

برنامه سازی پیشرفته-۹۸۱-دانشگاه بوعلی سینا
استاد : دکتر مرتضی یوسف صنعتی

محسن زارعین - ۹۷۱۲۳۵۸۰۱۹

مقدمه و توضیح پروژه :

هدف از این پروژه ساخت یک بازی استراتژیک شامل ده قهرمان با نام های mrsghost و alphaman و kratos و robi و giant و dr.marry و sniper و professor و leon و commander میباشد که هر یک دارای قدرت و جان متفاوتی هستند. در هر دور بازی هر بازیکن 5 قهرمان بصورت دل خواه انتخاب میکند و در هر دور بازی هر بازیکن تنها حق شلیک فقط با یک کاراکتر را دارد. بازی تا زمانی ادامه دارد که جان تمام کاراکتر های یک بازیکن زودتر تمام شود

توضیحات کلی کد بازی :

یک کلاس انتزاعی از hero داریم که 10 کلاس مختلف قهرمان ها از ان بصورت public ارث میبرند . این کلاس انتزاعی شامل توابع زیر است :

```
virtual ~Hero();  
virtual void Attack(sf::Vector2i,Grid*,Grid*,std::vector<Hero*>&) = 0;  
virtual sf::Vector2i& get_position_on_grid() ;  
virtual short int& getHealth();  
virtual unsigned short int& getPower();  
virtual bool isSelected();  
virtual bool getAttackPermission();
```

هر یک از کلاس های ارث برده توابع مجازی بالا را override میکنند . برای مثال کلاس الفا من بصورت زیر است :

```

class AlphaMan : public Hero
{
public:
    AlphaMan(sf::RenderWindow*);
    virtual ~AlphaMan();
    virtual void Attack(sf::Vector2i, Grid*, Grid*, std::vector<Hero*> &)
override;
    virtual short int& getHealth() override;
    virtual unsigned short int& getPower() override;
    virtual sf::Vector2i& get_position_on_grid() override;
    void set_position_on_grid(sf::Vector2i);
    AlphaManCard alphamancard ;
private:
    void SpecialPower(sf::Vector2i, Grid*, std::vector<Hero*> &);
};

```

از طرفی بازی یک کلاس منو دارد که کل بازی توسط شی ای که از این کلاس در تابع main ساخته میشود اجرا میشود . این کلاس متدی به اسم exec که روی شی ساخته شده از این کلاس در تابع main فراخوانی میشود . این کلاس یک متغیر از جنس enum دارد که تمام وضعیت های بازی در آن تعریف شده است و براساس حالت فعلی که این متغیر دارد گرافیک بازی لود میشود و این حالات مختلف در وضعیت های متفاوت بازی دست به دست میشوند و window مورد نظر نمایش داده میشود

نحوه استفاده از پلی مورفیسم یا چند ریختی در پروژه :

یک کلاس player داریم که در آن یک vector از جنس *hero (پدر قهرمان ها) داریم . حال قهرمان هایی که هر بازیکن انتخاب میکند در این vector چیده میشود طوری که حداکثر سایز این vector پنج است .

حال با استفاده از این وکتور حین بازی ، از طریق پوینتر پدری تابع اتم قهرمان مورد نظر فراخوانی میشود و تاثیرات لازم روی حریف از طریق این تابع اعمال میشود از طرفی بعضی از قهرمان ها شامل توابع خاصی هستند که پدر آن ها را ندارد لذا با تبدیل پوینتر پدری به بچه بصورت زیر آن تابع را فراخوانی میکنیم :

برای مثال mrs.ghost بصورت زیر است

```
mrsghost * MGH = dynamic_cast<mrsghost*>(father pointer);
```