اگر ورودی‌ها اعداد ۲ و ۵ باشند، خروجی به‌صورت زیر خواهد شد:

۹۶. الگوریتم، فلوچارت و برنامه‌ای که دو عدد طبیعی (بزرگ‌تر از ۰) را خوانده، عدد اول را به توان عدد می‌رساند (فقط با عملگر جمع). اگر ورودی‌ها اعداد ۲ و ۱۰ باشند، خروجی به‌صورت زیر خواهد شد:

۹۷. الگوریتم، فلوچارت و برنامه‌ای که n را خوانده ($n > 1$) و مجموع n جمله اول سری زیر را محاسبه کرده، نمایش می‌دهد:

$$s = 1 - \frac{1+3}{2+4} + \frac{1+3+5}{2+4+6} - \frac{1+3+5+7}{2+4+6+8} + \dots$$

۹۸. الگوریتم و برنامه برداشت وجه از ATM

۹۹. الگوریتم خواندن حقوق ناخالص، سن و سابقه کار یک کارگر کارخانه‌ای و نمایش حقوق پرداختی به کارگر مطابق با یکسری قوانین

۱۰۰. الگوریتم و برنامه خواندن n و سپس خواندن n عدد دیگر نمایش میانگین اعداد مثبت و منفی آن‌ها

۱۰۱. الگوریتم و برنامه خواندن n و سپس خواندن تعداد اعضای n خانوار محاسبه مجموع یارانه‌های پرداختی به این خانوارها

۱۰۲. الگوریتم و برنامه خواندن سه عدد مرتب‌سازی آن‌ها به ترتیب صعودی

۱۰۳. الگوریتم و برنامه خواندن سه عدد و نمایش کوچک‌ترین عدد و مکان آن

۱۰۴. الگوریتم و برنامه خواندن n امین روز سال و نمایش تاریخ روز

۱۰۵. الگوریتم و برنامه خواندن شماره روز و ماه و تعیین چندمین روز سال

۱۰۶. الگوریتم و برنامه‌ای که n را خوانده، حاصل سری زیر را نمایش می‌دهد:

$$S_n = \sum_{i=1}^n \frac{i}{i+1} = \frac{1}{2} + \frac{2}{3} + \dots + \frac{n}{n+1}$$

۱۰۷. الگوریتم و برنامه‌ای که n را خوانده، حاصل سری زیر را نمایش می‌دهد:

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

۱۰۸. الگوریتم و برنامه خواندن n، سپس n عدد دیگر و نمایش تعداد اعداد مثبت، صفر و منفی

۱۰۹. الگوریتم و برنامه خواندن دو عدد x و y و نمایش حاصل تقسیم صحیح عدد بزرگ‌تر بر عدد کوچک‌تر با استفاده از تفریق

۱۱۰. الگوریتم و برنامه خواندن دو عدد x و y و نمایش حاصل باقی‌مانده تقسیم صحیح عدد بزرگ‌تر بر عدد کوچک‌تر با استفاده از تفریق

۱۱۱. الگوریتم و برنامه روش حدس کولاتز

۱۱۲. الگوریتم و برنامه خواندن n و سپس خواندن n عدد دیگر، نمایش کوچک‌ترین عدد و تعداد تکرار آن

۱۱۳. الگوریتم و برنامه خواندن تاریخ فعلی و تاریخ تولد یک فرد و محاسبه سن فرد

۱۱۴. الگوریتم و برنامه خواندن صحیح و مثبت n و سپس، ارقام صفر آن را حذف می‌نماید و نمایش می‌دهد.

۱۱۵. الگوریتم و برنامه‌ای که حاصل سری زیر را نمایش می‌دهد:

$$\frac{1 \times 3 \times 5 \times \dots \times 99}{2 \times 4 \times 6 \times \dots \times 100}$$

۱۱۶. الگوریتم و برنامه‌ای که حاصل سری زیر را نمایش می‌دهد:

$$\frac{1 \times 2}{1+2} + \frac{3 \times 4}{3+4} + \frac{5 \times 6}{5+6} + \dots + \frac{99 \times 100}{99+100}$$

۱۱۷. الگوریتم و برنامه‌ای که n را خوانده، سپس n عدد را خوانده، با فرمول زیر واریانس را حساب می‌کند و نمایش می‌دهد:

$$v = \frac{1}{n} \sum x_i^2 - \left(\frac{1}{n} \sum x_i \right)^2$$

۱۱۸. الگوریتم و برنامه نمایش تمام اعداد سه‌رقمی که همه ارقام آن‌ها برابر هستند.

۱۱۹. الگوریتم و برنامه‌ای نمایش تمام اعداد سه‌رقمی که اختلاف بین هر رقم با رقم قبلی و بعدی خودش برابر ۱ است.

۱۲۰. الگوریتم و برنامه‌ای نمایش تمام اعداد پنج‌رقمی که رقم سمت راست آن‌ها صفر و رقم سمت چپ آن‌ها مضرب ۳ است.

۱۲۱. n به توان ۲ برابر با مجموع n تا اولین عدد فرد است. الگوریتم و برنامه‌ای که عدد n را خوانده، به روش بیان‌شده، توان ۲ آن را نشان می‌دهد

۱۲۲. الگوریتم و برنامه خواندن عدد صحیح و مثبت n و نمایش رقمی از آن که بیش‌ترین مقدار دارد

۱۲۳. الگوریتمی که اعداد صحیح و مثبت x و n را از ورودی می‌خواند، حاصل n جمله عبارت زیر را نمایش می‌دهد.

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

۱۲۴. الگوریتم و برنامه خواندن عدد صحیح و مثبت n ، نمایش بزرگ‌ترین رقم آن و تعداد تکرار آن
۱۲۵. الگوریتم و برنامه خواندن عدد صحیح و مثبت n سپس، به هر رقم آن یک واحد اضافه کند (اگر پس از اضافه شدن یک به رقم، آن رقم ۱۰ شد، به جای آن رقم ۰ قرار دهد) و آن را نمایش می‌دهد.
۱۲۶. الگوریتم خواندن n و r و محاسبه حاصل عبارت زیر (بدون استفاده از زیر الگوریتم‌ها).

$$p(n, r) = \frac{n!}{(n-r)!}$$

۱۲۷. الگوریتم و برنامه خواندن n و r و محاسبه حاصل عبارت زیر (با استفاده از زیر الگوریتم‌ها).

$$p(n, r) = \frac{n!}{(n-r)!}$$

۱۲۸. الگوریتم و برنامه خواندن x و n و نمایش مجموع n جمله سری زیر (با استفاده از زیر الگوریتم‌ها).

$$s = x - \frac{x^2}{2} + \frac{x^3}{1 \times 3} - \frac{x^4}{2 \times 4} + \frac{x^5}{1 \times 3 \times 5} - \dots$$

۱۲۹. الگوریتم و برنامه خواندن n و نمایش تمام عدد تام و مجموع اعداد تام ۱ تا n (با استفاده از زیر الگوریتم‌ها)
۱۳۰. الگوریتم و برنامه خواندن n و خواندن n عدد دیگر و نمایش هر عددی که مجموع ارقام آن تام باشد (با استفاده از زیر الگوریتم‌ها)
۱۳۱. الگوریتم و برنامه خواندن n و نمایش تمام عدد اول و مجموع اعداد اول ۱ تا n (با استفاده از زیر الگوریتم‌ها)
۱۳۲. الگوریتم و برنامه خواندن n و خواندن n عدد دیگر و نمایش حاصل ضرب ارقام غیر صفر هر عدد خوانده شده (با استفاده از زیر الگوریتم‌ها)
۱۳۳. الگوریتم و برنامه خواندن n و خواندن n عدد دیگر و نمایش تعداد اعدادی که حداقل دو رقم زوج داشته باشند (با استفاده از زیر الگوریتم‌ها)
۱۳۴. الگوریتم و برنامه خواندن n و خواندن n عدد دیگر و نمایش تعداد اعدادی که همه ارقام آن فرد است (با استفاده از زیر الگوریتم‌ها)
۱۳۵. الگوریتم و برنامه خواندن n و یک رقم d و نمایش تعداد تکرار رقم d در عدد n (با استفاده از زیر الگوریتم‌ها)
۱۳۶. الگوریتم و برنامه چاپ اعداد سه رقمی که مجموع فاکتوریل ارقام آن‌ها برابر با خود عدد باشد. (با استفاده از زیر الگوریتم‌ها)
۱۳۷. الگوریتم و برنامه خواندن عدد صحیح مثبتی و محاسبه مجموع ارقام آن (با استفاده از زیر الگوریتم‌ها)
۱۳۸. الگوریتم و برنامه خواندن یک عدد زوج و نمایش تمام حالت‌های که این عدد زوج برابر مجموع دو عدد فرد اول باشد (با استفاده از زیر الگوریتم‌ها)
۱۳۹. الگوریتم و برنامه تشخیص عدد مثلثی (با استفاده از زیر الگوریتم‌ها)
۱۴۰. الگوریتم و برنامه تشخیص عدد مربعی (با استفاده از زیر الگوریتم‌ها)
۱۴۱. الگوریتم و برنامه تشخیص عددی توانی از سه است یا خیر (با استفاده از زیر الگوریتم‌ها)

۱۴۲. الگوریتم و برنامه نمایش مغلوب عدد (با استفاده از زیر الگوریتم‌ها)

۱۴۳. الگوریتم و برنامه تعویض محتوی دو عدد (با استفاده از زیر الگوریتم‌ها)

۱۴۴. الگوریتم و برنامه مرتب‌سازی سه عدد (با استفاده از زیر الگوریتم‌ها)

۱۴۵. الگوریتم و برنامه چاپ اعداد متحابه بین دو عدد (با استفاده از زیر الگوریتم‌ها)

۱۴۶. الگوریتم محاسبه حاصل عبارت زیر: (با استفاده از زیر الگوریتم‌ها)

$$\frac{x^n (n)^n (n)^{x+2}}{x! n! (x+n)!} \sum_{i=1}^n \frac{x^i i^x x^n}{x! n! (n+2)!}$$

۱۴۷. الگوریتم و برنامه بنگاه‌داری (با استفاده از زیر الگوریتم‌ها)

۱۴۸. الگوریتم و فلوجارت خواندن ۱۰ عدد و نمایش آن‌ها از آخرین عنصر به اولین عنصر

۱۴۹. الگوریتم و فلوجارت خواندن ۱۰، سپس خواندن عدد دیگر (x) و تعیین می‌کند این عدد چند بار در آرایه تکرار شده است

۱۵۰. زیر الگوریتم و تابعی به نام readArray که آرایه a و تعداد عناصر آن n به‌عنوان پارامتر دریافت کرده، n عدد را خوانده، در آرایه قرار می‌دهد.

۱۵۱. زیر الگوریتم و تابعی به نام printArray که آرایه a و تعداد عناصر آن n به‌عنوان پارامتر دریافت کرده، عناصر آرایه a را نمایش می‌دهد.

۱۵۲. الگوریتم و برنامه‌ای که دو آرایه ۱۰ عنصری را تعریف کرده، عناصر آن‌ها را می‌خواند و توسط زیر الگوریتم AddArray عناصر آرایه a را نظیر به نظیر با عناصر آرایه b جمع کرده، و در عناصر آرایه c قرار می‌دهد و در پایان، از طریق زیر الگوریتم printArray عناصر آرایه‌های a، b و c را نمایش می‌دهد.

۱۵۳. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، عناصر آن را می‌خواند. توسط زیر الگوریتم reverse عناصر آن را برعکس می‌کند.

۱۵۴. الگوریتم و برنامه‌ای که دو آرایه ۱۰ عنصری را تعریف کرده، عناصر آن‌ها را می‌خواند و توسط زیر الگوریتم calculate حاصل A-B را در آرایه C قرار می‌دهد و در پایان، از طریق زیر الگوریتم printArray عناصر آرایه‌های A، B و C را نمایش می‌دهد.

۱۵۵. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، $A[0] = 2$ باشد، عناصر آن را با فرمول $A[i] = 2 * A[i-1] + 3 * i$ پر می‌کند.

۱۵۶. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، ۱۰ عدد را خوانده، اگر عدد فرد بود، آن را در آرایه ذخیره کرده، و گرنه ۳ برابر عدد وارد شده را ذخیره می‌نماید

۱۵۷. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس با استفاده از زیر الگوریتم changeSort عناصر آن را از کوچک به بزرگ مرتب کرده، نمایش می‌دهد.

۱۵۸. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس عناصر آن را از کوچک به بزرگ با روش حبابی مرتب کرده، نمایش می‌دهد.

۱۵۹. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس عناصر آن را از کوچک به بزرگ با روش انتخابی مرتب کرده، نمایش می‌دهد.

۱۶۰. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس یک عدد دیگر (x) را می‌خواند و مکان اولین وقوع x را در آرایه پیدا می‌نماید و نمایش می‌دهد.
۱۶۱. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس یک عدد دیگر (x) را می‌خواند و مکان آخرین وقوع x را در آرایه پیدا می‌نماید و نمایش می‌دهد.
۱۶۲. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس دو عدد دیگر (x و k) را می‌خواند و مکان k امین وقوع x را در آرایه پیدا می‌نماید و نمایش می‌دهد.
۱۶۳. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، ۱۰ عدد مرتب (از کوچک به بزرگ) را خوانده، سپس یک عدد دیگر (x) را می‌خواند و با استفاده از زیر الگوریتم binSearch مکان وقوع x را در آرایه پیدا می‌نماید و نمایش می‌دهد.
۱۶۴. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، عناصر آن را می‌خواند. دو عدد دیگر x و y را خوانده، به جای مقدار x در عناصر آرایه a مقدار y را جایگزین می‌کند. در پایان، عناصر آرایه a را نمایش می‌دهد.
۱۶۵. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، عناصر آن را می‌خواند. بیش‌ترین مقدار آرایه و مکان آن را نمایش می‌دهد.
۱۶۶. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، عناصر آن را می‌خواند، کم‌ترین مقدار آرایه و تعداد تکرار آن را نمایش می‌دهد.
۱۶۷. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس یک عدد دیگر (index) را می‌خواند، عنصر مکان index را در آرایه حذف می‌نماید و نمایش می‌دهد.
۱۶۸. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، ۱۰ عدد را خوانده، در آن قرار می‌دهد. عناصر غیر تکراری را نمایش می‌دهد.
۱۶۹. الگوریتم و برنامه‌ای که دو آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن‌ها را خوانده، عناصر مشترک دو آرایه را در آرایه سوم قرار داده و آن را نمایش می‌دهد.
۱۷۰. الگوریتم و برنامه‌ای که دو آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن‌ها را خوانده، سپس اجتماع عناصر دو آرایه را در آرایه سوم قرار داده و آن را نمایش می‌دهد.
۱۷۱. الگوریتم و برنامه‌ای که دو آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن‌ها را خوانده، سپس عناصری که در آرایه اولی باشند و در دومی نباشند، را نمایش می‌دهد.
۱۷۲. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، ۱۰ عدد را خوانده، در آن قرار می‌دهد. سپس عناصر تکراری را نمایش می‌دهد.
۱۷۳. الگوریتم و برنامه‌ای که آرایه‌ای با n عنصر که مقادیر آن‌ها ۰ یا یک را خوانده، ۰ها را به سمت چپ و ۱ها را به سمت راست آرایه منتقل کرده، نمایش می‌دهد.
۱۷۴. الگوریتم و برنامه‌ای که آرایه‌ای را تعریف کرده و مقدار عناصر آن‌ها را خوانده، به طوری که مقدار عناصر آن ابتدا افزایشی هستند (یعنی، عنصر قبلی کم‌تر از عنصر بعدی است) و سپس عناصر آن کاهشی می‌باشند. این برنامه بزرگ‌ترین مقدار این نوع آرایه را نمایش می‌دهد.

۱۷۵. الگوریتم و برنامه‌ای که آرایه‌ای را تعریف کرده و مقدار عناصر آن‌ها را خوانده، و در پایان، آرایه‌ای را ایجاد می‌کند که هر عنصر آن، تعداد عناصری کوچک‌تر از سمت راست خودش را شمارش می‌کند.
۱۷۶. الگوریتم و برنامه‌ای که یک آرایه دوبعدی 5×5 در 5 را تعریف کرده، عناصر قطر اصلی و فرعی را مقدار 1 داده، بقیه عناصر را با صفر پر می‌کند. در پایان، عناصر آرایه دوبعدی را نمایش می‌دهد.
۱۷۷. الگوریتم و برنامه‌ای که یک آرایه دوبعدی 5×5 در 5 را تعریف کرده، عناصر محیطی آن مقدار 1 داده، بقیه عناصر را با صفر پر می‌کند. در پایان، عناصر آرایه دوبعدی را نمایش می‌دهد.
۱۷۸. زیر الگوریتم و برنامه‌ای به نام `readAYD` که یک آرایه دوبعدی n در m را به‌عنوان پارامتر دریافت کرده، عناصر آن را می‌خواند.
۱۷۹. زیر الگوریتم و برنامه‌ای به نام `printAYD` که یک آرایه دوبعدی n در m را به‌عنوان پارامتر دریافت کرده، عناصر آن را نمایش می‌دهد.
۱۸۰. الگوریتم و برنامه‌ای که آرایه 4×4 در 4 را تعریف کرده، 16 عدد را با زیر الگوریتم `readAYD` خوانده، در آن قرار می‌دهد. سپس عناصر زیر قطر اصلی آن را نمایش می‌دهد. الگوریتم و برنامه‌ای که آرایه 4×4 را تعریف کرده، 16 عدد را خوانده، در آن قرار می‌دهد. سپس، کوچک‌ترین عنصر و تعداد تکرار آن را نمایش می‌دهد.
۱۸۱. الگوریتم و برنامه‌ای که یک آرایه دوبعدی 4×4 در 4 را تعریف کرده، 16 عدد را خوانده، در آن قرار می‌دهد. سپس بزرگ‌ترین عنصر، سطر و ستون آن را نمایش می‌دهد.
۱۸۲. الگوریتم و برنامه‌ای که عناصر آرایه‌ای 4×4 را خوانده، بزرگ‌ترین عنصر هر سطر را پیدا کرده، به همراه اطلاعات همان سطر در خروجی چاپ می‌کند.
۱۸۳. الگوریتم و برنامه‌ای که یک آرایه 4×4 تعریف کرده، سپس عناصر آرایه را خوانده و حاصل ضرب عناصر غیر صفر محیطی آرایه را محاسبه می‌کند و نمایش می‌دهد. ع
۱۸۴. الگوریتم و برنامه‌ای که یک ماتریس 4×4 را می‌خواند. اگر مجموع عناصر روی قطر اصلی و فرعی برابر مجموع عناصر محیطی ماتریس باشد، برنامه "Yes"، و گرنه "No" را نمایش می‌دهد.
۱۸۵. الگوریتم و برنامه‌ای که یک ماتریس 4×4 را می‌خواند. اگر همه عناصر آرایه برابر باشند، "Yes"، و گرنه "No" را نمایش می‌دهد.
۱۸۶. الگوریتم و برنامه‌ای که یک ماتریس 4×4 را می‌خواند. اگر عناصر همه عناصر آرایه به‌صورت صعودی مرتب باشند، "Yes"، و گرنه "No" را نمایش خواهد داد.
۱۸۷. الگوریتم و برنامه‌ای که یک ماتریس را خوانده، تشخیص می‌دهد که آیا این ماتریس یک ماتریسی اسپارس است یا نه؟
۱۸۸. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، کاراکترهای مکان‌های فرد آن را نمایش می‌دهد.
۱۸۹. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، مقلوب می‌کند و آن را نمایش می‌دهد.
۱۹۰. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، تمام کاراکترهای کوچک رشته را به بزرگ و کاراکترهای بزرگ را به کوچک تبدیل می‌کند.

۱۹۱. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، تعداد کاراکترهای صدا دار آن را شمارش کرده و نمایش می‌دهد.
۱۹۲. الگوریتم و برنامه‌ای که یک کلمه را دریافت کرده، حرف نقره‌ای آن را نمایش می‌دهد.
۱۹۳. الگوریتم و برنامه‌ای که نامی را خوانده، معادل عددی آن را نمایش می‌دهد.
۱۹۴. الگوریتم و برنامه‌ای که دو رشته را خوانده، رشته اول را در رشته دوم کپی می‌کند.
۱۹۵. الگوریتم و برنامه‌ای که دو رشته را خوانده، رشته دوم را به انتهای رشته اول اضافه می‌کند.
۱۹۶. الگوریتم و برنامه‌ای که یک رشته را خوانده، تعداد تکرار هر حرف آن را نمایش می‌دهد.
۱۹۷. الگوریتم و برنامه‌ای که یک رشته را خوانده تشخیص می‌دهد آیا متجانس است یا خیر؟
۱۹۸. الگوریتم و برنامه‌ای که یک رشته را خوانده، کلیه جاهای خالی از قبیل کاراکتر 'n'، 't' و ' ' در آن را حذف می‌کند و نمایش می‌دهد.
۱۹۹. الگوریتم و برنامه‌ای که یک رشته را خوانده، فضای خالی سمت چپ (ابتدای رشته) را حذف می‌کند و نمایش می‌دهد.
۲۰۰. الگوریتم و برنامه‌ای که رشته‌ای را از ورودی خوانده و بین حروف آن فاصله (Blank) ایجاد می‌کند.
۲۰۱. الگوریتم و برنامه‌ای که یک رشته را خوانده، سپس یک کاراکتر را می‌خواند و بین حروف رشته کاراکتر خوانده شده را قرار داده و رشته را نمایش می‌دهد.
۲۰۲. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، کاراکتری از رشته که بزرگ‌ترین کد اسکی را دارد، نمایش می‌دهد.
۲۰۳. الگوریتم و برنامه‌ای که ابتدا n را خوانده، سپس n رشته را می‌خواند و رشته‌هایی که حرف اول و آخر آن‌ها یکسان باشند را نمایش می‌دهد.
۲۰۴. الگوریتم و برنامه‌ای که ابتدا n را خوانده، سپس n رشته و یک کاراکتر را می‌خواند و رشته‌هایی که حرف اول آن‌ها برابر کاراکتر خوانده شده باشد را نمایش می‌دهد.
۲۰۵. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، سپس دو عدد را می‌خواند و با توجه به اعداد خوانده شده، حروف مکان‌های آن‌ها را جابه‌جا می‌نماید.
۲۰۶. الگوریتم و برنامه‌ای که تعدادی فاصله اضافی به سمت چپ رشته اضافه می‌کند تا اندازه رشته به n تغییر یابد. اگر اندازه رشته بزرگ‌تر یا مساوی n باشد، رشته هیچ تغییر نمی‌یابد.
۲۰۷. الگوریتم و برنامه‌ای که رشته‌ای را خوانده و بعد از هر حرف کوچک، حروف بزرگ آن کاراکتر را اضافه می‌کند.
۲۰۸. الگوریتم و برنامه‌ای که صحت کد ملی وارد شده را بررسی می‌کند
۲۰۹. الگوریتم و برنامه‌ای که صحت شابک وارد شده را بررسی می‌کند
۲۱۰. الگوریتم و برنامه‌ای که صحت شماره کارت اعتباری وارد شده را بررسی می‌کند

مقدمه

امروزه بیش تر مسائلی که با آن روبه رو هستیم توسط رایانه قابل حل هستند. سؤالی که مطرح می‌باشد این است که چگونه رایانه می‌تواند این مسائل را حل کند؟ در پاسخ باید گفت که رایانه به روش گام به گام حل مسئله که به آن الگوریتم گفته می‌شود، این مسائل را حل می‌کند. معمولاً برای حل مشکلات به دنبال ساده‌ترین و سریع‌ترین راه حل‌ها هستیم. سال‌ها است که علم با یافتن پاسخ سؤالات خود و استفاده از آن‌ها در پیشامدهایی که الگوی تکراری دارند، اهداف خود را پیش می‌برد و سریع‌تر از انتظار ما رازهای طبیعت را از دل آن بیرون می‌کشد. یکی از کلمات کلیدی تعریف مفهوم الگوریتم، «حل مسئله» است. الگوریتم، یکی از روش‌های رسیدن به جواب سؤال‌های ما و حل چالش‌های روزمره‌ی زندگی است. عملیات به ظاهر ساده‌ای همچون ضرب دو عدد تا چالش‌های تخصصی‌تر مانند پیش‌بینی وضع هوا در روزهای آینده، همه مبتنی بر الگوریتم‌ها هستند و همواره می‌کوشیم، الگوریتم‌هایی با دقت یا کارایی بیش‌تر برای آن‌ها ارائه دهیم. هوش مصنوعی، یادگیری ماشین، پردازش تصویر، داده کاوی، پیش‌بینی وضع هوا و پیش‌بینی افت و خیز بازارهای مالی، مثال‌های دیگری از کاربرد الگوریتم‌ها است که امروزه بیش از پیش مورد توجه قرار گرفته‌اند.

الگوریتم یکی از روش‌های حل مسئله از طریق ریاضی و منطق است. این روش اولین بار توسط خوارزمی مورد استفاده قرار گرفت. در حال حاضر اساس اجرای بسیاری از برنامه‌های کامپیوتری روش خوارزمی است. نام الگوریتم نیز به احترام دانشمند شهیر ایرانی از کلمه‌ی الخوارزمی گرفته شده است.

اگر بخواهیم معنی الگوریتم را در زمینه ریاضیات و علوم رایانه بررسی کنیم، می‌توان گفت الگوریتم‌ها مجموعه فرایندهایی هستند که به کمک آن‌ها می‌توان بسیاری از مسائل برنامه‌نویسی را به راحتی حل کرد. به عنوان مثال، الگوریتم یک موتور جستجو را در نظر بگیرید. الگوریتم موتور جستجو گوگل به طور ساده این گونه است که عبارت تایپ شده شما را دریافت کرده و آن را در پایگاه داده‌های خود جست و جو می‌کند. سپس صفحات وب مربوطه را پیدا کرده و به شما نشان می‌دهد. این روند کلی از ایجاد سؤال تا رسیدن به پاسخ یک الگوریتم محسوب می‌شود. استفاده از الگوریتم‌ها در کاهش هزینه‌های مالی و زمانی یک پروژه اهمیت زیادی دارد. الگوریتم‌ها با انجام سلسله اقدامات مشخصی و در ازای گرفتن ورودی تعریف شده، نتیجه‌ای مطابق انتظار به ما خواهند داد.

کتاب حاضر اکثر مفاهیم مورد نیاز برای ورود به دنیای برنامه‌نویسی را بیان کرده است. ترتیب مثال‌ها به گونه‌ای است که از ساده‌ترین مثال شروع کرده و هرچه جلوتر می‌رود مفاهیم کاربردی‌تر را در قالب مثال‌های سخت‌تر بیان می‌نماید.

از نقاط قوت این کتاب نسبت به کتاب‌های مشابه موجود در بازار، پیاده‌سازی الگوریتم‌های حل شده با زبان برنامه‌نویسی پایتون است.

از تمامی اساتید و دانشجویان عزیز تقاضا داریم، هرگونه اشکال، ابهام در متن کتاب، پیشنهاد و انتقادات را به آدرس پست الکترونیکی fanavarienovin@gmail.com ارسال نمایند.

مؤلفین

fanavarienovin@gmail.com

۱-۱. سیستم‌های عددنویسی

هر مبنایی به اندازه شماره مبنا، نماد (علامت) دارد. این نمادها از صفر شروع شده و تا عدد مبنا منهای یک ادامه دارند. مثلاً مبنای ۸ دارای ۸ نماد است که از صفر شروع شده تا ۷ ادامه می‌یابد. مبنای ۱۰ دارای ۱۰ نماد (۰ تا ۹) است. مبنای ۱۶ دارای ۱۶ علامت می‌باشد که از ۰ تا ۱۵ ادامه دارد. چون هر نماد باید با یک علامت نشان داده شود، اعداد دورقمی در این مبنا با حروف الفبا نشان داده می‌شوند (مانند جدول زیر):

مبنای ۱۰	۰	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	۱۱	۱۲	۱۳	۱۴	۱۵
مبنای ۱۶	۰	۱	۲	۳	۴	۵	۶	۷	۸	۹	A	B	C	D	E	F

سیستم ده‌دهی، روش عددنویسی است که در محاسبات روزمره استفاده می‌کنیم، با ده علامت ۰، ۱، ۲، ...، ۹ و ارزش‌گذاری متفاوت ارقام در مکان‌های مختلف (یکان، دهگان، صدگان و ...) می‌توانیم همه‌ی اعداد را بخوانیم و بنویسیم. در این سیستم، ارزش هر رقم بستگی به محلی دارد که رقم در آن قرار گرفته است و هر مکان در بخش صحیح عدد، ارزشی معادل ده برابر ارزش مکانی رقم سمت راست را دارد. اما، هر مکان در بخش اعشاری عدد، ارزشی معادل یک‌دهم برابر ارزش مکانی رقم سمت راست را دارد؛ مثلاً در عدد

۴۹۳۷۶.۸۲۳، ارزش‌های ارقام به صورت زیر است:

$$49376.823 = 4 \times 10^4 + 9 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 6 \times 10^0 + 8 \times 10^{-1} + 2 \times 10^{-2} + 3 \times 10^{-3}$$

$$= 40000 + 9000 + 300 + 70 + 6 + 0.8 + 0.02 + 0.003 = 49376.823$$

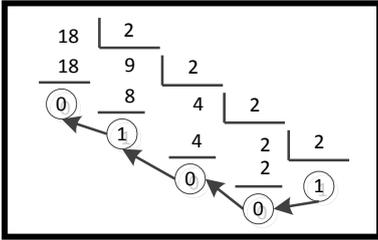
سیستم دودویی، اگر بخواهیم برای نمایش داده‌ها و اطلاعات از سیستم ده‌دهی استفاده کنیم، در پیاده‌سازی سخت‌افزار رایانه‌ها حداقل به ده سیگنال الکتریکی با سطوح متفاوت جهت تشخیص ده رقم مختلف سیستم ده‌دهی از یک‌دیگر نیاز است. این وضعیت، پیاده‌سازی سخت‌افزار را خیلی پیچیده می‌کند که هم هزینه‌ی طراحی سخت‌افزار را افزایش می‌دهد و هم احتمال وقوع خطا را زیاد می‌کند. به همین دلیل، باید دنبال روشی برای عددنویسی بگردیم که پیاده‌سازی آن، به کم‌ترین تعداد سیگنال‌ها نیاز داشته باشد و چون ساده‌ترین وضع سیگنال‌ها، وجود و یا وجود نداشتن آن‌ها است، باید روشی را پیدا کنیم که فقط دو نماد برای نوشتن اعداد در آن به کار روند. اگر بخواهیم مفاهیم مربوط به روش معمولی عددنویسی را برای این دستگاه بازسازی کنیم، باید از دو علامت برای نوشتن ارقام استفاده کنیم و ارزش مکانی هر رقم را دو برابر ارزش مکانی رقم سمت راستش در نظر بگیریم. با این حساب، مثلاً عددی که در این روش به صورت 11001.01

نوشته می‌شود، در روش معمولی عددنویسی به صورت زیر محاسبه می‌شود:

$$1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = 16 + 8 + 0 + 0 + 1 + 0 + 0.25 = 25.25$$

در حالت کلی، داده‌هایی که وارد رایانه می‌شوند، ابتدا به کد دودویی تبدیل می‌گردند و عملیات محاسبه و پردازش در مبنای دو صورت می‌گیرد و در نهایت، در هنگام نمایش در خروجی، نتیجه‌ها به کد ده‌دهی تبدیل

می‌شوند. برای این که عدد مبنای ۱۰ را به مبنای ۲ ببریم، قسمت صحیح و اعشاری عدد را جدا کرده، قسمت صحیح را با تقسیم‌های متوالی بر عدد ۲ به مبنای دو تبدیل می‌کنیم و عمل تقسیم را تا زمانی ادامه می‌دهیم که خارج قسمت از مبنا عدد بزرگ‌تر باشد و بعد، آخرین خارج قسمت را می‌نویسیم و باقی‌مانده‌ها را از انتها به ابتدا می‌نویسیم. اما برای قسمت اعشاری از ضرب متوالی در ۲ استفاده می‌کنیم و عمل ضرب را تا زمانی ادامه می‌دهیم تا بخش اعشار پر شود یا نتیجه ضرب صفر شود.



روشی که برای نوشتن اعداد در مبنای ۲ به کار بردیم، برای هر عدد دیگری (غیر از دو) هم قابل استفاده است؛ مثلاً $(18)_{10} = (10010)_2$.

عملیات پردازش در سیستم دودویی، مثل قوانین کلی محاسبه‌ی معمولی است؛ با این تفاوت که رقم نقلی و قرضی در محاسبات، به جای عدد ۱۰، عدد ۲ است. بنابراین، جمع دو عدد $(1011)_2$ و $(1001)_2$ در مبنای ۲ به صورت زیر انجام می‌شود:

$$\begin{array}{r} 1 \quad 1 \quad 1 \\ + 1001 \\ \hline 10100 \end{array}$$

توجه کنید که مجموع دو رقم ۱ و ۱ در مبنای دو به صورت ۱۰ نوشته می‌شود که ۰ را به عنوان حاصل جمع می‌نویسیم و ۱ را به عنوان رقم نقلی به واحد بعد منتقل می‌کنیم.

جمع عددهای یگرقمی در مبنای دو			
$\begin{array}{r} 0 \\ + 0 \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ + 1 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ + 0 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \quad 1 \\ + 1 \\ \hline 10 \end{array}$

تفریق عدد $(101)_2$ از عدد $(10011)_2$ هم به صورت زیر انجام می‌شود:

$$\begin{array}{r} 1 \\ 022 \\ 10011 \\ - 00101 \\ \hline 01110 \end{array}$$

همان‌طور که در این مثال مشاهده می‌کنید، اگر در طبقه‌ای، رقم بالایی از رقم پایینی کم‌تر باشد، یک واحد از طبقه‌ی سمت چپ (که معادل دو واحد در طبقه‌ی فعلی است) به این طبقه منتقل می‌شود.

تفریق عددهای یگرقمی در مبنای دو

آشنایی با الگوریتم و زبان برنامه‌نویسی پایتون ۱۹

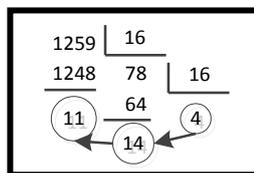
$\begin{array}{r} 0 \\ - 0 \\ \hline 0 \end{array}$	با یک رقم فرضی	$\begin{array}{r} 1 \\ - 0 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ - 1 \\ \hline 10 \end{array}$
---	----------------	---	--

همان‌طور که مشاهده می‌کنید در حالت $1 - 0$ به یک رقم قرضی نیاز است، بنابراین حاصل تفریق ۱ شده اما یک رقم قرضی نیز وجود دارد.

سیستم هشت‌تایی: در سیستم هشت‌تایی، برای نمایش اعداد از ارقام ۰ تا ۷ استفاده می‌شود و مثل دستگاه دودویی، برای تبدیل مبنای ۸ به ۱۰ از عمل ضرب و برای تبدیل از مبنای ۱۰ به ۸ از عمل تقسیم استفاده می‌کنیم. مثال زیر جمع و تفریق در مبنای ۸ را نشان می‌دهد:

$$\begin{array}{r} 11 \\ 605 \\ + 376 \\ \hline 1203 \end{array} \qquad \begin{array}{r} 7 \\ 3814 \\ - 157 \\ \hline 227 \end{array}$$

در این مثال مجموع دو عدد ۵ و ۶ برابر ۱۱ است که از بزرگ‌ترین نماد در مبنای ۸ (یعنی عدد ۷) بزرگ‌تر شده است؛ پس ۸ واحد از آن کم می‌کنیم و یک واحد به طبقه‌ی بالاتر اضافه می‌کنیم و عدد ۳ یعنی حاصل تفریق را در پایین می‌نویسیم و این کار را برای سایر سطوح نیز انجام می‌دهیم. همچنین در تفریق عدد ۷ از عدد ۶ ابتدا ۸ واحد به صورت قرضی به عدد ۶ داده می‌شود تا عدد ۱۴ حاصل شود (بنابراین باید یک واحد از طبقه‌ی سمت چپ کم شود اما چون رقم سمت چپ، صفر است و کم کردن از آن امکان‌پذیر نیست ابتدا یک واحد از عدد ۴ کم می‌کنیم، سپس ۸ واحد به عدد صفر اضافه کرده و در نهایت یک واحد از آن کم می‌کنیم).
سیستم شانزده‌تایی، سیستم شانزده‌تایی، کمی با سیستم‌های قبلی فرق دارد. چون مبنای ده بزرگ‌تر است، ارقام معمولی برای نمایش اعداد در این پایه کافی نیستند. بنابراین، برای نمایش ارقام این سیستم از ارقام ۰ تا ۹ و شش نماد (که به ترتیب A (10)، B (11)، C (12)، D (13)، E (14) و F (15)) استفاده می‌شود. در این سیستم ارزش هر طبقه، ۱۶ برابر ارزش طبقه‌ی سمت راست آن است. پس، عدد ۱۲۵۹ مبنای ۱۰ برابر EB؛ مبنای ۱۶ می‌باشد.



جدول زیر اعداد ۰ تا ۱۵ را در مبنای ۲، ۸ و ۱۶ نشان می‌دهد.

ده‌دهی	باینری	اکتال	هگزا دسیمال	ده‌دهی	باینری	اکتال	هگزا دسیمال
--------	--------	-------	-------------	--------	--------	-------	-------------

۸	10	1000	8	0	0	0000	0
۹	11	1001	9	1	1	0001	1
A	12	1010	10	2	2	0010	2
B	13	1011	11	3	3	0011	3
C	14	1100	12	4	4	0100	4
D	15	1101	13	5	5	0101	5
E	16	1110	14	6	6	0110	6
F	17	1111	15	7	7	0111	7

تبدیل مستقیم توان‌های دو، با توجه به آن که اعداد ۸ و ۱۶ توان‌هایی از ۲ هستند، می‌توان آن‌ها را به روش ساده‌ای به هم تبدیل کرد. در این روش، ابتدا عدد را در مبنای ۲ و سپس به مبنای موردنظر تبدیل می‌کنیم. مثلاً در تبدیل از مبنای ۲ به مبنای ۱۶، هر چهار رقم در مبنای ۲ معادل یک رقم در مبنای ۱۶ است. به همین ترتیب هر سه رقم در مبنای ۲ معادل یک رقم در مبنای ۸ است (مانند جدول فوق).

تبدیل مبنای ۲ به ۱۶، هر چهار رقم مبنای ۲ را با هم یک دسته در نظر گرفته (بخش صحیح را از سمت راست چهار رقم چهار رقم جدا کرده، اما، بخش اعشاری را سمت چپ چهار رقم چهار رقم می‌کنیم و دسته-ای که از چهار رقم کم‌تر باشد، آن قدر صفر اضافه می‌کنیم تا چهاررقمی گردد) و معادل هر دسته را طبق جدول به دست آورده و جایگزین می‌نماییم. به‌عنوان مثال، مقدار 10111110101.0001100011 به زیر دسته‌بندی می‌کنیم:

$$0101 \ 1111 \ 0101 \ . \ 0001 \ 1000 \ 1100$$

سپس هر دسته را جداگانه از مبنای ۲ به ۱۰ تبدیل می‌کنیم و به ازای اعداد از ۱۰ بالاتر معادل مبنای ۱۶ آن را قرار می‌دهیم:

$$010111110101.0001100011 \rightarrow 5 \ 15 \ 5.1 \ 8 \ 12 \rightarrow 5F5.18C$$

تبدیل مبنای ۱۶ به ۲، هر رقم مبنای ۱۶ را جداگانه به مبنای ۲ تبدیل می‌کنیم و به ازای هر رقم مبنای ۱۶ باید ۴ رقم در مبنای ۲ قرار دهیم:

$$(7DE.4B)_2 = (0111, 1101, 1110.0100, 1011)_2$$

مکمل یک، برای به دست آوردن مکمل یک عدد باینری کافی است تمام اها به ۰ و همه ۰ها به ۱ تبدیل شوند. به‌عنوان مثال، مکمل یک عدد 10010011 برابر با 01101100 است.

مکمل دو، برای به دست آوردن مکمل دو عدد باینری باید مکمل یک را حساب کرده، یک واحد به مکمل یک اضافه نمود. به‌عنوان مثال، برای محاسبه مکمل دو 10010011 مکمل یک را به دست آورده، که برابر با 01101100 است. اکنون به مقدار 011001100 یک واحد اضافه می‌کنیم تا 011001101 به دست آید. روش دیگر به دست آوردن متمم دو این است که تا اولین ۱ را از سمت راست به همان صورت می‌نویسیم. بقیه ارقام باقی‌مانده ۰ها به ۱ و ۱ها به ۰ تبدیل می‌گردند. مثلاً، متمم دو مقدار 10001010 برابر با 01110110 است.

نمایش اعداد منفی، معمولاً آخرین بیت (سمت چپ‌ترین بیت) به‌عنوان بیت علامت در نظر گرفته می‌شود که ۱ باشد عدد منفی است. برای به دست آوردن مقدار عدد کافی است عدد را متمم دو کرده و آن را به مبنای ۱۰ تبدیل نماییم. اکنون این عدد را در یک منفی ضرب می‌کنیم. به‌عنوان مثال، مقدار 11111010 معادل منفی 00000110 (متمم دو 11111010) است (یعنی ۶-).

۲-۱. حل مسئله

حل مسئله فرآیند پیچیده از تفکر است. روش‌های مختلفی از قبیل پنج چرا، روش پولیا، تریز و غیره برای حل مسئله وجود دارد. در این بخش روش پولیا را شرح می‌دهیم. در سال ۱۹۴۵ جورج پولیا برای اولین بار روش چهار مرحله‌ای را برای حل مسئله تدوین کرده است. این مراحل عبارت‌اند از:

۱. فهمیدن مسئله (شناخت مسئله)، شناخت و داشتن فهم درست از یک مسئله با مطالعه دقیق آن رخ می‌دهد. در این مرحله به سؤالات زیر پاسخ داده می‌شود:

✚ چه می‌خواهیم؟ مجهول‌ها چیست؟

✚ چه داریم؟ داده‌های مسئله چه هستند؟

✚ با چه شرایطی مواجه هستیم؟

حل مسئله، فرآیندی است که ورودی آن داده‌ها و خروجی آن تعیین مجهول‌ها است.

۲. طراحی نقشه، پیچیده‌ترین بخش حل مسئله است. در این بخش باید به سؤال زیر پاسخ داده شود:

✚ با توجه به شرایط مسئله، ارتباط بین داده‌ها و مجهول‌ها چیست؟

پس از یافتن پاسخ، نقشه طرح شده در قالب الگوریتم یا فلوچارت بیان می‌شود.

۳. اجرای نقشه، نقشه برای رسیدن به جواب اجرا می‌شود. برای افزایش دقت و سرعت می‌توان نقشه را با یک زبان برنامه‌نویسی پیاده‌سازی کرده و در رایانه اجرا نمود.

۴. بازنگری، در این مرحله درستی جواب بررسی شده و در صورت امکان برای بهبود و توسعه الگوریتم تلاش می‌شود. بررسی درستی الگوریتم‌ها موضوعی است که نیاز به ابزارهای پیشرفته ریاضی دارد.

۳-۱. شناخت مسائل و ارائه راه حل مناسب برای آنها

جهت ارائه راه حل مناسب برای یک مسئله، بهترین کار بررسی آن (تحلیل و طراحی مسئله) است. جهت نیل به این هدف، سه عامل مهم باید در نظر گرفته شود که عبارت‌اند از:

۱. مقادیر معلوم (فرضیات مسئله): مقادیری هستند که در اختیار مسئله قرار می‌گیرند (همان فرضیات مسئله نام دارند) که برای نیل به هدف مسئله (خروجی) مورد نیاز می‌باشند.

۲. محاسبات (پردازش‌ها): برای رسیدن به نتایج مورد نظر معمولاً لازم هستند تا عملیاتی (رابطه‌های) را بر روی مقادیر معلوم (فرضیات) انجام دهید. قسمت اعظم این عملیات معمولاً از طریق فرمول‌های مختلف قابل انجام است.

البته محاسبات می‌توانند با توجه به روابط منطقی که بین مقادیر معلوم و خواسته‌های مسئله (خروجی) وجود دارند، انجام شوند.

۳. **خواسته‌های مسئله (مجهولات یا خروجی)**، مقادیری هستند که معمولاً از طریق انجام عملیات روی مقادیر معلوم حاصل می‌گردند. البته مجهولات می‌توانند از روابط منطقی که در حل مسئله دخالت می‌نمایند، نیز به وجود آمده و مورد استفاده قرار گیرند.

به عنوان مثال، فرض کنید بخواهیم محیط و مساحت یک مستطیل را محاسبه کنیم. برای حل این مسئله و ارائه راه‌حل مناسب با روش ارائه شده، ابتدا مقادیر معلوم (مقادیر ورودی) برای حل این مسئله را مورد توجه قرار می‌دهیم؛ همان‌طور که می‌دانید برای محاسبه محیط و مساحت هر مستطیل باید **طول و عرض** آن را در اختیار داشته باشیم، بنابراین، طول مستطیل (x) و عرض مستطیل (y) به عنوان داده‌های مورد نیاز برای حل مسئله کافی هستند.

محاسباتی که برای به دست آوردن محیط و مساحت مستطیل لازم هستند، در واقع فرمول‌های زیر خواهند بود:

$$P = (x + y) \times 2 = 2 \times (\text{عرض} + \text{طول}) \quad (\text{محیط مستطیل})$$

$$S = x \times y = \text{عرض} \times \text{طول} \quad (\text{مساحت مستطیل})$$

همان‌طور که مشاهده می‌کنید، از طریق فرمول‌های بیان شده، روابط بین داده‌های ورودی و نتایج مورد نظر به خوبی مشخص می‌گردد. یعنی از طریق این روابط ریاضی خواسته‌های مسئله (مجهولات یا خروجی) که همان محیط (P) و مساحت (S) مستطیل هستند در متغیرهای متناظرشان ذخیره می‌گردند.

۴-۱. مراحل برنامه‌نویسی

برنامه‌نویسی دارای مراحل زیر است:

۱. **شناخت مسئله**، که شامل مراحل زیر می‌باشد:

✚ شناخت فرضیات و داده‌های ورودی مسئله (Data)

✚ تشخیص خواسته‌ها یا مجهولات مسئله

✚ تعیین ارتباط بین فرضیات و مجهولات

۲. **ارائه طرح یا نقشه حل مسئله**

روش‌های مختلفی برای حل مسئله وجود دارد. برخی از این روش‌ها عبارت‌اند از:

✚ طراحی الگوریتم

✚ طراحی فلوچارت

۳. نوشتن برنامه با یکی از زبان‌های برنامه‌نویسی (نظیر C، پاسکال، C++، C#، پایتون، جاوا یا GO).

۵-۱. الگوریتم چیست؟

وقتی می‌خواهید کاری را انجام دهید، به‌ویژه اگر آن کار پیچیده باشد، ابتدا، باید روش و مراحل انجام آن کار را مشخص نمایید. به‌عنوان مثال، فرض کنید مسئول برگزاری جشنی هستید، در این صورت باید خودتان را برای انجام کارهای لازم جهت برگزاری این جشن آماده کنید. در آن صورت چه می‌کنید؟ مطمئناً لیستی از کارهایی که باید انجام دهید، تعیین کرده، ترتیب انجام آن‌ها را مشخص می‌کنید. یعنی، روش و مراحل انجام کار را به‌دقت تنظیم خواهید کرد. چنانچه این کار را به نحو کامل و بدون نقص انجام دهید که تحت هر

شرایطی با هر نوع امکانات و خصوصیات بتوان از این رویه استفاده نمود، اکنون توانسته‌اید الگوریتم برگزاری مراسم را طراحی کنید. حتی می‌توانید این الگوریتم را به دیگران بدهید تا آن‌ها نیز بتوانند از این الگوریتم برای برگزاری جشن‌شان استفاده کنند.

الگوریتم، عبارت است از مجموعه‌ای از دستورالعمل‌ها که مراحل انجام کاری را به زبان دقیق با جزئیات کافی بیان نماید، ترتیب اجرای دستورات و شرط خاتمه آن مشخص باشد.

برای این که با مفهوم الگوریتم آشنا شویم، دستورالعمل استفاده از تلفن همگانی را در نظر بگیرید. این دستورالعمل در زیر آمده است:

۱. گوشی را بردارید.
۲. یک سکه در داخل تلفن قرار دهید.
۳. منتظر شنیدن بوق آزاد باشید.
۴. شماره را گرفته، صحبت کنید.
۵. در پایان، گوشی را در جایش قرار دهید.

مراحل بیان‌شده، دستورالعمل استفاده از تلفن همگانی است (الگوریتم نیست). زیرا:

۱. مرحله ۲ به زبان دقیق بیان‌نشده است. یعنی، استفاده‌کننده تلفن نمی‌داند چه سکه‌ای را در داخل تلفن قرار دهد. برای این که این مرحله به یک دستورالعمل الگوریتم تبدیل شود، این دستورالعمل باید به صورت زیر تغییر یابد:

👉 یک سکه ۲۵۰ یا ۱۰۰ ریالی سالم در داخل تلفن قرار دهید.

۲. شرط خاتمه الگوریتم مشخص نشده است. زیرا، اگر تلفن خراب باشد، در مرحله ۳ هیچ‌گاه بوق آزاد شنیده نمی‌شود. بنابراین، استفاده‌کننده تلفن تا کی منتظر شنیدن بوق آزاد باشد. این مرحله را به صورت زیر تغییر دهید تا به یک مرحله الگوریتم تبدیل شود:

👉 ۳۰ ثانیه منتظر شنیدن بوق آزاد باشید. چنانچه بوق آزاد را نشنیده‌اید، گوشی را در جایش قرار دهید. مراحل ۱ تا ۳ را اجرا کنید. چنانچه ۳ بار این مراحل را انجام داده‌اید و بوق آزاد را نشنیده‌اید، گوشی را در جایش قرار دهید.

بنابراین، الگوریتم استفاده از تلفن همگانی به صورت زیر است:

۱. شروع
۲. گوشی را بردارید.
۳. یک سکه ۱۰۰ یا ۲۵۰ ریالی سالم در داخل تلفن قرار دهید.
۴. ۳۰ ثانیه منتظر شنیدن بوق آزاد باشید، چنانچه بوق آزاد را نشنیده‌اید، گوشی را در جایش قرار دهید.
- مراحل ۱ تا ۴ را اجرا کنید. اگر ۳ مرتبه این مراحل را انجام داده‌اید و بوق آزاد را نشنیده‌اید، گوشی را جایش قرار دهید و دنبال تلفن دیگر بگردید.
۵. شماره‌گیری نموده و صحبت کنید
۶. در پایان گوشی را جایش قرار دهید.

۱-۶. انواع دستورالعمل‌ها در الگوریتم

بیان گردید که الگوریتم مجموعه‌ای از دستورالعمل‌ها است. انواع دستورالعمل‌هایی که می‌توانند در الگوریتم به کار روند عبارت‌اند از:

۱. **دستورالعمل‌های ورودی**، برای دریافت داده‌های ورودی از کاربر به کار می‌روند و معمولاً برای نمایش و اجرا آن‌ها در الگوریتم از واژه‌های از قبیل "خوانده"، "بخوان"، "دریافت کن"، "بگیر" استفاده می‌گردد. به‌عنوان مثال، x و y را بخوان یا x و y را دریافت کن، نمونه‌ای از این دستورالعمل‌ها هستند.

۲. **دستورالعمل‌های خروجی**، برای نمایش اطلاعات خروجی (مقادیر ثابت‌ها یا متغیرها) یا پیام‌های موردنیاز جهت راهنمایی کاربر روی صفحه‌نمایش یا اجرای آن‌ها در الگوریتم از عبارتی نظیر "به خروجی ببر"، "چاپ کن" یا "نمایش بده" به کار می‌رود. به‌عنوان مثال، نمونه‌ای از این دستورالعمل‌ها عبارت‌اند از: ۱. "Enter two numbers" را نمایش بده ۲. "P" را چاپ کن، ۳. "حاصل عبارت $x+y+z$ را به خروجی ببر".

۳. **دستورالعمل‌های محاسباتی**، روش ارائه، استفاده از فرمول‌ها، انجام عملیات ریاضی و محاسباتی را تعیین می‌نمایند. در این دستورالعمل‌ها برای انتساب به‌جای علامت = از علامت فلش (\leftarrow) استفاده می‌شود. برای این منظور، در سمت راست فلش یک عبارت و در سمت چپ آن یک متغیر قرار می‌گیرد. نمونه‌هایی از دستورالعمل‌های محاسباتی عبارت‌اند از:

$$1. P \leftarrow (x+y) \times 2$$

$$2. S \leftarrow x \times y$$

متغیرهای p و s ، مکان‌هایی هستند که برای ذخیره‌سازی و نگه‌داری انواع داده‌ها به کار می‌روند.

۴. **دستورالعمل‌های شرطی**، بعضی مواقع نیاز است مقادیر داده‌های ورودی، اطلاعات خروجی، متغیرها و غیره باهم مقایسه شوند تا عملیاتی را در الگوریتم هدایت کرده و دستورالعمل‌های خاصی را اجرا کرده و دستورالعمل‌های دیگری را اجرا نکنند. برای این منظور از ساختارهای زیر استفاده می‌شوند:

✚ **اگر شرط (ها) آنگاه دستورالعمل (ات)**

✚ **اگر شرط (ها) آنگاه دستورالعمل (ات) وگرنه دستورالعمل (ات)**

در ساختار اول، ابتدا شرط یا شرط‌های جلوی **اگر** بررسی می‌شوند، در صورتی که نتیجه بررسی درست (True) باشد، دستور یا دستورات جلوی **آنگاه** اجرا می‌گردند، وگرنه (اگر شرط یا شرط‌ها نادرست (False) باشند)، بدون این که دستورالعمل‌های جلوی **آنگاه** اجرا شوند، دستورالعمل پس از **اگر** اجرا می‌شود.

اما، در ساختار دوم، ابتدا **شرط یا شرط‌های جلوی اگر** بررسی می‌شوند، چنانچه نتیجه ارزیابی شرط یا شرط درست (True) باشد، دستور یا دستورات پس از **آنگاه** اجرا می‌گردند. در ادامه دستورالعملی که پس از دستورالعمل **اگر** قرار دارد، اجرا خواهد شد، در غیر این صورت (اگر نتیجه ارزیابی شرط یا شرط‌ها نادرست (False) باشد)، دستور یا دستورات جلوی **وگرنه** اجرا می‌گردند. سپس، دستورالعملی که پس از دستور شرطی قرار دارد، اجرا می‌شود.

۵.

با دستورالعمل‌های شرطی در فصل دوم بیش‌تر آشنا خواهید شد.

دستورالعمل‌های تکرار (حلقه‌ها)،

گاهی لازم است تا مجموعه‌ای از دستورالعمل‌ها به دفعات تکرار شوند. در این موارد از دستورالعمل‌های تکرار (حلقه‌ها) استفاده می‌شود.

دستورالعمل‌های حلقه (تکرار) از بخش‌های مختلفی تشکیل می‌شوند که عبارت‌اند از:

✚ **شمارنده حلقه**، یک متغیر عددی است که تعداد دفعات تکرار دستورالعمل‌ها را در حلقه کنترل می‌کند. مقدار شمارنده در هر بار اجرای حلقه **افزایش** (حلقه‌های با شمارنده صعودی یا افزایش) یا **کاهش** (حلقه‌های با شمارنده نزولی یا کاهشی) می‌یابد.

✚ **مقدار اولیه شمارنده**، مقدار اولیه حلقه قبل از شروع حلقه تعیین می‌شود و به وسیله آن می‌توان مقدار اولیه را برای شمارنده حلقه تعیین کرد (یعنی، در هنگام شروع حلقه، این مقدار در شمارنده حلقه قرار می‌گیرد).

✚ **شرط حلقه**، برای کنترل تعداد دفعات حلقه باید از یک شرط استفاده کرد. شرط موجود در حلقه، نقطه پایان تکرار دستورالعمل‌ها را در حلقه مشخص می‌کند و باید به گونه‌ای تنظیم کرد تا از ایجاد حلقه بی‌نهایت (نامحدود) جلوگیری شود. برای ایجاد شرط در یک حلقه می‌توان از دستورالعمل‌های شرطی استفاده کرد.

✚ **دستورات حلقه (بدنه حلقه)**، دستورالعمل‌هایی هستند که در داخل حلقه تکرار می‌شوند. این دستورالعمل‌ها با توجه به نیاز مسئله انتخاب می‌شوند.

✚ **گام افزایش یا کاهش شمارنده**، مقادیری هستند که پس از اجرای دستورات حلقه (بدنه حلقه) باید به شمارنده حلقه افزوده (برای حلقه‌های با شمارنده صعودی یا افزایشی) یا کاسته (برای حلقه‌های با شمارنده‌های نزولی یا کاهشی) شوند.

با دستورالعمل‌های حلقه تکرار و الگوریتم‌های مربوط به آن در فصل دوم بیش‌تر آشنا خواهیم شد.

مثال ۱-۱. الگوریتمی که مراحل آماده‌سازی نیمرو را بیان می‌کند.

۱. شروع
۲. ماهی‌تابه را بردارید.
۳. در ماهی‌تابه روغن بریزید.
۴. اجاق را با حرارت ملایم روشن نمایید.
۵. منتظر داغ شدن روغن بمانید.
۶. پس از داغ شدن روغن، تخم‌مرغ را در ظرف دیگر بشکنید.
۷. در صورت سالم بودن تخم‌مرغ آن را ماهی‌تابه بریزید.
۸. به تخم‌مرغ نمک و فلفل اضافه نمایید.
۹. تا زمانی که نیمرو به اندازه کافی بپزد، صبر کنید.
۱۰. اجاق را خاموش نمایید.

۱۱. نيمرو را در بشقاب قرار دهيد.

۱۲. پايان.

همان طور که در اين الگوريتم مشاهده مي کنيد، هر الگوريتم يك نقطه شروع و يك نقطه پايان دارد.

۷-۱. تفکر الگوريتمیک

آيا مفهوم الگوريتم فقط در دنياي برنامه‌نويسي تعريف مي‌شود؟ آيا تنها برنامه‌نويس براي طراحي و حل برنامه‌اش بايد از الگوريتم استفاده کند؟ يا اين که براي انجام تمام کارها، مي‌توان از ايده الگوريتمي استفاده نمود؟

در جواب اين سؤالات بايد بگوئيم که الگوريتم يك مفهوم عام و فراگير است. يعني، ايده الگوريتم، علاوه بر آن که در برنامه‌نويسي به کار مي‌رود، مي‌تواند در زندگي روزمره نيز استفاده شود.

يك برنامه‌نويس خوب، علاوه بر آن که به الگوريتم برنامه‌هايي که مي‌نويسد فکر مي‌کند، با الگوريتم زندگي مي‌کند. او تمام زندگي خودش را با الگوريتم انجام مي‌دهد. با الگوريتم فکر مي‌کند. يعني، تفکر او حتي در امور ساده زندگي نيز الگوريتمیک است. اين فرد احتمالاً برنامه‌نويس موفقي خواهد بود. اين تنها کافي نيست که بخوايد الگوريتم يك کار را پيدا کنيد. بلکه ماهيتاً بايد الگوريتمیک فکر کنيد. حتي به تمام مسائل زندگي نيز بايد به اين شيوه نگاه کنيد. مثال برنامه‌نويس که داراي تفکر الگوريتمیک است، نسبت به فردي که اين ويژگي را ندارد، مانند نسبت کسی است که ماهيتاً کارش داراي نظم است و کسی که کارش نظم خاصی ندارد. فرد اول، در پايان کار، کارگاه خود را مرتب و منظم مي‌کند. اما، کارگاه فرد دوم نه تنها در هنگام کار نامرتب است، بلکه بعد از انجام کار نيز نامنظم خواهد بود.

وقتي مي‌خواهيد کاري را انجام دهيد، بايد به الگوريتم مناسب براي انجام آن کار فکر کنيد. پس از يافتن الگوريتم مناسب، انجام آن کار براي‌تان بسيار ساده خواهد شد. چيزي را فراموش نمي‌کنيد و در انجام آن کار، همه ابعاد آن را در نظر خواهيد گرفت و مهم‌تر از همه اين‌ها، ساختار کار براي شما روشن خواهد بود. برنامه‌نويس بايد با الگوريتم زندگي کند و آن را به‌عنوان روش زندگي خود بشناسد. ارائه راه‌حل مناسب، نياز دقيق به شناخت مسئله دارد.

مثال ۲-۱. الگوريتمي که "Hello" را نمايش مي‌دهد.

۱. شروع

۲. عبارت "Hello" را نمايش بده

۳. پايان

```
1 print("Hello")
```

Hello

مثال ۳-۱. الگوريتمي که نام خودتان را خوانده و عبارت "Hello" را به همراه نام خودتان نمايش مي‌دهد.

۱. شروع

۲. name را بخوان

name : داده

۳. عبارت "Hello" ، " " ، name را نمایش بده
 "Hello" + " " + name
 (چاپ) : خواسته
 ۴. پایان

```
1 name = input()
2 print("Hello " + name)
```

Fanavarienovin
 Hello Fanavarienovin

۸-۱. عملگرها

عملگرها، نمادهایی هستند که اعمال خاصی را بر روی داده انجام می‌دهند. عملگرها انواع مختلف دارند که برخی از آنها عبارت‌اند از:

۱. عملگرهای محاسباتی
۲. عملگرهای رابطه‌ای (مقایسه‌ای)
۳. عملگرهای منطقی

جدول ۱-۱ عملگرهای محاسباتی.				
عملگر	نام عملگر	مثال	نتیجه	توضیحات
+	جمع	۱۲ + ۳	۱۵	عملوند اول را با عملوند دوم جمع می‌کند.
-	تفریق	۱۳,۵ - ۳	۱۰,۵	عملوند دوم را از عملوند اول کم می‌کند.
*	ضرب	۱۲ * ۲,۵	۳۰	عملوند اول را در عملوند دوم ضرب می‌کند.
/	تقسیم	۱۳ / ۲	۶	عملوند اول را بر عملوند دوم تقسیم می‌کند.
%	باقی‌مانده تقسیم صحیح	۱۳ % ۵	۳	باقی‌مانده تقسیم صحیح عملوند اول بر عملوند دوم را محاسبه می‌کند.

عملگرهای محاسباتی

این عملگرها برای انجام محاسبات بر روی داده‌های عددی به کار می‌روند (جدول ۱-۱). از جمله این عملگرها می‌توان عملگرهای + (جمع)، - (تفریق)، * (ضرب)، / (تقسیم) و % (باقی‌مانده تقسیم صحیح) را نام برد. با عملگرهای +، -، * و / از قبل آشنا هستید. عملگر % برای محاسبه باقی‌مانده تقسیم صحیح به کار می‌رود.

مثال ۴-۱. الگوریتمی که نتیجه عبارت $2 * 5 + 6 / 8 - 7$ را نمایش می‌دهد.

۱. شروع
 ۲. حاصل عبارت $2 * 5 + 6 / 8 - 7$ را نمایش بده
 ۳. پایان
- داده :-
 خواسته $2 * 5 + 6 / 8 - 7$
 رابطه :-

```
1 print(2 * 5 + 6 / 8 - 7)
```

3.75

مثال ۵-۱. الگوریتمی که یک عدد را خوانده، آن را نمایش می‌دهد.

۱. شروع
 ۲. با نمایش پیغام "Enter a:"، یک عدد را خوانده، در a قرار بده
 ۳. a را چاپ کن
 ۴. پایان
- a (عدد): داده
a (نمایش): خواسته
- رابطه

```
1 a = int(input("Enter a:"))
  print(a)
```

Enter a:10
10

مثال ۶-۱. الگوریتمی که دو عدد را خوانده، حاصل جمع آن‌ها را نمایش می‌دهد.

۱. شروع
 ۲. با نمایش پیغام "Enter a:"، یک عدد را خوانده، در a قرار بده
 ۳. با نمایش پیغام "Enter b:"، یک عدد را خوانده، در b قرار بده
 ۴. a + b را چاپ کن
 ۵. پایان
- a , b (دو عدد): داده
a + b (نمایش): خواسته
a + b : رابطه

```
1 a = int(input("Enter a:"))
2 b = int(input("Enter b:"))
3 print(a + b)
```

Enter a:10
Enter b:20
30

مثال ۷-۱. الگوریتمی که یک عدد a را خوانده، حاصل $b = a^3 / (a^2 + a + 1)$ را نمایش می‌دهد.

۱. شروع
 ۲. با نمایش پیغام "Enter a:" مقدار a را بخوان.
 ۳. $b \leftarrow a * a * a / (a * a + a + 1)$
 ۴. b را نمایش بده
 ۵. پایان
- a (عدد): داده
b (نمایش): خواسته
رابطه: $b \leftarrow a * a * a / (a * a + a + 1)$

```
1 a = int(input("Enter a:"))
2 b = (a ** 3) / (a ** 2 + a + 1)
3 print(b)
```

Enter a:10
9.00900900900901

مثال ۸-۱. الگوریتمی که سه عدد را خوانده، میانگین آن‌ها را نمایش می‌دهد.

۱. شروع
 ۲. با نمایش پیغام "Enter a:"، یک مقدار را خوانده، در a قرار بده
- a , b , c (۳ عدد): داده

۳. با نمایش پیغام "Enter b:" یک مقدار را خوانده، در b قرار بده
۴. با نمایش پیغام "Enter c:" یک مقدار را خوانده، در c قرار بده
۵.
$$ave \leftarrow \frac{a+b+c}{3}$$
۶. ave را چاپ کن
۷. پایان

```

1 a = int(input("Enter a:"))
2 b = int(input("Enter b:"))
3 c = int(input("Enter c:"))
4 ave = (a + b + c) / 3
5 print(ave)

```

```

Enter a:10
Enter b:15
Enter c:33
19.333333333333332

```

مثال ۹-۱. الگوریتمی که قاعده و ارتفاع متوازی‌الاضلاع را خوانده، مساحت آن را نمایش می‌دهد (مساحت متوازی‌الاضلاع برابر با قاعده * ارتفاع است).

۱. شروع
2. با نمایش پیغام "Enter base:" مقدار base را بخوان
3. با نمایش پیغام "Enter height:" مقدار height را بخوان area: خواسته
4. قاعده (base) ضرب در ارتفاع (height) را در مساحت (area) قرار بده.
5. مساحت (area) را نمایش بده. $area \leftarrow base * height$
6. پایان

```

1 base = float(input("Enter base:"))
2 height = float(input("Enter height:"))
3 area = base * height
4 print(area)

```

```

Enter base:20
Enter height:30
600.0

```

مثال ۱۰-۱. الگوریتمی که شعاع و ارتفاع استوانه‌ای را خوانده، حجم و مساحت کل استوانه را محاسبه می‌کند. حجم و مساحت کل استوانه به صورت زیر محاسبه می‌شود:

$$\pi = 22/7, \quad \text{حجم استوانه} = \pi * (\text{شعاع})^2 * \text{ارتفاع} \\
 (\text{شعاع})^2 * \pi * 2 + \text{ارتفاع} * \text{شعاع} * \pi * 2 = \text{حجم مساحت کل}$$

۱. شروع
۲. 22 تقسیم بر 7 را در pi قرار بده.
۳. با نمایش پیغام "Enter height:" ارتفاع را خوانده، در height قرار بده
۴. با نمایش پیغام "Enter radius:" شعاع را خوانده، در radius قرار بده

۵. حجم استوانه را با فرمول ($\pi * radius * radius * height$) حساب کرده، در volume قرار بده.
۶. با فرمول $((2 * \pi * radius) * height) + (2 * (\pi * radius * radius))$ حجم مساحت کل استوانه را حساب کرده، در surArea قرار بده.
۷. حجم استوانه (volume) و حجم مساحت کل استوانه (surArea) را نمایش بده.
- ۸ پایان

```

1 PI = 22.0 / 3
2 radius = float(input("Enter radius:"))
3 height = float(input("Enter height:"))
4 volume = height * radius * radius * PI
5 area = 2 * PI * radius * height + 2 * PI * radius * radius
6 print(volume, " ", area)
Enter radius:10
Enter height:20
14666.666666666666      4400.0
    
```

مثال ۱۱-۱. الگوریتمی که شعاع کره‌ای را خوانده، مساحت و حجم کره را محاسبه می‌کند. مساحت و حجم کره برابر است با:

$$\pi = 22/7, \quad \text{مساحت کره} = 4 * \pi * (\text{شعاع})^2,$$

$$\text{حجم کره} = 4 / 3 * \pi * (\text{شعاع})^3$$

۱. شروع
۲. 22 تقسیم بر 7 را در pi قرار بده.
۳. با نمایش پیغام "Enter radius:" شعاع کره را خوانده، در radius قرار بده.
۴. مساحت کره را با فرمول ($4 * \pi * radius * radius$) حساب کرده، در area قرار بده.
۵. حجم کره را با فرمول $((4/3) * (\pi * radius * radius * radius))$ حساب کرده، در volume قرار بده.
۶. مساحت کره (area) و حجم کره (volume) را نمایش بده.
۷. پایان

```

1 PI = 22.0 / 3
2 radius = float(input("Enter radius:"))
3 area = 4 * PI * radius ** 2
4 volume = 4.0/ 3 * PI * radius ** 3
5 print(volume, " ", area)
Enter radius:10
9777.777777777777      2933.333333333333
    
```

مثال ۱۲-۱. وزن یک مولکول آب $10^{-23} * 3,0$ گرم و وزن یک لیتر آب در حدود ۹۵۰ گرم است. الگوریتمی که وزن آب را برحسب لیتر از ورودی خوانده، تعداد مولکول‌های آن را محاسبه می‌کند.

۱. شروع
۲. با نمایش پیغام "Enter w:" وزن آب را خوانده، در w قرار بده.

آشنایی با الگوریتم و زبان برنامه‌نویسی پایتون ۳۱

۳. مقادیر 3.0×10^{-23} و 950 در m و 1 قرار بده.
۴. تعداد مولکول‌ها را با فرمول $(w*1)/m$ محاسبه کرده، در tedad قرار بده.
۵. تعداد مولکول‌ها (tedad) را نمایش بده.
۶. پایان

```
1 L = 950
2 M = 3.0E-23
3 w = float(input("Enter w:"))
4 tedad = w * L / M
5 print(tedad)
Enter w:10
3.1666666666666664e+26
```

مثال ۱۳-۱. هر سال برابر با $3/156 \times 10^7$ ثانیه است. الگوریتمی که سن شما را به سال دریافت کرده، به ثانیه تبدیل می‌کند.

۱. شروع
۲. با نمایش پیغام "Enter age:" سن را به سال خوانده، در age قرار بده.
۳. مقدار 3.156E7 را در ss قرار بده.
۴. سن برحسب ثانیه را با فرمول $age*ss$ محاسبه کرده، در second قرار بده.
۵. second را نمایش بده.
۶. پایان

```
1 ss = 3.156E7
2 age = int(input("Enter age:"))
3 second = age * ss
4 print(second)
Enter age:50
1578000000.0
```

مثال ۱۴-۱. الگوریتمی که حقوق کارمندی را خوانده، بیمه، مالیات و دریافتی او را محاسبه کرده، نمایش می‌دهد. بیمه و مالیات به ترتیب ۷ و ۱۰ درصد حقوق می‌باشند.

۱. شروع
۲. با نمایش پیغام "Enter s:" حقوق را خوانده، در s قرار بده.
۳. مقادیر $s * 0.07$ و $s * 0.1$ را در b و m قرار بده.
۴. حقوق دریافتی را با فرمول $s - b - m$ محاسبه کرده، در p قرار بده.
۵. p را نمایش بده.
۶. پایان

```
1 s = int(input("Enter s:"))
2 b = s * 0.07
3 m = s * 10 / 100
4 p = s - b - m
5 print(p)
```

```
Enter s:100000000
83000000.0
```

مثال ۱۵-۱. الگوریتمی که قیمت کالایی را در سال قبل و فعلی خوانده، نرخ تورم و قیمت سال بعد آن را چاپ می‌کند. نرخ تورم باید به صورت درصد (مثلاً ۵/۶٪) حساب گردد. (توضیح: برای محاسبه نرخ تورم و قیمت سال بعد به صورت زیر عمل کنید):

$$\text{نرخ تورم} = \frac{\text{قیمت سال قبل} - \text{قیمت سال فعلی}}{\text{قیمت سال قبل}}$$

$$\text{نرخ تورم} \times \text{قیمت سال فعلی} + \text{قیمت سال قبل} = \text{قیمت سال بعد}$$

۱. شروع
۲. با نمایش پیغام "Enter y1:" قیمت برای اولین سال را خوانده، در y_1 قرار بده.
۳. با نمایش پیغام "Enter y2:" قیمت برای دومین سال را خوانده، در y_2 قرار بده.
۴. با فرمول $(y_2 - y_1) / y_1$ ، نرخ تورم را محاسبه کرده، در t قرار بده.
۵. قیمت کالا در سال بعد را با توجه به فرمول $y_2 + y_2 * t$ محاسبه کرده، در y_3 قرار بده.
۶. y_3 را نمایش بده.
۷. پایان

```
1 y1 = int(input("Enter y1:"))
2 y2 = int(input("Enter y1:"))
3 t = (y2 - y1) / y1
4 y3 = y2 + y2 * t
5 print(y3)
```

```
Enter y1:100000000
Enter y1:104000000
10816000.0
```

مثال ۱۶-۱. مدیر شرکتی به هر متخصص خود ماهانه ۷۵۰۰۰۰ تومان حقوق پرداخت می‌کند. او می‌خواهد بداند که اگر ۱۳/۵ درصد به حقوق هر متخصص اضافه کند، سالانه چقدر به هزینه شرکت اضافه می‌شود. الگوریتمی که تعداد متخصصین شرکت را خوانده، این کار را انجام می‌دهد.

۱. شروع
۲. مقدار 750000 را در pay قرار بده.
۳. با نمایش پیغام "Enter tedad:" تعداد متخصصین شرکت را خوانده، در tedad قرار بده.
۴. هزینه‌ای که اضافه می‌شود را با فرمول $pay * 12 * tedad * 13.5 / 100$ محاسبه کرده، در extera قرار بده.
۵. extera را نمایش بده.
۶. پایان

```
1 PAY = 750000
2 tedad = int(input("Enter tedad:"))
3 extera = tedad * 12 * PAY * 13.5 / 100
4 print(extera)
```

```
Enter tedad:100  
121500000.0
```

مثال ۱۷-۱. الگوریتمی که عددی دورقمی را خوانده، مجموع ارقام و مغلوب آن را چاپ می‌کند.

۱. شروع
۲. با نمایش پیام "Enter num:" یک عدد صحیح را خوانده، در num قرار بده.
۳. باقی‌مانده تقسیم صحیح num بر 10 را محاسبه کرده (عملگر % باقی‌مانده تقسیم صحیح را محاسبه می‌کند)، در رقم یکان (n_1) قرار بده.
۴. تقسیم صحیح عدد num بر 10 را حساب کرده، در رقم دهگان (n_2) قرار بده.
۵. مغلوب عدد (n_2)، $n_1 * 10 +$ و مجموع ارقام ($n_1 + n_2$) را نمایش بده.
۶. پایان

```
1 num = int(input("Enter num:"))  
2 n1 = num % 10  
3 n2 = num // 10  
4 print(n1 + n2, "\t", n1 * 10 + n2)
```

```
Enter num:47  
11 74
```

مثال ۱۸-۱. الگوریتمی که وزن کالایی را بر حسب کیلوگرم دریافت می‌نماید و وزن آن را بر حسب گرم نمایش می‌دهد. هر کیلوگرم برابر ۱۰۰۰ گرم است.

۱. شروع
۲. با نمایش پیام "Enter kg:" وزن به کیلوگرم را خوانده، در kg قرار بده.
۳. kg را در 1000 ضرب کرده تا به گرم تبدیل شده و در g قرار بده.
۴. g را نمایش بده.
۵. پایان

```
1 kg = int(input("Enter kg:"))  
2 g = 1000 * kg  
3 print(g)
```

```
Enter kg:100  
100000
```

مثال ۱۹-۱. اگر کارمندی به اندازه ۱۵ درصد حقوق ماه گذشته خود پاداش بگیرد، الگوریتمی که حقوق کارمند را خوانده، پاداش او را محاسبه نموده، نمایش می‌دهد.

۱. شروع
۲. با نمایش پیام "Enter salary:" حقوق را خوانده، در salary قرار بده.
۳. salary را ضرب 0.15 (۱۵ درصد) می‌کند تا پاداش را حساب کرده، در reward قرار بده.
۴. reward را نمایش بده.
۵. پایان

```

1 salary = int(input("Enter salary:"))
2 reward = salary * 15 / 100
3 print(reward)

```

Enter salary:20000000
3000000.0

مثال ۲۰-۱. الگوریتمی که ارتفاع و قاعده مثلث را از ورودی خوانده، مساحت آن را محاسبه کرده، به خروجی می‌برد.

$$\text{قاعده} * \text{ارتفاع} * 0.5 = \text{مساحت مثلث}$$

۱. شروع
۲. با نمایش پیغام "Enter x:" ارتفاع مثلث را خوانده، در x قرار بده.
۳. با نمایش پیغام "Enter y:" قاعده مثلث را خوانده، در y قرار بده.
۴. ارتفاع (x) ضرب در قاعده (y) تقسیم بر 2 را حساب کرده، در مساحت (s) قرار بده.
۵. s را نمایش بده.
۶. پایان

```

1 x = int(input("Enter x:"))
2 y = int(input("Enter y:"))
3 s = x * y / 2
4 print(s)

```

Enter x:10
Enter y:20
100.0

مثال ۲۱-۱. الگوریتمی که دو عدد را از کاربر دریافت می‌کند و حاصل جمع، ضرب، تقسیم، تفریق و باقی‌مانده تقسیم صحیح آن‌ها را چاپ می‌کند.

۱. شروع
۲. با نمایش پیغام "Enter a:" یک عدد را خوانده، در a قرار بده.
۳. با نمایش پیغام "Enter b:" یک عدد را خوانده، در b قرار بده.
۴. a، "+", b، "="، جمع a و b را نمایش بده.
۵. a، "-", b، "="، تفریق a و b را نمایش بده.
۶. a، "*", b، "="، حاصل ضرب a در b را نمایش بده.
۷. a، "/", b، "="، حاصل تقسیم a بر b را نمایش بده.
۸. a، "%", b، "="، باقی‌مانده تقسیم صحیح a بر b را نمایش بده.
۹. پایان

```

1 a = int(input("Enter a:"))
2 b = int(input("Enter b:"))
3 print(a, " + ", b, " = ", a + b)
4 print(a, " - ", b, " = ", a - b)
5 print(a, " * ", b, " = ", a * b)
6 print(a, " / ", b, " = ", a / b)

```

```
7 print(a, " % ", b, " = ", a % b)
Enter a:20
Enter b:7
20 + 7 = 27
20 - 7 = 13
20 * 7 = 140
20 / 7 = 2.857142857142857
20 % 7 = 6
```

مثال ۲۲-۱. الگوریتمی که مقدار x را از ورودی خوانده، حاصل عبارت زیر را محاسبه می‌کند:

$$y = \frac{1}{x^2+x+1}$$

۱. شروع
۲. با نمایش پیغام "Enter x:" یک عدد را خوانده، در x قرار بده.
۳. $1.0 / (x * x + x + 1)$ را y قرار بده.
۴. y را نمایش بده.
۵. پایان

```
1 x = int(input("Enter x:"))
2 y = 1 / (x ** 2 + x + 1)
3 print(y)
Enter x:10
0.009009009009009009
```

مثال ۲۳-۱. الگوریتمی که دو عدد را از ورودی می‌خواند (عدد اول مقاومت یک مدار الکتریکی و عدد دوم جریان آن را نمایش می‌دهد). با فرمول زیر ولتاژ را محاسبه کرده، نمایش می‌دهد:

$$\text{جریان} * \text{مقاومت} = \text{ولتاژ}$$

۱. شروع
۲. با نمایش پیغام "Enter r:" مقاومت (r) را بخوان.
۳. با نمایش پیغام "Enter a:" جریان (a) را بخوان.
۴. $r * a$ (جریان * مقاومت) را حساب کرده، در v (ولتاژ) قرار بده.
۵. v را نمایش بده.
۶. پایان

```
1 r = int(input("Enter r:"))
2 a = int(input("Enter a:"))
3 v = r * a
4 print(v)
Enter r:10
Enter a:30
300
```

مثال ۲۴-۱. در شرکتی، سالانه ۱۵۰ خودکار، ۵۰ بسته کاغذ A۴ مصرف می‌شود. در پایان سال این شرکت می‌خواهد بدانند در سال آینده چقدر برای این بخش از تجهیزات اداری، افزایش هزینه دارد. الگوریتمی که قیمت این اقلام را در اسامال از ورودی دریافت می‌نماید، نرخ تورم در سال آینده را نیز می‌گیرد، هزینه اضافه‌شده شرکت

را در این بخش محاسبه می‌کند و به خروجی می‌برد. تورم به صورت درصد وارد می‌گردد که الگوریتم باید آن را به مقدار اعشاری تبدیل کند. به‌عنوان مثال، اگر تورم را ۵٫۶ وارد کنیم، الگوریتم باید از ۰٫۰۵۶ استفاده کند.

۱. شروع
۲. با نمایش پیغام "Enter a4:" قیمت کاغذ را خوانده، در متغیر a4 قرار بده.
۳. با نمایش پیغام "Enter pen:" قیمت خودکار را خوانده، در pen قرار بده.
۴. با نمایش پیغام "Enter t:" تورم را خوانده، در t قرار بده.
۵. هزینه اضافه‌شده را با فرمول $50 * pen * t / 100 + 150 * a4 * t / 100$ کرده، در coste قرار بده.
۶. coste را نمایش بده.
۷. پایان

```

1 a4 = int(input("Enter a4:"))
2 pen = int(input("Enter pen:"))
3 t = int(input("Enter t:"))
4 coste = 50 * pen * t / 100 + 150 * a4 * t / 100
5 print(coste)

```

```

Enter a4:100
Enter pen:150
Enter t:30
6750.0

```

مثال ۲۵-۱. یک دوچرخه‌سوار با سرعت x کیلومتر بر ساعت شروع به حرکت می‌کند و پس از n دقیقه سرعت آن به k کیلومتر در ساعت می‌رسد. الگوریتمی که با استفاده از فرمول زیر شتاب او را محاسبه کرده، نمایش می‌دهد:

$$\text{شتاب} = \frac{(x(\text{سرعت اولیه}) - k(\text{سرعت نهایی})) * 60}{n(\text{زمان})}$$

۱. شروع
۲. با نمایش پیغام "Enter k:" سرعت نهایی را خوانده، در k قرار بده.
۳. با نمایش پیغام "Enter x:" سرعت اولیه را خوانده، در x قرار بده.
۴. با نمایش پیغام "Enter n:" زمان را خوانده، در n قرار بده.
۵. با فرمول $(k-x) * 60 / n$ شتاب را حساب کرده، در acceleration قرار بده.
۶. acceleration را نمایش بده.
۷. پایان

```

1 k = int(input("Enter k:"))
2 x = int(input("Enter x:"))
3 n = int(input("Enter n:"))
4 int
5 acceleration = (k - x) * 60 / n
6 print(acceleration)

```

```

Enter k:100
Enter x:10
Enter n:10
540.0

```

مثال ۲۶-۱. الگوریتمی که شعاع دایره را خوانده، محیط، مساحت و قطر آن را نمایش می‌دهد (محیط، مساحت و قطر دایره به صورت زیر حساب می‌شوند):

$$\begin{aligned} \text{شعاع} &= 2 * \text{قطر} \\ \text{محیط دایره} &= 2 * \text{شعاع} * \pi, \quad \pi = 3.1415 \\ \text{مساحت دایره} &= \pi * (\text{شعاع})^2 \end{aligned}$$

۱. شروع
۲. مقدار pi را برابر 3.1415 قرار بده.
۳. با نمایش پیغام "Enter r:" شعاع دایره را خوانده، در r قرار بده.
۴. r ضرب در ۲ (همان قطر) را در n قرار بده.
۵. n (قطر) ضرب در pi را محاسبه کرده، در محیط (p) قرار بده.
۶. pi ضرب در r ضرب در r را در مساحت (a) قرار بده.
۷. n, p و a را نمایش بده.
۸. پایان

```

1 PI = 3.1415
2 r=int(input("Enter r:"))
3 n = 2 * r
4 p = n * PI
5 a = PI * r * r
6 print(n, "\t", p, "\t", a)

```

```

Enter r:20
40 125.66000000000001 1256.6000000000001

```

مثال ۲۷-۱. الگوریتمی که عددی را به سال خوانده، تعیین می‌کند چند ماه، چند روز و چند ثانیه است.

$$\begin{aligned} \text{سال} &= 365.25 * \text{تعداد روز} \\ \text{تعداد روز} &= \text{تعداد ماه} / 30 \\ \text{تعداد ثانیه} &= \text{تعداد روز} * 24 * 60 * 60 \end{aligned}$$

۱. شروع
۲. با نمایش پیغام "Enter year:" سال را خوانده، در متغیر year قرار بده.
۳. با فرمول 365.25*year تعداد روز را حساب کرده، در day قرار بده.
۴. day را نمایش بده.
۵. day را بر 30 به صورت صحیح تقسیم کرده و در month قرار بده.
۶. month را نمایش بده.
۷. day را در 24 (هر روز ۲۴ ساعت) در 60 (هر ساعت ۶۰ دقیقه) و در 60 (هر دقیقه ۶۰ ثانیه) ضرب کرده، در متغیر second قرار بده.
۸. مقدار second را نمایش بده.
۹. پایان

```

1 year = int(input("Enter year:"))
2 day = int(365.25* year)
3 print(day)
4 month = day // 30
5 print(month)
6 second = day * 24 * 60 * 60
7 print(second)

```

```

Enter year:50
18262
608
1577836800

```

مثال ۲۸-۱. الگوریتمی که یک عدد صحیح پنج رقمی را خوانده، ارقام آن را جدا می‌کند و هر یک از ارقام را با سه فاصله بین آن‌ها چاپ می‌کند. به‌عنوان مثال، اگر کاربر عدد ۴۲۳۳۹ را وارد کند، خروجی به‌صورت زیر باشد:

```

4      2      3      3      9

```

۱. شروع
۲. با نمایش پیغام "Enter n:" یک عدد پنج رقمی را خوانده، در متغیر n قرار بده.
۳. n را در temp قرار بده تا عدد n از دست نرود.
۴. باقی مانده تقسیم صحیح temp بر 10 را در r5 قرار بده تا رقم یکان temp در r5 قرار گیرد.
۵. temp تقسیم بر 10 را در temp قرار بده تا رقم یکان temp حذف گردد.
۶. باقی مانده تقسیم صحیح temp بر 10 را در r4 قرار بده تا رقم یکان temp (یعنی همان رقم دهگان n) در r4 قرار گیرد.
۷. temp تقسیم بر 10 را در temp قرار بده تا رقم یکان temp حذف گردد.
۸. باقی مانده تقسیم صحیح temp بر 10 را در r3 قرار بده تا رقم یکان temp (یعنی، همان رقم صدگان n) در r3 قرار گیرد.
۹. temp تقسیم بر 10 را در temp قرار بده تا رقم یکان temp حذف گردد.
۱۰. باقی مانده تقسیم صحیح temp بر 10 را در r2 قرار بده تا رقم یکان temp (یعنی، همان رقم هزارگان n) در r2 قرار گیرد.
۱۱. temp تقسیم بر 10 را در temp قرار بده تا رقم یکان temp حذف گردد.
۱۲. باقی مانده تقسیم صحیح temp بر 10 را در r1 قرار بده تا رقم یکان temp (یعنی، همان رقم ده هزارگان n) در r1 قرار گیرد.
۱۳. r1، r2، r3، r4 و r5 را با ۳ فاصله نمایش بده.
۱۴. پایان

```

1 n = int(input("Enter n:"))
2 temp = n
3 r1 = temp % 10
4 temp //= 10
5 r2 = temp % 10
6 temp //= 10

```

```

7 r3 = temp % 10
8 temp //= 10
9 r4 = temp % 10
10 temp //= 10
11 r5 = temp % 10
13 print( r5, " ", r4, " ", r3, " ", r2, " ", r1)

```

Enter n:13987
1 3 9 8 7

مثال ۲۹-۱. هر لیتر معادل ۰,۲۶۴۱۷۹ گالن است. الگوریتمی که میزان بنزین مصرفی اتومبیل کاربری را بر حسب لیتر و مسافت طی شده آن را به مایل گرفته، سپس مصرف به ازای هر مایل به گالن را نمایش می‌دهد.

۱. شروع
۲. با نمایش پیغام "Enter liter:" مصرف به لیتر را خوانده، در liter قرار بده.
۳. با نمایش پیغام "Enter mile:" مسافت طی شده را خوانده، در mile قرار بده
۴. با فرمول $(\text{mile} / \text{liter}) * 0.264179$ ، نسبت مصرف مایل به گالن را حساب کرده، در result قرار بده.
۵. result را نمایش بده.
۶. پایان

```

1 liter = int(input("Enter liter:"))
2 mile = int(input("Enter mile:"))
3 result = mile * 0.264179 / liter
4 print(result)

```

Enter liter:20
Enter mile:100
1.320895

مثال ۳۰-۱. الگوریتمی که m و n را خوانده، حاصل عبارات زیر را محاسبه کند:

$$a = m^2 - n^2$$

$$b = 2 * m * n$$

$$c = m^2 + n^2$$

۱. شروع
۲. با نمایش پیغام "Enter m:" یک عدد را خوانده، در m قرار بده
۳. با نمایش پیغام "Enter n:" یک عدد را خوانده، در n قرار بده
۴. $m ** 2 - n ** 2$ را در a قرار بده.
۵. $2 * m * n$ را در b قرار بده.
۶. $m ** 2 + n ** 2$ را در c قرار بده.
۷. a، b و c را نمایش بده.
۸. پایان