

# Documentação – Trabalho Prático

## **“Pac Fantasy”**

Universidade Federal de Minas Gerais

Programação e Desenvolvimento de Software I - Sistemas de Informação

Professor Pedro O. S. Vaz de Melo

08/2021

Lucas Roberto Santos Avelar

### **1.Introdução**

O Trabalho Prático da disciplina consiste na utilização dos conceitos aprendidos durante as aulas, com auxílio da biblioteca Allegro, para a construção de um jogo baseado nos antigos “Final Fantasy’s”.

“Pac Fantasy” mistura conceitos oriundos do jogo-base com elementos do clássico Pac Man. Ao iniciá-lo, o jogador será direcionado para o Menu Principal do jogo e deve escolher entre “Jogar” e “Instruções”, utilizando as setas cima/baixo ou as teclas W/S do teclado para navegar dentre as opções e apertando Enter para confirmar a opção desejada. “Instruções” apresenta alguns detalhes e informações sobre o jogo e como operá-lo, enquanto “Jogar” redireciona o usuário para o jogo em si. Uma vez dentro do jogo, o objetivo como Pac Man é alcançar o fantasma que está vulnerável do outro lado da tela. Para isso, o personagem pode ser movimentado com as teclas WASD ou as setas do teclado. No canto superior esquerdo, ficam presentes os pontos de vida e a pontuação do personagem.

Escondidos no cenário, encontram-se monstros e amigas. Ao chegar perto demais de um monstro, Pac Man entrará em batalha com ele. Serão exibidos, acima de cada um, seus respectivos pontos de vida. Na batalha, o personagem e o adversário alternam-se em turnos: caso o Pac Man ataque (seja um ataque normal ou um especial), o monstro revidará também atacando-o. O duelo continua até que: 1) o monstro morra, e nesse caso o personagem volta a navegar pelo mapa e obtém os pontos do monstro, que some do cenário; 2) o personagem morra, e nesse caso o jogo acaba com derrota; 3) o herói decida fugir e, caso tenha êxito, retorna para a navegação no mapa e o monstro continua ativo para batalha caso ocorra uma nova aproximação. O usuário pode navegar dentre as opções de batalha com as teclas WS ou as setas cima/baixo do teclado e confirma a opção ao pressionar Enter.

Os monstros possuem 4 níveis: Easy, Médio, Hard e Boss, em ordem crescente de poder. Quanto maior o poder, maior a força de ataque e a vida total do monstro, assim como maior é a pontuação obtida ao derrotá-lo. Quanto mais perto do objetivo, maiores as chances de encontrar inimigos com nível maior. No total, são 9 inimigos Easy, 5 Médios, 3 Hard e 1 Boss.

Já as amigas conferem um bônus de vida ao jogador. Aproximando-se de uma, será exibida uma tela demonstrando o encontro entre a amiga e o personagem e informando qual o valor desse bônus.

Após a prova oral, foram implementados 2 cheats ao jogo: no modo de Navegação, ao pressionar a tecla R, todos os monstros e amigas têm a posição revelada no mapa. Já no modo de Batalha, após pressionar a tecla I, todo ataque mata qualquer monstro instantaneamente. Esses cheats permanecem ativos até que a respectiva tecla seja pressionada novamente.

O jogador vence quando consegue que todo o personagem alcance o fantasma vulnerável. Nesse caso, são exibidas a tela de vitória e a pontuação final do jogador (caso a pontuação seja maior que o recorde, uma mensagem de aviso também será exibida).

## 2. Descrição do Código

A tabela abaixo apresenta um resumo do que está presente em determinados trechos do código:

<b>Linhas</b>	<b>Descrição</b>
1 ~ 30	Includes e DEFINES presentes durante o código do jogo
33 ~ 43	Declaração de variáveis globais
45 ~ 51	Função utilizada quando é necessária a geração de números aleatórios entre x e y
53 ~ 190	Criação das diversas estruturas presentes no jogo, dentre elas: a do Personagem principal, as de cada nível de monstro e a das amigas, possuindo no geral variáveis como posição x e y no mapa e pontos de vida. O herói apresenta mais variáveis, destinadas à controle do número de especiais e dos turnos da batalha, por exemplo. Também existem estruturas para cada ataque (ataque normal, ataque especial, ataque cheat e ataque de cada monstro) que apresentam posição x e y no cenário de batalha, dano, cor, entre outras variáveis. O círculo de escolha de uma opção também é uma estrutura aqui definida.
192 ~ 373	Funções de inicialização para cada uma das estruturas acima descritas, atribuindo valores às variáveis. O herói é inicializado com 100 de vida, os monstros Easy, Médio, Hard e Boss com 30, 45, 75 e 120, respectivamente,

	entre outros. Os ataques também são inicializados, com o ataque normal do personagem recebendo 10 de dano, o ataque cheat recebendo 200 e os outros ataques (incluindo o especial) recebendo valores aleatórios de acordo com a força do personagem/inimigo. Finalmente, as opções são inicializadas como um círculo de cor branca.
375	Início das funções de navegação no mapa
378 ~ 383	Função que desenha o personagem no mapa
385 ~ 399	Função que desenha o cenário de navegação, colorindo a tela de cinza e escrevendo informações do personagem no canto superior esquerdo (HP e Pontuação)
401 ~ 453	Função que movimenta o personagem na tela de acordo com a tecla pressionada: W/seta cima movimenta o personagem para cima, S/seta baixo para baixo e assim por diante
455 ~ 463	Função que verifica se o personagem chegou ao objetivo, retornando 1 em caso positivo
466	Início das funções de batalha
468 ~ 485	Função que desenha o cenário no caso do encontro entre o personagem e uma amiga, informando o jogador sobre esse encontro e sobre o valor do bônus por ele recebido
487 ~ 505	Função que desenha o cenário no caso do encontro entre o personagem e um monstro, exibindo a vida do personagem e desenhando a caixa de opções no canto inferior direito
507 ~ 523	Função que calcula as chances de fuga do herói, baseado no valor atual de seus pontos de vida
526 ~ 575	Função que permite a escolha de uma opção pelo herói e determina o que foi escolhido por ele baseado na posição do círculo de opções
577 ~ 624	Função que verifica a opção escolhida pelo personagem e libera a execução dessa ação no programa principal, alterando as variáveis necessárias
626 ~ 630	Função que desenha o círculo de opções
632 ~ 636	Função que calcula a distância euclidiana entre dois pontos
638 ~ 676	Funções que utilizam do cálculo da distância euclidiana para verificar se o personagem se aproximou de um monstro ou amiga
678	Início das funções de menu e final de jogo
679 ~ 713	Função que permite a interação com as opções presentes no menu: “Jogar” e “Instruções”.
715 ~ 724	Função que verifica a opção escolhida pelo usuário no Menu Inicial e libera a execução dessa ação no programa principal
727 ~ 731	Função que colore a tela em caso de vitória

733 ~ 740	Função que verifica se o personagem morreu
742 ~ 749	Função que desenha a tela final em caso de derrota (morte do personagem)
751	Início do programa principal
753 ~ 823	Rotinas padrão de inicialização da biblioteca Allegro, que criam a tela do jogo, criam e inicializam o timer do jogo, permitem a adição de imagens, instalam o teclado e permitem sua utilização dentro do jogo, criam e registram eventos na fila, entre outras funções iniciais
825 ~ 854	Criação das variáveis de imagem e atribuição de imagens presentes na pasta do jogo à essas variáveis. A imagem de cada monstro é carregada, assim como a imagem da amiga, do menu, das instruções e do personagem
856 ~ 857	Inicialização da variável que define o início do jogo e da variável que decide o primeiro modo de jogo como sendo o Menu Inicial
860 ~ 914	Criação de todos os personagens do jogo com seus respectivos ataques. É criado um personagem (baseado na estrutura do personagem) e ele é inicializado utilizando a função já descrita de inicialização. O mesmo se repete para cada nível de monstro (assim como para seus ataques), para as amigas e para as opções. Vale ressaltar que é criado um vetor de x posições para cada nível de monstro, e esse x depende do número de monstros de cada nível presente. Por exemplo, o vetor de Monstros Easy possui 9 posições. Para as amigas também é criado um vetor.
916	Início do while que se repete enquanto o jogo está ativo
922	Verifica se é um evento de tempo
924 ~ 936	Se o modo de jogo for Menu, desenha a tela do Menu e o círculo de opções do Menu
938 ~ 946	Se o modo de jogo for Instruções, desenha a tela de Instruções e o círculo de opções das Instruções
948	Verifica se o modo de jogo é Navegação
953 ~ 954	Desenha o cenário de navegação e o objetivo
956 ~ 971	Inicialmente oculto, esse trecho revela as posições dos monstros e das amigas no mapa. Pode ser ativado pressionando a tecla R enquanto no modo de Navegação
973 ~ 985	Verifica se o personagem mudou de direção e, em caso positivo, redesenha-o na nova posição. Por exemplo, se o herói estava indo para cima e foi para a direita, esse trecho desenha o personagem virado para a direita
989 ~ 1027	Trecho com vários for's que verificam se o personagem se aproximou de um monstro ou amiga e, em caso positivo, qual monstro/amiga foi
1029	Verifica a função que retorna se o personagem alcançou o objetivo
1031 ~ 1045	Abre o arquivo que contém o recorde obtido e verifica se a pontuação atual é maior do que esse recorde. Em caso

	positivo, edita o arquivo com o novo recorde e libera a exibição da mensagem de aviso para novo recorde
1047 ~ 1061	Chama a função do cenário final e exibe as mensagens finais de congratulações e de exibição do recorde (e de aviso de novo recorde, caso o antigo tenha sido batido). Finaliza o while e acaba o jogo
1065	Chama os devidos procedimentos quando o modo de jogo é Batalha
1068 ~1073	Se o herói não tiver encontrado uma amiga, chama a função de desenho do cenário de batalha com conflito, desenha o círculo de opções e chama a função para processar a ação do personagem
1075	Desenha o Pac Man em posição de batalha no cenário
1077 ~ 1088	Se o herói tiver encontrado uma amiga, desenha o cenário de encontro com amiga, desenha o Pac Man e a amiga, congela a tela por 3 segundos e depois retira a amiga do mapa, adiciona o bônus à vida do personagem e ele retorna para o mapa
1090 ~ 1431	Trecho que possui tamanho elevado mais pela repetição de elementos do que pela complexidade. Primeiro, verifica-se qual o nível do inimigo encontrado, desenhando no cenário suas respectivas imagem e pontos de vida. Depois, verifica-se a ação do personagem: se for um ataque, é desenhado um círculo de raio 10 que se movimenta até o inimigo e ao chegar nele, subtrai da sua vida o dano (mesmo valor do raio). No caso do ataque especial, o mesmo ocorre, com a diferença de que o raio e o dano são valores aleatórios. Após o inimigo receber o ataque, o teclado do jogador é travado (não pode escolher entre as opções) e o adversário realiza um ataque que funciona da mesma forma: é desenhado um círculo com raio igual ao dano e, quando esse círculo atinge o personagem, subtrai-se o dano da vida do herói e o teclado é destravado, continuando o ciclo. Ao final de cada ataque, sua respectiva função de inicialização é chamada para resetar o ataque à sua posição inicial original e atribuir um novo valor de dano. Esse procedimento explicitado foi escrito uma vez para cada nível de monstro, o que explica o tamanho do trecho
1433 ~ 1436	Em caso de retorno positivo da função que indica a morte do personagem, chama a função de desenho do cenário de derrota e quebra o while
1452	Verifica se alguma tecla foi pressionada
1454 ~ 1455	Se o modo de jogo for Menu, chama a função de processamento de tecla no Menu
1459 ~ 1460	Se o modo de jogo for Instruções, chama a função de processamento de tecla nas Instruções
1462 ~ 1463	Se o modo de jogo for Navegação, chama a função de processamento de tecla na Navegação

1465 ~ 1466	Se o modo de jogo for Batalha, chama a função de processamento de tecla na Batalha
1478 ~ 1484	Processos de finalização; destruição da tela, do timer e da fila de eventos e congelamento da tela por 3 segundos para verificação da tela final (mensagem de vitória ou derrota)