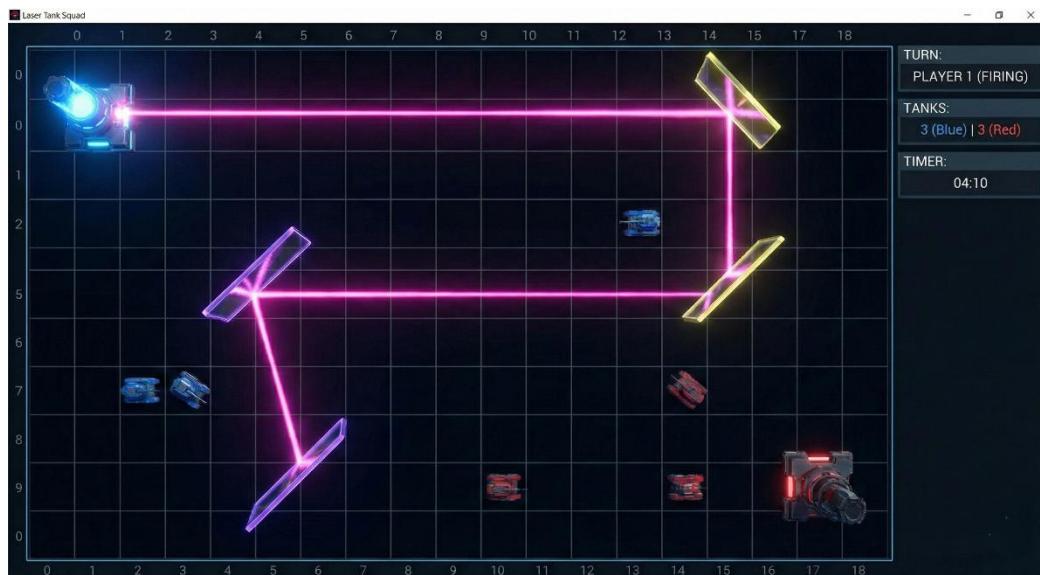




پروژه عملی درس مبانی برنامه‌نویسی کامپیوتر - دانشکده برق

پروژه شماره ۴: نبرد استراتژیک تانک‌های لیزری (Laser Tank Squad)



۱. اهداف پروژه

این پروژه با هدف ارزیابی توانایی در طراحی و پیاده‌سازی یک سیستم شبیه‌سازی کنسولی ارائه می‌شود. اهداف اصلی عبارت‌اند از:

- استفاده صحیح از مدیریت حافظه پویا (**Dynamic Memory Allocation**) برای کار با ماتریس‌های با ابعاد $m * n$.
- تسلط بر توابع بازگشتی (**Recursion**) برای شبیه‌سازی حرکت و بازتاب پرتو لیزر.
- استفاده از الگوریتم‌های آزمون و خطا (**Backtracking**) برای اعتبارسنجی نقشه بازی در زمان تولید.
- طراحی یک رابط کاربری کنسولی (**Console UI**) با روزرسانی لحظه‌ای و نمایش رنگی وضعیت‌ها.

۲. مقدمه

تا به حال در درس مبانی برنامه‌نویسی کامپیوتر با زبان C++ و داده‌ساختارها، دستورات آن و مفاهیم الگوریتمی آشنا شده‌اید. با توجه به آنچه تا اینجا یاد گرفته و تمرین کرده‌اید، از شما انتظار می‌رود با دانشی که دارید بتوانید پروژه‌هایی که نیاز به شبیه‌سازی دقیق، مدیریت داده‌ها و طراحی رابط کاربری (UI) در محیط ترمینال دارند را انجام دهید. در این پروژه، یک بازی استراتژیک نوبتی در محیط کنسول طراحی می‌شود. میدان نبرد یک جدول $m * n$ است که در آن تانک‌ها، آینه‌ها و منابع لیزر با یکدیگر تعامل دارند. هر بازیکن دارای یک منبع لیزر ثابت است و با استفاده از حرکت تانک‌ها و چرخش آینه‌ها تلاش می‌کند منبع یا تانک‌های حریف را نابود کند. تمرکز اصلی پروژه بر مدل‌سازی دقیق قوانین بازی، ترتیب رخدادها و مدیریت وضعیت سیستم است.

```
=====
P1 SCORE: 02 | P2 SCORE: 01 | TIME: 02:30 | TURN: P1
=====

      0   1   2   3   4
+---+---+---+---+
0 | S1 | . | \ | . | T2 |
+---+---+---+---+
1 | . | T1 | . | / | .
+---+---+---+---+
2 | \ | . | / | . | .
+---+---+---+---+
3 | . | . | . | T1 | \
+---+---+---+---+
4 | T2 | . | / | . | S2 |
+---+---+---+---+


[LOG]: Player 1 moved Tank to (1,1).
[LOG]: S1 Fixed Source ready for Horizontal/Vertical shot.
[CMD]: (M)ove Tank, (R)otate Mirror, (E)xit: _
```

نمایی از محیط کنسولی بازی

۳. ساختار صفحه بازی (Entities)

هر خانه از جدول $n * m$ دقیقاً می‌تواند شامل یکی از موارد زیر باشد:

3.1 منبع لیزر (Laser Source)

- برای هر بازیکن یک منبع لیزر ثابت وجود دارد.
- منبع بازیکن ۱ در مختصات (0,0) قرار دارد.
- منبع بازیکن ۲ در مختصات (m-1, n-1) قرار دارد.
- منبع قابل جابه‌جایی نیست.
- برخورد هر پرتو لیزر (خودی یا دشمن) با منبع باعث نابودی آن و پایان بازی می‌شود.

3.2 تانک‌ها (Tanks)

- تانک‌ها واحدهای متحرک هستند.
- تعداد تانک‌های هر بازیکن در ابتدای بازی توسط کاربر تعیین می‌شود و برای هر دو بازیکن برابر است.

- هر خانه در هر لحظه حداقل می‌تواند یک تانک داشته باشد.

3.3 آینه‌ها (Neutral Mirrors)

- آینه‌ها اشیای ثابت و بی‌طرف هستند.
- قابلیت جابه‌جایی ندارند.
- تنها عمل مجاز روی آن‌ها چرخش ۹۰ درجه‌ای است.
- هر آینه دارای عمر مفید محدود است (قانون فرسودگی).

4. تولید نقشه و اعتبارسنجی (Map Validation)

4.1 ابعاد صفحه

- ابعاد صفحه $n * m$ هستند.
- مقادیر (m) و (n) عددی صحیح بین ۴ تا ۱۰ بوده و در زمان اجرا از کاربر دریافت می‌شوند.

4.2 توزیع آینه‌ها

- در هر سطر جدول باید حداقل یک آینه به صورت تصادفی قرار داده شود.

4.3 محدوده امن (Safety Zone)

- در زمان تولید نقشه، هیچ تانک دشمنی نباید در محدوده مربعی به ضلع ۳ حول منبع لیزر هر بازیکن قرار گیرد.
- برای بازیکن ۱: محدوده $(0,0)$ تا $(2,2)$
- برای بازیکن ۲: محدوده $(m-3,n-3)$ تا $(m-1,n-1)$

4.4 تضمین مسیر تانک

- چیدمان اولیه آینه‌ها نباید مسیر عبور تانک‌ها از راست به چپ یا از بالا به پایین را به طور کامل مسدود کند.
- یعنی نباید در همه خانه‌های یک سطر یا ستون آینه قرار گرفته باشد

5. روند نوبت بازی (Turn Order)

هر نوبت بازی دقیقاً به ترتیب زیر انجام می‌شود:

۱. حرکت یا چرخش (انتخاب یکی) :

- بازیکن یا یک تانک خود را حرکت می‌دهد،
- یا یکی از آینه‌ها را ۹۰ درجه می‌چرخاند.

◦ یا یکی از تانک هایش به یکی از خانه های حول تانک به شعاع ۱ شلیک کند

۲. شلیک لیزر:

◦ پس از انجام عمل مرحله ۱، بازیکن جهت شلیک لیزر را انتخاب می‌کند:

- افقی (در راستای سطر)
- عمودی (در راستای ستون)

۳. بررسی برخوردها و اعمال نتایج:

◦ حرکت لیزر، برخوردها و حذف اشیاء بررسی و اعمال می‌شوند.

۶. قوانین حرکت و برخورد

6.1 حرکت تانک

- تانک می‌تواند به یکی از ۸ خانه مجاور (افقی، عمودی، قطری) حرکت کند.
- حرکت خارج از محدوده صفحه مجاز نیست.

6.2 برخورد تانک با تانک

- اگر یک تانک قصد ورود به یک خانه حاوی تانک (خودی یا دشمن) را داشته باشند:
 - هر دو تانک فوراً نابود شده و از صفحه حذف می‌شوند.

6.3 برخورد لیزر

- پرتو لیزر به صورت بازگشتی در صفحه حرکت می‌کند.
 - در برخورد با:
- تانک: تانک نابود می‌شود و پرتو متوقف می‌گردد.
- آینه: جهت پرتو تغییر کرده و از عمر آینه کاسته می‌شود.
- منبع لیزر: منبع نابود شده و بازی پایان می‌یابد.

6.3 اصابت گلوله تانک

- تانک می‌تواند به ۸ خانه اطرافش شلیک کند

۷. سیستم فرسودگی و بازتولید(Rusting & Respawn)

- هر آینه دارای عمر مفید برابر با ۴ بار بازتاب پرتو است.
- با هر بازتاب، عمر آینه یک واحد کاهش می‌یابد.
- پس از رسیدن عمر آینه به صفر:
 - آینه نابود می‌شود.
 - یک آینه جدید با عمر کامل در یک خانه تصادفی خالی ظاهر می‌شود.

۸. شرایط پیروزی و پایان بازی

بازی در یکی از حالت‌های زیر پایان می‌یابد:

1. نابودی کامل تانک‌های یکی از بازیکنان.
2. ورود یک تانک به خانه منبع لیزر حریف.
3. اصابت مستقیم پرتو لیزر به منبع لیزر حریف.
4. خروج توافقی (Exit): در این حالت بازیکنی که تانک بیشتری در صفحه دارد برنده است.

۹. رابط کاربری (UI)

- پس از هر نوبت، صفحه کسول باید پاک‌سازی (Clear Screen) شود.
- اطلاعات زیر در بالای صفحه نمایش داده شود:
 - بازیکن فعلی
 - تعداد تانک‌های باقی‌مانده هر بازیکن
 - زمان سپری شده از شروع بازی

۱۰. بخش امتیازی (Bonus Features)

1. نمایش مسیر لیزر با کلراکترهای \ / | - و رنگ صورتی.
2. نمایش رنگی وضعیت سلامت آینه‌ها:
 - بنفش: ۴
 - آبی: ۳
 - سبز: ۲

- زرد: ۱
 - قرمز: ۰ (شکسته)
۳. تایمر: نمایش زمان سپری شده از بازی
-

۱۱. نکات فنی پیاده‌سازی

- استفاده از یک ساختار داده مانند struct Cell برای نگهداری اطلاعات هر خانه الزامی است.
 - شبیه‌سازی حرکت لیزر باید به صورت بازگشتی انجام شود.
 - از ایجاد حلقه‌های بینهایت لیزر جلوگیری شود (کنترل وضعیت‌های تکراری).
-

۱۲. نحوه ارسال

- پروژه به صورت فایل ZIP با فرمت زیر ارسال شود:
StudentNumber1_StudentNumber2.zip
 - فایل ZIP باید شامل سورس‌کدها و یک فایل توضیحات (Documentation) باشد.
-

موفق باشید