



برنامه نویسی پایتون



```

while int(input('A: ')) == 0:
    print("A can't be zero!")
while int(input('B: ')) > 0:
    print("B is positive")
while int(input('C: ')) < 0:
    print("C is negative")
while True:
    n = int(input("Enter a number: "))
    if n in range(n):
        if n%4 == 0:
            turtle.forward(100)
            turtle.left(90)
        elif n%4 == 1:
            turtle.forward(100)
            turtle.right(90)
        elif n%4 == 2:
            turtle.forward(100)
            turtle.left(180)
        elif n%4 == 3:
            turtle.forward(100)
            turtle.right(180)
    else:
        print("Invalid input")
n = 1
while (i <= n):
    if (i%2 == 1):
        print("1", end=" ")
    else:
        print("0", end=" ")
    i = i+1
print()
if (i%2 == 0):
    h = 1
else:
    h = 0
i = i+1

```





مجموعه کتاب‌های علامه حلی

برنامه نویسی پایتون (۱)

امام علی علیه‌السلام فرمودند:
هر کس در حال طلب دانش مرگش فرا رسد، میان او و پیامبران
تنها یک درجه تفاوت باشد.

تقدیم به روح بلند شهدای پرواز به ویژه نخبگان متعهد آقای
محمد صالحه و همسرگرامیشان خانم زهرا حسنی سعدی.

- محمدرضا جهانگیر
- سید شهاب قدسی
- علیرضا صالحه





ششماه
کتاب

سرشناسه : جهانگیر، محمدرضا، ۱۳۵۴
عنوان و نام پدیدآور : برنامه‌نویسی: پایتون (۱) / محمدرضا جهانگیر، سید شهاب قدسی، علیرضا صالحه
مشخصات نشر : تهران: انتشارات حلی، ۱۴۰۱.
مشخصات ظاهری : ۱۰۴ص: مصور(بخشی رنگی)، جدول، نمودار؛ ۲۲ × ۲۹ س.م.
فروست : مجموعه کتاب علامه حلی
شابک : ۹۷۸-۶۰۰-۴۹۶-۲۴۱-۴
وضعیت فهرست نویسی : فیپای مختصر
یادداشت : پایتون (برنامه‌نویسی کامپیوتر).
رده‌بندی کنگره : ۱۳۹۴ ج ۹ پ/۷۳/۷۶ QA
رده بندی دیویی : ۰۵/۱۳۳
شماره کتابشناسی ملی : ۳۹۸۸۵۴۱



عنوان کتاب : برنامه‌نویسی: پایتون (۱)
ناشر : انتشارات حلی
ناشرهمکار : انتشارات دانش پژوهان جوان
مؤلفین : محمدرضا جهانگیر، سیدشهاب قدسی، علیرضا صالحه
صفحه آرا : راضیه‌سادات فرهانیان
طراح جلد : الهه شرفی
تصویرساز : محمدحسین صفدریان
هماهنگی : سمیه سادات فاطمی
سال چاپ : ۱۴۰۲
نوبت چاپ : نهم (ویرایش دوم)
شمارگان : ۲۰۰۰ جلد
قیمت : ۱۲۰۰۰۰ تومان
شماره شابک : ۹۷۸-۶۰۰-۴۹۶-۲۴۱-۴



تهران، خیابان انقلاب، میران فردوسی، ابتدای کویه براتی، پلاک ۱۶ ول ۱۴

تلفن دفتر مرکزی: ۵-۶۶۷۴۴۳۸۴









کلیه حقوق این اثر برای ناشر محفوظ است.

هیچ شخص حقیقی یا حقوقی حق برداشت تمام یا قسمتی از اثر را به صورت چاپ، فتوکپی، جزوه و مجازی ندارد.

متخلفان به موجب بند ۵ از ماده ۲ قانون حمایت از ناشران تحت پیگرد قانونی قرار می‌گیرند.



پالاب
براتی

	فصل ۱ مفاهیم پایه	درسنامه ۷	تمرین ۲۰
		۲۱ درسنامه	۳۰ تمرین
	فصل ۲ برنامه‌نویسی با پایتون		
	فصل ۳ خروجی، متغیر، ورودی	۳۱ درسنامه	۴۷ تمرین
		۴۹ درسنامه	۶۳ تمرین
	فصل ۴ شرط (if)		
	فصل ۵ لاکی قهرمان وارد می‌شود	۶۵ درسنامه	۷۳ تمرین
		۷۵ درسنامه	۸۶ تمرین
	فصل ۶ حلقه شمارشی (for)		
	فصل ۷ حلقه شرطی (while)	۸۹ درسنامه	۱۰۱ تمرین
		۱۰۳ درسنامه	۱۱۴ تمرین
	فصل ۸ برنامه‌نویسی پیشرفته		

به نام خدا

چند سال پیش، تعدادی از معلمان با دغدغه «آموزش استعدادهای درخشان»، دور هم جمع شدند و موسسه علامه حلی را تأسیس کردند. این معلم‌ها - که خودشان از دانش‌آموزان مدارس استعدادهای درخشان شهر تهران می‌باشند - سال‌ها در مدارس سمپاد (سازمان ملی پرورش استعدادهای درخشان)، به دنبال پیاده‌سازی روش‌های جدید و مؤثر آموزش بوده‌اند و در نهایت تصمیم گرفتند تا نتیجه این تجربیات را در موسسه علامه حلی در اختیار دیگر فعالان در عرصه آموزش بگذارند.

مجموعه کتاب‌های انتشارات علامه حلی، یکی از محصولات این تلاش جمعی است. در این کتاب‌ها تلاش شده است تا علاوه بر تأمین محتوای مناسب برای دانش‌آموزان برتر کشور، روش‌های جدیدتر و مؤثرتر آموزشی هم در انتقال این محتوا به کار گرفته شده و پیاده‌سازی شود. در پس این کتاب‌ها، ساعت‌ها کار فکری برای انتخاب ساختار و شیوه تدوین صرف شده است. فعال کردن دانش‌آموز در روند آموزش و ارجاع او به انجام مشاهدات، فعالیت‌ها و آزمایش‌های مناسب برای انتقال مفاهیم آموزشی و همچنین ترغیب دانش‌آموز برای مراجعه به منابع گسترده‌تر چون سایت‌های علمی اینترنتی و نرم‌افزارهای آموزشی، از ویژگی‌های این سیستم آموزشی است. علاوه بر این برای کمک به فرایند تدریس معلمان عزیز، محصولات جانبی چون متن راهنمای تدریس کتاب، محتوای الکترونیک و ... در کنار هر کتاب تولید شده است.

مجموعه کتاب‌های علامه حلی، با همکاری جمع زیادی از مؤلفین و معلمان باتجربه مدارس سمپاد - که به دقت انتخاب شده‌اند - تألیف و ویرایش گردیده است؛ اما آرزوی ما در این مؤسسه این است که از حضور تمامی معلمان دلسوز و باتجربه مدارس سمپاد و دیگر مراکز آموزشی برتر کشور عزیزمان، در تألیف کتاب‌ها و دیگر محصولات آموزشی، بهره ببریم؛ بنابراین از شما دبیران عزیز خواهشمندیم تجربه‌های خود را در زمینه استفاده از این کتاب و آموزش آن در کلاس، برای ما به آدرس الکترونیک: book@mhelli.ir ارسال فرمایید تا ما در چاپ‌های بعدی کتاب، از تجربیات، نظرات و حتی تصاویر ارسالی شما در انجام آزمایش‌ها، فعالیت‌ها، بازدیدها و ... در کتاب - و البته با ذکر نام ارسال‌کننده - استفاده کنیم. البته دانش‌آموزان خوب و پرتلاش هم می‌توانند در این کار همکاری کنند و با معلمان خود در اجرای این طرح همراه شوند.

عابدی جعفری

مدیر انتشارات حلی

مقدمه مؤلفان

چرا برنامه نویسی؟

یکی از پرسش‌هایی که برای دانش‌آموزان مطرح می‌شود این است چرا باید برنامه‌نویسی بیاموزیم؟ برنامه‌نویسی به چه دردی می‌خورد؟ مگر چند درصد از ما قرار است برنامه‌نویس شویم که این درس را برایمان در نظر می‌گیرند؟ و پرسش‌هایی از این دسته، چند پاسخ برای این پرسش‌ها متصور است:

۱. مهم‌ترین علت این است که برنامه‌نویسی نگرش ما را نسبت به مسائل پیش‌رو در زندگی تغییر می‌دهد. رایانه موجودی است که فقط منطق را متوجه می‌شود و برای پاسخ گرفتن از آن باید راه‌حل را به صورت دقیق و کاملاً صحیح برای رایانه تشریح کرد. این باعث می‌شود تا برنامه‌نویس بیاموزد چگونه در مواجهه به مشکلات و مسائل پیش‌رو حتی در خارج از رایانه و در ارتباط با سایر افراد چگونه می‌توان مسأله را تحلیل کرد، به قطعات قابل حل تجزیه نمود و در نهایت آن را حل کرد. به صورت خلاصه فراگیری برنامه‌نویسی نگرش منطقی و توان حل منطقی مسائل در برنامه‌نویس به‌طور چشم‌گیری بهبود می‌بخشد.

۲. مهارت برنامه‌نویسی در حد متوسط حتی برای غیر برنامه‌نویس‌ها هم مورد نیاز خواهد بود. فراگیری برنامه‌نویسی برای یک دانش‌آموز یا یک دانشجو با استعداد با خلاقیت و با رویکرد علمی و نیز شاغلین بسیاری از مشاغل فنی و مهندسی و حتی مشاغل مرتبط با آن‌ها یک مزیت است؛ چرا که بسیاری اوقات در پروژه‌ها یا امور شغلی، نیازمند برنامه‌های کوچکی خواهید شد که اگر برنامه‌نویسی بلد نباشید، کار شما لنگ یک برنامه‌نویس خواهد بود.

۳. آن دسته‌ای که برنامه‌نویسی علاقمند خواهند شد و آن را به‌عنوان رشته تحصیلی یا شغل انتخاب خواهند کرد، اگر از سنین مدرسه به برنامه‌نویسی روآورند، موفقیت‌های بیشتری در این حوزه کسب خواهند نمود و فرصت‌هایی نظیر مسابقات و المپیادها به موفقیت آینده‌ایشان کمک بسیاری خواهد نمود. و مهم‌تر اینکه حل مسائل زیاد و آشنایی با الگوریتم‌های متنوع نگرش منطقی-برنامه‌نویسی ایشان را به شکل صحیحی رشد خواهد داد.

چرا پایتون؟

این هم پرسش مهمی است. چرا از بین این همه زبان پایتون؟ پایتون چه مزایا و معایبی نسبت به زبان‌های دیگر دارد؟

۱. هر زبانی برای توسعه‌دهنده‌ای از نرم‌افزارها تولید شده است. مثلاً ایجاد سایت اینترنتی، برنامه‌های محاسباتی، برنامه‌های مبتنی بر پایگاه داده، بازی، برنامه‌های تلفن همراه و تبلت، تولید سیستم عامل و ... که برای هر کدام زبانی مناسب‌تر است. در این میان یکی از مزایای پایتون این است که در حوزه‌های متنوعی کاربرد دارد.

۲. سادگی پایتون از جمله عواملی است که در آغاز کار دست و پا گیر نخواهد بود و در ادامه نیز برنامه‌نویسی را آسان و شیرین خواهد کرد. مزیتی که بسیاری از زبان‌های پیشرفته هم از آن محرومند.

۳. به نظر می‌رسد جدید بودن، به‌روز بودن و پذیرفته شدن این زبان در شرکت‌های بزرگ کامپیوتری باعث پیشرفت و اقبال این زبان گردد؛ و کسانی که با این زبان آشنا هستند در دنیای دیجیتالی که همه چیز در آن با گذشت ساعت‌ها به سرعت قدیمی و ناسازگار می‌شود، شانس بیشتری در این زمینه خواهند داشت.

۴. برای برنامه‌نویسی به زبان پایتون رایانه پیشرفته نیاز نیست. از یک رایانه با مشخصات معمولی یا کمی پایین‌تر از معمول نیز می‌توان استفاده کرد. محیط برنامه‌نویسی پایتون روی سیستم عامل‌های مختلف از جمله ویندوز، لینوکس، اندروید قابل نصب است. حتی با تهیه یک تبلت ارزان قیمت اندرویدی می‌توان با استفاده از برنامه‌هایی مانند touchcode، QPython و ... که در بازارهای اندرویدی هم موجود است، به نوشتن برنامه پرداخت.

۵. پایتون یک زبان متن باز (open source) است و وجود کتابخانه‌های رایگان زیاد از مزایای آن است. اگر مایل باشید در حوزه خاصی برنامه‌نویسی کنید و نیازمند دستورات و توابع و محاسبات آن حوزه باشید، کافی است در اینترنت جستجو کنید تا کتابخانه‌ای رایگان که دستورات، توابع و فرمول‌های مورد نیاز شما را تولید و گردآوری کرده بیابید. البته شما نیز می‌توانید کتابخانه‌های تولیدی خودتان را در وبلاگ یا سایت خودتان منتشر کنید تا به نام شما مورد استفاده دیگران قرار گیرد.

قبل از شروع به مطالعه کتاب، این قسمت را بنویسید:

وقتی شروع به خواندن این کتاب کنید با بخش‌های مختلفی مواجه می‌شوید که غالباً یک لاک‌پشت متفاوت در اول هر کدام وجود دارد. برای هر کدام از این بخش‌ها از شما انتظار داریم کار متفاوتی انجام دهید. این قسمت‌ها بر اساس تئوری‌های نوین آموزش و تجارب موفق تدریس برای آموزش دانش‌آموزان مستعد طراحی شده است. این بخش‌ها شامل:

درخت دانش: در صفحه دوم هر فصل، نمودار دایره‌ای شکلی کشیده شده که به ما کمک می‌کند بفهمیم در آن فصل مطالب علمی چطور تقسیم‌بندی شده و ارتباط آن‌ها با هم چیست. در واقع این بخش نقشه‌ای است برای گم نشدن در موضوعات علمی.

اهداف رفتاری: زیر هر درخت دانش، چند جمله نوشته شده که از اول کار معلوم کند که این فصل را می‌خوانیم که چه بشود. خوب است در آخر فصل هم برگردیم و ببینیم که می‌توانیم کارهایی را که در این بخش گفته انجام دهیم یا نه.

پاسخگو باش: در این قسمت باید پاسخگو باشیم. پاسخگوی سؤالی که پرسیده شده و انتظار می‌رود بعد از خواندن درس تا آن قسمت، بتوانیم با کمی فکر کردن به آن جواب دهیم.

فسفر بسوزان: شاید لازم باشد مقدار بیشتری از مغز خودمان استفاده کنیم و قدری از فسفرهای ذخیره‌شده را بسوزانیم! سؤالاتی که در بخش فسفر بسوزان مطرح می‌شود فقط با خواندن مطالب درسی قابل پاسخگویی نیست و باید کمی بیش از معمول درباره آن‌ها فکر کنیم.

شهر فرنگ: همه چیز که کتاب و کلاس نیست. تقریباً همه ما ساعت‌هایی از روز را در اینترنت سر می‌کنیم و می‌شود علاوه بر سایر کارها، به سایت‌های علمی و جذاب هم سر زد. در بخش شهر فرنگ سایتی مربوط به موضوع فصل معرفی شده که توصیه مؤلفان بازدید از آن سایت است.

لغت‌نامه: ما دانش‌آموزان مستعد و متفاوت (!) دوست داریم بتوانیم علاوه بر مطالب درسی، جستجویی هم بکنیم و ببینیم در دنیا درباره موضوع درسی ما چه چیزی وجود دارد. برای همین در پایان هر فصل لغات مهم فصل با معادل انگلیسی آن آورده شده است.

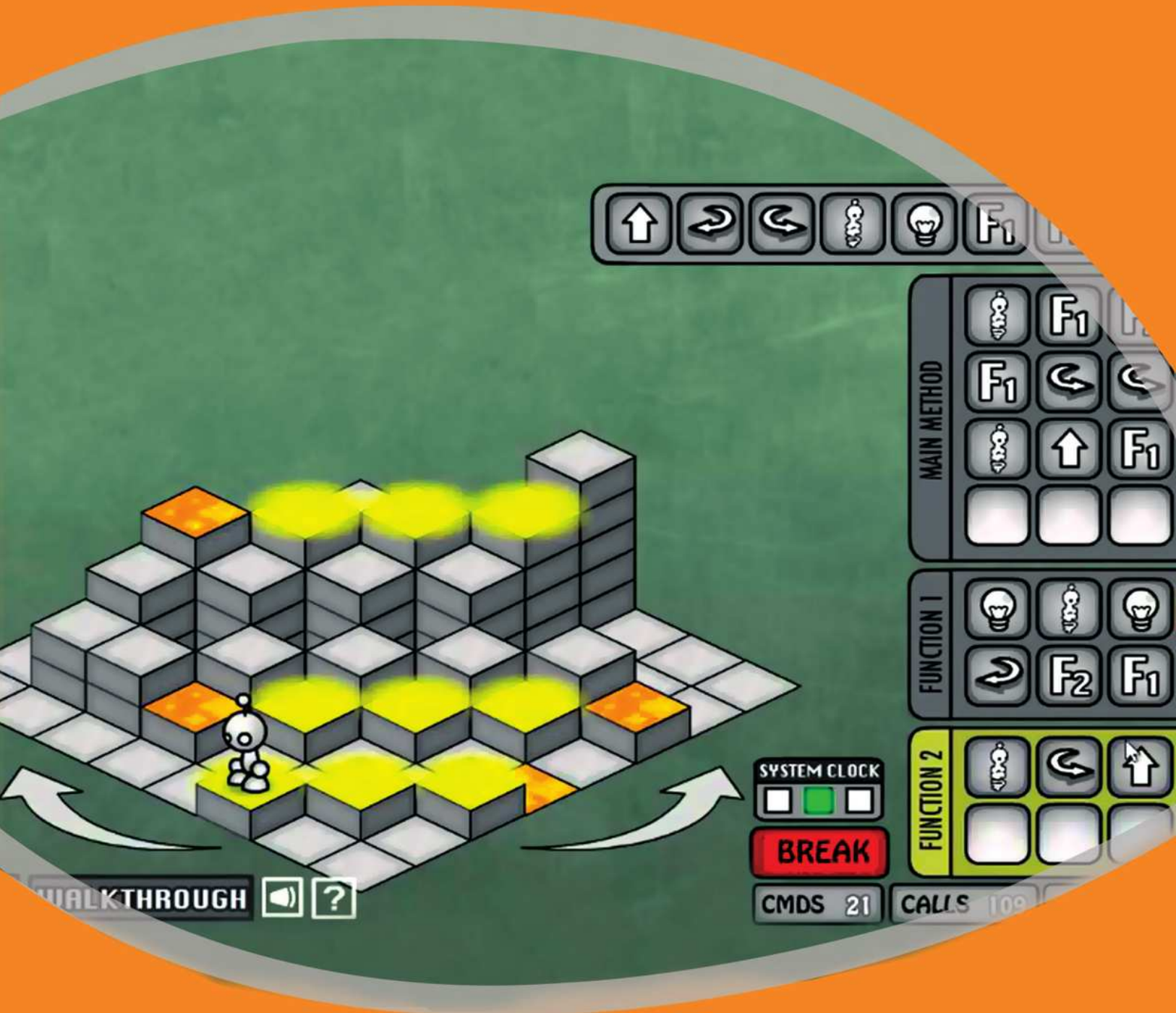
تمرین‌ها: در آخر هر فصل تمرین‌های مرتبط با آن آورده شده است. از آنجایی که مؤلفان کتاب از دبیران باسابقه هستند پس تعداد تمرین‌ها، وقت لازم برای انجام آن‌ها، تعداد سؤالات سخت و آسان و نوع سؤالات با برنامه و محاسبه تعیین شده است. پس خیالتان راحت باشد که همه تمرین‌ها را در طول سال می‌شود انجام داد. تمرین‌ها بر اساس موضوعات هر فصل بخش‌بندی شده، بنابراین لازم نیست برای تمرین منتظر پایان فصل باشید؛ در پایان هر مبحث می‌توانید به بخش تمرین‌ها مراجعه کنید و تمرین‌های همان مبحث را حل کنید.

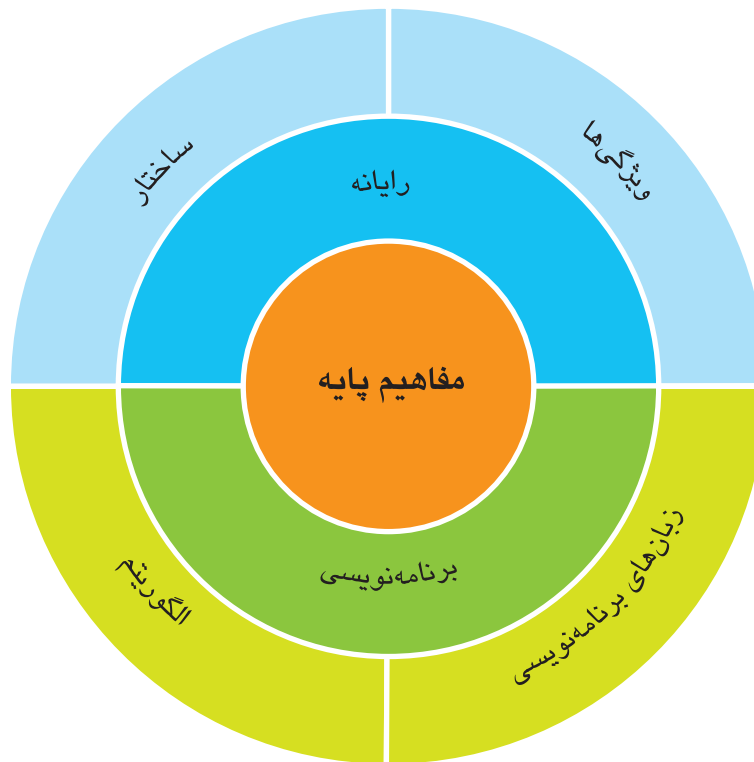
دست‌به‌کد شو: برنامه‌نویسی درسی کاربردی است که در حین آموزش آن لازم است شما هم دست به کد بشوید. در بخش دست به کد شو از شما خواسته شده تا سعی کنید خودتان برنامه را بنویسید. حواستان باشد این بخش، قسمت مهمی از روند درسی است و نمی‌شود بدون دست‌به‌کد شدن برنامه نویسی یاد گرفت.

در ضمن شما هم می‌توانید برای ما مطالب و مسئله ارسال کنید! مطالب و مسئله‌هایی که خودتان از آن‌ها لذت برده‌اید را به آدرس ketab@mhelli.ir ارسال نمایید.



مفلحیم پایه





اگر این فصل را به خوبی مطالعه کنی و کارهای فواسته شده را به دقت انجام دهی:

- با ویژگی‌های رایانه آشنا می‌شوی و می‌توانی تفاوت‌ها و شباهت‌های آن با انسان را توضیح دهی.
- با اجزای اصلی رایانه بهتر آشنا می‌شوی.
- با مشخصات زبان‌های برنامه‌نویسی و سه نماینده معروف از سه دسته مختلف آن‌ها آشنا می‌شوی.
- تعاریف الگوریتم، برنامه و هوش مصنوعی را یاد می‌گیری.
- با بازی جذاب لایت‌بات و نمونه کنترل روبات آن آشنا می‌شوی.

مقدمات رایانه

رایانه چیست؟

رایانه ابزاری است که بشر برای انجام بهتر بسیاری از کارهای خود ساخته است. اگر به اطراف خود با دقت نگاه کنید، ابزارها و دستگاه‌های الکترونیکی و دیجیتالی بسیاری را مشاهده می‌کنید؛ مانند تلویزیون، رادیو، یخچال، ماشین لباسشویی، تلفن همراه و از همه مهم‌تر خود رایانه! ویژگی‌های منحصر به فرد رایانه باعث شده تا در زمینه‌های مختلف، جایگزینی برای انسان باشد؛ اما این ویژگی‌ها از کجا آمده است؟ برای پاسخ به این سؤال، باید اطلاعات بیشتر و دقیق‌تری از ویژگی‌های رایانه داشته باشیم.



خستگی ناپذیری: رایانه تعطیل‌بردار نیست! برخلاف انسان که پس از مدتی خسته می‌شود، رایانه می‌تواند به‌طور مداوم و برای مدت‌های طولانی کار کند. دستگاه‌ها و رایانه‌های مختلف به‌طور شبانه‌روزی مشغول انجام وظایف و کارهایی هستند که به آن‌ها محول شده است. گاهی اوقات این دستگاه‌ها به قدری مشغول به فعالیت و خدمت‌رسانی هستند که هرگونه توقف در کار آن‌ها خسارات و ضررهای بزرگی به بار می‌آورد. از این نمونه دستگاه‌ها می‌توان به رایانه‌های خدمات‌دهنده در بانک‌ها، مراکز درمانی و اتاق‌های عمل و یا رایانه‌های موجود در برج‌های مراقبت فرودگاه‌ها اشاره کرد.



سرعت: برای انجام بسیاری از کارها از جمله پیش‌بینی هواشناسی، محاسبات رمزنگاری و ... نیازمند محاسبات پیچیده و طولانی‌مدت هستیم. رایانه که در لغت به معنای ماشین محاسبه‌گر است، این وظیفه مهم را بر عهده دارد. توانایی رایانه‌ها برای این منظور بسیار بالا است و با پیشرفت فناوری و ظهور سخت‌افزارها جدید، روزبه‌روز بالاتر هم می‌رود.

دقت: رایانه در انجام محاسبات علاوه بر سرعت از دقت بالایی هم برخوردار است. این ویژگی به ما اطمینان می‌دهد که حتی در موقعیت‌هایی که جان انسان‌ها در معرض خطر است به رایانه اعتماد کنیم و به آن اجازه دهیم به جای انسان تصمیم بگیرد. بهترین نمونه در این زمینه سیستم رایانه‌ای خلبان خودکار (Auto Pilot) است که برخی از وظایف مهم خلبان را به جای او انجام می‌دهد.

حافظه: حافظه انسان سالم برای مدت‌زمان کوتاهی اطلاعات را نگه می‌دارد و نگهداری حجم زیاد اطلاعات برای مدت‌زمان طولانی برای او کاری بسیار دشوار است؛ اما رایانه با این مشکل روبه‌رو نیست. رایانه می‌تواند انبوهی از اطلاعات را به مدت نامحدود در حافظه خود ذخیره کند و در صورت نیاز انسان، بدون هیچ‌گونه مشکل و یا تأخیر محسوسی اطلاعات درخواستی را در اختیار او قرار دهد.

ویژگی‌های بالا از مهم‌ترین مزایای رایانه نسبت به انسان است. این ویژگی‌ها موجب می‌شود تا انسان قرن حاضر در انجام بسیاری از کارهای روزمره خود از رایانه استفاده کند. از سوی دیگر رایانه انجام کارهایی را برای انسان ممکن ساخته که بدون استفاده از آن تقریباً برای ما غیرممکن بود. پیش‌بینی هواشناسی، سفر به سیارات دیگر، رمزگشایی برخی متون رمز شده از جمله کارهایی هستند که بدون استفاده از رایانه انجام آن‌ها برای ما ممکن نیست.



به نظرت رایانه چه ویژگی‌های دیگری دارد؟ چه کارهای دیگری می‌توانی مثال بزنی که انسان بدون کمک رایانه قادر به انجام آن‌ها نیست یا انجام آن‌ها برای انسان خطرناک است؟

امروزه رایانه تبدیل به یک ابزار ضروری و حیاتی در زندگی انسان شده است که استفاده و بهره‌مندی درست از آن نیازمند شناخت دقیق ساختار، توانایی‌ها و نحوه تعامل با رایانه می‌باشد.

انسان در برابر رایانه

برای درک بهتر ویژگی‌های رایانه‌ها مقایسه آن‌ها با انسان‌ها می‌تواند به ما کمک کند. در ادامه به بررسی شباهت‌ها و تفاوت‌های انسان و رایانه می‌پردازیم:

شباهت‌های انسان و رایانه

انسان اطلاعات را از محیط اطراف خود می‌گیرد و با محاسبه و سنجیدن این اطلاعات برای آینده خود تصمیم می‌گیرد. رایانه نیز مانند انسان می‌تواند با استفاده از اطلاعاتی که در اختیارش قرار می‌گیرد تصمیم‌گیری کند. شباهت دیگر آن‌ها قدرت بالای محاسبه و پردازش اطلاعات است. هرچند هنوز قدرتمندترین رایانه‌ها هم نمی‌توانند به سرعت مغز انسان در انجام محاسبات برسند.



تفاوت‌های انسان و رایانه

مهم‌ترین مزیت انسان نسبت به رایانه در داشتن خلاقیت و قوه ابتکار است. برخلاف ما انسان‌ها رایانه فاقد این ویژگی است؛ اما منظور از خلاقیت و ابتکار چیست؟ وقتی که ما با یک مسئله جدید ریاضی مواجه می‌شویم، با ایده‌ها و راه‌حل‌های مختلفی ممکن است آن را حل کنیم؛ ولی یک رایانه توانایی انجام این کار را ندارد و تنها قادر به انجام کارهایی است که انسان از ابتدا و در هنگام ساخت و برنامه‌ریزی آن برای او تعیین کرده است. از دیگر ویژگی‌های انسان، قابلیت یادگیری اوست. انسان پس از برخورد با اتفاقات و مسائل جدید، اطلاعات جدیدی را به حافظه خود می‌سپارد و بعدها در گرفتن تصمیمات خود، از اشتباهات گذشته و تجربیات تلخ و شیرین خود استفاده می‌کند؛ ولی رایانه این توانایی را ندارد. البته شاخه نسبتاً جدیدی از دانش رایانه به نام هوش مصنوعی به وجود آمده است که سعی در تهیه برنامه‌هایی برای رایانه دارد تا به وسیله آن‌ها یادگیری انسان را مدل کند تا در طی گذشت زمان رایانه هم بتواند یاد بگیرد.



چه شباهت‌ها و تفاوت‌های دیگری بین انسان و رایانه وجود دارد؟



پاسنگو باش

اجزای رایانه

همه شما ماشین حساب را دیده‌اید و با آن کار کرده‌اید، ماشین حساب شامل بخش‌های زیر است:

ورودی: صفحه کلیدی برای دریافت اطلاعات از انسان و انتقال آن به داخل ماشین حساب.

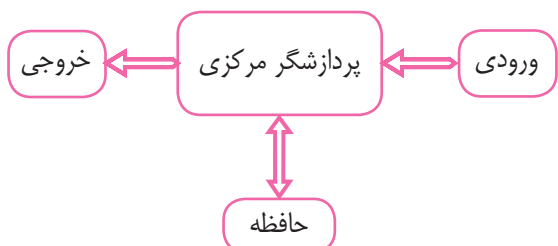
پردازشگر مرکزی: که اطلاعات ورودی را دریافت و با توجه به آن اعمال و محاسبات ریاضی لازم را انجام می‌دهد.

حافظه: برای نگهداری اطلاعات و نتایجی که از قبل محاسبه شده‌اند.

خروجی: یا صفحه‌نمایش LCD برای اینکه ماشین حساب نتایج خود را به انسان انتقال دهد.

این بخش‌ها در بسیاری از موجودات زنده هم وجود دارند. بدن شما از طریق حواس پنج‌گانه از محیط اطراف خود اطلاعاتی به دست می‌آورد و آن‌ها را از طریق نرون‌ها یا رشته‌های عصبی به سمت مغز ارسال می‌کند. مغز و یا واحد پردازشگر مرکزی بدن، با استفاده از اطلاعات دریافتی و اطلاعات قبلی که در حافظه شما ذخیره شده است، تصمیم می‌گیرد و طبق آن دستورات لازم را به اعضای مختلف بدن می‌دهد. اعضای مختلف بدن نیز وظیفه دارند با توجه به تصمیماتی که مغز می‌گیرد عمل کنند.

رایانه نیز از جهاتی بسیار شبیه یک ماشین حساب است. اما بخش‌های اصلی آن کدام‌اند؟ واحد پردازشگر مرکزی رایانه و یا در حقیقت مغز آن، درون جعبه رایانه قرار دارد و سایر قطعات را مدیریت می‌کند. دستگاه‌های دیگر مانند



صفحه کلید، موشواره و میکروفون نقش ورودی را ایفا می‌کنند و اطلاعات را از دنیای بیرون از رایانه گرفته و به داخل سیستم انتقال می‌دهند. دستگاه‌های دیگری مانند صفحه نمایش و چاپگر هم اطلاعات را از درون رایانه گرفته و به دنیای بیرون انتقال می‌دهند. به‌طور مشابه، برای ذخیره‌سازی اطلاعات در رایانه حافظه‌های مختلفی در نظر گرفته شده است.



پایب است
برای

انسان‌ها به زبان‌های متفاوتی صحبت می‌کنند: فارسی، عربی، ترکی، انگلیسی و ... برای اینکه انسان‌ها بتوانند باهم ارتباط برقرار کنند، باید بتوانند زبان یکدیگر را درک کنند. انسان‌هایی که به یک زبان مشترک سخن می‌گویند، مشکلی در این رابطه باهم ندارند، ولی کسانی که به یک زبان مشترک صحبت نمی‌کنند به یک مترجم نیاز دارند. برای مثال فرض کنید که به کشور آلمان سفر کرده‌اید ولی آلمانی بلد نیستید. در این صورت به یک مترجم نیاز دارید که هم آلمانی و هم فارسی بلد باشد. در این صورت شما می‌توانید به کمک این مترجم با افراد آلمانی زبان ارتباط برقرار کنید. دستگاه‌های مختلف در یک رایانه نیز همین‌طورند. موارد متعددی پیش می‌آید که زبان دو یا چند دستگاه مشترک نیست. پس آن‌ها نیز به یک مترجم برای فهم اطلاعات یکدیگر نیازمندند. مهم‌ترین قطعه در رایانه، ریزپردازنده (پردازنده مرکزی) است که باید بتواند با تمام بخش‌های موجود در یک رایانه ارتباط برقرار کند و آن‌ها را کنترل کند. برای این منظور از دستگاه‌هایی استفاده می‌شود که نقش یک واسطه (در حقیقت مترجم) را ایفا می‌کنند. این دستگاه‌ها (مدارات واسط) وظیفه دارند دستورات ریزپردازنده را به سایر بخش‌ها منتقل کنند و وضعیت و درخواست‌های آن‌ها را به ریزپردازنده برسانند. این دستگاه‌ها تقریباً برای تمام دستگاه‌های ورودی/خروجی لازم هستند.



از این به بعد می‌توانیم هر سیستم محاسبه‌گری که قابل برنامه‌ریزی باشد را رایانه بنامیم. این سیستم‌ها می‌توانند طیف گسترده‌ای از ماشین‌های لباس‌شویی تا ابررایانه‌های پیش‌بینی‌کننده وضع هوا را در بر بگیرند. هر رایانه از یک واحد پردازشگر مرکزی (CPU) برای اجرای دستورات و انجام محاسبات استفاده می‌کند. در کنار این واحد، قطعات ورودی، خروجی و واحد حافظه قرار دارند. داده‌ها از دنیای خارج به‌وسیله ورودی‌ها وارد رایانه شده و پاسخ نهایی توسط خروجی‌ها به کاربر ارائه می‌شود. واحد حافظه هم به ذخیره نتایج و اطلاعات موردنیاز برای انجام محاسبات می‌پردازد. مطمئناً بدون وجود حافظه‌ای که اطلاعات اولیه را در اختیار داشته باشد و پاسخ‌های نهایی را در خود ذخیره کند، محاسبات بی‌معنی و بی‌استفاده خواهند بود.

در رایانه سه نوع حافظه وجود دارد:

- حافظه پنهان
- حافظه موقت
- حافظه دائم

برای روشن شدن جایگاه هر یک از این سه نوع حافظه، از یک مثال استفاده می‌کنیم و بعد به شرح هر یک از این انواع می‌پردازیم. فرض کنید دانش‌آموزی در آزمون ریاضی شرکت کرده است. این دانش‌آموز برای پاسخ به سؤالات آزمون، مسیر زیر را طی می‌کند:

ابتدا با استفاده از ابزار ورودی یعنی چشم، صورت‌مسئله را مطالعه می‌کند. سپس داده‌های مسئله به حافظه مغز منتقل شده و مغز شروع به محاسبه می‌کند. تا جایی که مقدار محاسبات کم باشد، مغز می‌تواند از حافظه کوتاه‌مدت استفاده کرده و جواب این محاسبات را به‌دست آورد. حافظه کوتاه‌مدت مغز شبیه حافظه پنهان پردازنده است. برای انجام محاسبات پیچیده‌تر و یا حجیم‌تر، مغز به یک حافظه کمکی نیازمند است، این حافظه همان صفحه‌چرک‌نویس است که گنجایش بیشتری برای ذخیره‌سازی محاسبات دارد. این حافظه معادل حافظه موقت در رایانه می‌باشد. دانش‌آموز در پایان، پاسخ‌هایی که در هنگام انجام محاسبات در برگه چرک‌نویس نوشته است را به برگه پاسخ‌نامه منتقل می‌کند؛ برگه پاسخ‌نامه مثالی از حافظه دائم می‌باشد. با توجه به مثال فوق، تعریف دقیق‌تری از این سه نوع حافظه ارائه می‌دهیم:

حافظه پنهانی: این حافظه گنجایش بسیار کم و سرعت انتقال اطلاعات بسیار بالایی دارد، بنابراین برای انجام محاسبات سریع و کوچک استفاده می‌شود. مقدار این حافظه در هر رایانه محدود است و به‌نوعی نمایانگر قدرت پردازشی سیستم می‌باشد، نام دیگر این حافظه ثبات یا *Register* می‌باشد.

حافظه موقت: اطلاعات موردنیاز برای حل یک مسئله و یا محاسبات لازم برای حل یک مسئله بر روی این حافظه ذخیره می‌شوند. از ویژگی‌های این حافظه، سرعت کار بالا و پاک شدن اطلاعات آن در اثر قطع جریان برق است. حافظه *Ram* در این دسته از حافظه‌ها قرار دارد.

حافظه دائم: برنامه‌ها و نتایج نهایی محاسبات بر روی این حافظه ذخیره می‌شوند. می‌توان اطلاعات را بر روی این حافظه به‌طور دائمی ذخیره و نگهداری کرد چراکه محتوای این حافظه در اثر قطع جریان برق پاک نمی‌شود. هارددیسک‌ها نمونه مهمی از این دسته از حافظه‌ها هستند.

چگونه با رایانه ارتباط برقرار کنیم؟

حتماً شنیده‌اید که زبان رایانه صفر و یک است. اگر این‌طور باشد واقعاً زبان عجیبی است! در حقیقت در مدارهای الکترونیکی رایانه، محاسبات تنها با ارقام صفر و یک انجام می‌شوند، به این روش محاسبه، محاسبه در مبنای ۲ می‌گوییم. ما انسان‌ها محاسباتمان را در مبنای ۱۰ انجام می‌دهیم؛ به‌همین خاطر است که از ارقام ۰، ۱، ۲، ۳، ...، ۸ و ۹ استفاده می‌کنیم.

به‌عبارت‌دیگر رایانه تمام اعداد و دستورات را به‌صورت دنباله‌هایی از صفر و یک می‌فهمد؛ بنابراین برای ارتباط برقرار کردن با آن نیز باید به همین روش عمل کرد، یعنی باید با زبان صفر و یک (زبان ماشین) با رایانه صحبت کرد! این کار بسیار مشکل است و عملاً انجام کارهای کمی پیچیده با این روش غیرممکن خواهد بود؛ اما این مشکل راه‌حلی دارد: زبان‌های برنامه‌نویسی که در بخش بعدی به معرفی مختصر آن‌ها می‌پردازیم.



زبان‌های برنامه‌نویسی

زبان‌های برنامه‌نویسی، برقراری ارتباط ما با رایانه را راحت‌تر می‌کنند. ساختار این زبان‌ها بسیار شبیه به ساختار زبان‌های طبیعی طراحی شده تا کاربر بتواند به‌راحتی منظور خود را بیان کند. درنهایت، دستورات نوشته‌شده در یک زبان برنامه‌نویسی، به زبان رایانه یا زبان ماشین (ترکیبی از صفرها و یک‌ها) ترجمه می‌شود تا برای رایانه قابل‌فهم باشد. با این روش می‌توانیم به‌راحتی از رایانه بخواهیم که کارهای موردنظر ما را انجام دهد.

زبان‌های برنامه‌نویسی در سطوح مختلفی طبقه‌بندی می‌شوند، به‌این ترتیب که هر چه زبان برنامه‌نویسی به زبان انسان نزدیک‌تر باشد به آن زبان، سطح بالاتر می‌گوییم و هر چه به زبان رایانه (زبان ماشین) نزدیک‌تر باشد، آن زبان را سطح پایین‌تر می‌گوییم. براساس این تقسیم‌بندی، نگاهی به معروف‌ترین زبان‌های برنامه‌نویسی موجود (از سطح پایین به بالا) می‌اندازیم:

زبان اسمبلی (Assembly)

این زبان نزدیک‌ترین زبان به زبان ماشین یا زبان صفر و یکی رایانه است. برنامه‌نویسی با زبان اسمبلی، هرچند از کار کردن با صفر و یک‌ها ساده‌تر است، ولی هنوز فاصله زیادی با زبان انسان دارد. از طرف دیگر زبان اسمبلی زبان بسیار قدرتمندی است، زیرا به‌طور مستقیم با سخت‌افزار رایانه در ارتباط است و قادر است پردازنده و اجزای جانبی آن را کاملاً در کنترل خود بگیرد.



```
.model small
.stack
.data
message db "Hello
World!", "$"

.code

main proc
mov ax,seg message
mov ds,ax

mov eh,09
lea dx,message
int 21h

mov ax,4c00h
int 21h
main endp
end main
```

▶ برنامه‌ای برای نوشتن یک پیغام ساده در زبان اسمبلی!

```
#include <iostream.h>
int main()
{
    cout << "Hello World!" << endl;
    return 0;
}
```

همان برنامه در زبان C++



python

```
print("Hello World!")
```

زبان‌های C و C++

زبان C، زبان سطح بالاتر از زبان اسمبلی است، پس می‌توانید حدس بزنید که برنامه‌نویسی با آن باید ساده‌تر از اسمبلی باشد. همچنین این زبان، زبان قدرتمندی نیز هست. خوب است بدانید که زبان C را با استفاده از زبان اسمبلی ساخته‌اند. نسخهٔ جدیدتر زبان C به نام C++ شناخته می‌شود.

زبان پایتون (PYTHON)

دستورات زبان پایتون بسیار شبیه به کلمات زبان انگلیسی هستند، بنابراین یادگیری و برنامه‌نویسی با آن، بسیار راحت‌تر از زبان C است و در نتیجه این زبان از زبان C سطح بالاتر است. این زبان در عین ساده بودن یکی از قدرتمندترین و محبوبترین زبان‌های برنامه‌نویسی حال حاضر دنیا است. این زبان توسط فان روسوم در سال ۱۹۹۱ در کشور هلند طراحی شد و در توسعهٔ آن در این سال‌ها، همواره تأکید بر حفظ دو ویژگی سادگی و بازدهٔ بالای آن بوده است. یادگیری زبان پایتون و برنامه‌نویسی با آن بسیار ساده است. ما نیز به همین دلیل آن را برای آموزش در این کتاب انتخاب کرده‌ایم. شما نیز به‌زودی لذت برنامه‌نویسی در پایتون را حس خواهید کرد! و در آخر همان برنامه به زبان پایتون، آیا می‌توانید حدس بزنید که این برنامه چه کاری انجام می‌دهد؟!

سایر زبان‌ها

در دنیای برنامه‌نویسی از زبان‌های برنامه‌نویسی بسیاری استفاده می‌شود که هرکدام ویژگی‌ها و توانایی‌های مخصوص به خود را دارند. از جمله آن‌ها می‌توان به زبان‌های جاوا، C#، دلفی، پرل، فورترن، PHP و ... اشاره کرد.



الگوریتم

تعریف

شما در خانه نشسته‌اید و مشغول درس خواندن هستید. دوستان با شما تماس می‌گیرند و مشکلی را با شما در میان می‌گذارند: او می‌خواهد وضو بگیرد، ولی وضو گرفتن یادش رفته! دوستان از شما درخواست کمک می‌کنند، چگونه به او کمک خواهید کرد؟
حتماً روش انجام کار را مرحله‌به‌مرحله برای او شرح خواهید داد. مثلاً خواهید گفت: اول آب را در دست راستش بریزد، سپس آب را روی صورتش ریخته و تمام صورتش را با دست راستش بشوید. در مرحلهٔ دوم باید دست چپش را پر از آب کند، سپس آب را روی دست راستش بریزد و دستش را از آرنج به پایین بشوید؛ و همین‌طور مراحل را به‌صورت کامل تا مسح پای چپ به او توضیح می‌دهید.



دستورالعمل درست کردن چای چیست؟ اگر بخواهیم راه حل درست کردن چای را به کسی که بلد نیست، توضیح دهی، چگونه این کار را می‌کنی؟



محمّد ابن موسی خوارزمی دانشمند بزرگ ایرانی، اولین کسی است که این راه را برای بیان روش انجام کارها پیشنهاد داد. لفظ الگوریتم نیز از نام الخوارزمی گرفته شده است.

ویژگی‌ها

هر الگوریتم عبارت از مجموعه دستوراتی است که دارای ویژگی‌های زیر باشد:

۱. متن هر دستور باید برای اجراکننده قابل فهم و ساده باشد.

۲. متن هر یک از دستورات باید دقیق و غیرمبهم باشد.

۳. دستورات باید به ترتیب صحیح اجرا شوند.

۴. دستورات باید شرط پایان یا توقف داشته باشند.

وقتی می‌خواهیم به رایانه دستور بدهیم تا برای ما کاری انجام بدهد، ابتدا باید به الگوریتم آن کار فکر کنیم؛ یعنی باید ببینیم دستورالعمل چیست و با انجام چه مراحل، کاری که می‌خواهیم انجام می‌شود. پیدا کردن الگوریتم یکی از مهم‌ترین بخش‌های حل مسئله‌های رایانه‌ای است. گاهی اوقات لازم است چند ساعت و حتی چند روز به یک مسئله فکر کنیم و الگوریتم‌های مختلف را بررسی کنیم تا یک الگوریتم خوب و قابل اجرا بر روی رایانه پیدا کنیم.

بعد از پیدا کردن الگوریتم، باید آن را به زبان برنامه‌نویسی بنویسیم تا رایانه قادر به فهم و عمل کردن به آن باشد.

برنامه

رایانه دستگاهی است که توانایی اجرای برنامه‌های مختلف را دارد. اگر به این مسئله توجه داشته باشیم که رایانه یک ابزار است و مانند سایر ابزارها وظیفه‌آسان‌تر کردن کارهای ما را دارد، متوجه می‌شویم که برنامه‌ها باید کاربرد خاصی برای ما داشته باشند. مثلاً بتوانند تمرینات ریاضی ما را حل کنند (!) یا اطلاعات کارکنان یک شرکت را به سرعت و راحتی در اختیار ما بگذارند و یا به پیش‌بینی وضعیت هوا در ساعات آینده بپردازند.

در تمام این مثال‌ها برنامه اطلاعاتی را از کاربر می‌گیرد که به آن ورودی می‌گوییم، سپس این اطلاعات را پردازش می‌کند تا به جواب خواسته‌شده کاربر برسد و در نهایت پاسخ به‌عنوان خروجی برنامه به کاربر ارائه می‌شود.

(ورودی، خروجی، پردازش)

به نظرت در حال حاضر، مهمترین کارهایی که رایانه‌ها انجام می‌دهند چیست؟



پاسنگو باتش

یک برنامه از دستوراتی شکل گرفته که پشت سر هم اجرا می‌شوند. این دستورات در هر زبان برنامه‌نویسی شکل و کارکرد خاصی دارند. از پشت سر هم قرار گرفتن دستورات یک برنامه و اجرای آن توسط رایانه، کار خواسته‌شده برای ما انجام می‌شود. کسی که می‌خواهد با زبان خاصی برنامه بنویسد، باید ابتدا دستورات موردنیازش را بشناسد و شکل و کاربرد آن‌ها را بداند. در مرحله بعد برنامه‌نویس، با پشت سر هم قرار دادن این دستورات، الگوریتم موردنظرش را به برنامه تبدیل می‌کند.



گری کاسپاروف در حال مسابقه با دیپ جونیور در سال ۲۰۰۳
این مسابقه منجر به تساوی ۳-۳ گشت

هوش مصنوعی

شاید این خبر را شنیده باشید: «رایانه توانست گری کاسپاروف، مرد اول شطرنج جهان را شکست دهد». شطرنج، یکی از پیچیده‌ترین بازی‌های فکری است و برنده شدن در آن، نتیجه مستقیم هوش بازیکن می‌باشد. وقتی به دو عبارت بالا نگاه می‌کنیم، طبیعتاً نتیجه می‌گیریم که رایانه یک موجود هوشمند است. آیا این نکته به این معناست که انسان رقیبی سرسخت پیدا کرده است؟

بیا به نگاهی به الگوریتم بازی شطرنج بیندازیم. هنگام بازی، حریف با هر حرکت خود، ما را در موقعیتی جدید قرار می‌دهد. در این موقعیت جدید، ما امکان حرکات مختلفی را داریم که می‌توان تمام احتمالات ممکن را بر روی کاغذ نوشت. تعداد این حالات ممکن

است زیاد باشد، ولی وقتی کمی دقت می‌کنیم متوجه می‌شویم که حریف هم به ازای هر یک از این حالات، امکان حرکات مختلفی را دارد. پس تعداد کل حرکات احتمالی در این دو مرحله بسیار زیاد می‌شود. حال اگر این بحث را در مورد تمام مراحل یک بازی کامل شطرنج گسترش دهیم می‌بینیم احتمالات ممکن را باید با ارقام نجومی بیان کرد! در واقع راه صحیح شطرنج بازی کردن این است که ما در هر مرحله به‌ازای هر حرکت خود، بتوانیم حرکات حریف را در مراحل بعدی بسنجیم و بهترین حالت را انتخاب کنیم. وقتی بهترین حالت حاصل خواهد شد که ما بتوانیم تمام احتمالات ممکن را تا انتهای یک بازی پیش‌بینی کنیم. واضح است که این کار عملاً غیرممکن است.



می‌توانی بازی‌های دیگری مثال بزنی که پیش‌بینی بازی تا آخرین مرحله غیرممکن باشد؟ بازی‌ای که قابل پیش‌بینی باشد چه‌طور؟

روش بالا روشی است که هر شطرنج‌بازی در ذهن خود اجرا می‌کند. بازیکنان مبتدی می‌توانند یک یا دو مرحله بعد را پیش‌بینی کنند و هرچه بازیکن حرفه‌ای‌تر باشد، می‌تواند مراحل بیشتری را پیش‌بینی کند. رایانه هم با همین روش شطرنج بازی می‌کند. البته به علت سرعت بالای محاسبات، رایانه می‌تواند در زمان کوتاهی مراحل بعدی بازی را پیش‌بینی کند. رایانه برای انجام هر کاری، اگرچه به‌ظاهر هوشمندانه باشد، از یک الگوریتم مشخص پیروی می‌کند. به این الگوریتم‌ها که باعث می‌شوند رایانه رفتاری (به‌ظاهر) هوشمندانه داشته باشد، الگوریتم‌های هوش مصنوعی گفته می‌شود. هوش مصنوعی یکی از جالب‌ترین و پرکاربردترین حوزه‌های علم نرم‌افزار است. الگوریتم‌های هوش مصنوعی، قابلیت حل مسائل پیچیده در زمان کوتاه را برای ما فراهم می‌کنند. در ادامه این فصل با رباتی آشنا می‌شویم که قادر است تعداد محدودی از دستورات ما را اجرا کند.



کنترل روبات LIGHTBOT

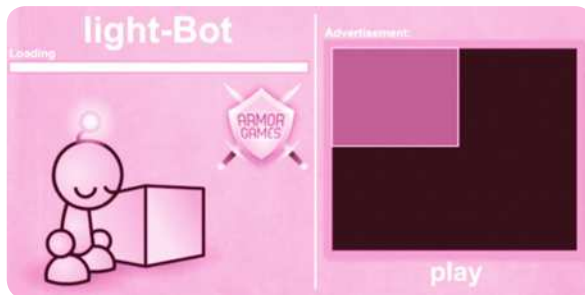
لایت‌بات کیست؟

لایت‌بات، نام رباتی است که ساخته شده تا ما به کمک آن، نحوه طراحی الگوریتم‌های رایانه‌ای و تبدیل آن‌ها با زبان قابل فهم برای رایانه را تمرین کنیم. دقیقاً مانند رایانه، این روبات از خود هیچ هوشی ندارد، بلکه تنها به دستوراتی که ما به او می‌دهیم عمل می‌کند. البته او به هر دستوری عمل نمی‌کند! بلکه فقط دستورات خاصی را می‌فهمد و قادر است به آن‌ها عمل کند. از این جهت هم دقیقاً مثل رایانه و زبان‌های برنامه‌نویسی است.

شاید فکر کنید لایت‌بات بدنی لاغر و ضعیف دارد، ولی او بدون خستگی و با دقت بسیار همه دستورات شما را اجرا می‌کند و هرگز از دستورات سرپیچی نمی‌کند. همان‌طور که گفتیم او هیچ هوش و خلاقیتی ندارد و فقط دستوراتی خاص که به زبان خودش به او داده می‌شود را اجرا می‌کند. پس باید زبان او را یاد بگیریم و بفهمیم به چه دستوراتی می‌تواند عمل کند.

شروع بازی

بازی را از روی سی دی همراه کتاب باز کنید. پس از باز کردن بازی، با این محیط روبرو می شوید:



سپس روی گزینه *New Game* کلیک کنید و کلید فاصله (*space*) را بزنید.




خوش آمدید! شما وارد اولین مرحله بازی شده‌اید. همان‌طور که می‌بینید لایت‌بات در یک محیط شطرنجی ایستاده و منتظر دستورات شما است.




دستورات قابل فهم


دستورات قابل فهم لایت‌بات در گوشه سمت راست بالای صفحه، قرار دارند. به لایت‌بات فقط ۷ دستور قابل فهم دارد! بیایید ببینیم معنی هر دستور چیست:

یک خانه به جلو حرکت کن. 

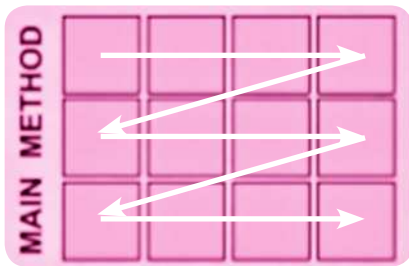
به سمت راست خودت بچرخ. 

به سمت چپ خودت بچرخ. 

ببر! 

کلید را بزن. 

در مورد دو دستور آخر بعداً توضیح می‌دهیم.



چگونه به لایت بات دستور بدهیم؟

لایت بات حافظه محدودی دارد و فقط می توان به او ۱۲ دستور داد تا او آن ها را پشت سر هم اجرا کند. ترتیب اجرای دستورات به شکل روبه رو است:

برای دستور دادن به او، از بخش دستورات قابل فهم، دستور مورد نظر خود را بگیرید و روی یکی از مربع های پایینی رها کنید. سپس برای آنکه به لایت بات بگویید تا دستورات را اجرا کند، دکمه *GO!* را بزنید.

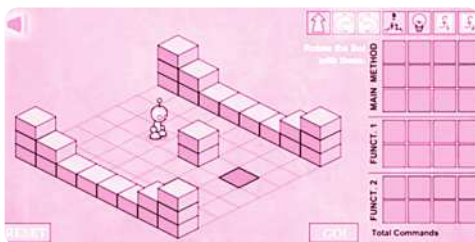


برای متوقف کردن لایت بات می توانید روی دکمه *STOP* بزنید.

چگونه بازی کنیم؟



کار لایت بات روشن کردن چراغ های نقشه است! در هر مرحله، باید به گونه ای به لایت بات دستور بدهیم تا با انجام آن ها، به خانه های آبی نقشه (که در آن ها چراغ وجود دارد) برود و کلید را بزند تا چراغ روشن شود. به مرحله اول بازی دقت کنید. خانه آبی دو خانه جلوتر قرار دارد. پس باید به لایت بات دستور بدهیم دو خانه به جلو برود و کلید را بزند:



حالا با زدن دکمه *GO!* به لایت بات می گوییم دستورات را اجرا کند. بله! موفق شدیم تا مرحله اول را بگذرانیم. با زدن کلید فاصله وارد مرحله بعد می شویم. مرحله دوم کمی پیچیده تر است.



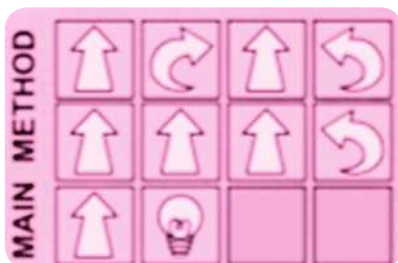
یک مانع بر سر راه وجود دارد که یا باید از روی آن بپریم و یا آن را دور بزنیم. ابتدا پریدن و سپس دور زدن را امتحان کنید و سعی کنید این مرحله را پشت سر بگذارید. دقت کنید که:

- پریدن فقط روی مانعی ممکن است که با موقعیت فعلی لایت بات یک واحد اختلاف ارتفاع داشته باشد.

- چرخش به معنی حرکت به راست یا چپ نیست، بلکه لایت بات فقط سر جای خودش می چرخد. همان طور که متوجه شدید، راه حل این مرحله یکتا نیست و می توان با راه های مختلف چراغ این مرحله را روشن کرد و مرحله را به پایان رساند. یکی از راه حل ها را در زیر می بینید: مراحل به تدریج پیچیده تر می شوند و مغز شما را بیشتر به کار خواهند انداخت. لایت بات که خسته نمی شود! شما هم سعی کنید بدون خستگی مراحل را تا پایان ادامه دهید.

دستورهای $f1$ و $f2$

شاید برایتان سؤال شده باشد که چهار ردیف مربعی که در زیر ۱۲ مربع اصلی چیدن دستورها وجود دارد برای چیست؟



دو ردیف بالا مربوط به دستور $f1$ و دو ردیف پایین مربوط به دستور $f2$ می شوند.

با استفاده از این مربع ها می توانیم یک «مجموعه دستور» تعریف کنیم که با استفاده از دستور $f1$ اجرا می شوند؛ یعنی هر جا به لایت بات دستور $f1$ را بدهیم، او هشت دستوری که در مربع های مربوط به $f1$ نوشتیم را اجرا می کند. هشت مربع پایینی هم مربوط به $f2$ است و هرگاه دستور $f2$ را دادیم اجرا می شوند.

						f_1	f_2
FUNCT. 2	MAIN METHOD						
FUNCT. 1	MAIN METHOD						
FUNCT. 2	MAIN METHOD						

FUNCT. 1	MAIN METHOD		f_1	f_1	f_1
MAIN METHOD					



پایه می‌کنند؟!

FUNCT. 2	MAIN METHOD		f_1	f_1	f_1
FUNCT. 1	MAIN METHOD		f_2	f_2	
MAIN METHOD					

اگر درون f_1 دستور f_2 را استفاده کنیم چه اتفاقی می‌افتد؟
دستورات زیر چه می‌کنند؟



فشار بسوزان

لغت نامه



واژه علمی	ترجمه	واژه علمی	ترجمه
Display	صفحه نمایش	Register	تَبَّات
LCD	صفحه نمایش کریستال مایع	Case	جعبه
Mouse	موشواره	Printer	چاپگر
Microphone	میکروفون	RAM	حافظه موقتی
CPU	واحد پردازشگر مرکزی	Keyboard	صفحه کلید
		Monitor	صفحه نمایش

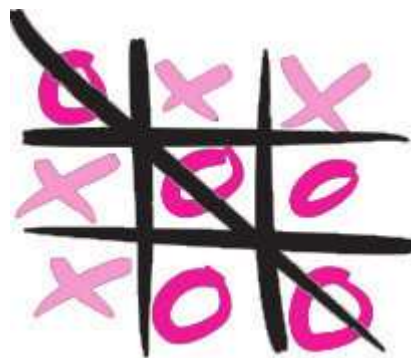
لمح بنی کن



۱. زبان در عین ساده بودن یکی از قدرتمندترین و محبوبترین زبان‌های برنامه‌نویسی حال حاضر دنیا است.
۲. یک برنامه از شکل گرفته که پشت سر هم اجرا می‌شوند.
۳. به الگوریتم‌هایی که باعث می‌شود رایانه رفتاری (به‌ظاهر) هوشمندانه داشته باشد، الگوریتم‌های گفته می‌شود.

۱. ویژگی‌های،، و از مهم‌ترین مزایای رایانه نسبت به انسان است.
۲. مهم‌ترین مزیت انسان نسبت به رایانه در داشتن و قوه است.
۳. هر سیستم که قابل باشد را رایانه می‌نامیم.
۴. حافظه گنجایش و سرعت بسیار بالا دارد.

۱. اشکالات الگوریتم زیر را مشخص کنید.
۲. می‌خواهم الگوریتم بازی دوز را به تو بگویم! ابتدا بگذار خانه وسط، سپس سعی کن ببری!



۱. اطلاعات در حافظه به‌صورت دائمی ذخیره و نگهداری می‌شوند.
۲. از ویژگی‌های حافظه سرعت کار بالا و پاک شدن اطلاعات آن در اثر قطع جریان برق است.
۳. در نهایت نوشته‌شده در یک زبان برنامه‌نویسی، به زبان رایانه یا (ترکیبی از صفرها و یک‌ها) ترجمه می‌شود تا برای رایانه باشد.
۴. هر چه زبان برنامه‌نویسی به زبان نزدیک‌تر باشد
۵. به آن زبان، سطح بالاتر می‌گوییم و هر چه به زبان نزدیک‌تر باشد، آن زبان را سطح پایین‌تر می‌گوییم.
۶. زبان نزدیک‌ترین زبان به زبان ماشین یا زبان صفر و یکی رایانه است.



سطح مبتدی

۱. برای رفتن به مدرسه (از خانه) چه مراحلی را طی می‌کنید؟ آن‌ها را به ترتیب بنویسید.
۲. برای ضرب دو عدد دو رقمی در هم چه مراحلی را باید انجام دهیم؟ الگوریتم آن را بنویسید.

سطح متوسط

۳. می‌خواهیم از بین ۱۰۰ عدد زیر، بزرگ‌ترین عدد را پیدا کنیم. ابتدا سعی کنید خودتان این کار را بکنید. سپس روش (الگوریتم) خود را بنویسید.

۱۴۴, ۷۸۱, ۷۶۸, ۳۵۵, ۹۲۹, ۲۵۳, ۲۶۴, ۷۱۲, ۷۶۵, ۷۶۸, ۹۵۱, ۷۶۹, ۹۹۰, ۵۷۹, ۴۶۰, ۲۱, ۱۰۹, ۹۶۳, ۸۱۲, ۱۱۴, ۴۷۶, ۸۵۰, ۲۸, ۵۸۰, ۶۵۹, ۱۲۵, ۸۴۴, ۶۰, ۲۳۴, ۳۷۸, ۶۰۶, ۳۹۱, ۶۹۳, ۸۴۴, ۴۶۰, ۸۵, ۸۸۱, ۲۲۲, ۸۴۶, ۹۹۰, ۶۶۱, ۵۹۶, ۶۲۲, ۳۸۷, ۳۵۳, ۴۷۱, ۲۹۳, ۴۳۳, ۴۵۰, ۶۴۸, ۱۸۹, ۵۷۶, ۹۷۲, ۲۱۱, ۸, ۴۷۴, ۶۳۲, ۴۷۵, ۶۹۴, ۶۴۶, ۴۳۲, ۴۲۵, ۹۵۲, ۱۹۸, ۸۲۳, ۱۸۶, ۹۰۳, ۶۲۸, ۶۳۶, ۸۷۹, ۵۷۲, ۹۵۶, ۷۰۱, ۸۷۰, ۷۵۱, ۴۳۶, ۴۲۲, ۶۴۹, ۸۵۴, ۳۹, ۱۲۳, ۲۷۷, ۹۲۵, ۶۲۲, ۶۲۰, ۵۸۵, ۶۲۴, ۹۵۸, ۹۵۷, ۵۷۶, ۴۰۶, ۹۸۷, ۶۱۰, ۱۳۵, ۴۷۴, ۵۲, ۸۶۲, ۸۲, ۹۰۹, ۱۶۸

سطح پیشرفته

۴. فرض کنید در جایی از یک دشت صاف یک هویج قرار گرفته است و در فاصله‌ای از آن یک خرگوش غیر خلاق قرار گرفته است به گونه‌ای که نمی‌تواند آن هویج را ببیند. این خرگوش دستورات زیر را می‌فهمد:
 - یک درجه به راست بچرخ
 - یک درجه به چپ بچرخ
 - بو بکش
 - بوی فعلی را با بوی قبلی مقایسه کن (تا بفهمد کدام بیشتر است. تذکر: تنها امکان مقایسه به بوی قبلی وجود دارد نه سایر بوها)
 - در امتدادی که قرار داری بدو تا به هویج برسی.
 الگوریتمی بنویسید که این خرگوش با اجرای آن بتواند هویج را در دشت پیدا کند. (راهنمایی: ابتدا باید جهت بوی بیشینه را پیدا کند و سپس به سمت آن بدود)
۵. اگر بخواهیم ۱۰۰ عدد سوال ۳ را به ترتیب، از کوچک به بزرگ بنویسیم، چگونه این کار را می‌کنیم؟ ابتدا سعی کنید خودتان این کار را بکنید. سپس روش (الگوریتم) خود را بنویسید.