

# The Eternal Battle

توضیحات پروژه: پروژه بصورت بازی دو نفره ای می باشد که در آن هر بازیکن دارای 6 مهره که یک مهره آن مهره ی ثابت می باشد. 5 مهره دیگر هر کدام قابلیت حرکت به خانه های اطراف را دارند. اگر هر بازیکن بتواند مهره ثابت حریف را بزند پیروز میشود.

توضیحات پروژه: بازی از هفت کلاس صفحه، مهره، بازیکن، حرکت، موزیک و نمایش و بررسی حرکت و یک کلاس اصلی تشکیل شده است. در ادامه به توضیح هر کلاس میپردازیم:

## 1-صفحه(Board)

کلاس صفحه مسئولیت ساخت صفحه بازی را به عهده دارد. این کلاس از چند متد برای جایگذاری و برگرداندن طول و عرض صفحه، چاپ کردن صفحه و برگرداندن آن به حالت ابتدایی تشکیل شده است. این کلاس نیز دارای یک سازنده می باشد که با گرفتن طول و عرض ویژگی های کلاس صفحه را میتواند برگرداند.

- rest(): بازسازی صفحه
- print(): چاپ صفحه
- SetDefault(): قرار گیری خودکار صفحه
- setX(),setY(): جایگذاری طول و عرض

•  $getX(), getY()$ : برگرداندن طول و عرض

1	2		...	x-1	X
—	—		...	—	2
			....		.
					.
					.
—	—		...	—	y-1
—	—		...	—	Y

## 2- مهره (Mohreh)

در ابتدای پروژه مهره های بازی هر کدام به عنوان یک شی جدا در نظر گرفته شده بود و با استفاده از آن صفحه پر میشد. همچنین قابلیت جایگذاری دستی مهره ها نیز وجود داشت.

اما در ادامه ساخت بازی مهره ها را به عنوان بخشی از کلاس بازیکن در نظر گرفتیم (با ایجاد بازیکن مهره ها نیز بصورت خودکار ایجاد میشوند).

## 3- بازیکن (Player)

این کلاس مسئولیت ساخت مهره های بازی را دارد. این کلاس دارای ویژگی های  $x$  و  $y$  می باشد که با آن مختصات هر مهره در صفحه را مشخص میکند. این کلاس دارای چند متد برای جایگذاری و برگرداندن

مختصات مهره و مهمتر از هر چیز متدی برای جایگذاری خودکار مهره ها در صفحه می باشد.

- `setX(),setY()`:جایگذاری طول و عرض مهره
- `getX(),getY()`:برگرداندن طول و عرض مهره
- `setDefault(Board b)`: قرارگیری مهره ها در صفحه

مثالی برای کاربرد متد `setDefault(Board b)`:

```
Player player=new Player();  
Board board=new Board(5,7);  
player.setDefault(Board board);  
board.print();
```

1	2	3	4	5	
	R	+	R		1
	R	R	R		2
					3
	B	B	B		4
	B	*	B		5

(Move)

4-حرکت

کلاس حرکت یکی از مهمترین کلاس های بازی می باشد که جا به جایی تک مهره ها را به عهده دارد. حرکت در بازی به دو نوع می باشد.

- 1-حرکت بر اساس مختصات اولیه و ثانویه
- 2-حرکت بر اساس مختصات اولیه و جهت حرکت

برای این منظور این تابع از متد های زیر تشکیل شده است:

1-setX(),setX2(),setY(),setY2(): جایگذاری مختصات

2-getX(),getX2(),getY(),getY2():برگرداندن مختصات

3-moveR(int x1,int y1,int x2,int y2,Board b),-

moveB(..): جا به جایی مهره با توجه به مختصات خانه ای که در آن قرار دارد و خانه ای که میخواهد به آن برود. moveR برای مهره قرمز و moveB برای مهره ی آبی می باشد.

4-moveR2(int x,int y,int direction,Board b)-

moveB2(...):جا به جایی مهره با توجه به مختصات خانه ای که در آن قرار دارد جهت خانه ای که میخواهد به آن برود. moveR2 برای مهره قرمز و moveB2 برای مهره ی آبی می باشد. هر مهره حداکثر دارای 8 انتخاب می باشد.

5-moveB2Check(int x,int y,int direction,Board b),-

moveR2Check(..):متد چک یکی از متد های نیازی کلاس حرکت می باشد. این مدت با کمک کلاس نمایش و بررسی حرکت(harekat)خانه های مجاز برای حرکت را به صورت ترو و فالس ریترن میکند و سپس در در کلاس اصلی با استفاده از یک حلقه for میتوان تمامی خانه های مجاز را بررسی کرد. به این صورت که نتیجه بررسی خانه های مجاز اطراف را گرفته و با توجه به اینکه امکان جا به جایی به آن خانه می باشد یا نه مقادیر صحیح و غلط را برمی گرداند. یکی دیگر از مسئولیت های این متد بررسی خانه های گوشه می باشد. خانه های گوشه و ردیف و ستون اخر انتخاب های محدودی با توجه به برد صفحه دارند. این متد محدودیت

ها را بررسی کرده و سپس انتخاب های مجاز را به متد مورد نظر می رساند ( جلوتر به آن پرداخته می شود ). برای نوشتن این متد باید چند نکته بسیار مهم رعایت میشد برای مثال مهره ای که در گوشه پایین صفحه وجود دارد حق پایین رفتن ندارد زیرا اگر بخواهد این کار را بکند از مقادیر مجاز آرایه خارج میشود و کل بازی کرش میکند و مشکلات ایجاد میکند، پس به همین علت باید شرایط خاص گوشه بودن مهره ها حتما در نظر گرفته بشود و اینکه اگر برای مثال مهره ها در ستون راست صفحه باشند حق ندارند که دوباره به سمت راست بروند و اگر بخواهند این عمل را انجام بدهند باعث میشود که از آرایه بیرون بروند و دوباره بازی کرش بکند!

### **5-نمایش و بررسی حرکت(harekat)**

کلاس نمایش و بررسی حرکت یک کلاس با قابلیت های مختلف می باشد. بطور کلی این کلاس به کاربر کمک می کند تا انتخاب صحیحی برای جا به جایی داشته باشد. این کلاس دارای متد هایی است که مختصات مهره ها را میدهد. ویژگی جالب و مهم این کلاس متد های بررسی و بازسازی است.

- **Barresi(int x,int y,Board b):** با مشخص کردن مهره ی انتخابی کاربر این متد خانه های مجاز برای حرکت را به صورت دایره های تو خالی نشان میدهد.
- **Bazsazi(Board b):** پس از انتخاب حرکت توسط کاربر این متد صفحه را به حالت عادی برمیگرداند. (دایره های تو خالی را پاک میکند).
- **redlist(Board b),bluelist(Board b):** مختصات مهره های باقی مانده را به کاربر نمایش میدهد.

برای نمونه میتوان به مثال زیر که بخشی از کلاس اصلی که باشد، اشاره کرد. میخواهیم برای مثال انتخاب های خانه مهره آبی (2و4) در برد 5\*5 را ببینیم:

```
Board board=new board(5,5);  
x1=2;  
y1=4;  
harekat hr=new harekat();  
hr.barresi(y1 - 1, x1 - 1, board);  
board.print();
```

1	2	3	4	5	
	R	+	R		1
	R	R	R		2
O	O	O			3
O	<u>B</u>	B	B		4
O	B	*	B		5

## 6-کلاس آهنگ (musicStuff)

این کلاس وظیفه پخش آهنگ را دارد. این کلاس محل فایل آهنگ را دریافت کرده و آهنگ را بدون وقفه تا موقعی که بازی بسته بشود ادامه میدهد.

## 7-کلاس اصلی (Main)

این کلاس برای اجرای برنامه می باشد. در ابتدا اشیاء صفحه، بازیکن، حرکت آبی و قرمز و بررسی حرکت تشکیل می شود و به صورت پابلیک تعریف میشوند زیرا که می خواهیم توسط تمامی متد ها قابل دسترسی باشند. همچنین متغیر flag برای اجرای نوبت بازیکنان نیز ایجاد میشود و متغیر flag2 برای این مورد که ابتدا برد ساخته بشود و سپس باقی کارها انجام بشود.

در متد اصلی ابتدا فهرستی از گزینه ها به نمایش در می آید که بصورت زیر می باشد:

- 0 - To print choice options.(نمایش گزینه ها)
- 1 - To create a Board.(ساخت صفحه)
- 2 - To print the Board.(چاپ صفحه)
- 3 - To add The Players to the Board.  
(افزودن بازیکنان به صفحه)
- 4 - To Move The Red Team Players.

(حرکت دادن مهره قرمز)

5 - To Move The Blue Team Players.

(حرکت دادن مهره آبی)

6 - To Restart the Board.(بازسازی صفحه)

7 - To Count the remaining Blue Balls.

(شمارش تعداد مهره آبی)

8 - To Count the remaining Red Balls.

(شمارش تعداد مهره قرمز)

9 - To quit the Game.( خروج از بازی)

چاپ این فهرست بر عهده متد () printInstructions می باشد و با انتخاب شماره 0 به نمایش در می آید.

با انتخاب شماره 1 متد () createBoard اجرا می شود. این متد با گرفتن طول و عرض از کاربر صفحه ای با این مشخصات را می سازد.(تذکر:حداقل مقادیر طول و عرض 5 می باشد،در غیر اینصورت خطای ورود مجدد داده می شود) و نکته مهم دیگری که در این سوییچ کیس رعایت شده این است که اگر کاربر برد(صفحه) را نسازد، حق استفاده از باقی موارد منور و ندارد زیرا آنها بدون برد(صفحه)عمل نمیکنند.

کلید شماره 2 مربوط به متد () printBoard و چاپ صفحه می باشد.

کلید شماره 3 نیز بازیکنان را با صدا زدن متد () addPlayer به صفحه می افزاید.



کلید شماره 4 و 5 برای حرکت قرمز و آبی می باشد. (moveRed(), moveBlue())

با انتخاب هر یک از این دو ابتدا متد (blueList())(redList()) که قبل تر به آن پرداخته بودیم اجرا می شود و لیست مهره به همراه مختصات آنها چاپ می شود.

سپس از کاربر خواسته می شود که مختصات مهره ای که میخواهد جا به جا کند را وارد کند. با وارد کردن مختصات متد Barresi() نیز اجرا می شود و مسیر های ممکن به نمایش در می آید.

برای حرکت دو انتخاب 1- از طریق جهت و 2- از طریق مختصات داده می شود.

اگر 1 انتخاب شود جهت های مجاز برای کاربر نمایش داده میشود و کاربر با انتخاب گزینه مورد نظر مهره را حرکت می دهد و اگر 2 را انتخاب کند لیستی از مختصات خانه های مجاز برای حرکت داده می شود و با انتخاب مناسب مهره حرکت می کند.

با حرکت هر مهره نوبت هر بازیکن عوض می شود و بازکن حق ندارد دوباره بازی کند.

در صورت انتخاب شماره 6 از فهرست صفحه مجدد باز سازی می شود. (متد rest())

شماره 7 و 8 تعداد مهره های باقی مانده را نشان می دهد.

و شماره 9 نیز امکان خروج از بازی را فراهم می کند.

موفق باشید.