

به نام خدا

امتحان میانترم برنامه سازی پیشرفته

مدت امتحان : ۶۰ دقیقه

در این سوال می خواهیم بازی tic tac toe را پیاده سازی کنیم و از شما خواسته شده است که توابعی که برای این کار لازم است را پیاده سازی کنید و در نهایت با استفاده از این توابع بازی را پیاده سازی نمایید. در صورت نیاز توابع دیگری را پیاده سازی کرده و از آنها در پیاده سازی توابع نامبرده استفاده کنید.

زمین بازی tic tac toe را به صورت dictionary زیر در نظر بگیرید. که در آن tl, tm, tr, ml, mm, mr, ll, lm, lr کلیدهای دیکشنری هستند، و مقادیر کلید های می توانند O, X یا رشته شامل space باشند. که به ترتیب نشان دهنده مقدار X در آن مکان، مقدار O در مکان و هیچ مقدار در آن مکان هستند.

tl(top left)	tm(top middle)	tr (top right)
ml(middle left)	mm (middle middle)	mr(middle right)
Ll(low left)	lm (low middle)	lr (low right)

```
board = {'tl': 'X', 'tm': ' ', 'tr': ' ',  
'ml': ' ', 'mm': ' ', 'mr': ' ',  
'll': ' ', 'lm': ' ', 'lr': 'O'}
```

توصیف بازی

در هنگام شروع بازی، کاربر باید دو مقدار را به ترتیب و با فاصله وارد کند. ۱- نام کاربری، ۲- آدرس سیستم

نام کاربری می تواند هر ترکیبی از حروف، اعداد و کاراکتر باشد. و همچنین نباید تکراری باشد. فرض کنید فایلی به نام user.txt در کنار برنامه قرار دارد که نام های کاربری افراد مختلف به همراه آدرس سیستم آنها (هر کدام در یک خط) در آن موجود است. (نام کاربر و آدرس سیستم آن با - از یک دیگر جدا شده اند). هنگامی که کاربر نام کاربری خود را وارد کرد، در صورتیکه نام تکراری بود از او خواسته می شود که دوباره نام خود را وارد نماید. و در صورتیکه تکراری نبود، باید نام کاربری کاربر به همراه آدرس آن (که با فاصله از هم جدا شده اند) هم به نام های موجود در فایل user اضافه شود.

ورودی بعدی آدرس سیستم کاربر است که باید به فرمت زیر باشد.

آدرس کاربر از شش قسمت تشکیل شده است که قسمت ها با : از هم جدا شده اند. و هر قسمت نمایش هگزادسیمال یک عدد بین 0 تا 2¹⁶ می باشد. مقدار زیر یک نمونه آدرس را نشان می دهد.

A0B3:8012:D0E5:0000:0000:0000

نام کاربری و آدرس در یک خط خوانده می شوند که با فاصله از یکدیگر جدا شده اند. در صورتیکه هر کدام مشکل داشتند باید از کاربر خواسته شود که آنرا دوباره وارد کند و این کار آنقدر تکرار شود تا مشکل برطرف شود.

پس از وارد کردن نام کاربری و آدرس، نوبت بازی انتخاب می شود (که X باید اول بازی کند یا O) کاربر همیشه X است. بنابراین در صورتیکه نوبت بازی X باشد، از کاربر خواسته می شود که وارد کند، در کدام خانه می خواهد مهره خود را قرار دهد. پس از انتخاب کاربر، سیستم به صورت تصادفی از بین گزینه های موجود یکی را انتخاب می کند. و مهره سیستم در آنجا قرار داده می شود.

و بازی به همین ترتیب ادامه پیدا می کند که یا یک نفر ببرد و یا اینکه مهره های هر دو طرف که هر کدام ۳ مهره است، تمام شود.

وقتی بازی تمام شد، به کاربر نشان داده می شود که چند بار تاکنون برده است. و سپس از او پرسیده می شود که آیا میخواهد باز هم بازی را ادامه دهد؟ کاربر باید از بین (Y-Y-n-N) یکی را انتخاب کند. در صورتیکه پاسخ کاربر Y یا y بود، بازی دوباره از ابتدا شروع می شود. در پایان هر دور بازی، تعداد دفعاتی که کاربر تا کنون برده است به او نشان داده می شود.

پیاده سازی

قسمت main بازی تا حدودی برای شما نوشته شده است. شما نیاز است که توابع خواسته شده را پیاده سازی کند و سپس در صورت لزوم در قسمت main بازی تغییراتی دهید تا به صورت درست اجرا شود.

در ادامه توابع و شرح هر کدام آورده شده است.

file_redear	ورودی: file_name : نام فایلی که اطلاعات را از آن می خواند	این تابع یک generator است. تمام خطوط فایل در این generator خوانده می شوند و هر بار که next آن صدا زده می شود، به عنوان خروجی username و address که در یک خط فایل قرار دارند، برگشت داده می شود.
get_user_inputs		این تابع ورودی های کاربر که نام کاربری و آدرس سیستم آن است را در یک خط و با فاصله از کاربر می گیرد. در صورتیکه نام کاربری تکراری باشد و یا آدرس سیستم کاربر با فرمت درست وارد نشده باشد، باید به او خطا داده شود و از او خواسته شود ورودی ای خطا داشته دوباره وارد کند. در صورتیکه اطلاعات کاربر درست باشد، این اطلاعات و فایل user اضافه شده و در نهایت این تابع username و address را به صورت خروجی برمی گرداند.
get_choice(turn)	turn دارای مقدار x یا o است که مشخص می کند نوبت سیستم است یا نوبت user	در این تابع در صورتیکه نوبت کاربر باشد، از او ورودی گرفته می شود (ورودی باید یکی از مقادیر ... ,tl,tm باشد و در صورتیکه خانه ای کاربر وارد می کند، قبلا پر شده باشد، باز از او خواسته می شود که مقدار درست را وارد کند. در صورتی که نوبت سیستم باشد، از بین خانه های خالی مقدار به صورت تصادفی انتخاب می شود. مقدار انتخاب شده، چه توسط کاربر باشد چه توسط سیستم به عنوان خروجی تابع برگشت داده می شود.
choose_first_turn		از بین دو مقدار x یا o یکی را به صورت تصادفی انتخاب می کند.
who_win		هر بار که صدا زده می شود، مشخص می کند که چه کسی برده است. برنده می تواند x، o یا هیچ کدام باشد، که به عنوان خروجی برگشت داده می شود.