

Matrix adivinhation

Sival Leão de Jesus

Departamento de Ciências Exatas – Universidade Estadual de Feira de Santana (UEFS)
Av. Transnordestina, s/n, Novo Horizonte, 44036-900, Feira de Santana, BA, Brasil

Abstract. *This report will present the procedures made for the creation of the software responsible for the game "Somas Esquecidas" at the request of Neo, where the Python programming language was used for the realization of the software, made in Microsoft's Windows operating system. The software developed has as main objective to be able to play a board game with two players.*

Resumo. *Neste relatório será apresentado os procedimentos feitos para a criação do software responsável pelo jogo "Somas esquecidas" a pedido de Neo, onde foi utilizado a linguagem de programação Python para a realização do software, feito no sistema operacional Windows da Microsoft. O software desenvolvido tem como principal objetivo ser capaz de jogar com dois jogadores um jogo de tabuleiro.*

1. Introdução

Ao ressuscitar e descobrir que vivia em uma simulação, Neo sai em busca dos seus companheiros. Porém, Smith, seu inimigo, acaba o capturando e o prendendo em um labirinto de enigmas onde ele tem que resolver todos para escapar e assim poder ir atrás de seus companheiros. No limite da sua capacidade, Neo consegue resolver quase todos os enigmas, onde resta apenas um e o mais difícil de todos, a construção do Jogo das Somas Esquecidas. sabendo que não é capaz de conseguir sozinho, Neo sai em busca de novos aliados que possam lhe ajudar nessa missão extremamente difícil e complicada, e acaba pedindo ajuda aos alunos de Engenharia de Computação da UEFS.

Neo oferece duas pílula aos alunos, onde a vermelha faria com que eles acordarem da simulação e uma azul para continuarem nela, depois de ingerir a pílula vermelha os alunos Engenharia de Computação da UEFS tem como objetivo principal ajudar Neo na criação de um software na linguagem de programação Python, capaz de executar o jogo somas esquecidas, jogo de tabuleiro jogado por dois jogadores, que tem como propósito adivinhar a soma de uma linha ou coluna e ganha quem tiver o maior número de acertos.

Diante disto, este relatório tem como finalidade esclarecer e explicar como foi desenvolvido o software capaz de se jogar somas esquecida e deste modo libertar Neo do labirinto que impede de buscar por seus companheiros

2. Metodologia

Como o uso de matrizes e funções era algo novo para os membros e importante para o funcionamento do código, já que as matrizes é usadas para a montagem do tabuleiro do

jogo e as funções evitar repetições desnecessárias de trechos de código, foi se discutido inicialmente nas sessões tutoriais como usar essas estruturas e manipulá-las no código e em seguida como armazenar números pseudo aleatórios sem repetição dentro da matriz.

Ao iniciar o processo de criação do foi devolvidos inicialmente os menus do jogo, onde o usuário pode escolher com quantos tabuleiro jogar, com o limite de dois tabuleiros simultaneamente, o nível de dificuldade do jogo, onde no nível fácil o tabuleiro apresenta dimensões 3x3 com números no intervalo de 1 a 30, no médio com dimensões 4x4 com números no intervalo de 1 a 60 e no difícil com dimensões 5x5 com números no intervalo de 1 a 100 e por último como o jogador deseja que a partida acabe, sendo por número de rodadas, sendo possíveis só número ímpares de rodadas ou até que o tabuleiro seja completados.

Em seguida foi se feito a parte principal do código para que o jogo ocorra, foi definido todas as matrizes necessárias e em seguida foi adicionados os números pseudo-aleatórios, números gerados pelo "import random" porém como os números não poderia se repetir, antes de colocá-los na matriz foi feito uma condição utilizado o comando "if" que significa "se" ou seja feito uma checagem para adicionar o número, onde se houver um número na matriz semelhante ao número gerado, esse número não entrava e um novo era gerado. Esse processo ocorria até que a matriz estivesse completa, e para descobrir isso, todo os comandos descritos acima foi feito dentro de um "while" que significa enquanto, junto com uma variável contadora que só era contada se um número novo entrasse na matriz.

Depois de gerada a matriz com todos os números, foi feito o processo de soma de cada coluna e cada linha é adicionado os resultados em uma outra matriz para que assim fizesse a checagem com o chute do jogador.

Com tudo definido, foram feitas as interações e checagem de valores. Como o software desejado tinha como um dos requisitos de revelar um casa do tabuleiro ao jogador que mais se aproxima da soma, tanto para mais como para menos, foi feito a subtração do valor absoluto entre o resultado da soma da linha ou coluna desejada pelo usuário com o valor do seu chute e em seguida comparado com o outro jogador, depois de passar pelo mesmo processo, quem tiver o menor número ganhar a rodada e terá uma casa no tabuleiro revelada, segundo duas condições, se o número chutado for maior que a soma o número maior da linha ou coluna que será mostrado também será o maior e se for menor que a soma o número menor que será revelado, caso o número chutado for exatamente o valor da soma, toda a coluna ou linha é revelada. E para fazer todo esse processo de mostrar o número maior ou menor, inicialmente foi criado duas matrizes cópia da matriz de números aleatórios em seguida foi se utilizando o dois comando "for", comando de repetição, para percorrer a coluna ou linha da matriz e procurar o maior e o menor número e depois de encontrar o número ele é adicionado a uma variável e se essa variável fosse um número diferente de um determinado número que nenhum tabuleiro aleatório geraria, exemplo para verificar o maior, o valor posto a essa

variável inicialmente é de -1 (menos um) assim qualquer número gerado será maior que ele e substituído pelo novo, número maior que ele, até ele possuir o valor do maior, ou seja se essa variável possuir um valor diferente de -1, esse valor seria adicionado à uma matriz vazia responsável pelo tabuleiro e pelo que os jogadores vê, e depois esse número é substituído das duas matrizes copias, por um número que não possa ser gerado pelo software de acordo com os níveis para que ele não volte a ser o maior na próxima rodada, esse processo ocorre a cada rodada e os pontos são atribuídos assim, como cada casa revelada vale um ponto, o jogador só irar receber se a variável for diferente de menos um, no caso deste exemplo, pois o número só é diferente se houver números a serem revelados na linha ou coluna desejada, evitando assim o acúmulo de pontos desnecessário e injusto. Quando uma linha ou coluna está completa o valor da soma será mostrado ao jogador e quando ela está completa ganha o jogador com mais pontos ou seja que revelou o maior número de casas.

3. Resultados, discussão e erros

O processo de desenvolver deste código foi bem complicado, já que utilizavam conceitos desconhecidos aos membros das sessões tutoriais e ao decorrer da produção surgiram várias dúvidas, principalmente com o uso de matrizes e como usá las para chegar no objetivo principal que é o funcionamento do jogo, mas ao decorrer do tempo e das experiências obtidas durante a construção do software foi se aprendendo e usando da melhor forma possível de cada membro, visando uma boa experiência para o jogador.

3.1 dados de entrada e saída.

Ao iniciar o software os jogadores irão se deparar com os menus do jogo onde apenas valores numéricos são aceitos, valores de 1 até a quantidade de opções que o menu apresenta. Exemplo, se houver apenas 3 opções será de 1 a 3.

Em seguida o jogo começa e será a vez do jogador 1 onde primeiramente ele deve escolher a linha ou coluna digitando apenas L ou C, onde L é usado para linha e C para coluna, com o número correspondente na frente. Exemplo C3 que é coluna 3, tanto letras maiúsculas como minúsculas são aceitas, em seguida a opção para digitar o número que ele achar ser a soma aparecerá, onde ele deve digitar valores numéricos, logo após será a vez do jogador 2 e o processo descrito para o jogador 1 se repete.

E como saída temos os números revelados no tabuleiro. Todo esse procedimento feito pelo jogador 1 e 2 se repete até que o tabuleiro complete.

3.2 metas não alcançadas e erros.

Infelizmente muitos dos requisitos solicitados por Neo não foram cumpridos, como jogar com dois tabuleiros simultaneamente, um para cada jogador, e como eles desejam que a partida acabe, se por número de rodadas ou completar a tabela e a escolha do nível do jogo fácil, médio e difícil. Infelizmente os jogadores só podem jogar no modo fácil e

até completar o tabuleiro, também não foi feita nenhuma condição de empate caso os dois chutem o mesmo valor na mesma linha ou coluna. Se os valores que eles chutarem resultar no valor absoluto da subtração for igual, ambos não pontuam e nenhuma casa é revelada. O código também possui alguns "bugs" e falhas, se um jogador acertar a soma exata da linha ou coluna todas as casas dela são reveladas e ele ganhará 3 pontos. Porém se ele repetir o procedimento ele ganhará mais 3 pontos, outra falha presente no código é no finalização do jogo, assim que o tabuleiro estiver completo será mostrado o ganhador e em seguida ele retorna ao tabuleiro completo ao invés do menu inicial, fazendo com que os jogadores tenham que fechar o terminal e iniciar novamente o software para uma nova partida.

3.3 Import

Para a realização deste software foi utilizado 3 "Import", código com uma função específica já pronta feito por outra pessoas, para gerar os números pseudo aleatórios do código foi se utilizando o import random e a função randint, também foi usado o import os, "os" sigla para sistema operacional, responsável por limpar o terminal a cada rodada das partidas e o import time e a função sleep, para gera um atraso na execução do código.

4. Conclusão

Infelizmente não foram cumpridos todos os objetivos propostos, onde o jogador consegue apenas jogar o modo fácil do jogo até que o tabuleiro seja totalmente completo, faltando várias opções para o jogador como o nível médio, difícil, escolha de como encerrar o jogo e a possibilidade de jogar com dois tabuleiros.

5. Referências

Import os - código fonte:

<https://github.com/python/cpython/tree/3.10/Lib/os.py>, Acesso em: 05 de maio de 2022

Import time - código fonte:

<https://github.com/python/cpython/blob/main/Modules/timemodule.c>, Acesso em: 05 de maio de 2022

Import random - código fonte:

<https://github.com/python/cpython/tree/3.10/Lib/random.py>, Acesso em: 05 de maio de 2022