



INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA
CAMPUS CAMPINA GRANDE
CURSO TÉCNICO EM INFORMÁTICA INTEGRADO

JÔNATAS TAVARES DOS SANTOS
LUCAS SILVA NASCIMENTO
MARIA GABRIELA PEREIRA DA SILVA

PROJETO FINAL DO 4º BIMESTRE

PROJETO: JOGO DO LABIRINTO

CAMPINA GRANDE – PB
2018

JÔNATAS TAVARES DOS SANTOS
LUCAS SILVA NASCIMENTO
MARIA GABRIELA PEREIRA DA SILVA

PROJETO FINAL DO 4º BIMESTRE

PROJETO: JOGO DO LABIRINTO

Projeto apresentado ao Curso Técnico em Informática Integrado do Instituto Federal de Educação, Ciência e Tecnologia da Paraíba – Campus Campina Grande, como requisito parcial para obtenção de nota na 4ª etapa, para aprovação na disciplina de Algoritmos e Lógica de Programação.

Orientadores: Profª. Elaine Cristina Juvino de Araújo e Prof. Marcos Vinícius Cantidiano Marques de Andrade.

CAMPINA GRANDE – PB

2018

1. INTRODUÇÃO

O Jogo do Labirinto é um jogo que consiste em tentar encontrar um caminho certo para chegar à saída, partindo de um determinado ponto, em meio a um emaranhado de passagens e caminhos confusos e embaralhados de tal modo que se torna difícil encontrar a sua respectiva saída.

Para fazer nosso projeto final do 4º bimestre na disciplina de Algoritmos e Lógica de Programação, optamos por fazer o jogo de duas formas:

- **Modo Tradicional:** Nesse modo, como já conhecemos, o usuário pode movimentar-se dentro do labirinto de um dos 3 níveis de dificuldade para onde desejar por meio da inserção das **teclas** "W", "A", "S" e "D" de seu teclado como entrada;
- **Modo Quiz:** Neste para se chegar à saída por um caminho já predefinido, o usuário deve responder corretamente a uma série de perguntas de múltipla escolha sobre conhecimentos gerais, divididas por 3 níveis de dificuldade.

Jogo do Labirinto no Modo Quiz iniciado no nível médio:

```

C:\Users\ZMAX\AppData\Local\Programs\Python\Launcher\py.exe
#####
#
#                                JOGO DO LABIRINTO - MODO QUIZ
#
#                                Aguardando resposta...
#
#####
#
#                                +-----+ +-----+ +-----+
#                                *           !           !
#                                +---+ +-----+ +-----+ +---+
#                                !       !           !       !
#                                +---+ +-----+ +-----+ +---+
#                                !           !           !
#                                +-----+ +-----+ +-----+ +---+
#                                !       !           !       !
#                                +---+ +---+ +-----+ +---+ +-----+
#                                !           !           !
#                                +-----+ +-----+ +-----+
#                                !
#                                +-----+
#
#####
#
#                                Atualmente, quantos elementos químicos possuem na tabela periódica?
#                                a) 113
#                                b) 118
#                                c) 108
#                                d) 109
#
#####                                Resposta #####

```

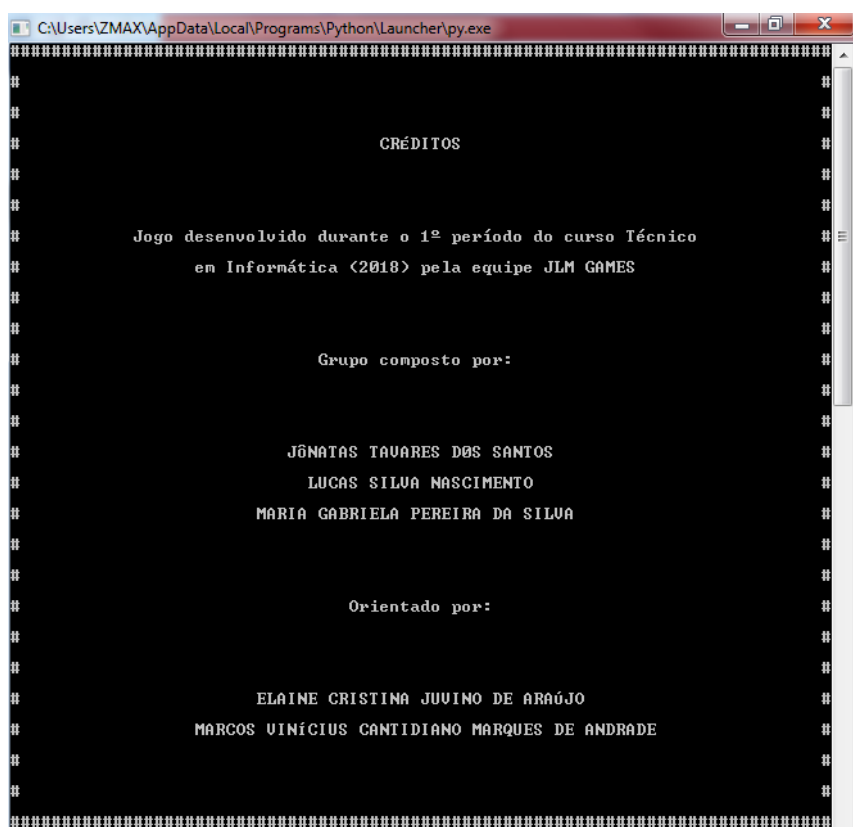
Tela de instruções:

```

C:\Users\ZMAX\AppData\Local\Programs\Python\Launcher\py.exe
#####
#
#                                COMO JOGAR:
#
#
#                                MODO TRADICIONAL:
#                                !
#                                !
#                                !   Você poderá escolher um, dos três níveis de dificuldade do
#                                !   jogo com diferentes labirintos, sendo eles: FÁCIL, MÉDIO e
#                                !   DIFÍCIL. E utilizando as teclas "W", "A", "S" e "D" do seu
#                                !   teclado você poderá mover-se livremente pelo labirinto até
#                                !   encontrar a respectiva saída do labirinto escolhido.
#                                !
#                                !
#                                MODO QUIZ:
#                                !
#                                !
#                                !   Você poderá escolher um, dos três níveis de dificuldade do
#                                !   jogo com diferentes labirintos, sendo eles: FÁCIL, MÉDIO e
#                                !   DIFÍCIL. Para avançar no labirinto em direção a saída pelo
#                                !   caminho correto, deverá responder corretamente a perguntas
#                                !   de múltipla escolha, em caso de erro voltará pra a posição
#                                !   anterior.
#                                !
#                                !   PRESSIONE "ENTER" PARA VOLTAR AO MENU
#
#####

```

Tela de Créditos:



No modo Tradicional o usuário utiliza as teclas “W”, “A”, “S” e “D” de seu teclado para se mover para cima, esquerda, baixo e direita respectivamente, movendo-se para onde desejar, se esse movimento for possível, claro. No modo quiz o usuário dará como entrada as teclas “A”, “B”, “C” e “D” de seu teclado que corresponde a sua respectiva alternativa, caso a resposta esteja correta o usuário avança para o próximo ponto, caso contrário retrocede ao anterior (este último caso ocorre apenas se o usuário tiver se movimentado pelo menos uma vez). Ao chegar à saída o usuário poderá decidir se deseja jogar novamente ou encerrar o programa.

2.1. ANÁLISE

As partes mais importantes de nosso programa são as nossas funções, elas se encontram em nosso arquivo em formato .py (funcoes.py) secundário. O nosso arquivo principal basicamente chamará as funções necessárias para sua execução.

Antes de tudo precisamos descobrir o sistema operacional da máquina em que o programa está sendo executado para saber quais comandos devem ser utilizados para poder limpar a tela sempre que necessário. O módulo “platform()” cumpriu bem essa função, e ao invés de darmos diretamente o comando

“D”): Primeiro será feito um teste para saber se o movimento é possível, se for o asterisco avança.

```
elif (movimento=='d'):
    if (labirinto[0][estado+2]=='|' or labirinto[0][estado+2]=='-' or labirinto[0][estado+2]=='+' or labirinto[0][estado+2]=='.'):
        situacao='|' # Não é possível realizar esse movimento.
    Tente novamente... # Atribui esse valor a variável para ser printada.
    else:
        labirinto[0][estado]=' ' # Remove o asterisco da posição onde estava adicionando um espaço vazio em seu lugar.
        labirinto[0][estado+2]='*' # O coloca na "posição atual mais 2" espaços a frente (2 porque usamos 2 espaços por questões de
```

No modo Quiz de nosso jogo, optamos por escrever os labirintos em string, com “.” representando os lugares onde o “*” deve parar, e “~” para marcar o caminho onde o asterisco passa até chegar ao ponto. Nós usamos o método replace() para substituir os caracteres, visto que não é possível alterá-los por índice sendo eles uma string, e imprimi-los em uma variável secundária onde são realizadas mudanças apenas com objetivo estético em um intervalo de tempo, também é feito o “salvamento” do movimento na matriz “voltar” para ser impresso em caso de erro.

```
voltar.append([]) #Adiciona uma lista vazia a lista ja existente.
for i in range(labirinto[:labirinto.find('.')].count('~')): #Enquanto houverem "*" e "." há possibilidade de
    voltar[-1].append(labirinto) #Adiciona o movimento a última lista.
    labirinto=labirinto.replace('*', ' ') #Substitui o asterisco por um espço vazio.
    labirinto=labirinto.replace('~','*',1) #Substitui o próximo passo por asterisco uma vez.
    imprimir=labirinto.replace('.', ' ') #substitui todos os pontos por espaços vazios para imprimir.
    print('#####')
#
#                               JOGO DO LABIRINTO - MODO QUIZ
# Resposta correta!
# Siga em frente.
#
#
#
#
#
#
#
#                               JOGO DO LABIRINTO - MODO QUIZ
# Resposta correta!
# Siga em frente.
#
#
#
#
#
#
#
#                               Pressione "ENTER" para prosseguir
# prox=input() #Aguarda algo ser digitado para mudar de tela.
os.system(limpar) #Limpa a tela.
```

Nós também usamos o método randint() para sortear as perguntas e evitar repetimentos de perguntas.

```
|sorteio=random.randint(1,len(perguntas))#Sorteia a pergunta.
sorteio=str(sorteio) #Converte o número inteiro sorteado em string pois é o tipo de chave do dicionário.
if len(sorteados)==len(perguntas):
    sorteados=[] #Se todas as perguntas ja forem feitas será zerado e poderão se repetir.
while sorteio in sorteados:
    sorteio=random.randint(1,len(perguntas))#Sorteia a pergunta.
    sorteio=str(sorteio) #Converte o número inteiro sorteado em string pois é o tipo de chave do dicionário.
    sorteados.append(sorteio) #Adiciona o número da pergunta sorteada a lista
```

Função para encerrar o programa que é executada se o usuário escolher a opção 5 na tela inicial, ou a opção 2 no fim da execução do jogo.

```
def encerrar (limpar): #Procedimento para encerrar o programa.
    os.system(limpar) #Limpa a tela.
    print(tela_de_creditos()) #Printa a tela de créditos.
    time.sleep(5) #Aguarda 5 segundos para prosseguir.
    telacreditos=tela_de_creditos() #Recebe a função tela de créditos como parâmetro.
    telacreditos=list(telacreditos) #Transforma cada caractere de "telacreditos" em uma elemento de uma lista.
    x=0 #Atribui um valor inicial a x.
    for i in range (telacreditos.count('\n')+1): #Loop pela quantidade de linhas que existe na tela de crédito.
        os.system(limpar) #Limpa a tela.
        for i in range (len(telacreditos[:telacreditos.index('\n')])): #Loop pela quantidade de caracteres que existe em uma linha da tela de crédito.
            telacreditos[x]=' ' #Substitui cada um dos caracteres de cada linha por um espaço vazio
            x+=1 #Adiciona +1 a x.
        x+=1 #Adiciona mais um a x para poder pular de linha.
        print(''.join(telacreditos)) #Printa a tela de créditos (agora com uma linha em branco) juntamos todos os elementos dessa lista.
        time.sleep(0.3) #Aguarda 0.3 segundos para prosseguir.
    sys.exit() #Encerra o programa e fecha a janela.
```

Função para evitar entradas inválidas:

```
def entrada (opcao,opcoes,tela,limpar): #Função para o ncdsoo da entrada ser inválida.
    while ((opcao in opcoes)==False): #Faz se o teste.
        os.system(limpar) #Limpa a tela.
        print(tela,end='') #Printa a referida tela.
        opcao=input('''##### Opção inválida #####
''' ) #Lê o novo valor.
        opcao=opcao.strip() #Remove espaços vazios antes e depois do texto.
        opcao=opcao.lower() #Converte toda string em letra minúscula.
        os.system(limpar) #Limpa a tela.
    return(opcao) #Retorna o novo valor da opção.
```

Conseguimos executar o que desejávamos respeitando o cronograma proposto na etapa 1, conseguindo até realizar algumas tarefas antes do prazo estabelecido.

AÇÕES	CRONOGRAMA PROPOSTO		CRONOGRAMA CUMPRIDO	
	INÍCIO	FIM	INÍCIO	FIM
Planejamento	12/11	17/11	12/11	17/11
Etapa 1 - Apresentação das ideias	19/11	19/11	19/11	19/11
Elaboração dos 3 labirintos	19/11	21/11	19/11	21/11
Elaboração das perguntas	19/11	22/11	19/11	21/11
Criação da tela inicial	20/11	23/11	20/11	23/11
Criação da tela de créditos	22/11	24/11	22/11	24/11
Etapa 2 – Metade das ideias em prática	26/11	26/11	26/11	26/11
Criação das funções	28/11	30/11	24/11	28/11
Etapa 3 – Análise do programa	03/12	03/12	03/12	03/12
Preparação pra apresentação	05/12	10/12	03/12	-
Etapa Projeto Final – Apresentação final	10/12	10/12	-	-

3. CONCLUSÃO

Este projeto foi de fundamental importância para aplicação do que foi aprendido na linguagem de programação Python3, na disciplina de Algoritmos e Lógica de Programação, durante o 1º período do Curso Técnico em Informática Integrado.

Pudemos aproveitar da companhia de um ao outro dentro de nossa equipe para compartilhar aprendizado e lógicas próprias de cada um, juntando tudo isso em um único e harmonioso programa de jogo clássico. Conseguimos executar o que tínhamos planejado na etapa 1, tivemos que abrir mão de algumas coisas, mas nós também acrescentamos coisas novas, que adicionaram algo a mais em nosso projeto.

Vivenciar experiências práticas desse tipo, em que é estimulado o raciocínio lógico e trabalho em equipe, é fundamental para a formação de um profissional na área de TI.