بسمه تعالی



عنوان:

**گزارشکار فاز آخر درس برنامه نویسی پیشرفته**

استاد:

**سرکار خانم مهندس بطحائیان**

دانشجو:

**نسترن منصوری**

**9612358041**

ترم تحصیلی:

**4001**

***فهرست***

[هدف: 3](#_Toc95215082)

[توضیحات کد: 3](#_Toc95215083)

[کلاس state: 3](#_Toc95215084)

[State.h: 3](#_Toc95215085)

[State.cpp: 4](#_Toc95215086)

[کلاس Shape: 5](#_Toc95215087)

[Shape.h: 5](#_Toc95215088)

[Shape.cpp: 6](#_Toc95215089)

[کلاس playGame: 7](#_Toc95215090)

[playGame.h: 7](#_Toc95215091)

[playGame.cpp: 8](#_Toc95215092)

[Main(): 12](#_Toc95215093)

# هدف:

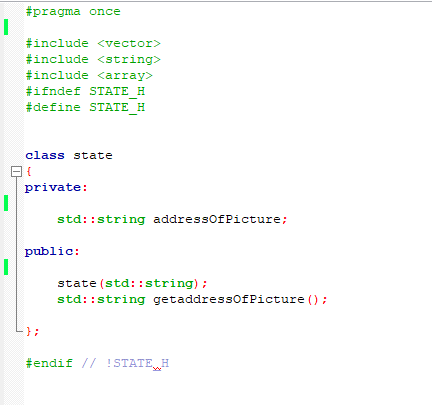
هدف از پیادهسازی این فاز افزودن بخش گرافیکی به کد پیاده سازی شده در فازهای قبلی است.

# توضیحات کد:

برای پیاده سازی این قسمت کتابخانه SFML استفاده شده است. علاوه بر این سه کلاس تحت نام های State، shape و playGame هم اضافه شده است که در ادامه توضیح داده خواهد شد.

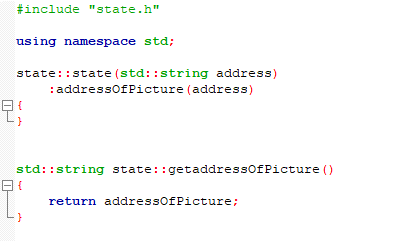
# کلاس state:

## State.h:



در این کلاس ادرس تصاویر و جایی که هستند در متغیر خصوصی ای ذخیره می شود.

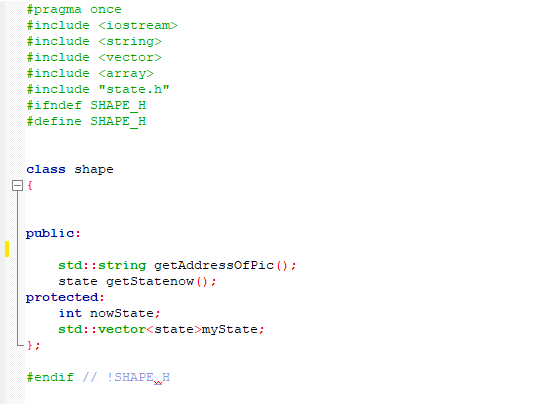
## State.cpp:



در پیاده سازی ایم کلاس یک کانستراکتو رداریم که در آن با گرفتن آدرس تصویر مقدار آن را در متغیر addressOfPicture ذخیره می کند. در تابعی دیگر مقدار ذخیره شده در متغیر خصوصی addressOfPicture برمی گرداند.

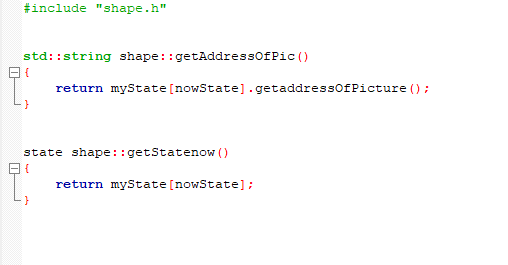
# کلاس Shape:

## Shape.h:



همانطور که در تصویر مشخص است در این کلاس دو متغیر protected داریم که یکی از آنها برای ذخیره state عکس ها و دیگری وکتوری برای ذخیره ی stateها است.

## Shape.cpp:

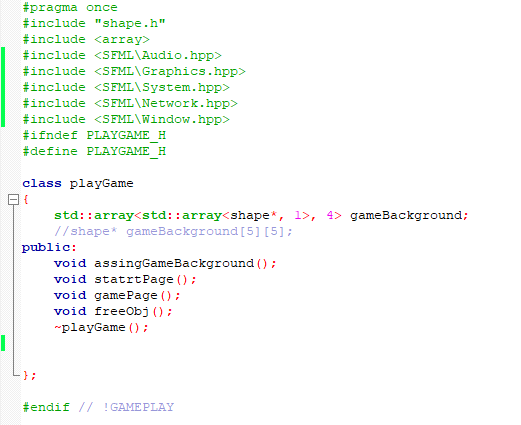


در پیاده سازی کلاس shape دو تابع داریم. یکی از توابع آدرس عکس ها را برمی گرداند که نام آن getAddressOfPic است.

تابع دوم state فعلی تصاویر را فراخوانی می کند.

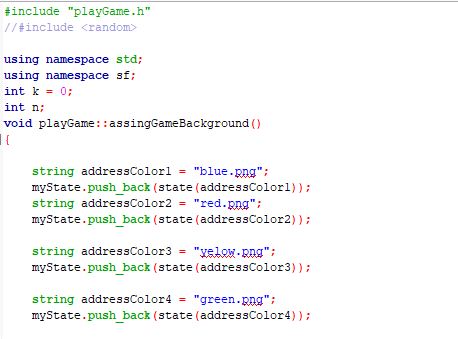
# کلاس playGame:

## playGame.h:



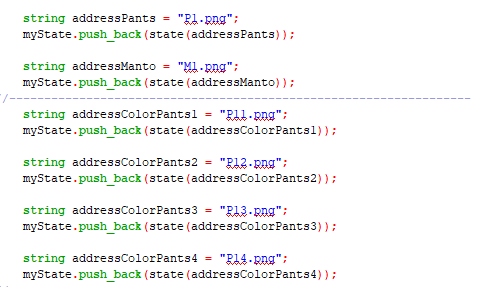
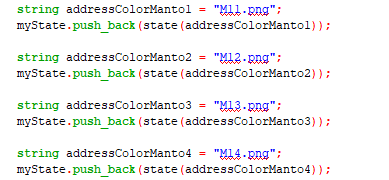
در این کلاس یک ارایه دو بعدی تعریف شده است برای قرار دادن تصاویر رنگ هایی که کاربر می تواند انتخاب کند. همچنین یکسری توابع در این کلاس موجود است که در ادامه توضیح داده شده است.

## playGame.cpp:



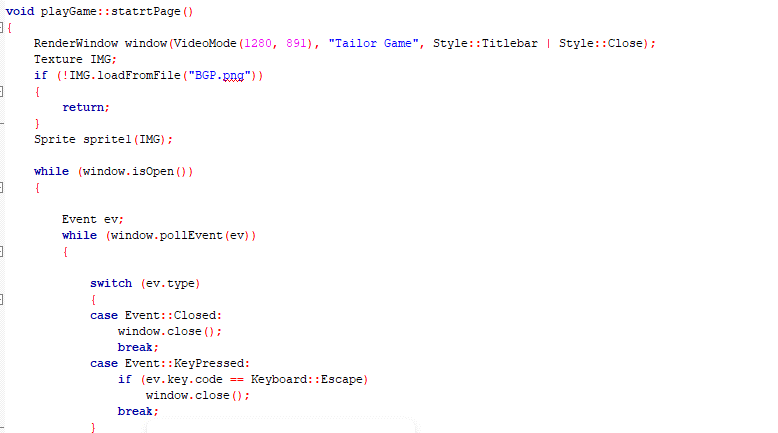
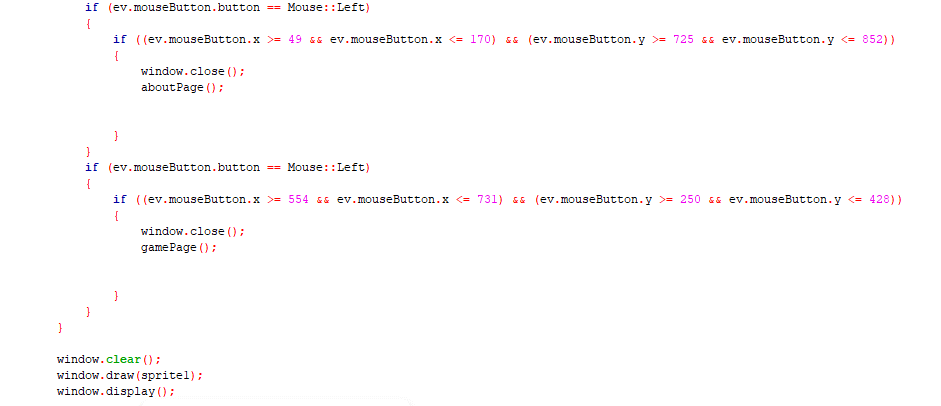
اولین تابع، تابع assingGameBackground است. در قسمت اول این تابع آدرس تصاویری که به کاربر نمایش داده می شود در state ها ذخیره می شوند. که در ابتدا تصاویر رنگ ها برای کاربر نمایش داده می شود تا کاربر رنگ پارچه موردنظر خود را از بین چهار رنگ موجود انتخاب کند.

بعد از انتخاب رنگ پارچه شلوار و مانتو همان رنگ به کاربر نشان داده می شود تا طراحی مورد نظر خود را از بین آن دو انتخاب کند.

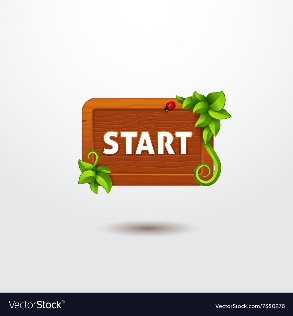
 

همانطور که میبیند آدرس تمامی تصاویر ذخیره می شوند تا در صورت انتخاب کاربر به او نمایش داده شوند.

تابع بعدی، تابع startPage می باشد.

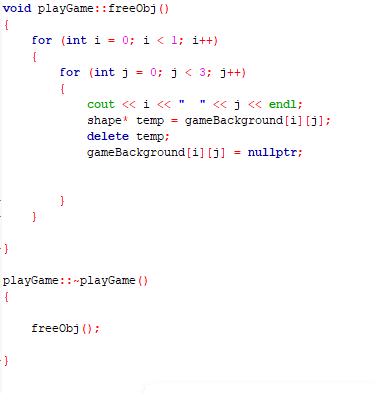
 

این تابع در ابتدا تصویر شروع بازی را نمایش می دهد. صفحه start بازی که به شکل زیر است و کاربر با کلیک کردن بر آن وارد محیط بازی می شود.



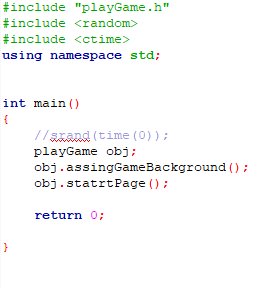
تابع gamePage یک دیگر از توابع این کلاس و درواقع اصلی ترین تابع است؛ که پیاده سازی و کلیک ردن ر.ی تصا.یر همگی در این تابع پیاده سازی شده اند.

و دوتابع اخر بازی که یکی دستراکتوری است که تابع دیگر را فراخوانی می کند.



تابع freeObj برای رها سازی حافظه ی گرفته شده یا همان پاک سازی تصاویر از صفحه نمایش پیاده سازی شده است که آرایه gameBackground را در یک حلقه تو در تو تخلیه می کند.

# Main():



همانطور که مشخص است در تابع main برنامه هم تغییراتی بوجود امده به این صورت که با اجرای برنامه ایتدا یک شی از کلاس playGame ایجاد می شود سپس با این شی توبع assignGameBackground و startPage فراخوانی می شود.