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

تألیف:

دکتر جواد وجیدی
دکتر رمضان عباس نژادورزی
دکتر مجید ایرانمنش



فن‌آوری نوین

سرشناسه	: وحیدی، جواد، ۱۳۴۸ - Vahidi, Javad
عنوان و نام پدید آور	: مبانی کامپیوتر و برنامه‌سازی با رویکرد الگوریتم و فلوجارت، پیاده‌سازی شده با جاوا/تالیف جواد وحیدی، رمضان عباس‌نژاد وری، مجید ایرانمنش.
مشخصات نشر	: بابل: فناوری نوین، ۱۴۰۰.
مشخصات ظاهری	: ۳۱۸ ص: مصور.
شابک	: ۹۵۰۰۰ ریال: ۶-۴۴-۷۳۹۳-۶۲۲-۹۷۸
وضعیت فهرست نویسی	: فیبا
یادداشت	: کتابنامه: ص. ۳۱۴.
موضوع	: جاوا (زبان برنامه‌نویسی کامپیوتر)
موضوع	: Java (Computer program language)
موضوع	: الگوریتم‌های کامپیوتری -- راهنمای آموزشی
موضوع	: Computrer algorithms -- Study and teaching
موضوع	: نمودار جریان کار -- راهنمای آموزشی
موضوع	: Flow charts -- Study and teaching
شناسه افزوده	: عباس‌نژاد وری، رمضان، ۱۳۴۸ -
شناسه افزوده	: ایرانمنش، مجید، ۱۳۴۸ -
رده بندی کنگره	: ۷۶/۷۳QA
رده بندی دیویی	: ۰۰۵/۱۳۳
شماره کتابشناسی ملی	: ۸۴۴۴۰۲۸
اطلاعات رکورد کتابشناسی	: فیبا

@fanavarienovinpub

تلفن: ۰۱۱-۳۲۲۵۶۶۸۷

بابل، کد پستی ۷۳۴۴۸-۷۱۶۷۴

فن آوری نوین

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

تألیف: جواد وحیدی، رمضان عباس‌نژاد وری، مجید ایرانمنش.

نوبت چاپ: چاپ اول

سال چاپ: تابستان ۱۴۰۰

شمارگان: ۲۰۰

قیمت: ۹۵۰۰۰ تومان

نام چاپخانه و صحافی: دفتر فنی سورنا

شابک چاپی: ۶-۴۴-۷۳۹۳-۶۲۲-۹۷۸

شابک الکترونیکی: ۹-۴۳-۷۳۹۳-۶۲۲-۹۷۸

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

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

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

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

فهرست مطالب

فصل اول: مفاهیم مقدماتی	۱۷
فصل دوم: فلوجارت، ساختار تصمیم و تکرار	۶۳
فصل سوم: الگوریتم‌های فرعی و توابع	۱۵۶
فصل چهارم: آرایه‌ها	۱۹۵
فصل پنجم: پردازش رشته‌ها	۲۸۶
منابع:	۳۱۴

الگوریتم‌ها و برنامه‌های حل شده در کتاب در یک نگاه

۱. الگوریتم آماده‌سازی نیمر و
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 $m * n^2$

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 ن 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 $11 = y = 3x + c$ و $12 = y = ax + b$
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 کلیه 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 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 باقی مانده 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 چاپ اعداد متحابه بین دو عدد (با استفاده از زیر الگوریتم‌ها)

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

$$\frac{x^n (n)^3 (n)^{x+2}}{x! n! (x+n)!} \sum_{i=1}^n \frac{x^i i^x x^n}{x! n! (n+i)!}$$

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

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

۱۴۷. الگوریتم و فلوجارت خواندن ۱۰، سپس خواندن عدد دیگر (x) و تعیین می‌کند این عدد چند بار در

آرایه تکرار شده است

۱۴۸. 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.

۱۴۹. 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.

۱۵۰. 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.

۱۵۱. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، عناصر آن را می‌خواند. توسط زیر

الگوریتم reverse عناصر آن را برعکس می‌کند.

۱۵۲. 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.

۱۵۳. 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.

۱۵۴. 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 واردشده را ذخیره می‌نماید

۱۵۵. 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.

۱۵۶. 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.

۱۵۷. 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.

۱۵۸. 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.

۱۵۹. 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.
۱۶۰. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، داده‌های آن را خوانده، سپس دو عدد دیگر (k و x) را می‌خواند و مکان kامین وقوع x را در آرایه پیدا می‌نماید و نمایش می‌دهد.
۱۶۱. 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.
۱۶۲. 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.
۱۶۳. الگوریتم و برنامه‌ای که یک آرایه ۱۰ عنصری را تعریف کرده، عناصر آن را می‌خواند. بیش‌ترین مقدار آرایه و مکان آن را نمایش می‌دهد.
۱۶۴. 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.
۱۶۵. 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.
۱۶۶. 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.
۱۶۷. 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.
۱۶۸. 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.
۱۶۹. 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.
۱۷۰. 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.
۱۷۱. 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.
۱۷۲. 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.
۱۷۳. 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.

۱۷۴. الگوریتم و برنامه‌ای که یک آرایه دوبعدی ۵ در ۵ را تعریف کرده، عناصر قطر اصلی و فرعی را مقدار داده، بقیه عناصر را با صفر پر می‌کند. در پایان، عناصر آرایه دوبعدی را نمایش می‌دهد.
۱۷۵. الگوریتم و برنامه‌ای که یک آرایه دوبعدی ۵ در ۵ را تعریف کرده، عناصر محیطی آن مقدار ۱ داده، بقیه عناصر را با صفر پر می‌کند. در پایان، عناصر آرایه دوبعدی را نمایش می‌دهد.
۱۷۶. زیر الگوریتم و برنامه‌ای به نام readAYD که یک آرایه دوبعدی n در m را به‌عنوان پارامتر دریافت کرده، عناصر آن را می‌خواند.
۱۷۷. زیر الگوریتم و برنامه‌ای به نام printAYD که یک آرایه دوبعدی n در m را به‌عنوان پارامتر دریافت کرده، عناصر آن را نمایش می‌دهد.
۱۷۸. الگوریتم و برنامه‌ای که آرایه ۴ در ۴ را تعریف کرده، ۱۶ 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 ۴ در ۴ را تعریف کرده، ۱۶ عدد را خوانده، در آن قرار می‌دهد. سپس، کوچک‌ترین عنصر و تعداد تکرار آن را نمایش می‌دهد.
۱۷۹. الگوریتم و برنامه‌ای که یک آرایه دوبعدی ۴ در ۴ را تعریف کرده، ۱۶ عدد را خوانده، در آن قرار می‌دهد. سپس بزرگ‌ترین عنصر، سطر و ستون آن را نمایش می‌دهد.
۱۸۰. الگوریتم و برنامه‌ای که عناصر آرایه‌ای ۴×۴ را خوانده، بزرگ‌ترین عنصر هر سطر را پیدا کرده، به همراه اطلاعات همان سطر در خروجی چاپ می‌کند.
۱۸۱. الگوریتم و برنامه‌ای که یک آرایه ۴×۴ تعریف کرده، سپس عناصر آرایه را خوانده و حاصل ضرب عناصر غیر صفر محیطی آرایه را محاسبه می‌کند و نمایش می‌دهد. ع
۱۸۲. الگوریتم و برنامه‌ای که یک ماتریس ۴×۴ را می‌خواند. اگر مجموع عناصر روی قطر اصلی و فرعی برابر مجموع عناصر محیطی ماتریس باشد، برنامه "Yes"؛ وگرنه "No" را نمایش می‌دهد.
۱۸۳. الگوریتم و برنامه‌ای که یک ماتریس ۴×۴ را می‌خواند. اگر همه عناصر آرایه برابر باشند، "Yes"؛ وگرنه "No" را نمایش می‌دهد.
۱۸۴. الگوریتم و برنامه‌ای که یک ماتریس ۴×۴ را می‌خواند. اگر عناصر همه عناصر آرایه به‌صورت صعودی مرتب باشند، "Yes"؛ وگرنه "No" را نمایش خواهد داد.
۱۸۵. الگوریتم و برنامه‌ای که یک ماتریس را خوانده، تشخیص می‌دهد که آیا این ماتریس یک ماتریسی اسپارس است یا نه؟
۱۸۶. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، کاراکترهای مکان‌های فرد آن را نمایش می‌دهد.
۱۸۷. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، مقلوب می‌کند و آن را نمایش می‌دهد.
۱۸۸. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، تمام کاراکترهای کوچک رشته را به بزرگ و کاراکترهای بزرگ را به کوچک تبدیل می‌کند.
۱۸۹. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، تعداد کاراکترهای صدا دار آن را شمارش کرده و نمایش می‌دهد.
۱۹۰. الگوریتم و برنامه‌ای که یک کلمه را دریافت کرده، حرف نقره‌ای آن را نمایش می‌دهد.

۱۹۱. الگوریتم و برنامه‌ای که نامی را خوانده، معادل عددی آن را نمایش می‌دهد.
۱۹۲. الگوریتم و برنامه‌ای که دو رشته را خوانده، رشته اول را در رشته دوم کپی می‌کند.
۱۹۳. الگوریتم و برنامه‌ای که دو رشته را خوانده، رشته دوم را به انتهای رشته اول اضافه می‌کند.
۱۹۴. الگوریتم و برنامه‌ای که یک رشته را خوانده، تعداد تکرار هر حرف آن را نمایش می‌دهد.
۱۹۵. الگوریتم و برنامه‌ای که یک رشته را خوانده تشخیص می‌دهد آیا متجانس است یا خیر؟
۱۹۶. 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.
۱۹۷. الگوریتم و برنامه‌ای که یک رشته را خوانده، فضای خالی سمت چپ (ابتدای رشته) را حذف می‌کند و نمایش می‌دهد.
۱۹۸. الگوریتم و برنامه‌ای که رشته‌ای را از ورودی خوانده و بین حروف آن فاصله (Blank) ایجاد می‌کند.
۱۹۹. 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.
۲۰۰. الگوریتم و برنامه‌ای که رشته‌ای را خوانده، کاراکتری از رشته که بزرگ‌ترین کد اسکی را دارد، نمایش می‌دهد.
۲۰۱. 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.
۲۰۲. 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.
۲۰۳. 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.
۲۰۴. 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.
۲۰۵. الگوریتم و برنامه‌ای که رشته‌ای را خوانده و بعد از هر حرف کوچک، حروف بزرگ آن کاراکتر را اضافه می‌کند.
۲۰۶. الگوریتم و برنامه‌ای که صحت کد ملی وارد شده را بررسی می‌کند
۲۰۷. الگوریتم و برنامه‌ای که صحت شابک وارد شده را بررسی می‌کند
۲۰۸. 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 جاوا 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

مفاهیم مقدماتی

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:

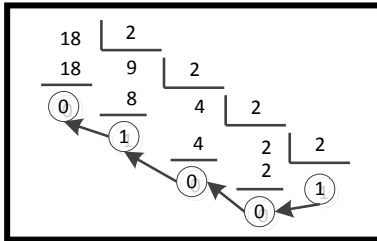
$$49376.823 = 4 \times 10^4 + 9 \times 10^3 + 3 \times 10^2 + 7 \times 10^1 + 6 \times 10^0 + 8 \times 10^{-1} + 2 \times 10^{-2} + 3 \times 10^{-3}$$

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

➤ 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 هزینه 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 مبنای

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 $(1011)_2$ $(1001)_2$ در مبنای ۲ به صورت زیر انجام می‌شود:

$$\begin{array}{r} \\ + \\ \hline 1 \end{array}$$

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 یک‌رقمی در مبنای دو			
$\begin{array}{r} 0 \\ + 0 \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ + 1 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ + 0 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ + 1 \\ \hline 10 \end{array}$

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

$$\begin{array}{r} \\ - \\ \hline 0 \end{array}$$

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

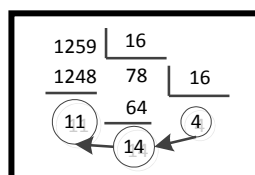
همان طور که مشاهده می کنید در حالت $1 - 0$ به یک رقم قرضی نیاز است، بنابراین حاصل تفریق ۱ شده اما یک رقم قرضی نیز وجود دارد.

سیستم هشت تایی: در سیستم هشت تایی، برای نمایش اعداد از ارقام ۰ تا ۷ استفاده می شود 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 می کنیم. مثال زیر جمع و تفریق در مبنای ۸ را نشان می دهد:

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

در این مثال مجموع دو عدد ۵ و ۶ برابر ۱۱ است که از بزرگ ترین نماد در مبنای ۸ (یعنی عدد ۷) بزرگ تر شده است؛ پس ۸ واحد از آن کم می کنیم و یک واحد به طبقه ی بالاتر اضافه می کنیم و عدد ۳ یعنی حاصل تفریق را در پایین می نویسیم و این کار را برای سایر سطوح نیز انجام می دهیم. همچنین در تفریق عدد ۷ از عدد ۶ ابتدا ۸ واحد به صورت قرضی به عدد ۶ داده می شود تا عدد ۱۴ حاصل شود (بنابراین باید یک واحد از طبقه ی سمت چپ کم شود اما چون رقم سمت چپ، صفر است و کم کردن از آن امکان پذیر نیست ابتدا یک واحد از عدد ۴ کم می کنیم، سپس ۸ واحد به عدد صفر اضافه کرده و در نهایت یک واحد از آن کم می کنیم).

سیستم شانزده تایی، سیستم شانزده تایی، کمی با سیستم های قبلی فرق دارد. چون مبنای ده بزرگ تر 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 می شود. در این سیستم ارزش هر طبقه، ۱۶ برابر ارزش طبقه ی سمت راست آن است. پس، عدد ۱۲۵۹ مبنای ۱۰ برابر EB مبنای ۱۶ می باشد.



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

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

✚ 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 دسته‌ای 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:

$$1100 \quad 1000 \quad 0001 \quad . \quad 0101 \quad 1111 \quad 10101$$

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:

$$10111110101.0001100011 \rightarrow 5 \quad 15 \quad 5.1 \quad 8 \quad 12 \rightarrow 5F5.18C$$

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 راه‌حل مناسب برای آن‌ها

جهت ارائه راه‌حل 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) موردنیاز می‌باشند.

۲. محاسبات (پردازش‌ها): برای رسیدن به نتایج موردنظر 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.

به‌عنوان مثال، فرض کنید بخواهیم محیط و مساحت یک مستطیل را محاسبه کنیم. برای حل این مسئله و ارائه راه‌حل مناسب با روش ارائه‌شده، 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) به‌عنوان 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:

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

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

همان‌طور که مشاهده می‌کنید، از طریق فرمول‌های بیان‌شده، روابط بین داده‌های ورودی و نتایج مورد نظر به‌خوبی 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 توانسته- اید 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 دقیقی 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.

۶-۱. 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 دستور(ات)**

➤ **اگر شرط (ها) آنگاه دستور(ات) وگرنه دستور(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.

با دستورالعمل‌های شرطی در فصل دوم بیش‌تر آشنا خواهید شد.

۵. **دستورالعمل‌های تکرار (حلقه‌ها)**، گاهی لازم است تا مجموعه‌ای از دستورالعمل‌ها به‌دفعات 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 به‌وسیله 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 به‌گونه‌ای 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 به‌اندازه کافی بپزد، صبر کنید.
۱۰. اجاق را خاموش نمایید.

۱۱. نیمرو را در بشقاب قرار دهید.

۱۲. پایان.

همان طور که در این الگوریتم مشاهده می کنید، هر الگوریتم یک نقطه شروع و یک نقطه پایان دارد.

۷-۱. تفکر الگوریتمیک

آیا مفهوم الگوریتم فقط در دنیای برنامه نویسی تعریف می شود؟ آیا تنها برنامه نویس برای طراحی و حل برنامه اش باید از الگوریتم استفاده کند؟ یا این که برای انجام تمام کارها، می توان از ایده الگوریتمی استفاده نمود؟

در جواب این سؤالات باید بگوییم که الگوریتم یک مفهوم عام و فراگیر است. یعنی، ایده الگوریتم، علاوه بر آن که در برنامه نویسی به کار می رود، می تواند در زندگی روزمره نیز استفاده شود. یک برنامه نویس خوب، علاوه بر آن که به الگوریتم برنامه هایی که می نویسد فکر می کند، با الگوریتم زندگی می کند. او تمام زندگی خودش را با الگوریتم انجام می دهد. با الگوریتم فکر می کند. یعنی، تفکر او حتی در امور ساده زندگی نیز الگوریتمیک است. این فرد احتمالاً برنامه نویس موفقی خواهد بود. این تنها کافی نیست که بخواهید الگوریتم یک کار را پیدا کنید. بلکه ماهیتاً باید الگوریتمیک فکر کنید. حتی به تمام مسائل زندگی نیز باید به این شیوه نگاه کنید. مثال برنامه نویسی که دارای تفکر الگوریتمیک است، نسبت به فردی که این ویژگی را ندارد، مانند نسبت کسی است که ماهیتاً کارش دارای نظم است و کسی که کارش نظم خاصی ندارد. فرد اول، در پایان کار، کارگاه خود را مرتب و منظم می کند. اما، کارگاه فرد دوم نه تنها در هنگام کار نامرتب است، بلکه بعد از انجام کار نیز نامنظم خواهد بود.

وقتی می خواهید کاری را انجام دهید، باید به الگوریتم مناسب برای انجام آن کار فکر کنید. پس از یافتن الگوریتم مناسب، انجام آن کار برای تان بسیار ساده خواهد شد. چیزی را فراموش نمی کنید و در انجام آن کار، همه ابعاد آن را در نظر خواهید گرفت و مهم تر از همه این ها، ساختار کار برای شما روشن خواهد بود. برنامه نویسی باید با الگوریتم زندگی کند و آن را به عنوان روش زندگی خود بشناسد. ارائه راه حل مناسب، نیاز دقیق به شناخت مسئله دارد.

مثال ۲-۱. الگوریتم و برنامه ای که "Hello" را نمایش می دهد.

۱. شروع
 ۲. عبارت "Hello" را نمایش بده
 ۳. پایان
۱. شروع
۲. عبارت "Hello" را نمایش بده
۳. پایان

```

1 public class Program
2 {
3     public static void main(String[] args)
4     {
5         System.out.println("Hello");
6     }
7 }

```

Hello

مثال ۳-۱. الگوریتم و برنامه‌ای که نام خودتان را خوانده و عبارت "Hello" را به همراه نام خودتان نمایش می‌دهد.

۱. شروع
۲. name را بخوان
۳. عبارت "Hello", " ", name را نمایش بده " + " + name "Hello" (چاپ): خواسته
۴. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         String name = new Scanner(System.in).nextLine();
7         System.out.println("Hello " + name);
8     }
9 }
    
```

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 باقی مانده تقسیم صحیح به کار می رود.

مثال ۴-۱. الگوریتم و برنامه ای که نتیجه عبارت $2 * 5 + 6 / 8 - 7$ را نمایش می دهد.

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

```

1 public class Program
2 {
3     public static void main(String[] args)
4     {
5         System.out.println(2 * 5 + 6 / 8 - 7);
6     }
7 }

```

3

مثال ۵-۱. الگوریتم و برنامه ای که یک عدد را خوانده، آن را نمایش می دهد.

۱. شروع
 ۲. پیام "Enter a:" را نمایش بده
 ۳. a را بخوان
 ۴. a را چاپ کن
 ۵. پایان
- a (عدد): داده
a (نمایش): خواسته
رابطه :-

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter a:");
7         int a = Integer.parseInt(new Scanner(System.in).nextLine());
8         System.out.println(a);
9     }
10 }

```

Enter a:12

12

مثال ۶-۱. الگوریتم و برنامه ای که دو عدد را خوانده، حاصل جمع آن ها را نمایش می دهد.

۱. شروع
 ۲. پیام "Enter a, b:" را نمایش بده
 ۳. a را بخوان
 ۴. b را بخوان
- a, b (دو عدد): داده
a + b (نمایش): خواسته

رابطه: $a + b$

۵. $a + b$ را چاپ کن

۶. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter a, b:");
8         int a = Integer.parseInt(input.next());
9         int b = Integer.parseInt(input.next());
10        System.out.println(a + b);
11    }
12}

```

Enter a, b:10 20
30

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

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

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter a:");
7         float a = Float.parseFloat(new Scanner(System.in).nextLine());
8         float b = a * a * a / (a * a + a + 1);
9         System.out.println(b);
10    }
11}

```

Enter a:10
9.009009

مثال ۸-۱. الگوریتم و برنامه‌ای که سه عدد را خوانده، میانگین آن‌ها را نمایش می‌دهد.

۱. شروع
۲. پیغام "Enter a, b, c:" را نمایش بده
۳. a را بخوان

۴. b را بخوان

۵. c را بخوان

ave (میانگین): خواسته

$$\text{رابطه: } \text{ave} \leftarrow \frac{a + b + c}{3}$$

$$\text{ave} \leftarrow \frac{a + b + c}{3} \quad ۶.$$

۷. ave را چاپ کن

۸. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter a, b, c:");
8         float a = Float.parseFloat(input.next());
9         float b = Float.parseFloat(input.next());
10        float c = Float.parseFloat(input.next());
11        float ave = (a + b + c) / 3;
12        System.out.println(ave);
13    }
14 }

```

Enter a, b, c:25 20 10
18.333334

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

۱. شروع base , height (۲ عدد): داده

۲. پیغام "Enter base, height:" را نمایش بده

۳. قاعده را خوانده، در base قرار بده. area: خواسته

۴. ارتفاع را خوانده، در height قرار بده.

۵. قاعده (base) ضرب در ارتفاع (height) را در مساحت (area) قرار بده.

۶. مساحت (area) را نمایش بده. رابطه: $\text{area} \leftarrow \text{base} * \text{height}$

۷. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter base1, height:");
8         float base1 = Float.parseFloat(input.next());
9         float height = Float.parseFloat(input.next());
10        float area = base1 * height;
11        System.out.println(area);

```

```

12 }
13 }
Enter base1, height:10 20
200.0
    
```

مثال ۱۰-۱. الگوریتم و برنامه‌ای 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:

$$\pi = 22/7$$

$$\text{حجم استوانه} = \text{ارتفاع} * (\text{شعاع})^2 * \pi$$

$$\text{حجم مساحت کل} = 2 * \pi * (\text{شعاع})^2 + \text{ارتفاع} * \text{شعاع} * \pi$$

۱. شروع
۲. 22.0 تقسیم بر 7 را در pi قرار بده.
۳. پیغام "Enter radius, height:" را نمایش بده
۴. ارتفاع استوانه را خوانده، در height قرار بده.
۵. شعاع استوانه را خوانده، در radius قرار بده.
۶. حجم استوانه را با فرمول (pi*radius*radius*height) حساب کرده، در volume قرار بده.
۷. با فرمول ((2*pi*radius)*height)+(2*(pi*radius*radius)) حجم مساحت کل استوانه را حساب کرده، در surArea قرار بده.
۸. حجم استوانه (volume) و حجم مساحت کل استوانه (surArea) را نمایش بده.
۹. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         final float PI = 22.0f / 7.0f;
7         Scanner input = new Scanner(System.in);
8         System.out.print("Enter radius, height:");
9         float radius = Float.parseFloat(input.next());
10        float height = Float.parseFloat(input.next());
11        float volume = height * radius * radius * PI;
12        float area = 2 * PI * radius * height + 2 * PI * radius * radius;
13        System.out.printf("%1$s %2$s" + "\r\n", volume, area);
14    }
15 }
    
```

```

Enter radius, height:10 20
6285.7144 1885.7142
    
```

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

$$\pi = 22/7$$

$$\text{مساحت کره} = 4 * \pi * (\text{شعاع})^2$$

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

۱. شروع
۲. 22.0 تقسیم بر 7 را در pi قرار بده.
۳. پیغام "Enter radius:" را نمایش بده
۴. شعاع کره را خوانده، در radius قرار بده.
۵. مساحت کره را با فرمول $(4 * \pi * \text{radius} * \text{radius})$ حساب کرده، در area قرار بده.
۶. حجم کره را با فرمول $((4.0/3) * (\pi * \text{radius} * \text{radius} * \text{radius}))$ حساب کرده، در volume قرار بده.
۷. مساحت کره (area) و حجم کره (volume) را نمایش بده.
۸. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         final float PI = 22.0f / 7.0f;
7         System.out.print("Enter radius:");
8         float radius = Float.parseFloat(new Scanner(System.in).nextLine());
9         float area = 4 * PI * radius * radius;
10        float volume = 4.0f / 3 * PI * radius * radius * radius;
11        System.out.printf("%1$s %2$s" + "\r\n", volume, area);
12    }
13}

```

Enter radius:10
4190.476 1257.1428

مثال ۱۲-۱. وزن یک مولکول آب 3.0×10^{-23} گرم و وزن یک لیتر آب در حدود ۹۵۰ گرم است. الگوریتم و برنامه‌ای که وزن آب را برحسب لیتر از ورودی خوانده، تعداد مولکول‌های آن را محاسبه می‌کند.

۱. شروع
۲. پیغام "Enter w:" را نمایش بده
۳. وزن آب را خوانده، در w قرار بده.
۴. مقادیر 3.0×10^{-23} و 950 را به ترتیب در m و l قرار بده.
۵. تعداد مولکول‌ها را با فرمول $(w * l) / m$ محاسبه کرده، در tedad قرار بده.
۶. تعداد مولکول‌ها (tedad) را نمایش بده.
۷. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         final double L = 950;
7         final double M = 3.0E-23;

```



```

8      System.out.print("Enter w:");
9      double w = Double.parseDouble(new Scanner(System.in).nextLine());
10     double tedad = w * L / M;
11     System.out.println(tedad);
12     }
13 }

```

```

Enter w:100
3.1666666666666664E27

```

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

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

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         final double ss = 3.156E7;
7         System.out.print("Enter age:");
8         int age = Integer.parseInt(new Scanner(System.in).nextLine());
9         double second = age * ss;
10        System.out.println(second);
11    }
12 }

```

```

Enter age:51
1.60956E9

```

مثال ۱۴-۱. الگوریتم و برنامه‌ای 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. پیغام "Enter s:" را نمایش بده
۳. حقوق را خوانده، در s قرار بده.
۴. مقادیر $s * 0.07$ و $s * 0.1$ را در b و m قرار بده.
۵. حقوق دریافتی را با فرمول $s - b - m$ محاسبه کرده، در p قرار بده.
۶. p را نمایش بده.

۷. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter s:");
7         int s = Integer.parseInt(new Scanner(System.in).nextLine());
8         int b = s * 7 / 100;
9         int m = s * 10 / 100;
10        int p = s - b - m;
11        System.out.println(p);
12    }
13}

```

Enter s:1000000
830000

مثال ۱۵-۱. الگوریتم و برنامه‌ای 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 ۶/۵٪) 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. پیغام "Enter y1, y2:" را نمایش بده.
۳. قیمت برای اولین سال را خوانده، در y1 قرار بده.
۴. قیمت برای دومین سال را خوانده، در y2 قرار بده.
۵. با فرمول $(y2 - y1) / y1$ ، نرخ تورم را محاسبه کرده، در t قرار بده.
۶. قیمت کالا در سال بعد را با توجه به فرمول $y2 + y2 * t$ محاسبه کرده، در y3 قرار بده.
۷. y3 را نمایش بده.
۸. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter y1, y2:");
8         int y1 = Integer.parseInt(input.next());
9         int y2 = Integer.parseInt(input.next());
10        float t = (float)(y2 - y1) / y1;
11        float y3 = y2 + y2 * t;
12        System.out.println(y3);

```

```

13 }
14 }
Enter y1, y2:10000000 15000000
2.25E7
    
```

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. الگوریتم و برنامه‌ای که تعداد متخصصین شرکت را خوانده، این کار را انجام می‌دهد.

۱. شروع
۲. مقدار ۷۵۰۰۰۰ را در pay قرار بده.
۳. پیغام "Enter tedad:" را نمایش بده
۴. تعداد متخصصین شرکت را خوانده، در tedad قرار بده.
۵. هزینه‌ای که اضافه می‌شود را با فرمول $pay * 12 * tedad * 13.5 / 100$ محاسبه کرده، در extera قرار بده.
۶. extera را نمایش بده.
۷. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         final int PAY = 750000;
7         System.out.print("Enter tedad:");
8         int tedad = Integer.parseInt(new Scanner(System.in).nextLine());
9         float extera = tedad * 12 * PAY * 13.5f / 100;
10        System.out.println(extera);
11    }
12 }
Enter tedad:50
6.075E7
    
```

مثال ۱۷-۱. الگوریتم و برنامه‌ای که عددی دو رقمی را خوانده، مجموع ارقام و مغلوب آن را چاپ می‌کند.

۱. شروع
۲. پیغام "Enter num:" را نمایش بده
۳. یک عدد صحیح را خوانده، در num قرار بده.
۴. باقی‌مانده تقسیم صحیح num بر ۱۰ را محاسبه کرده (عملگر % باقی‌مانده تقسیم صحیح را محاسبه می‌کند)، در رقم یکان (n1) قرار بده.
۵. تقسیم صحیح عدد num بر ۱۰ را حساب کرده، در رقم دهگان (n2) قرار بده.
۶. مغلوب عدد (n1, n2) و مجموع ارقام (n1 + n2) را نمایش بده.
۷. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter num:");
7         int num = Integer.parseInt(new Scanner(System.in).nextLine());
8         int n1 = num % 10;
9         int n2 = num / 10;
10        System.out.printf("%1s\t%2s%3s" + "\r\n", n1 + n2, n1, n2);
11    }
12 }

```

```

Enter num:78
15      87

```

مثال ۱۸-۱. الگوریتم و برنامه‌ای 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. پیغام "Enter kg:" را نمایش بده.
۳. وزن به کیلوگرم را خوانده، در kg قرار بده.
۴. kg را در 1000 ضرب کرده تا به گرم تبدیل شده و در g قرار بده.
۵. g را نمایش بده.
۶. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter kg:");
7         int kg = Integer.parseInt(new Scanner(System.in).nextLine());
8         int g = 1000 * kg;
9         System.out.println(g);
10    }
11 }

```

```

Enter kg:1000
122000

```

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

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

۵. reward را نمایش بده.

۶. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter salary:");
7         int salary = Integer.parseInt(new Scanner(System.in).nextLine());
8         int reward = salary * 15 / 100;
9         System.out.println(reward);
10    }
11 }

```

```

Enter salary:1000000
1500000

```

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

$$\text{قاعده} * \text{ارتفاع} \times 0,5 = \text{مساحت مثلث}$$

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

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter x, y:");
8         int x = Integer.parseInt(input.next());
9         int y = Integer.parseInt(input.next());
10        int s = x * y / 2;
11        System.out.println(s);
12    }
13 }

```

```

Enter x, y:100 150
7500

```

مثال ۲۱-۱. الگوریتم و برنامه‌ای که دو عدد را از کاربر دریافت می‌کند و حاصل جمع، ضرب، تقسیم، تفریق و باقی مانده تقسیم صحیح آن‌ها را چاپ می‌کند.

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

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter a, b:");
8         int a = Integer.parseInt(input.next());
9         int b = Integer.parseInt(input.next());
10        System.out.printf("%1$s + %2$s = %3$s" + "\r\n", a, b, a + b);
11        System.out.printf("%1$s - %2$s = %3$s" + "\r\n", a, b, a - b);
12        System.out.printf("%1$s * %2$s = %3$s" + "\r\n", a, b, a * b);
13        System.out.printf("%1$s / %2$s = %3$s" + "\r\n", a, b, a / b);
14        System.out.printf("%1$s %% %2$s = %3$s" + "\r\n", a, b, a % b);
15    }
16 }
17 }

```

```

Enter a, b:100 150
100 + 150 = 250
100 - 150 = -50
100 * 150 = 15000
100 / 150 = 0
100 % 150 = 100

```

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

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

۱. شروع
۲. پیغام "Enter x:" را نمایش بده
۳. x را بخوان.

۴. $1.0 / (x * x + x + 1)$ را y قرار بده.

۵. y را نمایش بده.

۶. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         System.out.print("Enter x:");
7         int x = Integer.parseInt(new Scanner(System.in).nextLine());
8         float y = 1.0f / (x * x + x + 1);
9         System.out.println(y);
10    }
11 }

```

```

Enter x:10
0.009009009

```

مثال ۲۳-۱. الگوریتم و برنامه‌ای 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:

جریان * مقاومت = ولتاژ

۱. شروع

۲. پیغام "Enter r, a:" را نمایش بده

۳. مقاومت (r) را بخوان.

۴. پیغام "Enter a:" را نمایش بده

۵. جریان (a) را بخوان.

۶. $r * a$ (جریان * مقاومت) را حساب کرده، در v (ولتاژ) قرار بده.

۷. v را نمایش بده.

۸. پایان

```

1 import java.util.*;
2 public class Program
3 {
4     public static void main(String[] args)
5     {
6         Scanner input = new Scanner(System.in);
7         System.out.print("Enter r, a:");
8         int r = Integer.parseInt(input.next());
9         int a = Integer.parseInt(input.next());
10        int v = r * a;
11        System.out.println(v);
12    }
13 }

```

```

Enter r, a:100 150
15000

```