



درس برنامه سازی پیشرفته

تمرین سوم بخش اول

دانشکده مهندسی کامپیوتر

دانشگاه صنعتی شریف

نیم سال دوم ۹۹-۹۸

مبحث:

گرافیک و جنریک

مهلت ارسال:

۱۶ خرداد

ساعت ۲۳:۵۹

ویراستار فنی:

صابر ظفرپور و محمد فراهانی



به موارد زیر توجه کنید:

* به‌ازای هر سوال در سامانه‌ی کوئرا، یک بخش جداگانه برای بارگذاری برنامه‌ی شما وجود دارد. فایل برنامه‌ی خود با پسوند zip را در بخش مربوط به هر سوال بارگذاری کنید.

* پس از ارسال فایل مربوط به هر سوال، سامانه‌ی کوئرا به‌صورت لحظه‌ای برنامه‌ی شما را داوری کرده و نمره‌ی آن سوال را به شما اعلام می‌کند که در صورت کم بودن نمره‌تان، می‌توانید آن را تصحیح کرده و دوباره ارسال کنید. به جز سوال‌های دوم و سوم که در هنگام تحویل حضوری مورد ارزیابی قرار می‌گیرد.

* هم‌فکری و هم‌کاری در پاسخ به تمرینات اشکالی ندارد و حتی توصیه نیز می‌شود؛ ولی پاسخ ارسالی شما باید حتماً توسط خود شما نوشته شده باشد. در صورت هم‌فکری در مورد یک سوال، نام افراد دیگر را به‌صورت کامنت در ابتدای کد هر سوال بنویسید. این نکته رو در نظر بگیرید که هم‌فکری تنها مربوط به بخش ایده سوال هست نه پیاده‌سازی آن و در صورت محرز شدن تقلب برای فرد خاطی بدون مسامحه منفی نمره تمرین منظور می‌گردد.

* شما می‌توانید تمامی سوالات و ابهامات خود را در سایت کوئرا در بخش مشخص شده برای این تمرین بپرسید.

* به‌ازای هر روز تاخیر در ارسال پاسخ هر سوال، ۳۰ درصد از نمره‌ی کسب‌شده‌ی شما در آن سوال کم می‌شود. به عنوان مثال اگر پاسخ یک سوال را با دو روز تاخیر ارسال کنید، فقط ۴۰ درصد از نمره‌ای که برای آن سوال گرفته‌اید برای شما لحاظ خواهد شد.

* در کل شما می‌توانید سه روز تاخیر بدون کسر نمره داشته باشد.

* مهلت ارسال تمرین تا ساعت ۲۳:۵۹ روز ۱۶ خرداد ۱۳۹۹ است.

* تنها به یکی از دو سوال دو یا سه پاسخ دهید.



۱ در دسرهای ناتمام اکبر دلاک

اکبر دلاک از دانشجویان پرکار و رودی ۹۸ است که به تازگی برای انتخابات SSC نامزد شده است. ولی به دلیل بسته بودن دانشگاه کل برنامه زندگی اش به هم ریخته است. او حتی یادش نیست که کلا چه کارهایی داشته و چه کارهایی را میخواست انجام دهد. او از آن جا که می‌داند در صورت رای آوردن احتمالی در انتخابات SSC کارهای او چند برابر می‌شود، تصمیم گرفت برای مدیریت کارهایش با اصغر دلاک مشورت کند. بعد از تفکر و مشورت فراوان با اصغر دلاک به این نتیجه رسید که نیاز به یک داده ساختار دارد که برای مدیریت کارهایش از آن استفاده کند، طوری که آخرین کاری که داشته یا اولین کاری که داشته و هنوز انجامش نداده یا کارهایی که برایش بیشتر از یک کار مشخص اولویت داشته را بتواند دریافت کند. از آن جایی که خیلی از دوستان اکبر دلاک درگیر این مشکل هستند، او تصمیم دارد که داده ساختاری را پیاده‌سازی کند که دوستانش نیز در عوض رای دادن به او در انتخابات (!) از آن استفاده کنند. با توجه به این که دوستان اکبر برای ذخیره کردن کارهایشان کلاس مخصوص به خودشان را دارند، او باید داده ساختارش را به صورت generic پیاده‌سازی کند، اما او به دلیل مسئولیت‌های فراوانی که دارد، وقت انجام این کار را ندارد و شما باید داده ساختار را به شکل زیر برایش پیاده‌سازی کنید. به طور دقیق نام داده ساختار و تعریف آن این‌گونه است:

```
public class AkbarWorks<T extends Comparable> {}
```

این کلاس باید عملیات زیر را انجام دهد که تعریف دقیق توابع متناظر آورده شده است.

- یک عنصر به مجموعه عناصری که تاکنون وجود دارند اضافه شود.

```
public void add (T item);
```

- کوچکترین عنصر را حذف و آنرا خروجی دهد.

```
public T getMin () throws IllegalStateException;
```

- عنصری که دیرتر از سایر عناصر اضافه شده است را خروجی دهد. اگر مقدار remove برابر با true بود، این عنصر را حذف هم بکند.



```
public T getLast (boolean remove) throws IllegalStateException;
```

- عنصری که زودتر از سایر عناصر اضافه شده است را خروجی دهد. اگر مقدار `remove` برابر با `true` بود این عنصر را حذف هم بکند.

```
public T getFirst (boolean remove) throws IllegalStateException;
```

- همه عناصری که کوچکتر از عنصر `element` هستند را خروجی دهد. اگر مقدار `remove` برابر با `true` بود این عناصر را حذف هم بکند.

```
public Comparable[ ] getLess (T element, bool remove);
```

- `n` عنصری که اخیراً حذف شده‌اند را (به ترتیب از اخیر ترین) خروجی دهد. (برای مقاصد بازیابی اطلاعات!) اگر `n` از تعداد کل عناصر حذف شده بیشتر بود، تمام عناصر حذف شده را خروجی دهد.

```
public Comparable[ ] getRecentlyRemoved (int n);
```

نکات:

- در توابع `getMin` ، `getLast` و `getFirst` اگر هیچ عنصری وجود نداشته باشد باید یک استثنا از نوع `IllegalStateException` پرتاب شود.
- توابع `getMin` ، `getLast` ، `getFirst` و `getFirst` روی عناصر موجود کار می‌کنند و اگر عنصری حذف شده باشد، این توابع آن عنصر را بررسی نمی‌کنند.
- پیاده‌سازی داخلی و سایر متغیرها یا توابع کلاس اختیاری است.



۲ شطرنج گرافیکی

در تمرین سری قبل، مجبور شدید که برای این که جلوی خورده شدن خودتان توسط آدم‌خواران را بگیرید، برای آن‌ها یک بازی شطرنج طراحی کنید. آن‌ها بعد از مدتی بازی با آن شطرنج، به دلیل رابط کاربری متنی از آن خسته شدند و حال از شما خواسته‌اند که برای این که شما را نخورند، یک بازی شطرنج گرافیکی برای آنان تهیه کنید. به همین دلیل شما هم مجبور هستید که بازی شطرنج خود را به شکل گرافیکی تبدیل کنید و امیدوار باشید که بازی طراحی شده توسط شما آن‌قدر جذاب باشد که آنان به قدری مشغول بازی بشوند که بعد از خسته شدن از بازی، به قدری دچار ضعف بشوند که امکان خوردن شما برایشان مهیا نباشد!

همان طور که از توضیحات بند قبل مشخص است، هدف در این سؤال طراحی یک بازی شطرنج گرافیکی است. در این تمرین، یکسری موارد اجباری و یکسری موارد اختیاری برای پیاده سازی وجود دارد. موارد اختیاری باعث کسب نمره اضافی علاوه بر نمره کامل سؤال می‌شوند.

■ توجه: برای راحتی کار، در صورتی که کدی که برای تمرین قبل زدید از کیفیت خوبی برخوردار باشد، می‌توانید از بخش‌هایی زیادی از آن کد برای این سؤال هم استفاده کنید ولی لزومی به این کار ندارید و اگر هم بخواهید، می‌توانید کل کد برنامه را دوباره بنویسید.



بخش اجباری (۳۰ نمره)

۱. صفحه لاگین (ساختن و ورود به اکانت) - ۵ نمره
امکاناتی که باید در صفحه مربوط به لاگین و ساخت اکانت موجود باشند:

- حذف اکانت (۱ نمره)
- ساخت اکانت (۱ نمره)
- لاگین (۱ نمره)
- تغییر رمز عبور (۱ نمره)
- خروج از بازی (۱ نمره)
- ۲. منوی اصلی (۵ نمره)

○ نیوگیم

⇐ در ابتدای ورود به بازی باید turn limit و تعداد دفعات مجاز Undo
پرسیده شود؛ منظور از turn limit چیست، در داک تمرین ۲ توضیح
داده شده است (۲ نمره)

○ اسکوربورد که به همان ترتیب گفته شده در تمرین ۲ مرتب بشود و همه موارد
تأثیرگذار در مرتب‌سازی (برد و باخت و...) برای هر بازیکن به شکل جدول
یا هر شکل مناسب دیگری نمایش داده شوند. (۲ نمره)

○ قابلیت لاگ اوت (۱ نمره)

⇐ بعد از لاگ اوت کردن، طبیعتاً باید به منوی قبلی یعنی منوی لاگین و ثبت
نام برگردید.

۳. خود بازی (۲۰ نمره)

○ همه موارد مربوط به بازی که در داک تمرین ۲ گفته شده‌اند، می‌بایستی پیاده
سازی بشوند. از جمله حرکت درست مهره‌ها مطابق قوانین شطرنج، انتخاب
مهره‌ها، زدن مهره‌ها، نمایش درست صفحه شطرنج با اسامی درست و واقعی
خانه‌ها یعنی ردیف‌های افقی اعداد ۱ تا ۸ و ستون‌های عمودی حروف A تا
H انگلیسی. (۱۰ نمره)

○ وجود تصویر مناسب و متفاوت برای هر کدام از مهره‌ها؛ یعنی مهره‌ها فقط به
صورت یک حرف الفبا نظیر Q و... روی صفحه نباشند و شکل مناسب برای
آنان در نظر گرفته بشود. (۲ نمره)

○ امکان انصراف از بازی به وسیله یک Button یا هر شکل دیگری که علاقه
دارید. (۱ نمره)



- پیاده سازی درست برد و باخت (۱ نمره)
- امکان Undo کردن به یک حرکت قبل و مشخص کردن تعداد Undo های مجاز در ابتدای بازی و بازگشت به حالت قبل با انیمیشن (۲ نمره)
- قرار دادن یک Button یا هر شکل دیگری برای پایان نوبت (۱ نمره)
- پیام‌های مناسب برای هر رویداد (مثلاً عدم امکان حرکت به یک نقطه - برد و اتمام بازی - تغییر نوبت و...) (۲ نمره)
- داشتن انیمیشن برای حرکت مهره‌ها (۱ نمره)

بخش امتیازی (۱۵ نمره)

هر مورد که پیاده سازی شود، ۳ نمره خواهد داشت. بنابراین نیازی به پیاده سازی تمامی موارد نیست و پس از رسیدن به سقف ۱۵ نمره امتیازی، نمره بیشتری برای سایر بخش‌ها داده نمی‌شود.

۱. امکان زمان‌دار کردن بازی. هر فرد یک زمان‌سنج جدا دارد که با به اتمام رسیدن زمان یک نفر در نوبت وی، آن فرد بازنده شده و فرد دیگر برنده می‌شود. زمان‌سنج هر نفر تنها در نوبت خود وی کار می‌کند.

۲. پیاده سازی قابلیت Drag and Drop برای مهره‌ها. توجه کنید که در صورت پیاده سازی این مهره، طبیعتاً مسئله Select کردن مهره به شکلی شبیه تمرین ۲ موضوعیت ندارد و در صورت پیاده سازی Drag and Drop نمره آن داده می‌شود. همچنین در این حالت نیازی به پیاده سازی انیمیشن برای حرکت مهره‌ها (به جز در مورد Undo) نیست. چون مهره‌ها با Drag شدن توسط موس حرکت خواهند کرد.



۳. پیاده سازی قابلیت‌های قلعه کردن و آن پاسان. (هر مورد ۱.۵ نمره)

⇐ لینک توضیحات قلعه کردن در ویکی‌پدیا

⇐ لینک توضیحات آن پاسان در ویکی‌پدیا

■ توجه: در مورد قلعه کردن، نیازی به رعایت شرط کیش نبودن شاه که در لینک‌ها ذکر شده نیست. همچنین در صورتی که بخش امتیازی مربوط به کیش و مات را پیاده سازی نکنید، نیازی به رعایت این که شاه بعد از حرکت در وضعیت کیش قرار نگیرد هم نیست. در صورتی که کیش و مات را به عنوان بخش امتیازی پیاده سازی کنید، باید این موضوع که شاه بعد از این حرکت در وضعیت کیش قرار نگیرد هم بررسی شود. سایر شرایط باید رعایت شوند.

۴. نگهداری لیست کامل حرکت‌ها و توانایی بازگشت به هر کدام از آن‌ها و ادامه دادن بازی از آنجا یعنی به نوعی قابلیت Undo کردن به هر حرکت از حرکات قبلی وجود داشته باشد. طبیعتاً تعداد دفعات انجام این کار، همانند شکل ساده Undo محدودیت دارد. در صورت پیاده سازی این قابلیت، منطقاً نیازی به پیاده سازی Undo یک حرکتی به صورت جداگانه نیست؛ چون پیاده سازی این مورد، به خودی خود این موضوع را هم شامل می‌شود.

۵. پخش موسیقی و صداگذاری

این موضوع دو جنبه دارد:

⇐ موسیقی پس زمینه در کل بازی

⇐ صداگذاری برای حرکت مهره‌ها و کلیک روی دکمه‌ها

موسیقی پس زمینه، ۱.۵ نمره و صداگذاری هم ۱.۵ نمره دارد.



۶. پیاده سازی کیش و مات. طبیعتاً در صورت پیاده سازی این موضوع، مسئله زده شدن شاه که برای سادگی در داک تمرین ۲ مطرح شده بود، دیگر مطرح نیست و نیازی به پیاده سازی زده شدن شاه که در شطرنج واقعی وجود ندارد نیز نخواهد بود.

⇐ [لینک توضیحات کیش کردن در ویکی‌پدیا](#)

⇐ [لینک توضیحات کیش و مات در ویکی‌پدیا](#)

۷. نمایش حرکت‌های موجود برای هر مهره با انتخاب آن؛ یعنی مثلاً رنگ خانه‌هایی که این مهره می‌تواند به آن‌ها برود، کمی تغییر کند و از بقیه صفحه متمایز شود)

۸. تبدیل سرباز به مهره با ارزش تر در صورت رسیدن به انتهای زمین

⇐ در صورتی که فقط امکان تبدیل به یک نوع مهره باشد ۲.۵ نمره به شما تعلق می‌گیرد.

⇐ در صورتی که امکان انتخاب میان وزیر و رخ و اسب و فیل وجود داشته باشد ۳ نمره به شما تعلق می‌گیرد.

۹. زیبایی بازی



۳ فضانوردی

در این تمرین قرار است حالت ساده‌ای از بازی space invaders را پیاده‌سازی کنید. برای آشنایی با این بازی می‌توانید به این [صفحه](#) مراجعه کنید. تفاوت بازی اصلی با بازی که شما باید پیاده‌سازی کنید، این است که نیازی به پیاده‌سازی موانع نیست و همچنین نیازی نیست که هدف‌ها تیراندازی کنند و یا به چپ و راست حرکت کنند (این قسمت جزو موارد امتیازی محسوب می‌شود). در صورت رسیدن هدف‌ها به پایین صفحه (جایی که سفینه‌ی هدف در آن قرار دارد)، بازی تمام می‌شود و پیام متناسب با آن نمایش داده می‌شود.

بخش اجباری (۳۰ نمره)

۱. جابه‌جا شدن سفینه‌ی بازیکن به چپ و راست با استفاده از کلیدهای جهت (۳ نمره)
۲. تیراندازی توسط سفینه‌ی بازیکن با استفاده از یکی از کلیدهای کیبورد به دلخواه (۳ نمره)
۳. حرکت کردن تیرها با استفاده از انیمیشن (۴ نمره)
۴. قرار دادن هدف‌ها در ردیف‌های متفاوت و اینکه سفینه‌های هر ردیف رنگ متفاوتی داشته باشند. (۳ نمره)
۵. جابه‌جا شدن هدف‌ها پس از بازه‌ی زمانی مشخص به یک ردیف پایین‌تر (۳ نمره)
۶. نابود شدن سفینه‌های هدف در صورت برخورد تیر سفینه‌ی بازیکن به آن‌ها (۳ نمره)
۷. در نظر گرفتن امتیاز برای هر بازی و نشان دادن آن در صفحه‌ی بازی (امتیاز هر بازی برابر با تعداد هدف‌هایی است که بازیکن به آنها شلیک و آنها را نابود می‌کند) (۴ نمره)
۸. پیاده‌سازی منوی ساده برای ورود به بازی و خروج از آن با استفاده از کلیک کردن بر روی گزینه‌های آن (۳ نمره)
۹. در نظر گرفتن حساب کاربری برای بازیکن‌های متفاوت و قراردادن نام کاربری برای هریک از آن‌ها (در صورتی که نام کاربری تکراری بود، پیام مناسب نشان داده شود) و امکان تغییر نام کاربری برای هر بازیکن و قرار دادن گزینه‌ی آن در منوی بازی (۴ نمره)



بخش امتیازی (۱۵ نمره)

هر مورد که پیاده‌سازی شود، ۳ نمره خواهد داشت. بنابراین نیازی به پیاده‌سازی تمامی موارد نیست و پس از رسیدن به سقف ۱۵ نمره امتیازی، نمره بیش‌تری برای سایر بخش‌ها داده نمی‌شود.

۱. تیراندازی توسط هدف‌ها به صورت رندوم که در صورت برخورد به سفینه‌ی هدف، بازی تمام می‌شود (در این حالت نیز پیام مناسب برای پایان بازی باید نمایش داده شود).

۲. جابه‌جا شدن سفینه‌های هدف به چپ و راست (همانند بازی اصلی)

۳. پیاده‌سازی جدول امتیازات و قرار دادن گزینه‌ی آن در منوی بازی

۴. بعد از اینکه سفینه‌های هدف یک ردیف پایین‌تر آمدند، یک ردیف از بالا به این سفینه‌ها اضافه شود. همچنین مدت زمانی که طول میکشد تا یک ردیف به هدف‌ها اضافه شود، هر بار کاهش پیدا کند تا به مقدار معینی برسد. بیشترین و کمترین این مقدار زمانی را در ابتدای بازی باید از بازیکن ورودی بگیرد.

۵. برای سفینه‌ی بازیکن یک قدرت مخصوص پیاده‌سازی کنید. این قدرت به صورت یک هدف در فضای خالی بین سفینه‌ی بازیکن و سفینه‌های هدف قرار می‌گیرد که بازیکن با تیراندازی به آن می‌تواند آن قدرت را دریافت کند (اگر فضای خالی وجود نداشت ظاهر نمیشود). توجه کنید که پس از دریافت آن، برای مدت زمان معینی میتوان از آن استفاده کرد و سپس از بین میرود. این قدرت به همراه اهداف دیگر در بازه‌ی زمانی مشخصی پایان می‌آید و اگر تا ردیف آخر گرفته نشود از بین میرود. از بین دو قدرت زیر، یکی را به دلخواه میتوانید پیاده‌سازی کنید :

○ مسلسل : سرعت تیرزدن را چندبرابر می‌کند که بعد از مدت زمان مشخصی این قدرت از دست می‌رود.

○ بمب : قابلیت پرتاب بمب به بازیکن می‌دهد که در صورت شلیک آن، چند هدف کنار هم نابود می‌شوند.

۶. قراردادن تعدادی مانع بین بازیکن و سفینه‌های هدف که در صورت برخورد هرکدام از تیرهای سفینه‌های هدف یا سفینه‌ی بازیکن با یکی از موانع، آن مانع از بین می‌رود.

۷. صداگذاری برای بازی (مثل صدای شلیک تیر) و همچنین موسیقی پس زمینه

۸. در کل زیباسازی هر چه بیشتر بازی با استفاده از نکته‌های گرافیکی