



deti

universidade de aveiro
departamento de eletrónica,
telecomunicações e informática

DIGDUG



GABRIEL MONTEIRO 107987

LUÍS SOUSA 108583

RICARDO PINTO 108598

FUNÇÕES DESENVOLVIDAS

PARA O DESENVOLVIMENTO OTIMIZADO DESTE PROJETO FORAM CRIADAS DIVERSAS FUNÇÕES DE MODO A ORGANIZAR O CÓDIGO:

- .ORDER_DIRECTIONS_PRIORITIES() - DEFINE A QUALIDADE DE UMA DIREÇÃO
- .ORDER_SAFE_DIRECTIONS() - ORGANIZA UMA LISTA DE UMA DIREÇÃO POR CUSTO
- .GET_DIRECTIONS_COSTS() - DEFINE OS CUSTOS DE UMA DIREÇÃO
- .CALCULATE_DISTANCES() - CALCULA AS DISTÂNCIAS DO DIGDUG A UM INIMIGO
- .GETROCKS() - CALCULA AS ROCHAS A UMA DISTÂNCIA DE 1 DO DIGDUG
- .DEF_LIMITS() - FUNÇÃO AUXILIAR PARA DEFINIR DIREÇÕES IMPOSSÍVEIS
- .DEF_MAP_LIMITS() - DEFINE DIREÇÕES IMPOSSÍVEIS CAUSADAS PELO MAPA
- .DEF_ROCK_LIMITS() - DEFINE DIREÇÕES IMPOSSÍVEIS CAUSADAS POR ROCHAS
- .DEF_FIRE_LIMITS() - DEFINE DIREÇÕES SUÍCIDAS CAUSADAS POR FOGO
- .CHOOSE_DIRECTION() - ESCOLHA DE UMA DIREÇÃO PARA JOGAR CONFORME QUALIDADE
- .GET_AIM() - DEFINE A MIRA E O OPOSTO ENTRE O DIGDUG E O INIMIGO

STUDENT

No ficheiro `student.py` é efetuado o envio do comando ideal do digdug para o servidor. Assim o projeto foi desenvolvido agrupando tudo de maneira a funcionar sem qualquer problema.

Ao iniciar o jogo, o state do game vai ser extraído e analisado, de forma ao digdug saber a posição de cada inimigo e saber reagir a cada um usando as condições extraídas, como a distância.

Sabendo a posição de cada elemento do mapa o digdug consegue reagir a tudo o que se estiver a passar, conseguindo evitar, matar e fugir de uma maneira rápida e eficaz.

CUSTOS DE JOGADA E QUALIDADE DE DIREÇÃO

OS CUSTOS DE CADA JOGADA PROVÊM DE UMA FÓRMULA DEPENDENTE DA DISTÂNCIA NORMALIZADA, QUE INDICA O PERIGO RELATIVO DE CADA EIXO.

ASSIM O DIGDUG VAI EVITAR SITUAÇÕES DE ALTO RISCO QUE PODERIAM LEVAR À SUA MORTE, ESPECIALMENTE EM NÍVEIS ELEVADOS. QUANDO O CUSTO ATINGE UM CERTO "THRESHOLD" IRÁ AFASTAR-SE, TENTANDO ATRAIR O INIMIGO PARA UMA ÁREA MAIS SEGURA QUE O PERMITIRÁ MATAR DE UMA MANEIRA QUE REDUZ BASTANTE A CHANCE DE PERDA DE VIDAS.

A JOGADA DE FUGA SERÁ CONSIDERADA "IMPORTANTE" NO DICIONÁRIO DE QUALIDADE, QUE VAI DOS VALORES 1 A 23, UTILIZANDO A ESCALA DE FIBONACCI. DESTE DICIONÁRIO VAI SER EXTRAÍDO A JOGADA DE MAIOR QUALIDADE E ESTA SERÁ EFETUADA (SE HOUVER DUAS JOGADAS DE MESMA QUALIDADE SERÁ UTILIZADA A PRIMEIRA INSERIDA).

PROÍBIDA (1) -> TERRÍVEL (2) -> PIOR (3) -> MÁ (5) -> POSSÍVEL (8) -> BOA (13) -> IMPORTANTE (21)

COMO PODEMOS VER NA FOTO EM BAIXO O DIGDUG RECEBER UM DICIONÁRIO DE CUSTOS DE CADA DIREÇÃO E UM DICIONÁRIO DE QUALIDADE DE CADA JOGADA, RESPECTIVAMENTE. ASSIM SABERÁ O MELHOR MOVIMENTO A EXECUTAR. NESTE CASO NÃO FOI ATINGIDO O THRESHOLD.

```
{ 'w': 0.25000000000000006, 'd': 0, 's': 0, 'a': 1.7500000000000002 }  
{ 'w': [8, 0], 'd': [8, 0], 's': [8, 0], 'a': [13, 9], 'A': [8, 0] }  
{ 'cmd': 'key', 'key': 'a' }
```


DEBUGGING

PARA VERIFICAR QUE O CÓDIGO CORRIA COMO PRETENDIDO FIZEMOS DEBUGGING DO CÓDIGO PARA VERIFICAR SE CERTAS CONDIÇÕES ESTAVAM A SER FEITAS COM SUCESSO. ISTO AJUDOU EM DIVERSOS CASOS PRINCIPALMENTE NAS FUNÇÕES QUE NOS PERMITEM DESVIAR DE PEDRAS E EVITAR O FOGO DO FYGAR. SEMPRE QUE A FUNÇÃO ESTAVA A SER EXECUTADA RECEBIAMOS UM AVISO NO TERMINAL. SE O COMPORTAMENTO FOSSE O ESPERADO PODIAMOS DAR A FUNÇÃO COMO BEM IMPLEMENTADA CASO CONTRÁRIO ENTENDERIAMOS O QUE ESTAVA A GERAR CONFLITOS DE MANEIRA A CONSERTAR.