



STAR  
WARS

# گزارش کار مینی پروژه

**تدوین گران :**

متین امیرپناه فر و نیما مخملی

زمستان 1402



## فهرست

1. معرفی بازی -----
2. ساختار پروژه -----
  1. کتابخانه ها -----
    1. iostream -----
    2. windows.h -----
    3. ctime -----
  2. ساختمان ها -----
    1. mySpaceShip -----
  3. توابع -----
    1. gameRun -----
    2. grandDraw -----
    3. horizontalDraw -----
    4. action -----
    5. Move -----
    6. Shoot -----
3. ثابت ها -----
4. منابع -----



## معرفی بازی

بازی به صورت انفرادی در یک زمین  $10 \times 10$  انجام می شود.

در سطر اول نام بازی درج شده است.

بازیکن در بازی دارای یک سفینه است که با کاراکتر # در صفحه مشخص می شود که قابلیت حرکت و یا شلیک در بازی را دارا است و سفینه بازیکن دارای سه فرصت بازی می باشد که به نام سلامتی در سطر دوم بازی مشخص شده است.

در صفحه بازی ده دشمنان وجود دارند که با کاراکتر (\*) مشخص شده اند.

سفینه با دستور حرکت (m) و سپس با تعیین کردن مسیر حرکت خود به راست (r)، چپ (l)، بالا (u) و یا پایین (d) می تواند موقعیت خود در زمین بازی را تغییر دهد در صورتی که سفینه به اشتباه بر روی یکی از سفینه های دشمن قرار بگیرد از سلامتی سفینه کم شده و سفینه به مکان شروع بازی باز می گردد .

با دستور شلیک (s) و سپس تعیین جهت شلیک به راست (r) یا چپ (l) سفینه می تواند سفینه ای از دشمن که در جهت شلیک قرار دارد را نابود کند.

بازی تا زمانی ادامه پیدا می کند که سفینه پیروز شود و یا شکست بخورد.

سفینه زمانی موفق می شود به پیروزی دست یابد که تمامی سفینه های دشمن موجود در نقشه را نابود کند.

در صورتی که بازیکن تمام سلامتی خود را از دست بدهد شکست می خورد.



## ساختار پروژه

برنامه به صورت ماژولار و از توابع و ساختمان های جداگانه طراحی شده تا قابلیت بهینه سازی و توسعه را داشته باشد.

کتابخانه ها و ساختمان ها

**<iostream>**

این کتابخانه کلاس ها و توابع کنترل ورودی و خروجی در ترمینال را در اختیار ما قرار می دهد.

**<windows.h>**

این کتابخانه دستورات کلی کار با ترمینال ویندوز را در اختیار ما قرار می دهد.

توابع استفاده شده از این کتابخانه موارد زیر است.

**<sleep>**

مسئولیت ایجاد تاخیر زمانی در برنامه را دارا است

**<system>**

پاک کردن صفحه ترمینال را با پارامتر cls و ایجاد توقف را با

پارامتر pause را انجام می دهد.

**<ctime>**

**<spaceship>**

این ساختمان داده های سفینه ما را از جمله موقعیت اولیه، موقعیت کنونی و سلامتی را نگهداری می کند.

توابع

**(gameRun)**

این تابع وظیفه ایجاد مقدمات بازی را برعهده دارد، و پیشنیازها برنامه را برای بازی کردن کاربر آماده میکند



## **()horizontalDraw**

این تابع وظیفه طراحی قسمت افقی صفحه بازی را دارد.

```
void horizontalDraw(void)
{
    cout << BOLDBLUE;
    for (int i=0; i < col; i++)
        cout << " ---";
    cout << RESET;
}
```

## **()granddraw**

این تابع وظیفه طراحی صفحه بازی را بر عهده دارد که کاربر میتواند با اعمال دستورات سفینه را روی آن به حرکت یا حمله وا دارد.

```
horizontalDraw();
cout << endl;
for (int j = 0; j < col; j++)
{
    cout << BOLDBLUE << "| " << RESET;
    if (j == mySpaceShip.x && i == mySpaceShip.y)
        cout << BOLDGREEN << mySpaceShip.c << ' ' << RESET;
    else if (condition[j][i] == 2)
    {
        cout << BOLDRED << "*" << RESET;
        win = true;
    }
    else
        cout << " ";
    cout << BOLDBLUE << "| " << RESET << endl;
}
horizontalDraw();
if (win == false)
{
    system("cls");
    cout << BOLDGREEN << "*****YOU WIN*****" << endl << RESET;
    system("pause");
    exit(0);
}
```

## **()Action**

این تابع وظیفه گرفتن و انتقال دستور از کاربر و تعیین حرکت یا شلیک را بر عهده دارد.

## **()Move**

این تابع وظیفه گرفتن و انتقال دستور تعیین نوع حرکت را بر عهده دارد، کاربر در این بازی میتواند با اعمال دستورات (u,d,r,l) سفینه را به

حرکت درآورد اما اگر سفینه در حرکت خود با دشمن برخورد کند یک جان از دست خواهد داد.

### ()Shoot

این تابع مسئولیت گرفتن و انتقال دستور شلیک و نابودی سفینه های دشمن را برعهده دارد، که کاربر میتواند با دستور `ج` به سمت چپ و راست خود شلیک کند تا در صورت وجود دشمن آن را از بین ببرد.

```
switch (shoot)
{
case 'r':
do
{
x++;
if (2 == condition[x][mySpaceShip.y])
{
condition[x][mySpaceShip.y] = 0;
break;
}
}while(x < col);
flag = true;
break;
case 'l':
do
{
x--;
if (2 == condition[x][mySpaceShip.y])
{
condition[x][mySpaceShip.y] = 0;
break;
}
}while(x >= 0);
flag = true;
break;
default:
cout << RED << "undefined please try again" << RESET << endl;
flag = false;
}
}while (flag == false);
```

3 با استفاده از دستور switch,case

## ثابت ها

&lt; define# &gt;

```
#define RESET "\033[0m"
#define BLACK "\033[30m" /* Black */
#define RED "\033[31m" /* Red */
#define GREEN "\033[32m" /* Green */
#define YELLOW "\033[33m" /* Yellow */
#define BLUE "\033[34m" /* Blue */
#define MAGENTA "\033[35m" /* Magenta */
#define CYAN "\033[36m" /* Cyan */
#define WHITE "\033[37m" /* White */
#define BOLDBLACK "\033[1m\033[30m" /* Bold Black */
#define BOLDRED "\033[1m\033[31m" /* Bold Red */
#define BOLDGREEN "\033[1m\033[32m" /* Bold Green */
#define BOLDYELLOW "\033[1m\033[33m" /* Bold Yellow */
#define BOLDBLUE "\033[1m\033[34m" /* Bold Blue */
#define BOLDMAGENTA "\033[1m\033[35m" /* Bold Magenta */
#define BOLDCYAN "\033[1m\033[36m" /* Bold Cyan */
#define BOLDWHITE "\033[1m\033[37m" /* Bold White */

#define row 10
#define col 10
#define NumberOfEnemies 10
```

ما در این پروژه از ثوابتی با کلمات کلیدی **define#** استفاده کرده ایم تعدادی از این ثوابت رنگ ها و **bold** های آنان است و سه ثابت دیگر مختص به صفحه ده درده بازیمان است **row** و **col** که برای نمایش صفحه هستند را مقدار 10 داده ایم و همچنین تعداد دشمنان ما که به صورت رندوم چیده خواهند شد هم مقدار 10 داده ایم.



## output //

```
*****STARWARS*****
health:3
| | | | | | | | | | |
| | | | | | | | | |
| * | | * | | * | | * | |
| | | | | | | | | |
| | | | | | | * | | | |
| | | | | | | | * | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | | | |
| | | | | | | # | | | |
| | | | | | | * | | * | |
| | | | | | | | | |
| | * | | | | | * | | |
| | | | | | | | | |
move or shoot[m/s]:
```





## پیوندها و منابع

### Books

cpp-How to program Deitel 1997 edition

### Websites

c++ programming

<https://www.w3schools.com/cpp/default.asp>

<https://www.geeksforgeeks.org/c-plus-plus/?ref=gcse>

learning git

<https://faradars.org/courses/fvgit9609-git-github-gitlab>



## آدرس پروژه در گیت هاب

<https://github.com/Matin0789/StarWar-mini-project-of-FCP-.git>