



پروژه عملی درس مبانی برنامه‌نویسی (مهندسی برق)

پروژه شماره چهار

بهار ۱۴۰۱

اهداف پروژه:

۱. آشنایی با پیاده‌سازی بازی کوریدور (Quoridor)
۲. استفاده و به کارگیری مناسب از داده ساختارهای زبان برنامه‌نویسی C++
۳. تلاش در جهت ارائه روش‌ها و الگوریتم‌های مناسب و بهینه برای حل چالش‌های پیاده‌سازی پروژه
۴. مهارت مدیریت زمان در آماده‌سازی پروژه در مدت زمان محدود با کیفیت مطلوب
۵. آشنایی با نحوه تقسیم کار و کار تیمی

مقدمه:

همان‌طور که تابه‌حال در درس مبانی برنامه‌نویسی به زبان C++، با داده ساختارهای مناسب جهت ذخیره و بازیابی اطلاعات (متغیرهای int، double، string، آرایه‌ها و...) آشنا شده‌اید، در این پروژه از شما انتظار می‌رود با توجه به مطالبی که تاکنون آموخته‌اید، بازی کوریدور را شبیه‌سازی کنید.

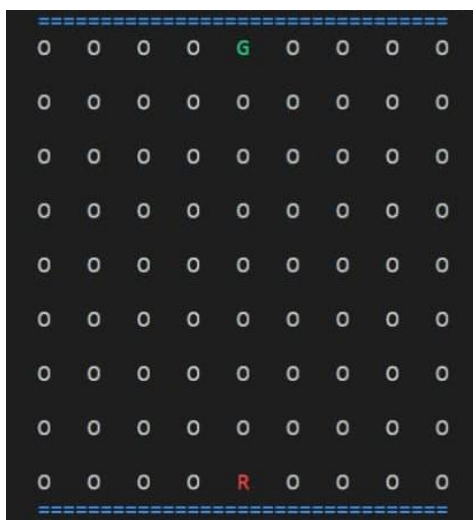


معرفی و روند بازی:

کوریدور بازی فکری مهیجی است که به صورت دو نفره نفره انجام می شود.

- قوانین برای بازی دو نفر:

در ابتدا موانع را در مکان خود قرار می دهیم. هر بازیکن آدمک خود را در خانه وسط اولین ردیف خود، قرار می دهد.

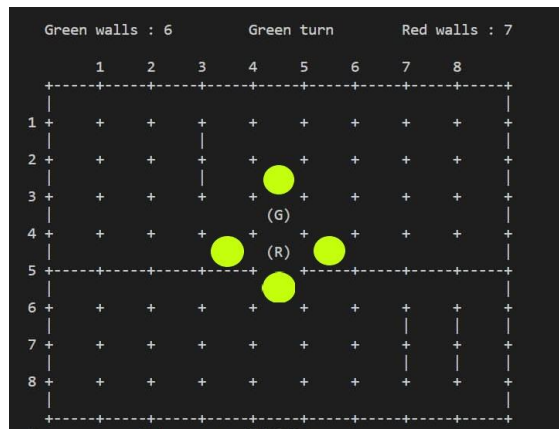


- طریقه بازی:

هر بازیکن می تواند در نوبت خود، یا آدمک خود را حرکت دهد و یا یک دیوار در صفحه قرار دهد. در حالتی که بازیکن هیچ دیواری نداشته باشد، باید آدمک خود را حرکت دهد.

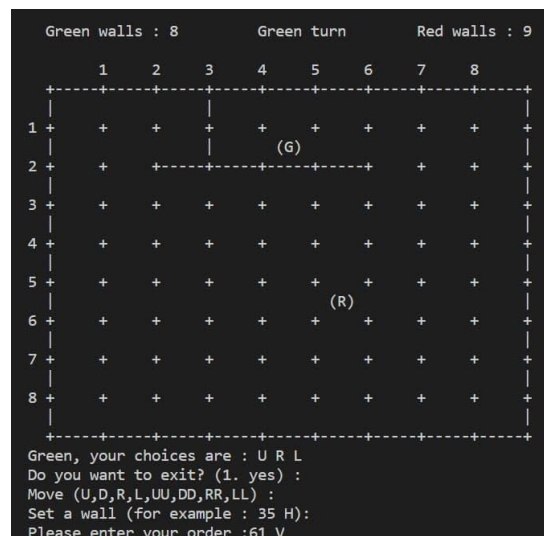
- حرکت آدمک ها

در هر نوبت آدمک ها فقط می توانند یک خانه در راستای افقی یا عمودی حرکت کنند. آدمک ها نمی توانند از روی دیوار ها بپرند. وقتی دو آدمک در حالتی مقابل هم قرار می گیرند که هیچ مانعی بین آنها نباشد آدمکی که نوبتش رسید می تواند از روی آدمک دیگر بپرد. اگر مانعی در پشت آدمک حریف قرار داشته باشد، بازیکن باید آدمک خود را به سمت راست یا چپ آدمک حریف ببرد.

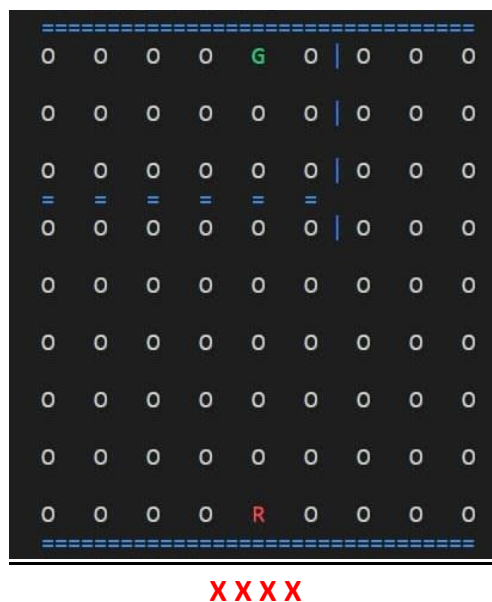


● قرار دادن دیوارها

این بازی ۲۰ دیوار دارد که به طور مساوی بین بازیکنان تقسیم شده تا آخر بازی دست هر بازیکن محفوظ است. موانع باید با هدف تسهیل حرکت آدمک بازیکن و یا در جهت جلوگیری از دسترسی آسان بازیکن مقابل به پایان صفحه استفاده شود. دیوارها باید به گونه ای در شیارهای خالی بین هر خانه قرار بگیرد قرار داده شوند که فقط دو خانه را کامل بپوشانند.



نکته مهم: آدمک حریف نباید به طور کامل محصور شود و باید راهی برای حرکت به سمت مقصد داشته باشد. (شکل زیر نباید رخ دهد).



- برنده بازی

برنده بازی کسی است که بتواند زودتر به شمالی ترین ردیف مقابل خود برسد.

- موده‌های بازی

در ابتدای بازی، بازیکنان می‌توانند یکی از حالت‌های تعد حرکت بی نهایت یا تعداد حرکت محدود را انتخاب کنند. در حالت اول، محدودیتی روی تعداد حرکت آدمک‌ها وجود ندارد. در حالت دوم، مثلاً عدد ۲۰ به عنوان حداکثر تعداد حرکت‌های ممکن در نظر گرفته می‌شود. اگر هیچ کدام از دو آدمک پیش از ۲۰ حرکت به مقصد نرسیدند، بازی متوقف شده و عملاً مساوی می‌شود.

نکته: تعیین تعداد حرکت‌ها به چند روش قابل انجام است. به عنوان مثال می‌توان در ابتدای بازی از کاربر خواست تا آن را تعیین کند یا این که خود بازی یک حالت پیش فرض برای تعیین آن داشته باشد. اما حداقل چیز مهم این است که اگر کاربر مقدار را مشخص می‌کند، یک سقفی برای آن تعیین شود تا کاربر اعداد غیر منطقی مثلاً ۱۰۰۰ را به عنوان محدودیت وارد نکند.

نکته: به عنوان بخش امتیازی می‌توانید بازی را در حالت چهارنفره پیاده‌سازی کنید. قوانینی که برای بازی چهار نفره وضع شده‌اند، همان قوانین دو نفره است ولی تفاوتی در آن وجود دارد و آن هم این است که پرشش از روی بیش از یک آدمک ممنوع است.

- مراحل کلی پیاده‌سازی بازی

نکته: توجه کنید اسامی و جملات به کاررفته فقط به عنوان نمونه و جهت فهم بهتر شما می‌باشند و گرنه شما می‌توانید از جملات و مفاهیم مشابه برای نشان دادن خروجی به کاربر استفاده کنید.

۱- در ابتدای کار که برنامه اجرا می‌شود، منویی به کاربر نشان داده شود که شامل دو گزینه باشد:

1- start game

2- exit

در صورتی که کاربر گزینه اول را انتخاب کند باید مود بازی را انتخاب کند و سپس بازی شروع شود و در صورتی که گزینه دوم وارد شود از بازی خارج شود.

۲- بعد از شروع بازی باید منویی به کاربر نمایش داده شود که شامل صفحه بازی، شخصی که نوبتش هست و سه گزینه زیر می‌باشد:

1- Add Wall

2- Move

3- exit to menu

```

** WELCOME TO QUORIDOR **
you have two choices
  1 . Start game
  2 . Exit
Please enter an option to : 1
-----

You know the rules let's go straight to the GAME!
=====
0 0 0 0 G 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 R 0 0 0 0
=====
1 . Add Wall
2 . Move
3 . Exit Game

```

۳- در صورتی که گزینه اول وارد شود باید یه پیغام نمایش داده شود که دیوار، در چه راستایی (عمودی یا افقی) و کجا باید قرار داده شود:

Enter your wall place and Type of wall placement ('start-x start-y Horizontal/ Vertical') :

برای مثال اگر کاربر به صورت زیر ورودی را وارد کند:

Enter your wall place and Type of wall placement ('start-x start-y Horizontal/ Vertical') :

1 2 Horizontal

یعنی در ردیف اول و ستون دوم دیوار را به صورت افقی قرار بده.

✓ در صورتی که دیوارها در جای نامعتبری قرار بگیرند و یا امکان گذاشتن دیوار در مکان موردنظر وجود نداشته باشد و یا تعداد دیوارهای بازیکن مورد نظر تمام شده باشد، باید خطای مناسب به کاربر نمایش داده شود.

❖ نکته:

شما برای نمایش صفحه بازی نیازی به استفاده از محیط گرافیکی نداشته و کافیت صفحه بازی را به صورت خروجی از ۰ و ۱ ها و یا کاراکترهای دلخواه خودتان نمایش دهید:

```

Do you want to exit? (1. yes) :
Move (U,D,R,L,UU,DD,RR,LL) :
Set a wall (for example : 35 H):
Please enter your order :D

Green walls : 6      Red turn      Red walls : 7

      1  2  3  4  5  6  7  8
+---+---+---+---+---+---+---+
1 +   +   +   +   +   +   +   +
  |   |   |   |   |   |   |   |
2 +   +   +---+---+---+---+---+
  |   |   |   |   |   |   |   |
3 +   +   +   +   +   +   +   +
  |   |   |   |   |   |   |   |
4 +   +   +   +   +   +   +   +
  |   |   |   |   |   |   |   |
5 +   +   +   +---+---+---+---+
  |   |   |   |   |   |   |   |
6 +   +   +   +   +   +   +   +
  |   |   |   |   |   |   |   |
7 +   +   +   +   +   +   +   +
  |   |   |   |   |   |   |   |
8 +   +   +   +   +   +   +   +
  |   |   |   |   |   |   |   |
+---+---+---+---+---+---+---+
Red, your choices are : U D L

```

نحوه ارسال:

کل فایل های موردنیاز برای اجراشدن پروژه را به همراه یک فایل PDF که شامل توضیحات برنامه است، در قالب یک فایل ZIP با فرمت Name&LastName_StudentID.zip ارسال کنید.

نکته: توجه کنید که پس از ارسال فایل پروژه در سایت، قادر به تغییر کدها تا روز ارائه نیستید و در ارائه، پروژه ارسالی در سایت مورد بررسی قرار می گیرد.

موفق باشید