

در دانشگاه فردوسی مشهد اپلیکیشنی جهت هم‌پیمایی (Carpooling) توسعه داده شده است که در آن دانشجویان سفر به دانشگاه را برای روز های بعد اعلام می‌کنند و زمان و موقعیت مکانی مبدا خود را ثبت می‌نمایند. هم‌چنین برای هر سفر مشخص می‌کنند که آیا خودرو دارند یا خیر. در واقع تعیین می‌کنند که راننده هستند و یا مسافر. هدف از توسعه این اپلیکیشن کاهش ترافیک و تردد خودروها در پردیس دانشگاه است. پس از ثبت اطلاعات، الگوریتمی براساس فاصله مسافرها را به راننده‌ها تخصیص می‌دهد و هر یک از مسافرها می‌توانند در خصوص این سفر نظر خود را ثبت نمایند.



برنامه ای بنویسید که

[StuNum, X, Y, T, D]

StuNum : شماره دانشجویی (۱۰۰ الی ۹۹۹)

X, Y : مختصات مکانی (طول و عرض جغرافیایی بین ۱۰ الی ۹۹)

T : نوع درخواست که یکی از حالت های راننده (R)، مسافر (P) یا هر دو (B) است.

• B به این موضوع اشاره دارد که ترجیح دانشجوی به مسافر بودن است اما در صورت لزوم می‌تواند راننده نیز باشد.

D : تاریخ سفر (بصورت پیش فرض روز بعد از سیستم را بگیرد و در صورت وارد کردن روز های بعد اولویت با تاریخ وارد شده است).

• بدیهی است که کاربر نمی‌تواند تاریخ های گذشته و یا همان روز را وارد کند.

• می‌توانید برای تاریخ از کتابخانه های مرتبط استفاده کنید.

منو ایجاد کنید:

۱- ثبت درخواست: از کاربر سوال کند که بصورت فردی (I) یا بصورت گروهی (G) ثبت می‌کند، در صورت ثبت فردی

یک لیست شامل [StuNum, X, Y, T, D] و در صورت گروهی یک لیست است که خود شامل n لیست

[StuNum, X, Y, T, D] است و n تعداد افراد گروه است

۲- اصلاح یک درخواست: شماره دانشجویی را از کاربر می‌گیرد و درخواست‌های همان روز و روزهای بعد را نمایش

دهد (حداکثر سه درخواست)، کاربر یکی را انتخاب کرده و بجز شماره دانشجویی بقیه موارد را می‌تواند اصلاح و یا

حذف کند.

• بدیهی است با هر بار اصلاح درخواست لیست خواسته شده در بخش سوم باید به روز رسانی گردد.

۳- تخصیص مسافرها به راننده: براساس فاصله و ظرفیت مسافران هر خودرو ۴ نفر در نظر گرفته شود. راننده‌ها را

براساس فاصله‌شان از دانشگاه اولویت شوند (مختصات دانشگاه [0, 0] در نظر گرفته شود). ابتدا مسافرها به خودرویی

که دورترین فاصله را از دانشگاه دارد، تخصیص داده شوند. ۴ مسافری که به خودرو کمترین فاصله را دارند، به خودرو تخصیص داده میشوند و سپس همین روند برای سایر خودروها و براساس اولویت آنها تکرار می‌شود. در این روند ابتدا مسافرها به راننده‌ها تخصیص داده می‌شوند و در صورتی که مسافری باقی مانده باشد، از افرادی که در گروه (B) قرار دارند بعنوان راننده استفاده می‌شود. به طور مشابه، اگر راننده‌ای باقی مانده باشد افراد گروه B به عنوان مسافر به آنها تخصیص داده می‌شوند. نتیجه در یک دیکشنری ذخیره می‌شود که شامل تاریخ، شماره سفر، راننده، مسافر می‌باشد (شماره سفر در الگوریتم تخصیص تعیین میشود و راننده-مسافرایی که با هم مسیر را طی میکنند که شماره سفر یکسان دارند).

- اگر تعدادی دانشجو بدون راننده ماندند لیست اسامی دانشجویان را بصورت کلیدی دلخواه در دیکشنری ذخیره کنید و زیر منویی برای این بخش درست کنید که فقط دانشجویان بدون راننده را نشان دهد.

۴- ثبت نظر کاربران برای هر سفر: این نظرات متن هستند و در یک دیکشنری ذخیره میشوند که کلید آن شماره سفر است.

۵- اصلاح نظر: کاربر با وارد کردن شماره دانشجویی خود و تاریخ سفر، ابتدا مشخصات سفر و نظر قبلی خود را مشاهده میکند و بعد متن نظر را اصلاح می‌نماید.

- فرض شده است که در هر روز هر فرد فقط یک سفر دارد.
- خطاهای زیر نیز کنترل شوند :
 - اگر درخواست سفری ثبت نشده است، کاربر نمیتواند از منو اصلاح درخواست استفاده نماید.
 - اگر تخصیص انجام نشده است، کاربر نمیتواند ثبت و اصلاح نظر انجام دهد.

-
- تمرین باید بصورت function نوشته شود.
 - وردی ها از طریق تابع دریافت میشوند و به پیش فرض ها دقت کنید.
 - این تمرین مربوط به بخش پایتون مقدماتی است و واضح است باید از دستورهایی آن حداکثر استفاده شود و نمیتوانید از کتابخانه هایی مثل Math , Pandas , Numpy استفاده کنید.
 - در صورت استفاده از Editor ها دقت کنید که `charset = utf-8` تنظیم شده باشد.
 - برای چاپ زیباتر جدول‌ها میتوانید از کتابخانه های مرتبط استفاده نمایید (اختیاری و فقط برای ساده شدن)
 - برای راحتی کار با تاریخ ها میتوانید از کتاب خانه های مرتبط با تاریخ و زمان استفاده کنید.
 - استفاده از شی گرایي با توجه به مفاهیم آن دارای امتیاز مثبت خواهد بود.

- استفاده از ظاهر مناسب در تهیه گزارشات دارای امتیاز مثبت خواهد بود.
- فایلها با صورت اعلام شده نامگذاری شوند و در سامانه VU بارگذاری شوند. (شماره دانشجویی + P۱)
- واضح است به تمرینهایی که پس موعده مقرر تحویل داده شوند و یا همراه با خطا باشند، نمرهای تخصیص داده نمیشود.
-