

در ادامه تعریف دو پروژه (دفترچه یادداشت و ترسیم لاک پشتی) آمده است. شما می‌توانید یکی از آنها را به دلخواه انتخاب کرده و انجام دهید. توجه فرمایید که پروژه اول (دفترچه یادداشت) به صورت تک‌نفره می‌باشد ولی پروژه دوم (ترسیم لاک پشتی) را می‌توانید در گروه‌های دونفره نیز انجام دهید.



## دفترچه یادداشت

### مقدمه

در این پروژه سعی داریم با مفهوم ساختارها و فایل‌ها برنامه‌ای بنویسیم که یک دفترچه یادداشت ساده را شبیه‌سازی نماید.

### قابلیت‌ها و خصوصیات دفترچه یادداشت

الف) هر یادداشت باید دارای بخش‌های زیر باشد:

۱) عنوان

۲) تاریخ ذخیره‌ی آخرین تغییرات

۳) متن اصلی

۴) تاریخ یادآوری

توجه کنید که تاریخ شامل روز، ماه، سال، ساعت و دقیقه می‌باشد.

ب) برنامه‌ی شما باید دارای امکانات زیر باشد:

۱) اضافه کردن یک یادداشت جدید: با اضافه کردن یک یادداشت جدید، تاریخ ذخیره‌ی آخرین تغییرات یادداشت باید به‌صورت خودکار با تاریخ جاری سیستم مقارنه‌ی شود. سعی کنید اضافه کردن به‌صورتی انجام شود که یادداشت‌های موجود بر اساس تاریخ یادآوری، مرتب شده باشند.

۲) جستجوی یادداشت بر اساس:

✓ عنوان

✓ تاریخ ذخیره‌ی آخرین تغییرات یادداشت

✓ متن اصلی

✓ تاریخ یادآوری

پس از نمایش نتیجه جستجو باید امکاناتی برای نمایش یادداشت‌های بعدی و قبلی (از نظر تاریخ یادآوری) باید وجود داشته باشد.

۲) حذف یک یادداشت

- ۳) نمایش یادداشتهای روز (یادداشتهایی که تاریخ یادآوری آنها روز جاری می باشد)
- ۴) نمایش تمام یادداشتهای موجود که بر حسب تاریخ یادآوری مرتب شده باشند. در این نمایش یادداشتهایی که تاریخ یادآوری آنها گذشته باشد، نباید نمایش داده شوند.
- ۵) ویرایش یک یادداشت: توجه کنید پس از ویرایش یک یادداشت، تاریخ ذخیره ی آخرین تغییرات یادداشت باید به صورت خودکار با تاریخ جاری سیستم مقارن شود.
- ۶) یک منو برای انتخاب یکی از موارد فوق

## تذکرات

- ۱- برنامه ها باید به زبان **C++** نوشته شوند.
- ۲- در نوشتن این پروژه الزاماً باید از ساختارها و فایل استفاده شود.
- ۳- به برنامه های شبیه به هم نمره ی منفی تعلق خواهد گرفت.
- ۴- همه ی دانشجویان باید به صورت حضوری در موعد مقرر پروژه خود را تحویل دهند.
- ۵- اضافه کردن امکانات اضافه به برنامه ( گرافیکی کردن و ...) نمره ی اضافه خواهد داشت.
- ۶- میزان ساده بودن کار با برنامه در بالاتر رفتن نمره تاثیر خواهد داشت.



## ترسیم لاک پشتی

### مقدمه

در این پروژه شما قرار است برنامه‌ای مانند **Python Turtle** شبیه‌سازی نمایید. در ابتدا برای آشنایی با این برنامه به لینک زیر مراجعه کرده و آن را دانلود کنید:

<http://pythonturtle.org/>

برنامه‌ی **Python Turtle** هر دستوری را که از ورودی می‌گیرد درجا کامپایل کرده و اجرا می‌کند ولی برنامه‌ای که شما می‌نویسید کل ورودی را یک جا از ورودی استاندارد یا فایل خوانده و پس از خواندن آخرین ورودی، خروجی نهایی را نمایش می‌دهد.

لاک‌پشت در شروع کار در خانه‌ی مرکزی جدول قرار دارد و رویش به سمت بالاست و با دستورات داده شده حرکت می‌کند. شما لاک‌پشت را با کاراکتر **T** در صفحه نمایش نشان می‌دهید.

### دستورات

دستوراتی که لاک‌پشت می‌تواند اجرا کند عبارتند از:

- ۱- **set\_board(a,b)**: اندازه‌ی صفحه نقاشی را تعیین می‌کند. توجه کنید که **a** و **b** اعدادی فرد می‌باشند. **a** تعداد سطرها و **b** تعداد ستون‌ها را مشخص می‌کند. این دستور تنها می‌تواند اولین دستور ورودی باشد و اگر اولین دستور نبود اندازه‌ی صفحه به صورت پیش فرض  $101 \times 101$  در نظر گرفته می‌شود.
- ۲- **save\_picture(MyPic)**: نقاشی را که تاکنون کشیده‌اید در فایلی که نام آن به وسیله‌ی رشته‌ی **MyPic** تعیین می‌شود ذخیره می‌کند.
- ۳- **go(n)**: باعث حرکت لاک‌پشت به اندازه‌ی **n** خانه بر روی صفحه‌ی نمایش به سمت جلو می‌شود.
- ۴- **Turn(n)**: این دستور باعث چرخش لاک‌پشت به اندازه‌ی **n** درجه در جهت عقربه‌های ساعت می‌شود. توجه داشته باشید که **n** باید ضربی از ۹۰ باشد.
- ۵- **color(c)**: رنگ ردپای لاک‌پشت را تعیین می‌کند. در این برنامه منظور از رنگ ردپا، کاراکتری است که به عنوان ردپای لاک‌پشت نمایش داده می‌شود (**c**). به عنوان مثال **color('#')** باعث می‌شود ردپای لاک‌پشت با کاراکتر **#** نمایش داده شود. مقدار پیش فرض کاراکتر **\*** می‌باشد.
- ۶- **reset()**: این دستور صفحه نمایش را پاک می‌کند. بعد از اجرا این دستور لاک‌پشت در محل اولیه خود قرار می‌گیرد.

۷- invisible(): این دستور لاک پشت را نامرئی می کند.

## ۸- `visible()`: لاک پشت را مرئی می کند.

۹- `pen_up()`: با اجرای این دستور لاک پشت ردپایی از خود بر صفحه نمی‌گذارد.

۱۰- pen\_down() : با اجرای این دستور لاک پشت دوباره بر روی صفحه ردپا می‌گذارد.

۱۱- `clear()`: با اجرای این دستور تمام ردپاهای لاک پشت پاک می شود ولی لاک پشت جابه جا نمی شود.

۱۲- `for i in range(a,b):` معادل با عبارت `for( i=a ; i<b ; i++ )` می باشد.

۱۳- `for i in range(a)`: معادل با عبارت `for( i=0 ; i<a ; i++ )` می باشد.

۱۴ - end\_for: پایان حلقه‌ی for را تعیین می‌کند.

۱۵- end: با اجرای این دستور ترسیم خاتمه یافته و نتیجه اجرای دستورات نمایش داده می‌شود.

## نمونه‌ای از اجرا

```
set_board(35,35)
for i in range(1,10)
    go(3*i)
    turn(90)
    go(3*i)
    turn(90)
end_for
end
```

[illegible]

- ۱- برنامه شما باید دارای انتخاب‌هایی برای دریافت دستورات از ورودی استاندارد و یا خواندن از یک فایل باشد.
- ۲- دستورات باید حساس به بزرگی یا کوچکی کاراکترها نباشند.
- ۳- توصیه می‌شود قبل از نوشتن برنامه، `python turtle` را دانلود کرده و با آن کار کنید.
- ۴- برنامه‌ها باید به زبان `C++` نوشته شوند.
- ۵- به برنامه‌های شبیه به هم نمره‌ی منفی تعلق خواهد گرفت.
- ۶- همه‌ی دانشجویان باید به‌صورت حضوری در موعد مقرر پروژه خود را تحویل دهند.
- ۷- اضافه کردن امکانات اضافه به برنامه ( گرافیکی کردن و ...) نمره‌ی اضافه خواهد داشت.
- ۸- میزان ساده بودن کار با برنامه در بالاتر رفتن نمره تاثیر خواهد داشت.