

Project Description:

- Name: AI RISK
- My term project will be the game RISK where a user competes with the computer. It will have an interactive user interface where the user can decide where to deploy their troops, whether to attack, and whether to maneuver troops. The computer will calculate the best strategy based on the current state of the map.

Structural Plan:

- This project will use object orientation to simplify and better categorize the complex, moving pieces. For instance, I have written a class for the territories so I can store information (like who is currently in that region, how many troops are on, etc.) as attributes of the object. I have also created multiple files in my project to break up my code into smaller and more manageable pieces. For instance I have a drawProject.py file that does all of the drawing components but does not contain the definitions of classes or instantiations of objects.

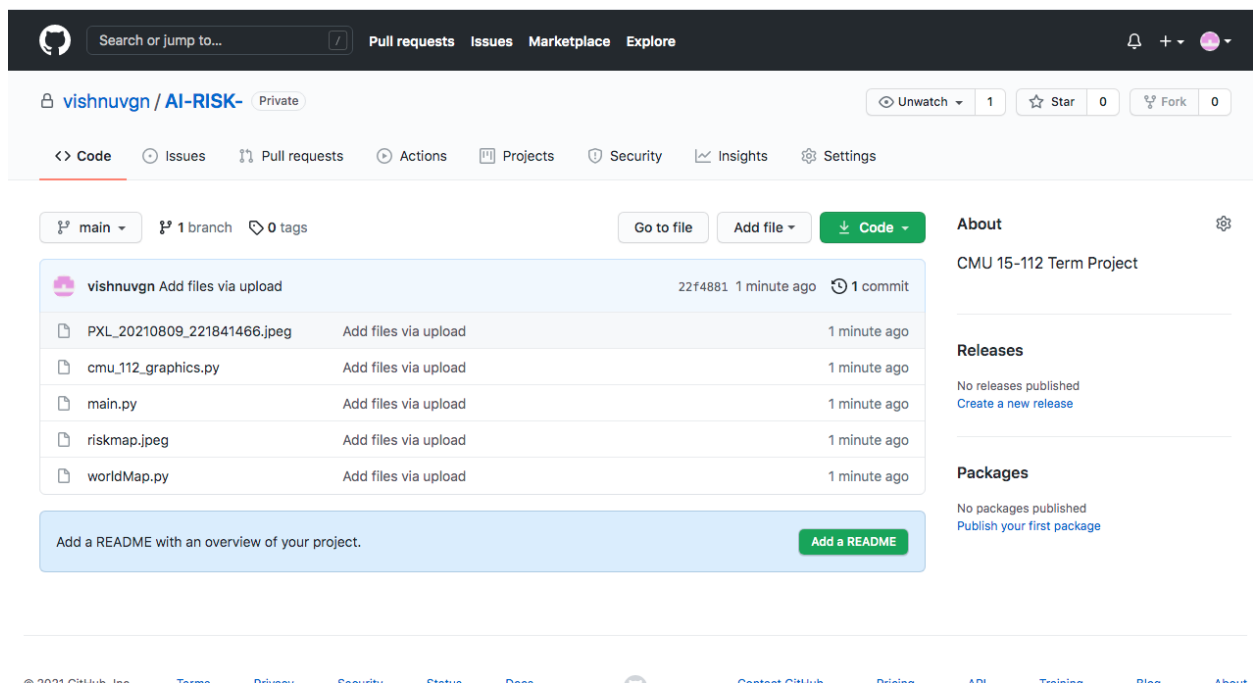
Algorithmic Plan:

- The decision making of the computer is the trickiest part of the project. The decision making algorithm should scan the current state of the map and take into account many pieces of information such as where enemy troops are, how many of them there are, and if any territories that the computer controls are weak. It should then decide where to place the troops that it gets at the beginning of its turn, decide whether to attack or not (if so, where), and, finally, whether to maneuver its troops or not. I will write this algorithm as if I were the one making the decision. So, I will try to mathematically rate how strong or weak a position is taking in the number of troops the enemy has and the number of troops

the computer has. If it is not enough to withstand an attack, then the computer should add troops in that area. Similarly, I write the algorithm so that the computer can pinpoint areas of weakness in its enemies territories and decide how to attack.

Version Control:

For version control, I am using GitHub and uploading my files in the cloud after I am done coding (when I stop for the time being and close VSCode).



main - AI-RISK- / main.py

Go to file



vishnuvgn Add files via upload

Latest commit 22f4881 2 minutes ago History

1 contributor

83 lines (60 sloc) | 2.49 KB

Raw

Blame



```
1 #####
2 # Term Project: RISK
3 #
4 # Your name: Vishnu Venugopal
5 # Your andrew id: vvenugop
6 #
7 #####
8
9 # import cs112_n21_week4_linter
10 import math, copy, random
11
12 import worldMap
13
14 from cmu_112_graphics import *
15
16 #####
17 # Helper functions
18 #####
19
20 def almostEqual(d1, d2, epsilon=10**-7):
21     # note: use math.isclose() outside 15-112 with Python version 3.5 or later
22     return (abs(d2 - d1) < epsilon)
23
24 import decimal
25 def roundHalfUp(d):
26     # Round to nearest with ties going away from zero.
27     rounding = decimal.ROUND_HALF_UP
28     # See other rounding options here:
29     # https://docs.python.org/3/library/decimal.html#rounding-modes
30     return int(decimal.Decimal(d).to_integral_value(rounding=rounding))
31
32
33 # riskmap => https://www.ultraboardgames.com/risk/continents.php
34
35
36 class Player(object):
37     def __init__(self):
```

```

34
35
36 class Player(object):
37     def __init__(self):
38         self.cardCount = 0 # how many territory cards they have at a certain time
39         self.cards = [] # what the cards are - list of Card objects
40         self.territories = [] # list of territories under Player control
41         self.troopPlaceCount = None # how many troops Player gets to place on the map at the start of their turn
42         self.controlContinent = False
43
44         # methods for relieving (placing) troops, attacking, defending, maneuvering
45
46 class Card(object):
47     def __init__(self, territory, icon): # EX: c1 = Card(Alaska, Cavalry)
48         self.territory = territory
49         self.icon = icon
50
51 def appStarted(app):
52     app.map = app.loadImage('riskmap.jpeg')
53
54     # code to get size of image
55     # https://newbedev.com/python-get-width-and-height-of-image-tkinter-code-example
56     img = Image.open('riskmap.jpeg')
57     mapPic = ImageTk.PhotoImage(img)
58     app.mapHeight = mapPic.height()
59     app.mapWidth = mapPic.width()
60
61 def gameDimensions():
62     pass
63
64 def redrawAll(app, canvas):
65     drawMap(app, canvas)
66     drawSideBar(app, canvas)
67
68 def drawMap(app, canvas):
69     canvas.create_image(app.mapWidth/2, app.mapHeight/2, image=ImageTk.PhotoImage(app.map))
70
71 def drawSideBar(app, canvas):
72     canvas.create_rectangle(app.mapWidth + 10, 10, app.width - 10, app.mapHeight, outline="black", fill="black")
73     canvas.create_text((app.width - 10 + (app.mapWidth + 10) )/2,
74                        (app.mapHeight - 10)/2, text="hello", fill="white")
75
76 def main():
77     runApp(width=1280, height=755)
78
79 if __name__ == '__main__':
80     main()
81
82 """

```

TP2 Update:

- I did not make any design changes.