

The Mind

The human mind is amazingly powerful. It can do things for real that sometimes feel like magic.

در این تمرین شما باید شبه‌بازی (!) جذاب The Mind را پیاده‌سازی کنید. یک بازی خیلی ساده، که مهارت ذهن‌خوانی شما را به چالش می‌کشد.



کلیت تمرین

در این تمرین شما بازی ذهن را به صورت چند کاربره و با استفاده از مفاهیم شبکه و برنامه‌نویسی چندنخی در گروه‌های دونفره پیاده‌سازی خواهید کرد.

توضیح بازی

بازی فکری ذهن، یک بازی همکاری مشترک است. در آن بازیکنان تلاش می کنند تا با مشارکت یکدیگر بر اعداد پیروز شوند. به این شکل که بازیکنان بدون دیدن کارتهای هم، باید تشخیص دهند آیا کمترین کارت در دست خودشان قرار دارد یا نه.

کارتها

- شمارهها: کارتها دارای شمارههای ۱ تا ۱۰۰ هستند که در ابتدای هر دور بر میخورند و در هر مرحله با توجه به شمارهی مرحله در دست بازیکنان قرار میگیرند.
- کارت مرحله: نشان دهندهی مرحله کنونی بازی است.
- کارت قلب: به تعداد بازیکنهای بازی باید در بازی کارت قلب داشته باشیم. این کارتها بین تمام بازیکنان مشترکاند.
- کارت نینجا: بازیکنان به صورت دلخواه میتوانند این کارت را مصرف کنند و بعد از آن هر بازیکن موظف است کمترین شماره ای که دارد را بازی کند. در ابتدای بازی دو کارت نینجا داریم که بین همه مشترک است.

قوانین

هر بازیکن در آغاز هر مرحله به تعداد شمارهی مرحله کارت برمیدارد.

بازی کردن کارتها به نوبت نیست و هر یک از بازیکنان پس از صبر کردن به مدت دلخواه، کمترین شماره کارتی را که دارد بازی می کند. به عبارتی بازیکنان باید حدس بزنند در دست دیگر بازیکنان چه کارتهایی وجود دارد تا بتوانند به ترتیب کارت هایشان را رو کنند.

اگر کارتی بازی شود ولی در دست یکی از بازیکنان کارتی با شمارهی کمتر وجود داشته باشد منجر به از دست دادن یک کارت قلب می شود و کارتهایی که شمارهی کمتری از کارت بازی شده داشتند میسوزند و از دست همه بازیکنان خارج میشوند.

بازیکنان دست یکدیگر را نمیبینند اما آخرین کارت بازی شده روی میز را میتوانند ببینند.

بازیکنان باید بتوانند مراحل مشخص شده را طی کنند بدون آن که تمام کارت های قلب خود را از دست بدهند.

همچنین بازیکنان می توانند تصمیم بگیرند که یک کارت نینجا استفاده کنند. با استفاده کردن این کارت هر بازیکن کوچک ترین kartی که دارد را بازی می کند. با این کار تعداد کارت های هر بازیکن کم شده و ایده ای در مورد شماره ی کارت های در دستان بقیه بازیکنان به ما می دهد. در این حالت نیز تمام کارت های کوچکتری که در دست بازیکن ها باقی مانده از دستشان خارج می شود ولی کارت قلب از دست نمی رود.

بازیکنان در ابتدای راندهای ۳، ۶، ۹ یک کارت قلب و در ابتدای راندهای ۲، ۵، ۸ یک کارت نینجا دریافت خواهند کرد که بین همه آنها مشترک است.

با به پایان رسیدن مرحله ۱۲ تمامی بازیکنان برنده اعلام خواهند شد.

ساختار کلی

- کاربر (کلاینت): نحوه ی ورود به بازی به این صورت است که هر کاربر پس از اجرا شدن به صورت خودکار به سرور بازی متصل می شود. هر کاربر در ابتدا نام خود را به سرور اعلام می کند و به لابی اضافه می شود. اگر هیچ بازیکن دیگری حضور نداشته باشد، کاربر نقش میزبان (host) را ایفا خواهد کرد و حداکثر تعداد بازیکن های بازی را نیز اعلام می کند. تنها میزبان می تواند به سرور درخواست شروع بازی بدهد.

امتیازی: اگر هنگام ورود کاربر به لابی، بازی ای در جریان بود و فضای کافی داشت، کاربر باید بتواند در صورت تمایل به آن اضافه شود.

- سرور: سرور پس از درخواست آغاز بازی توسط کاربر میزبان، با توجه به حداکثر تعداد بازیکنی که اعلام شده است و تعداد افرادی که به بازی اضافه شده اند بات ایجاد می کند و بازی را در سمت سرور آغاز می کند. از این به بعد سرور، پل ارتباطی میان کاربران خواهد بود و با هر حرکت جدید، وضعیت جدید بازی را به همه بازیکن ها اعلام خواهد کرد.

امتیازی: برنامه شما باید قادر به اجرای چند بازی به طور همزمان باشد. یعنی سرور پس از ایجاد یک بازی جدید آن را به لیست بازی های در حال اجرا اضافه می کند.

- بات: شما موظف هستید بات‌ها را به نحوی طراحی کنید که مانند بازیکن‌های دیگر استراتژی برای بازی داشته باشند و بتوانند به طور مستقل بازی کنند. برای این کار هر بات را به این صورت طراحی کنید که پس از زمان مشخصی که صبر می‌کند، کمترین کارت خود را بازی کند. برای تعیین مدت زمان صبر، برای مثال می‌توانید آن عدد را خروجی تابعی تصادفی از کمترین شماره کارت در دست بات و همچنین عدد سرعت بازی در نظر بگیرید. این زمان صبر کردن را می‌توانید به دلخواه خود پیاده‌سازی کنید و از توزیع دلخواه استفاده کنید، اما باید این زمان بر اساس قوانین بازی و استراتژی‌های مختلف باشد و مجاز نیستید به صورت کاملاً رندم این زمان صبر کردن را به دست آورید.

توجه کنید این زمان صبر کردن پس از بازی کردن هر بازیکن (انسان یا بات) برای تمامی بات‌ها reset می‌شود و نسبت به زمان آخرین حرکت انجام شده سنجیده می‌شود. همچنین بات‌ها توانایی درخواست استفاده از کارت نینجا را ندارند در نتیجه لازم به پیاده‌سازی درخواست آن برای بات‌ها نیستید.

جزئیات پیاده‌سازی

بازی چند نفره: در هر بازی تعدادی بازیکن و تعدادی بات هستند که مجموع آن‌ها مقداری ثابت است. به عبارتی با کم یا افزوده شدن بازیکن‌ها، جای آن‌ها را بات می‌گیرد و یا از تعداد بات‌ها کم می‌شود.

احراز هویت: شما باید در سرور بازی، بعد از اتصال کاربر، یک کلید رمز برای کاربر بفرستید که کابر در تمام پیام‌های خود به سرور بازی این کلید را در فیلدی تحت عنوان AuthToken بفرستد. AuthToken یک کلید برای ارتباط بین کاربر و سرور است. در واقع این فیلد مشخصه‌ی هر کاربر خواهد بود و سرور با AuthToken کاربران اتصال آن‌ها را تشخیص خواهد داد. با ورود دوباره یک کاربر این مقدار از نو باید تولید شود. برای تولید AuthToken می‌توانید از کلاس SecureRandom استفاده کنید.

سایر جزئیات: توقع می‌رود در یک اجرای معمولی برنامه شما دچار کرش نشود و این که بعد اتمام برنامه همه‌ی سوکت‌های باز، بسته شوند. نحوه پیاده‌سازی بخش‌هایی که در داک مشخص نشده است (مانند مدیریت خطاها) تماماً به اختیار خودتان است.

رابط کاربری

برای این تمرین می‌توانید به دلخواه خود در کنسول و یا به صورت گرافیکی رابط کاربری را طراحی کنید. تنها توجه کنید که هر رابطی که پیاده‌سازی می‌کنید باید در آن تعداد کارتهای باقی‌مانده از هر نوع (قلب و نینجا و پشت دست بازیکن‌های دیگر)، دست بازیکن (شما) و آخرین کارتی که بازی شده را به‌روزرسانی کنید و نمایش دهید. این رابط در سمت کاربر (کلاينت) نمایش داده می‌شود.

برای سرور رابط کاربری نیاز نیست اما پیشنهاد می‌شود رویدادهای مهم را در یک فایل log ذخیره کنید.

امتیازی

واکنش (امتیازی): بازیکن‌ها باید بتوانند در حین بازی برای یک‌دیگر اموجی بفرستند. طریقه ارسال آن می‌تواند تایپ متنی یا با انتخاب گزینه‌ها باشد. کافی است سه تا از اموجی‌های پرکاربرد (لبخند D: ، غم:) ، شادی:) ، پوکرفیس: | و ... را پیاده‌سازی کنید. سرور باید هرگونه محتوای متنی و عددی را (که مصداق تقلب کاربر است) نادیده بگیرد و تنها اموجی‌ها را انتشار دهد.

سایر قوانین و قواعد برنامه نویسی شبکه

رعایت اصول clean code در این تمرین اجباری است. علاوه بر مواردی که تا کنون یاد گرفته‌اید به موارد زیر نیز باید دقت کنید:

- کاربر و سرور شما باید به طور کامل مجزا از هم طراحی شوند ، به طوری که اگر کاربر و سرور در دو دستگاه مختلف اجرا شوند، هیچ مشکلی در برنامه به وجود نیاید و هر دو به درستی اجرا شوند.
- ارتباط کاربر و سرور فقط باید از طریق شبکه صورت بگیرد و هر گونه روش دیگر مانند صدا زدن مستقیم توابع یک‌دیگر مردود است. ارتباط کاربرها و سرور از طریق سوکت برقرار خواهد شد. این ارتباط می‌تواند از نوع TCP یا UDP باشد.
- توجه کنید که سرور و کاربر به صورت جداگانه اجرا می‌شوند و پورت و آدرس مذکور را از یک فایل config مشخص شده می‌خوانند. فایل کانفیگ کاربر و سرور مستقل از یک‌دیگر است.
- علاوه بر این، یک پورت به صورت پیش‌فرض 8000 و آدرس به نشانی localhost در برنامه خواهد بود که در صورت عدم تامین پورت و آدرس توسط فایل، استفاده خواهند شد.
- برای اتصال کافی است سرور روی یک پورت خاص گوش بایستد و کاربرها با اتصال به آدرس سرور و آن پورت به سرور متصل شوند.

آنچه باید تحویل بدهید

محتوای پروژه خود را zip کرده و آپلود کنید.