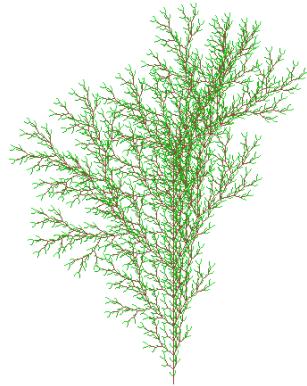




دانشکده مهندسی کامپیوتر

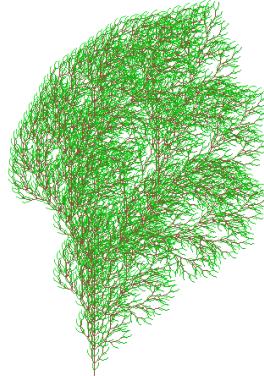
امتحان عملی اول



سید صالح اعتمادی

ملیحه حاجی حسینی

نیمسال اول ۱۴۰۲-۱۴۰۳



زمان امتحان:

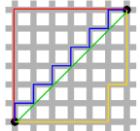
۱۸۰ دقیقه

نکات قابل توجه:

- استفاده از هیچ تابع آماده‌ای مجاز نیست. اگر شک دارید، از یکی از استادی سوال کنید.
- صفحه‌های باز روی کامپیوتر شما فقط صفحه cmd و صفحه vscode می‌تواند باشد. هیچ صفحه/برنامه دیگری نباید باز باشد.
- جز برگه‌ای که همراه دارید، استفاده از هیچ‌گونه منبع دیگری مجاز نیست.

۱ فاصله منهتن

تابع `q0_manhattan_distance` دو مختصات را در صفحه مختصات دریافت کرده و فاصله منهتن آن دو را برمی‌گرداند. فاصله منهتن بین دو نقطه عبارت است از کوتاهترین مسافتی که در راستای محور افقی یا عمودی طی می‌شود تا به نقطه دیگر برسیم. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.



۲ نزدیکترین همسایه

تابع `q1_closest_neighbor` را به گونه‌ای پیاده‌سازی کنید که یک لیستی از اعداد به عنوان ورودی دریافت کرده و فاصله نزدیکترین دو عددی متولی را برگرداند. برای این سوال از تابع `sort`, `min`, `max` نمی‌توان استفاده کرد. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.

۳ جمع دو لیست

تابع `q2_add_nums` دو لیست با طول یکسان از اعداد به عنوان ورودی دریافت کرده و یک لیست با همان طول برمی‌گرداند که شامل جمع اعداد متناظر در دو لیست ورودی می‌باشد. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.

۴ نزدیکترین دو دوست

تابع `q3_closest_two_friends` لیستی از اعداد به عنوان ورودی دریافت کرده و دو عدد که از همه به هم نزدیکترین هستند را انتخاب می‌کنند. سپس مکان (اندیس) عدد اول، اندیس عدد دوم و فاصله آنها را برمی‌گرداند. دقت کنید، ترتیب اعداد برگشتی: ابتدا اندیس کوچکتر، سپس اندیس بعدی و سپس فاصله می‌باشد. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.

۵ اجرای قانون

تابع `q4_apply_rule_once` یک رشته حرفی و یک قانون جایگزینی به عنوان ورودی دریافت کرده و سپس این قانون را روی رشته ورودی اعمال کرده نتیجه را برمی‌گرداند. قانون جایگزینی عبارت است از یک لیست با دو عضو: عضو اول: یک کاراکتر، عضو دوم یک رشته حرفی. روش اعمال قانون به این شکل است که در رشته حرفی ورودی از ابتدا به ازای هر کاراکتر، اگر با کاراکتر قانون یکسان بود، رشته قانون در خروجی قرار داده می‌شود، اگر نه، همان کاراکتر رشته ورودی بدون تغییر در خروجی قرار داده می‌شود. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.

۶ اجرای قانون‌ها

تابع `q5_apply_rules_once` یک رشته حرفی و لیستی از قانون‌های جایگزینی را به عنوان ورودی دریافت کرده و قانون‌ها را به ترتیب روی رشته ورودی اعمال می‌کند. روش اعمال قانون شبیه سوال قبل است. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.

۷ اجرای پیاپی قانون‌ها

تابع `q6_apply_rules_n` یک رشته حرفی، لیستی از قانون‌ها و تعداد تکرار را به عنوان ورودی دریافت کرده و قانون‌ها را به ترتیب روی رشته ورودی اعمال می‌کند و اعمال قانون‌ها را به اندازه تعداد تکرار، اعمال می‌کند. دقت کنید همه قانون‌ها یکبار اعمال بشوند، سپس همه قانون‌ها بار دیگر. نه اینکه یک قانون را چند بار تکرار کنیم، بعد قانون بعدی. برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید.

۸ لاکپشت حرف گوش کن

تابع q7_turtle_follows_instruction دستورهای موجود در رشته را طبق مسافت، زاویه و لیستی از رنگ‌های را به عنوان ورودی دریافت کرده و دستورهای شامل مکان‌ها، زاویه‌ها و رنگ‌ها را بر می‌گرداند. دستورها عبارتند از:

- F: برو جلو به اندازه مسافت داده شده
- +: بپیچ به راست به اندازه زاویه داده شده
- -: بپیچ به چپ به اندازه زاویه داده شده
- CX: رنگ خودت رو عوض کن به رنگ داده شده در مکان X در لیست رنگ‌های ورودی

برای نمونه ورودی و خروجی به تست مربوطه مراجعه کنید. گرفتن زاویه، مکان و رنگ برای یک متغیر لاکپشت با دستورهای زیر انجام می‌شود:

- t.position()
- t.heading()
- t.color()

۹ لاکپشت مورد اعتماد

تابع q8_turtle_follows_instruction مشابه تابع سوال قبل است با دو تفاوت: یک: چیزی بر نمی‌گرداند. دو: دو دستور جدید داریم. یکی ']' (براکت باز) و یکی '[' (براکت بسته). کاری که لاکپشت باید انجام دهد این است که با دیدن براکت باز مکان، زاویه و رنگ خود را به یک لیست اضافه کند. سپس با دیدن براکت بسته باید آخرین مکان/زاویه/رنگ را از آخر لیست حذف کند و مکان/زاویه/رنگ لاکپشت را طبق آن تنظیم کند. در هنگام تغییر مکان نباید «رد» لاکپشت بماند. یعنی قبل از تغییر مکان از دستور penup و پس از تغییر مکان از دستور pendown استفاده شود. دستورهای تغییر مکان/زاویه/رنگ عبارتند از:

- t.setpos(pos)
- t.setheading(angle)
- t.color(color)

۱۰ اعتماد به لاکپشت حرف گوش کن برای اجرای پیاپی قانون‌ها زیباست

همانطور که متوجه شدید سوال های ۸ و ۹ به هم مربوط بوده و سوال‌های ۵، ۶ و ۷ هم به هم مربوط هستند. این سوال، به سوال‌های ۹ و ۷ مربوط می‌شود. به این ترتیب که یک رشته ورودی (شامل دستورهای لاکپشتی) و تعداد قانون به عنوان ورودی داده می‌شود. این قانون‌ها باید روی رشته ورودی با تعداد تکرار خواسته شده اجرا شده و نتیجه را به لاکپشت مورد اعتماد برای اجرا تحويل دهد. لاکپشت‌های مورد اعتماد با اجرای قانون، محیط مناسب خودشان برای زندگی را خواهند ساخت. خروجی این فایل بصورت خودکار در فایل‌های beautiful_rules1.ps و beautiful_rules1.py ذخیره می‌شوند و توسط اساتید حل تمرین تصحیح می‌شوند. اگر بخواهید از درستی آن اطمینان حاصل کنید، دستور turtle.mainloop() را از کامنت خارج کنید تا خروجی آنرا قبل از نوشته شدن توی فایل ببینید. فایل‌های ps به همراه فایل‌های امتحان با add/commit/push باید در آژور فرستاده شوند.

موفق باشید!