

بازی Stratego یا فتح پرچم

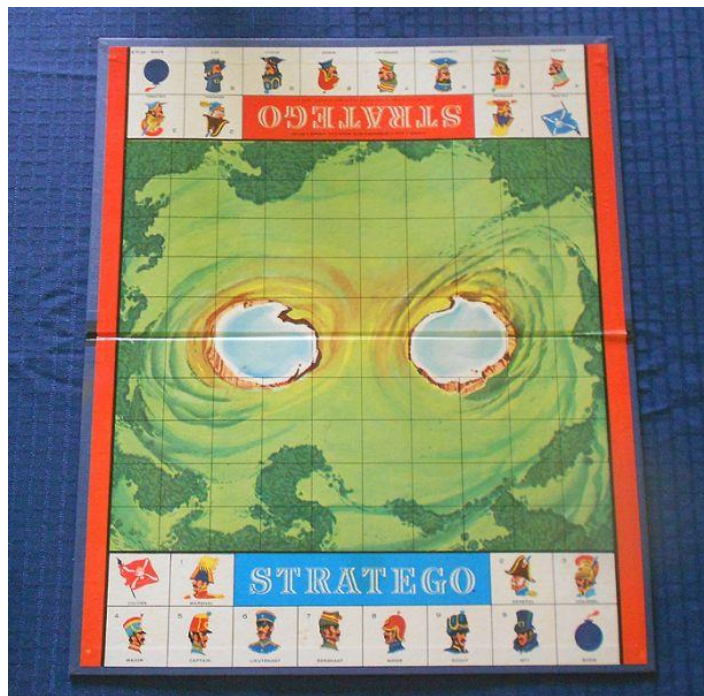
بازی Stratego یک بازی مهیج است که دو نفر آن را در یک صفحه 10×10 بازی می‌کنند که در شکل ۱ نمایی از آن نشان داده شده است.

	1	2	3	4	5	6	7	8	9	10
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										

شکل ۱: نمایی ساده از صفحه بازی Stratego

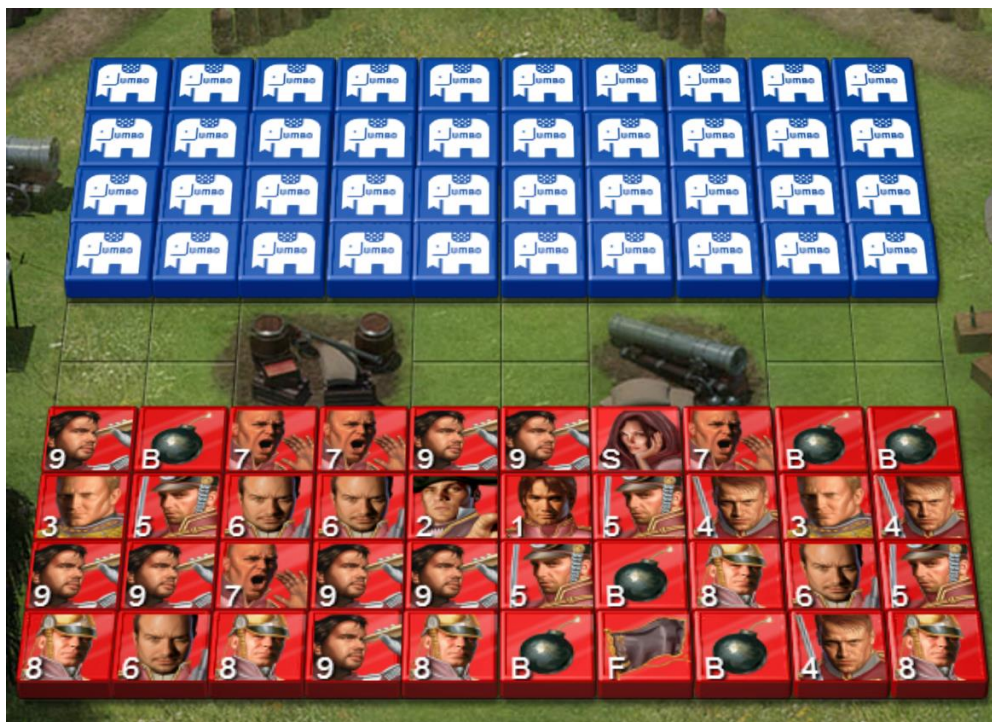
در این بازی هر شخص ۴۰ مهره در اختیار دارد که می‌تواند به اختیار خود آنها را در ۴۰ خانه روبروی خود بچیند. یعنی یک بازیکن مهره‌های خود را در سطرهای ۱، ۲، ۳ و ۴ و بازیکن دیگر مهره‌های خود را در سطرهای ۷، ۸، ۹ و ۱۰ می‌چیند. در ابتدای بازی هیچ مهره‌ای از هیچ بازیکنی در سطرهای ۵ و ۶ وجود ندارد.

مهره‌ها دو وجه دارند که در یک وجه آن شماره مهره مورد نظر و نام آن نوشته شده است، اما در یک وجه آن هیچ چیزی نوشته نشده است. نمایی از صفحه اولیه بازی در شکل ۳ نشان داده شده است. مهمترین نکته بازی این است که هر بازیکن مهره‌های خود را می‌بیند، اما از مهره‌های حریف اطلاعی ندارد و نمی‌بیند هر مهره حریف در چه خانه‌ای قرار گرفته است. این نکته در شکل ۳ نشان داده شده است. تنها در هنگام زدن مهره حریف برای بازیکنان مشخص خواهد شد.



شکل ۲: صفحه کاغذی بازی Stratego

در کل زمان انجام بازی مهره های هیچ یک از دو بازیکن اجازه ورود به خانه هایی که در شکل ۱ با رنگ آبی نمایش داده شده اند، در واقع رنگ آبی نشاندهنده دریاچه است. در شکل ۳ این خانه ها Block شده نشان داده شده اند.



شکل ۳: یک حالت ابتدایی نمونه از وضعیت قرار گرفتن مهره ها در ابتدای بازی Stratego

قوانین حرکت مهره ها در بازی Stratego

۴۰ مهره ای که هر بازیکن در بازی Stratego در اختیار دارد در جدول ۱ نمایش داده شده است. قوانین حرکت در این بازی به این صورت می باشد که هر مهره می تواند ۱ خانه به یکی از چهار جهت بالا، پایین، راست و چپ حرکت کند. همانطور که در جدول ۱ مشخص است، مهره شماره ۹ تنها مهره ای است که می تواند به تعداد دلخواه حرکت به یکی از چهار جهت راست، چپ، بالا و پایین حرکت کند.

جدول ۱: مهره های بازی Stratego برای یک بازیکن

قابلیت ویژه	تعداد موجود برای هر بازیکن	شماره مهره	نام مهره
زدن همه مهره ها	1	1	Marshal
زدن همه مهره ها به جز Marshal	1	2	General
-	2	3	Colonel
-	3	4	Major
-	4	5	Captain
-	4	6	Lieutenant
-	4	7	Sergeant
خنثی کردن بمب	5	8	Miner
حرکت چند خانه ای	8	9	Scout
زدن مهره Marshal	1	S	Spy
غیر قابل حرکت	6	B	Bomb
غیر قابل حرکت	1	F	Flag

مهره Flag و مهره Bomb قابل حرکت نیستند و پس از قرار داده شدن در ابتدای بازی دیگر نمی توان در طول بازی آنها را حرکت داد.

قوانین زدن مهره و چگونگی برنده شدن در بازی Stratego

در صورتی که مهره ای وارد خانه ای شود که مهره حریف در آن قرار دارد در واقع اقدام به زدن مهره حریف کرده است. قوانین زدن به این شکل است که بعد از زدن مهره حریف، هر دو بازیکن مهره های خود را نشان داده اند، و مهره ای که شماره کمتری داشته باشد از بازی بیرون می رود. در صورتی که شماره دو مهره برابر باشد، هر دو مهره بیرون خواهند رفت.

اگر مهره X یک بازیکن مهره Y بازیکن دیگر را بزند و $X < Y$ باشد، مهره Y بیرون می رود و مهره X در بازی نگه داشته می شود. در این حالت مهره X حرکت کرده است و وارد خانه ای شده است که مهره Y قبلا در آنجا قرار داشته است.

اما اگر مهره X یک بازیکن مهره Y بازیکن دیگر را بزند و $X > Y$ باشد، مهره X آن بازیکن بیرون می‌رود و مهره Y حریف در بازی نگه داشته می‌شود. در این حالت مهره Y حریف حرکت نکرده است و مهره X بازیکن مهاجم بیرون رفته است.

اگر مهره X یک بازیکن مهره Y بازیکن دیگر را بزند و $X = Y$ باشد، در اثر زدن هر دو مهره بیرون خواهند رفت.

مهره شماره ۸ تنها مهره ای است که می‌تواند Bomb یا همان مهره B را بزند. بقیه مهره ها در صورت زدن بمب خواهند سوخت و بیرون خواهند رفت. فقط در صورتی که مهره شماره ۸، مهره B را بزند، Bomb بیرون خواهد رفت.

مهره S تنها مهره ای است که می‌تواند مهره شماره ۱ یا همان Marshal را بزند. یعنی اگر مهره S، مهره شماره ۱ را بزند، مهره شماره ۱ بیرون خواهد رفت. اما مهره S در زمین بازی باقی خواهند ماند.

تمامی مهره ها می‌توانند مهره S را بزنند، حتی اگر مهره شماره ۱ یا Marshal زودتر اقدام به زدن مهره S بکند، مهره S از بازی بیرون رفته و مهره Marshal در زمین بازی باقی می‌ماند.

تمامی مهره‌ها غیر از Bomb ها که قابل حرکت نیستند، حتی مهره S می‌توانند مهره F یا پرچم را بزنند.

بازی هنگامی تمام می‌شود که یکی از بازیکنان مهره F یا پرچم حریف را بزند که در این صورت برنده خواهد شد. همچنین در صورتی که یک بازیکن تمام مهره های قابل حرکت طرف مقابل را بزند، برنده خواهد شد. بازی تحت شرایط خاصی مساوی خواهد شد که هر دو بازیکن تمام مهره های قابل حرکت یکدیگر را بزنند که کمتر اتفاق خواهد افتاد.

نحوه طراحی بازی

برنامه شما در این بازی باید بتواند به جای یکی از بازیکنان بازی Stratego بازی کند. برای این کار شما باید یک کلاس به نام Stratego بنویسید. کلاس شما باید دارای توابع زیر باشد.

تابع اول تابع Initialize است که Prototype آن به صورت زیر است.

`vector<string> Initialize()`

خروجی این تابع یک vector است که دقیقا دارای ۴ عدد رشته یا همان string می‌باشد. این رشته ها در ۴ ردیف نحوه چیده شدن مهره ها را نمایش می‌دهند. این تابع کلاس شما باید نحوه چیده شدن مهره ها را در ابتدای بازی نشان دهد. شما باید فرض کنید همیشه به جای بازیکنی که مهره های خود را در سطرها ۷، ۸، ۹ و ۱۰ قرار می‌دهد، بازی خواهید کرد. همچنین در این vector مهره ها باید به ترتیب از سطر ۱۰ به بالا و با شروع از ستون اول شروع کند.

به عنوان مثال مهره های بازیکنی که در شکل ۳ نشان داده شده است باید به ترتیب زیر در vector خروجی تابع مورد نظر قرار بگیرند. رشته ها از چپ به راست در خانه های اول تا چهارم vector مورد نظر قرار می‌گیرند.

86898BFB48, 997995B865, 3566215434, 9B7799S7BB

برنامه شما باید فقط از حروف بزرگ انگلیسی برای مهره های B, S و F استفاده کند. در صورتی که اندازه vector خروجی برابر با ۴ نباشد و یا یکی از رشته های آن کمتر از ۱۰ طول داشته باشد و یا از حروف بزرگ استفاده نکرده باشید و یا کاراکتر نامشخصی ارسال کرده باشید، برنامه شما دارای خطا در نظر گرفته خواهد شد.

تابع دوم کلاس شما تابع Play که Prototype آن به صورت زیر خواهد بود.

```
void Play(int & x1, int & y1, int & x2, int & y2)
```

این تابع یک حرکت را برای بازیکنی که برنامه شما آن را کنترل می کند، نشان می دهد که در آن خانه (x1, y1) به عنوان مبدأ حرکت و خانه (x2, y2) به عنوان مقصد باید در نظر گرفته شود. تابع شما باید به x1, y1, x2 و y2 را مقدار بدهد. در صورتی که حرکت شما جز حرکات مجاز بازی Stratego نباشد، برنامه شما دارای خطا در نظر گرفته خواهد شد. همچنین در صورتی که مقادیر داده شده به متغیرهای ورودی بین اعداد ۱ تا ۱۰ نباشند، برنامه شما دارای خطا در نظر گرفته خواهد شد.

تابع سوم کلاس شما تابع Capture خواهد بود که Prototype آن به صورت زیر است.

```
void Capture(char piece)
```

این تابع به شما نشان می دهد که در حرکت آخر توسط تابع Play چه اتفاقی افتاده است. ورودی این تابع یک کاراکتر است که نشان دهنده مهره حریف در خانه ای است که مقصد حرکت قبل با تابع Play بوده است. به عنوان مثال، در تابع Play شما دستور حرکت مهره خود از خانه (x1, y1) به خانه (x2, y2) را داده اید. در صورتی که این خانه دارای یکی از مهره های حریف باشد، این مهره در ورودی تابع Capture به شما داده خواهد شد. مثلاً اگر شما با مهره شماره ۴ خود وارد خانه ای شده اید که مهره شماره ۳ حریف در آن قرار دارد، ورودی تابع Capture یعنی کاراکتر piece برابر با '3' خواهد بود. در صورتی که خانه مقصد خالی باشد، کاراکتر piece برابر با '0' خواهد بود. شما خودتان با توجه به قوانین بازی باید وضعیت بازی را به روز کنید.

تابع چهارم کلاس شما تابع Update خواهد بود که Prototype آن به صورت زیر است.

```
void Update(int x1, int y1, int x2, int y2, char piece)
```

این تابع آخرین حرکت حریف شما را نشان می دهد و به شما کمک می کند تا وضعیت صفحه بازی را Update کنید. این تابع به شما نشان می دهد حریف مهره خود را از خانه (x1, y1) به عنوان مبدأ حرکت به خانه (x2, y2) به عنوان مقصد حرکت داده است. در صورتی که حرکت حریف به زدن منجر شده باشد، مهره حریف با ورودی piece نشان داده می شود. مثلاً اگر حریف شما با مهره شماره ۴ خود وارد خانه ای شده است که مهره شماره ۵ شما در آن قرار دارد، ورودی آخر تابع Update یعنی کاراکتر

piece برابر با '4' خواهد بود. در صورتی که حرکت به زدن منجر نشده باشد، کاراکتر piece در این تابع برابر با '0' خواهد بود. شما خودتان با توجه به قوانین بازی باید وضعیت بازی را به روز کنید.

این تابع برای این خواهد بود که وضعیت صفحه بازی بعد از هر حرکت بازیکنان به روز رسانی شود و به نوعی آنها را راهنمایی کند.

نحوه کنترل بازی

بازی Stratego در این پروژه بین کلاس شما و کلاس یک بازیکن دیگر انجام خواهد شد. این بازی توسط یک Application کنترل خواهد شد. قسمتی از برنامه این Application که مربوط به کنترل بازی است، به صورت Pseudo code در زیر نشان داده شده است. در این Pseudo code متغیرهای Player1 و Player2 از نوع کلاس نوشته شده توسط شما و حریف شما خواهند بود.

```
Player1.Initialize();  
Player2.Initialize();  
....  
while (game is not finished)  
{  
    Player1.Play();  
    ....  
    Player1.Capture();  
    ....  
    Player2.Update();  
    ....  
    Player2.Play();  
    ...  
    Player2.Capture();  
    ....  
    Player1.Update();  
}
```

در صورتی که برنامه شما با خطا مواجه شود و خروجی را به صورت خطا برگرداند. به عنوان مثال دستور حرکت غیرمجازی را بدهد فقط تا ۳ بار در کل بازی به برنامه شما فرصت داده خواهد شد تا حرکت خود را اصلاح کند، در غیر این صورت برنامه شما بازنده خواهد شد. برای انجام هر حرکت برنامه شما فقط ۲۰ ثانیه زمان دارد و در صورتی که بیشتر طول بکشد، نوبت به حریف شما داده خواهد شد. در صورتی که بازی تحت شرایط خاصی مساوی شود، بازی فقط یک بار تکرار خواهد شد. در صورتی که بازی باز هم مساوی شود بازیکنی که آخرین اقدام به زدن را انجام داده است برنده اعلام خواهد شد.

شما می‌توانید در این پروژه یک تیم حداکثر دو نفره داشته باشید. از داشتن تیم‌های تک نفره جدا خودداری کنید. تیم‌های خود را تا تاریخ ۱۳۹۴/۱/۲۲ اعلام کنید. نحوه برگزاری مسابقات در ادامه اعلام خواهد داشت. Application کنترل کننده بازی به زودی بر روی سامانه ivut.iut.ac.ir قرار خواهد گرفت.