

به نام خدا



دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران)
دانشکده‌ی مهندسی کامپیوتر و فناوری اطلاعات

پروژه‌ی چهارم مبانی برنامه‌سازی بازی Battleship (کشتی جنگی)

مدرس: احسان ناظر فرد



در پروژه‌ی پیش رو از شما خواسته شده است که بازی دو نفره و معروف battleship را آن گونه که توضیح داده خواهد شد، پیاده‌سازی نمایید. برنامه‌ای که شما باید بنویسید شبیه آن چیزی است که در این جا و این جا می‌توانید بازی کنید. مواردی که به این صورت نوشته شده‌اند امتیازی هستند و به صورت جداگانه توضیح داده خواهند شد.

قوانین بازی

۱. در حین بازی، هیچ کدام از بازیکنان محل کشتی‌های رقیبشان را نمی‌بینید.

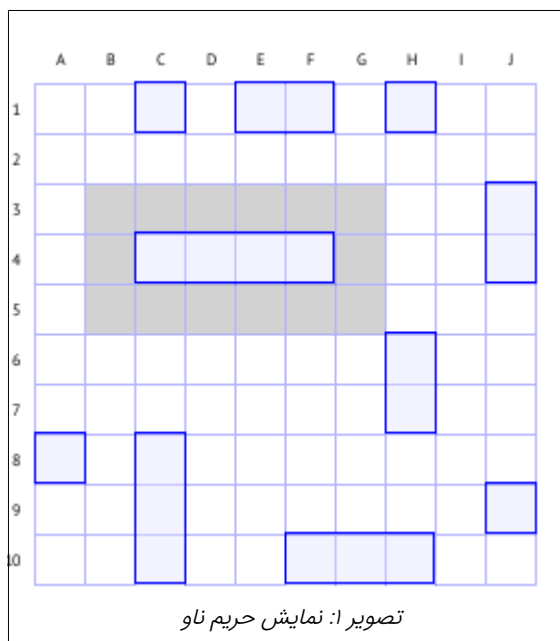
۲. ابعاد نقشه‌ی بازی ۱۰×۱۰ است.

۳. هر بازیکن ۱۰ کشتی از قرار زیر دارد:

نام	طول	تعداد
ناو	۴	۱
رزم ناو	۳	۲
ناوشکن	۲	۳
زیر دریایی	۱	۴

۴. هر کشتی می‌تواند به صورت عمودی و یا افقی در صفحه قرار بگیرد.

۵. هر کشتی به اندازه‌ی یک خانه اطراف خود حریم دارد و کشتی دیگری نباید در آن فضا قرار بگیرد.



تصویر ۱: نمایش حریم ناو

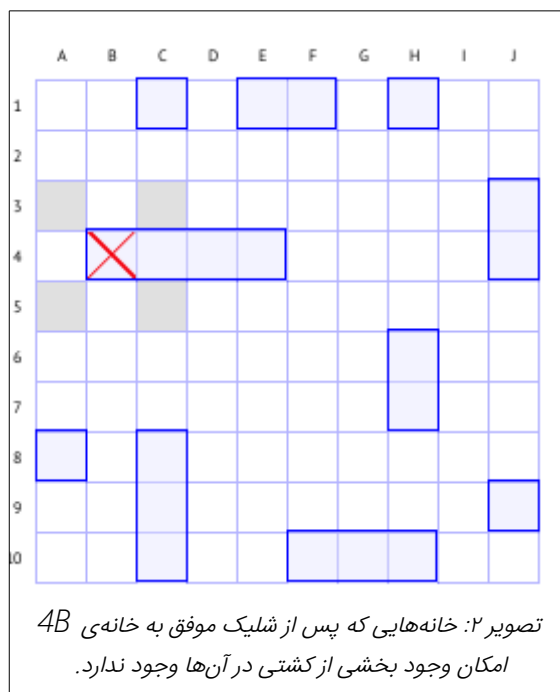
۶. هر بازیکن در نوبت خود به یکی از خانه‌های نقشه‌ی حریف شلیک می‌کند و مادامی‌که شلیک یک شلیک موفق باشد



می‌تواند به شلیک کردن ادامه دهد.

۷. بازیکن این امکان را دارد که خودش کشتی‌هایش را در صفحه بچیند.

۸. پس از هر شلیک موفق، خانه‌هایی که ممکن نیست ادامه‌ی کشتی اصابت شده در آن‌ها باشد در نقشه مشخص می‌شوند. این خانه‌ها بخشی از حریم کشتی هستند که در راستای شمالی-جنوبی و یا شرقی-غربی، که احتمال دارد ادامه‌ی کشتی در آن‌جا باشد، قرار ندارند. برای روشن شدن موضوع تصویر زیر را ببینید. در حقیقت این خانه‌ها به عنوان خانه‌هایی که مورد اصابت قرار گرفته‌اند ولی خالی بودند مشخص می‌شوند.



۹. بازی تا جایی ادامه پیدا می‌کند که یکی از بازیکنان تمامی کشتی‌های بازیکن دیگر را غرق کرده باشد. در انتهای بازی و با اعلام برنده، بازنده از محل کشتی‌های برنده آگاه می‌شود. (هر دو نقشه به صورت کامل نشان داده می‌شوند).



نکات پیاده‌سازی بازی

- در این بازی در حقیقت بازیکن دوم، رایانه است که با شما بازی خواهد کرد.
- هر ۲ نقشه، (نقشه‌ی شما و نقشه‌ی رایانه) را در ابتدای بازی تولید کنید و کشتی‌ها را با در نظر داشتن حریم آن‌ها و ابعاد نقشه به صورت تصادفی در آن‌ها بچینید. دقت کنید که مثلاً کشتی‌ای به طول ۳ نمی‌تواند از خانه‌ی 2B شروع شود و به سمت شمال ادامه داشته باشد! (در اینصورت از نقشه بیرون خواهد زد). شما می‌توانید کشتی‌هایتان را خودتان در نقشه‌تان بچینید. (با گرفتن مختصات x و y نقطه‌ی شروع هر کشتی و جهت قرار گرفتن آن‌ها از کاربر؛ شمال به جنوب، شرق به غرب، جنوب به شمال و یا غرب به شرق)
- برای بازی کردن هر چه راحت‌تر می‌توانید به جای گرفتن مختصات نقطه‌ی شلیک در حین بازی و یا مشخص کردن موقعیت کشتی‌ها در آغاز بازی، به جای استفاده از cin به صورت زیر عمل کنید:
یک کاراکتر مثل * برای نشان دادن موقعیت کنونی در نظر بگیرید (در حقیقت به جای ماوس خودمان این کاراکتر را در نظر می‌گیریم و جابه‌جا می‌کنیم). و مختصات آن را در دو متغیر به نام‌های `CURSOR_X` و `CURSOR_Y` نگهداری کنید و با استفاده از تابع `kb_hit` و کلیدهای بالا، پایین، چپ و راست مقادیر ذخیره شده در این متغیرها را به صورت مناسب تغییر دهید و محل جدید * را با پاک کردن صفحه و چاپ مجدد نقشه، نشان دهید. برای شلیک کردن هم از کلید `spacebar` استفاده کنید. * باید تنها در محدوده‌ی نقشه‌ی بازی جابه‌جا شود و خارج از آن نباید برود.
- با زدن کلید `escape` باید باید اجرای برنامه متوقف شود.
- هر بازیکن پس از انجام شلیک موفق، جایزه خواهد داشت، چنانچه نوبت حرکت رایانه باشد و جایزه داشته باشد باید با استفاده از توابع `delay` حرکت‌های او را نشان دهید. (هر حرکت مشخص شود و قابل تعقیب باشد مثلاً بین حرکت‌هایش ۳۰۰ میلی‌ثانیه وقفه باشد).
- صفحه را پس از هر حرکت به صورت کامل پاک کنید و مجدداً نقشه‌ها را در حالت جدید چاپ کنید. هر دو نقشه باید چاپ شوند ولی محل کشتی‌های رایانه نباید مشخص باشد. در انتهای بازی و در صورت باخت شما، محل کشتی‌های رایانه مشخص می‌شود.
- خانه‌هایی که به آن‌ها شلیک شده است، شلیک‌های موفق و خانه‌هایی که به آن‌ها شلیک نشده است باید با کاراکترها و رنگ‌های مناسب از هم قابل تمایز باشند و همچنین محل کشتی‌های خودتان در نقشه‌تان باید مشخص باشد.



- حرکت‌های رایانه کمی هوشمندانه باشند و تصادفی نباشند.
- شروع کننده‌ی بازی به صورت تصادفی انتخاب شود و در ابتدای بازی نام بازیکنان از کاربر گرفته شود (نام رایانه و نام خودتان). در پایان بازی با اعلام نام برنده برنامه خاتمه یابد.
- تصاویر استفاده شده در این فایل تنها به منظور بیان دقیق‌تر مسأله آورده شده‌اند.

نکات تحویل

- موعد انجام این پروژه تا ساعت ۲۳:۵۵ روز جمعه ۲۶ آذرماه ۹۵ در نظر گرفته شده است.
- قبل از آپلود کردن پروژه حتماً از صحت کدهایتان و اجرا شدن برنامه‌تان اطمینان حاصل کنید.
- حتماً حالت‌های مختلف با کد خود تست کنید و سعی کنید اشکالات کدتان را خودتان پیدا و رفع کنید. (به کمک trace کردن کد و یا break point)
- قبل از آپلود این دو کار را حتماً انجام دهید:
 - Clean کردن پروژه
 - حذف فایل SDF در پوشه‌ی پروژه
- تمرین خود را به صورت فایل فشرده و با فرمت PRJ4-9531111 در Edmodo آپلود کنید.