



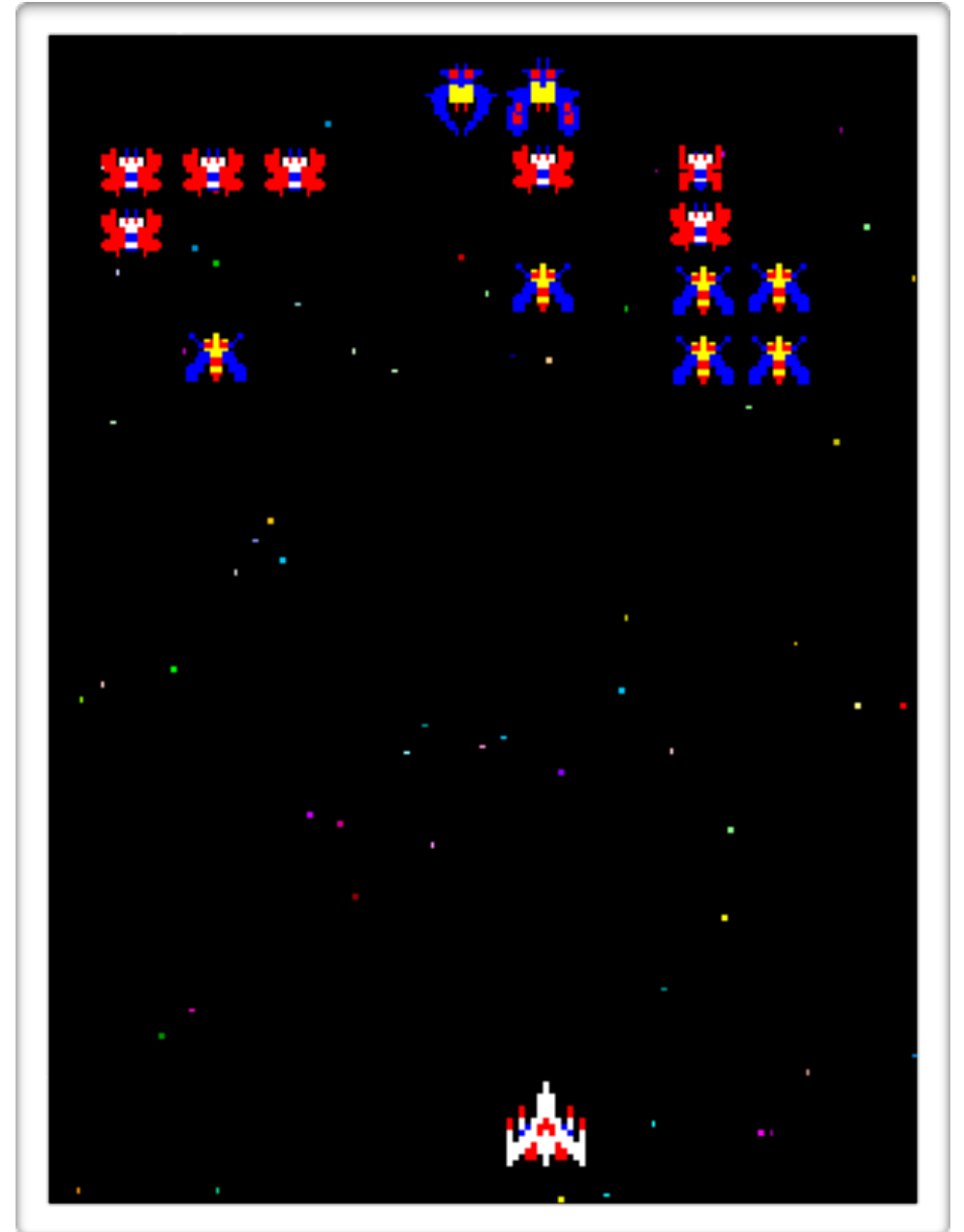
Rafael Vieira Coelho

rafaelvc2@gmail.com



Projeto

Jogo Alien Invasion



Uma Espaçonave que Atira

- Vamos criar um jogo com a biblioteca (coleção de módulos) **Pygame**
- Permite administrar imagens gráficas, animações e sons.
- Criaremos uma nave que se move da direita para a esquerda e atira em resposta à entrada do usuário.
- Crie uma pasta chamada **alien_invasion** em seu computador. Todos os arquivos do projeto serão salvos nesta pasta.

Planejando o seu Projeto

- Ao desenvolver um projeto grande, deve-se criar um plano antes de começar a escrever o código.
- No caso de um jogo, precisamos descrever o gameplay do jogo:

“Na Invasão Alienígena, o jogador controla uma espaçonave que aparece na parte inferior central da tela. O jogador pode mover a espaçonave para a direita e para a esquerda usando as teclas de direção e atirar usando a barra de espaço. Quando o jogo começa, uma frota de alienígenas enche o céu e se desloca na tela para os lados e para baixo. O jogador atira nos alienígenas e os destrói. Se o jogador atingir todos os alienígenas, uma nova frota, que se moverá mais rapidamente que a frota anterior, aparecerá. Se algum alienígena atingir a espaçonave do jogador ou alcançar a parte inferior da tela, o jogador perderá uma nave. Se o jogador perder três espaçonaves, o jogo terminará.”

Instalando o Pygame: Gerenciador pip

Instalando pacotes Python com o pip

As versões mais recentes de Python vêm com pip instalado, portanto verifique antes se ele já está em seu sistema. Em Python 3, o pip às vezes é chamado de *pip3*.

Verificando se o pip está instalado no Linux e no OS X

Abra uma janela do terminal e digite o seguinte comando:

```
$ pip --version  
u pip 7.0.3 from /usr/local/lib/python3.5/dist-packages (python 3.5)
```

Verificando se o pip está instalado no Windows

Abra uma janela do terminal e digite o seguinte comando:

```
$ python -m pip --version  
u pip 7.0.3 from C:\Python35\lib\site-packages (python 3.5)
```

Instalando o Pygame no Linux

Instalando o Pygame no Linux

Se você usa Python 2.7, instale o Pygame utilizando o gerenciador de pacotes. Abra uma janela de terminal e execute o comando a seguir, que fará o download e instalará o Pygame em seu sistema:

```
$ sudo apt-get install python-pygame
```

Teste sua instalação em uma sessão de terminal com o seguinte:

```
$ python  
>>> import pygame  
>>>
```

Se nenhum resultado aparecer, é sinal de que Python importou o Pygame e você estará pronto para ir para a seção “Dando início ao projeto do jogo”.

Dependências do Pygame

Se usar Python 3, dois passos são necessários: instalar as bibliotecas das quais o Pygame depende e fazer o download e a instalação do Pygame.

Dê o comando a seguir para instalar as bibliotecas necessárias ao Pygame (Se você usa um comando como `python3.5` em seu sistema, substitua `python3-dev` por `python3.5-dev`.)

```
$ sudo apt-get install python3-dev mercurial
```

```
$ sudo apt-get install libSDL-image1.2-dev libSDL2-dev libSDL-ttf2.0-dev
```

Esses comandos instalarão as bibliotecas necessárias para executar a Invasão Alienígena com sucesso. Se quiser habilitar algumas funcionalidades mais sofisticadas do Pygame, por exemplo, a capacidade de adicionar sons, acrescente também as bibliotecas a seguir:

```
$ sudo apt-get install libSDL-mixer1.2-dev libportmidi-dev
```

```
$ sudo apt-get install libswscale-dev libsmpeg-dev libavformat-dev libavcodec-dev
```

```
$ sudo apt-get install python-numpy
```

Instalando o Pygame no Windows

Instalando o Pygame no Windows

O projeto Pygame está hospedado em um site de compartilhamento de código chamado Bitbucket. Para instalar o Pygame em sua versão de Windows, encontre um instalador para Windows em <https://bitbucket.org/pygame/pygame/downloads/> que corresponda à versão de Python que você utiliza. Se você não encontrar um instalador apropriado em Bitbucket, dê uma olhada em <http://www.lfd.uci.edu/~gohlke/pythonlibs/#pygame>.

Depois de baixar o arquivo apropriado, execute o instalador se for um arquivo `.exe`.

Se você tiver um arquivo terminado com `.whl`, copie esse arquivo para o diretório de seu projeto. Abra uma janela de comandos, navegue até a pasta em que você copiou o instalador e use o pip para executá-lo:

```
> python -m pip install --user pygame-1.9.2a0-cp35-none-win32.whl
```


Dando Início ao Projeto do Jogo

```
1  import sys, pygame
2
3  def main():
4      pygame.init()
5      tela = pygame.display.set_mode((1200, 800))
6      pygame.display.set_caption("Alien Invasion")
7      while True:
8          for evento in pygame.event.get():
9              if evento.type == pygame.QUIT:
10                 sys.exit()
11                 pygame.display.flip()
12
13  main()
14
```

Atualiza a tela

Adicionando a Cor de Fundo

```
1  import sys, pygame
2
3  def main():
4      pygame.init()
5      tela = pygame.display.set_mode((1200, 800))
6      pygame.display.set_caption("Alien Invasion")
7      while True:
8          for evento in pygame.event.get():
9              if evento.type == pygame.QUIT:
10                 sys.exit()
11                 cor_fundo = (230, 230, 230)
12                 tela.fill(cor_fundo)
13                 pygame.display.flip()
14
15  main()
```

Cores Disponíveis

| | | | | | |
|---|---|---|---|--|---|
| www.Rags-Int-Inc.com | | | | | |
| 94.28.13 11° 86% 37% 01 Dark Skin | 241.148.108 18° 55% 95% 02 Light Skin | 97.119.171 222° 43% 67% 03 Blue Sky | 90.103.39 72° 62% 40% 04 Foliage | 164.131.196 270° 33% 77% 05 Blue Flower | 140.253.153 127° 46% 99% 06 Bluish Green |
| 255.116.21 24° 92% 100% 07 Orange | 7.47.122 219° 94% 48% 08 Purplish Blue | 222.29.42 356° 87% 87% 09 Moderate Red | 69.0.68 301° 100% 27% 10 Purple | 187.255.19 77° 93% 100% 11 Yellow Green | 255.142.0 33° 100% 100% 12 Orange Yellow |
| 0.0.142 240° 100% 56% 13 Blue | 64.173.38 108° 78% 68% 14 Green | 203.0.0 0° 100% 80% 15 Red | 255.217.0 51° 100% 100% 16 Yellow | 207.3.124 324° 99% 81% 17 Magenta | 0.148.189 193° 100% 74% 18 Cyan |
| 255.255.255 0° 0% 100% 19 White | 249.249.249 0° 0% 98% 20 Neutral 8 | 180.180.180 0° 0% 71% 21 Neutral 6.5 | 117.117.117 0° 0% 46% 22 Neutral 5 | 53.53.53 0° 0% 21% 23 Neutral 3.5 | 0.0.0 0° 0% 0% 24 Black |
| Numeric color values from Macbeth reference chart Adobe RGB 1998 | | | | | |

https://www.rapidtables.com/web/color/RGB_Color.html

Criando uma Classe de Configurações

alien_invasion.py

```
1 import sys, pygame
2
3 import settings as s
4
5 def main():
6     pygame.init() #inicia o modo gráfico do pygame
7     config = s.Settings() #cria o objeto de configurações
8     tela = pygame.display.set_mode((config.largura, config.altura)) #janela de 1200 x 800 pixe
9     pygame.display.set_caption("Alien Invasion") #define o título
10    while True:
11        for evento in pygame.event.get(): #verifica todos os eventos ocorridos
12            if evento.type == pygame.QUIT: #verifica se a janela foi fechada
13                sys.exit()
14            cor_fundo = (230, 230, 230) # define a cor cinza (R - Red, G - Green, B - Blue): 0-255
15            tela.fill(cor_fundo) #preenche a janela com a cor de fundo
16            pygame.display.flip() #redesenha a tela
17
18    main()
```

settings.py

```
1 class Settings():
2
3     def __init__(self):
4         self.largura = 1200
5         self.altura = 800
6         self.cor_fundo = (230, 230, 230)
7
```

Qual o erro do arquivo alien_invasion.py?

Adicionando a Imagem da Espaçonave: ship.py

```
1 import pygame
2
3 from random import randint
4
5 class Ship():
6
7     def __init__(self, tela):
8         self.tela = tela
9         #carrega a imagem
10        self.numero_imagem = randint(1,6)
11        self.nome_imagem = 'imagens/nave' + str(self.numero_imagem) + '.png'
12        self.imagem = pygame.image.load(self.nome_imagem)
13        self.retangulo = self.imagem.get_rect()
14        self.retangulo_tela = tela.get_rect()
15        #coloca a imagem na parte inferior da tela
16        self.retangulo.centerx = self.retangulo_tela.centerx
17        self.retangulo.bottom = self.retangulo_tela.bottom
18
19        #desenha a nave na sua posição atual
20    def desenha(self):
21        self.tela.blit(self.imagem, self.retangulo)
```

Adicionando a Imagem da Espaçonave: alien_invasion.py

```
1 import sys, pygame
2 import settings as s
3 from ship import Ship
4
5 def main():
6     pygame.init() #inicia o modo gráfico do pygame
7     config = s.Settings() #cria o objeto de configurações
8     tela = pygame.display.set_mode((config.largura, config.altura)) #janela de 1200 x 800 pixels
9     pygame.display.set_caption("Alien Invasion") #define o título
10    nave = Ship(tela)
11    while True:
12        for evento in pygame.event.get(): #verifica todos os eventos ocorridos
13            if evento.type == pygame.QUIT: #verifica se a janela foi fechada
14                sys.exit()
15        cor_fundo = (230, 230, 230) # define a cor cinza (R - Red, G - Green, B - Blue): 0-255
16        tela.fill(cor_fundo) #preenche a janela com a cor de fundo
17        nave.desenha() #desenha a nave na tela
18        pygame.display.flip() #redesenha a tela
19
20 main()
21
```

Refatorando o Código: game_functions.py

```
1  #!/usr/bin/env python
2  # coding: utf8
3
4  __AUTHOR__ = "Rafael Vieira Coelho"
5  __DATE__ = "19/05/2019"
6
7  #funções auxiliares
8  import sys, pygame
9
10 #responde a eventos de pressionamento de teclas e mouse
11 def testa_eventos():
12     for evento in pygame.event.get(): #verifica todos os eventos ocorridos
13         if evento.type == pygame.QUIT: #verifica se a janela foi fechada
14             sys.exit()
15
16 #atualiza as imagens na tela e redesenha a mesma
17 def atualiza_tela(config, tela, nave):
18     tela.fill(config.cor_fundo) #preenche a janela com a cor de fundo
19     nave.desenha() #desenha a nave na tela
20     pygame.display.flip() #redesenha a tela
21
```


Refatorando o Código: alien_invasion.py

```
1  #!/usr/bin/env python
2  # coding: utf8
3
4  __AUTHOR__ = "Rafael Vieira Coelho"
5  __DATE__ = "19/05/2019"
6
7  import sys, pygame
8  import settings as s
9  from ship import Ship
10 import game_functions as g
11
12 def main():
13     pygame.init() #inicia o modo gráfico do pygame
14     config = s.Settings() #cria o objeto de configurações
15     tela = pygame.display.set_mode((config.largura, config.altura)) #janela de 1200 x 800 pixels
16     pygame.display.set_caption("Alien Invasion") #define o título
17     nave = Ship(tela)
18     while True:
19         g.testa_eventos()
20         g.atualiza_tela(config, tela, nave)
21
22 main()
```


Pilotando a Nave: game_functions.py

```
10  #responde a eventos de pressionamento de teclas e mouse
11  def testa_eventos(nave):
12      for evento in pygame.event.get(): #verifica todos os eventos ocorridos
13          if evento.type == pygame.QUIT: #verifica se a janela foi fechada
14              sys.exit()
15          elif evento.type == pygame.KEYDOWN:
16              if evento.key == pygame.K_RIGHT: # Move a espaçonave para a direita
17                  nave.move_direita = True
18              elif evento.key == pygame.K_LEFT: # Move a espaçonave para a esquerda
19                  nave.move_esquerda = True
20              elif evento.key == pygame.K_UP: # Move a espaçonave para cima
21                  nave.move_cima = True
22              elif evento.key == pygame.K_DOWN: # Move a espaçonave para baixo
23                  nave.move_baixo = True
24          elif evento.type == pygame.KEYUP:
25              if evento.key == pygame.K_RIGHT:
26                  nave.move_direita = False
27              elif evento.key == pygame.K_LEFT:
28                  nave.move_esquerda = False
29              elif evento.key == pygame.K_UP:
30                  nave.move_cima = False
31              elif evento.key == pygame.K_DOWN:
32                  nave.move_baixo = False
33
34  #atualiza as imagens na tela e redesenha a mesma
35  def atualiza_tela(config, tela, nave):
36      tela.fill(config.cor_fundo) #preenche a janela com a cor de fundo
37      nave.desenha() #desenha a nave na tela
38      pygame.display.flip() #redesenha a tela
```

```
11 class Ship():
12
13     def __init__(self, tela):
14         self.velocidade = 5
15         self.tela = tela
16         #carrega a imagem
17         self.numero_imagem = randint(1,6)
18         self.nome_imagem = 'imagens/nave' + str(self.numero_imagem) + '.png'
19         self.imagem = pygame.image.load(self.nome_imagem)
20         self.retangulo = self.imagem.get_rect()
21         self.retangulo_tela = tela.get_rect()
22         #coloca a imagem na parte inferior da tela
23         self.retangulo.centerx = self.retangulo_tela.centerx
24         self.retangulo.bottom = self.retangulo_tela.bottom
25         self.move_direita = False
26         self.move_esquerda = False
27         self.move_cima = False
28         self.move_baixo = False
29
30     #desenha a nave na sua posição atual
31     def desenha(self):
32         self.tela.blit(self.imagem, self.retangulo)
33
34     #atualiza a posição da espaçonave de acordo com a flag de movimento
35     def atualiza(self):
36         if self.move_direita:
37             self.retangulo.centerx += self.velocidade
38         elif self.move_esquerda:
39             self.retangulo.centerx -= self.velocidade
40         elif self.move_baixo:
41             self.retangulo.centery += self.velocidade
42         elif self.move_cima:
43             self.retangulo.centery -= self.velocidade
```

Pilotando a Nave: alien_invasion.py

```
1  #!/usr/bin/env python
2  # coding: utf8
3
4  __AUTHOR__ = "Rafael Vieira Coelho"
5  __DATE__ = "19/05/2019"
6
7  import sys, pygame
8  import settings as s
9  from ship import Ship
10 import game_functions as g
11
12 def main():
13     pygame.init() #inicia o modo gráfico do pygame
14     config = s.Settings() #cria o objeto de configurações
15     tela = pygame.display.set_mode((config.largura, config.altura)) #janela de 1200 x 800 pixels
16     pygame.display.set_caption("Alien Invasion") #define o título
17     nave = Ship(tela)
18     while True:
19         g.testa_eventos(nave)
20         g.atualiza_tela(config, tela, nave)
21         nave.atualiza()
```

Limitando o Alcance da Nave

```
7 import pygame
8
9 from random import randint
10
11 class Ship():
12
13     def __init__(self, tela):
14         self.velocidade = 5
15         self.tela = tela
16         #carrega a imagem
17         self.numero_imagem = randint(1,6)
18         self.nome_imagem = 'imagens/nave' + str(self.numero_imagem) + '.png'
19         self.imagem = pygame.image.load(self.nome_imagem)
20         self.retangulo = self.imagem.get_rect()
21         self.retangulo_tela = tela.get_rect()
22         #coloca a imagem na parte inferior da tela
23         self.retangulo.centerx = self.retangulo_tela.centerx
24         self.retangulo.bottom = self.retangulo_tela.bottom
25         self.move_direita = False
26         self.move_esquerda = False
27         self.move_cima = False
28         self.move_baixo = False
29
30         #desenha a nave na sua posição atual
31     def desenha(self):
32         self.tela.blit(self.imagem, self.retangulo)
33
34     #atualiza a posição da espaçonave de acordo com a flag de movimento
35     def atualiza(self, config):
36         if self.move_direita and self.retangulo.right < config.largura:
37             self.retangulo.centerx += self.velocidade
38         elif self.move_esquerda and self.retangulo.left > 0:
39             self.retangulo.centerx -= self.velocidade
40         elif self.move_baixo:
41             self.retangulo.centery += self.velocidade
42         elif self.move_cima:
43             self.retangulo.centery -= self.velocidade
```

ship.py

```

8 import sys, pygame
9
10 #responde a eventos de pressionamento de teclas e mouse
11 def testa_eventos(nave):
12     for evento in pygame.event.get(): #verifica todos os eventos ocorridos
13         if evento.type == pygame.QUIT: #verifica se a janela foi fechada
14             sys.exit()
15         elif evento.type == pygame.KEYDOWN: #quando aperta uma tecla
16             testa_eventos_aperta_tecla(evento, nave)
17         elif evento.type == pygame.KEYUP: #quando solta a tecla
18             testa_eventos_solta_tecla(evento, nave)
19
20 #atualiza as imagens na tela e redesenha a mesma
21 def atualiza_tela(config, tela, nave):
22     tela.fill(config.cor_fundo) #preenche a janela com a cor de fundo
23     nave.desenha() #desenha a nave na tela
24     pygame.display.flip() #redesenha a tela
25
26 def testa_eventos_aperta_tecla(evento, nave):
27     if evento.key == pygame.K_RIGHT: # Move a espaçonave para a direita
28         nave.move_direita = True
29     elif evento.key == pygame.K_LEFT: # Move a espaçonave para a esquerda
30         nave.move_esquerda = True
31     elif evento.key == pygame.K_UP: # Move a espaçonave para cima
32         nave.move_cima = True
33     elif evento.key == pygame.K_DOWN: # Move a espaçonave para baixo
34         nave.move_baixo = True
35
36 def testa_eventos_solta_tecla(evento, nave):
37     if evento.key == pygame.K_RIGHT:
38         nave.move_direita = False
39     elif evento.key == pygame.K_LEFT:
40         nave.move_esquerda = False
41     elif evento.key == pygame.K_UP:
42         nave.move_cima = False
43     elif evento.key == pygame.K_DOWN:
44         nave.move_baixo = False

```

Para parar de
se mexer
quando
soltar a tecla

game_functions.py

Recapitulando

- Na próxima aula aprenderemos como fazemos nossa nave atirar (bullet.py).
- Até o momento, temos:
 - alien_invasion.py
 - settings.py
 - game_functions.py
 - ship.py

Tarefas

1. Crie um jogo que contenha o fundo da tela azul
2. Encontre uma imagem de um personagem que você goste e coloque no centro da tela (use imagens pequenas, ex. 50 x 87).
3. Faça com que o personagem se mova por toda a tela e que não extrapole a tela.
4. Escolha duas teclas e faça com que quando o usuário aperte uma tecla, aumente a velocidade da nave e que com outra, diminua.