**Daniel Dantas De Carvalho Bezerra - 32610785**

**Mateus Ieno Ramalho - 34374621**

**Miguel Arcanjo Barreto Siqueira - 34317414**

**Pedro Henrique Gomes Da Silva Morais - 34256059**

**Victor Chaves Moreira - 34148736**

**PROJETO DE TÉCNICAS E DESENVOLVIMENTO DE ALGORITMOS**

**JOGO DA FORCA**

**INTRODUÇÃO**

O projeto consiste em versão simplificada do tradicional Jogo da Forca, no qual uma pessoa deve escolher uma palavra em segredo, desenhar traços que representam cada letra da palavra, e os demais participantes devem adivinhar a palavra, chutando uma letra por vez. A cada erro, o desenho de um boneco (representando o jogador enforcado, como punição por ter perdido), vai de formando membro a membro. No caso projeto, por ser um software para ser rodado no terminal, cuja interface consiste apenas em linhas de comando por texto, não há desenho do boneco, apenas os traços que representam cada um a letra da palavra.

**RESULTADOS**

O código fonte foi feito na Linguagem C e faz uso das seguintes bibliotecas: stdio.h, stdlib.h, string.h, locale.h. logo de início são definidas as variáveis palavra (char, máximo de 50 caracteres), tentativas (char, 50 caracteres) e tentativasFeitas (int, valor 0). Os nomes das variáveis e das funções utilizadas no código fonte seguem o padrão Camel Case. A função principal (main), começa com função setlocale(0, “portuguese”), a qual permite que sejam impressas na tela caracteres especiais da língua portuguesa. Ao todo, são usadas 9 funções fora a função principal main: letraCorreta(char letra), tentativasErradas(), chute(char letra), derrota(), vitória(), adivinha(), desenha(), escolhePalavra(), bemVindo().

A segunda função em execução é bemVindo(), a qual não recebe parâmetros, apenas imprime na tela o cabeçalho do jogo e comando para que uma pessoa escolha a palavra a ser adivinhada (“palavra misteriosa”).

Em seguida é executada a função escolhePalavra(), a qual não recebe parâmetros. Essa função imprime o comando para que o usuário, nesse momento o jogador que vai escolher a palavra secreta, digite qual palavra deverá ser adivinhada pelos demais jogadores. Após a entrada da palavra, utilizando a função scanf(), são puladas várias linhas para esconder a palavras secreta dos demais jogadores na tela do terminal.

Uma vez definida a palavra secreta, é feito um laço de repetição do-while, no qual são executadas as funções desenha() e adivinha(), enquanto é feito o seguinte teste lógico: (!victoria() && !derrota()), i. e., as funções desenha() e advinha() são executadas enquanto as funções vitória() e derrota() não retornaram valores booleanos True (verdadeiro), ou seja, enquanto os jogadores não vencem ou perdem o jogo, o programa seguirá apresentando tentativas. A função desenha começa com uma variável tipo int que recebe como valor o retorno da função tentativasErradas(), a qual começa retornando valor 0 e coleta o número de erros dos jogadores.

A função desenha() executa um estrutura de repetição for que toma como parâmetros uma variável tipo int i que começa com valor 0 e servirá como contador para as repetições, e o retorno de strlen(palavra), que retorna o número de caracteres da palavra secreta. A estrutura for começa com uma estrutura condicional if-else, na qual é feito o teste o teste lógico (chute(palavra[i]). A função chute recebe como parâmetro a letra da palavra que acabou de ser digitada, definida por palavra[i]. É feita a analise se tentativa da vez está correta, retornando o valor 1 caso esteja, ou 0 caso não. Sendo o valor retornado 1 (True), será impressa as letras da palavra que já foram adivinhadas, caso 0 (False), será impresso o caráter “\_”, representando os o espaço da letra ainda não adivinhada.

A função adivinha(), executada logo em seguida a função desenha(), começa imprimindo o comando para que o jogador da vez adivinhe uma letra que será lida pelo scanf() e o valor da tentativa é atribuído a variável chute (char). É executada uma condicional if-else, no qual o teste lógico analisa se a letra atribuída a variável chute está na palavra secreta, em caso sim, é impressa a mensagem de que o jogador acertou a letra, caso contrário, é impressa a mensagem de que o jogador errou. Em seguida, um array chamado tentativas recebe a letra que está na variável chute, e uma variável chamada tentativasFeitas recebe um valor inteiro a mais (+ 1), na contagem.

Essa estrutura do-while permanecerá em execução até que as funções victoria() e derrota() retornem valores True. A função vitoria() retornará 1 (true) caso todas as letras da palavra secreta já tiverem sido adivinhadas e função derrota() e chamará a função tentativasErradas(), a qual, por sua vez, irá percorrer o array tentativasFeitas e contas quantas tentativas erradas aconteceram, retornando assim o valor. Caso o número de erros dos jogadores seja maior ou igual a 5, a função derrota() retornará 1 (True).

Encerrada a estrutura do-while, caso vitória retorne True, será impressa a mensagem de que o jogador ganhou, caso contrário, será impressa a mensagem de que o jogador perdeu, e qual era a palavra secreta. Por fim, é impressa a mensagem de que o jogo foi encerrado.

**APÊNDICE**

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <locale.h>

char palavra[50];

char tentativas[50];

int tentativasFeitas = 0;

//verifica se a letra pertence a palavra

int letraCorreta(char letra){

for(int i = 0; i < strlen(palavra); i++) {

if(letra == palavra[i]) {

return 1;

}

}

return 0;

}

//adiciona um erro, caso a letra digitada não pertença a palavra

int tentativasErradas() {

int erros = 0;

for(int i = 0; i < tentativasFeitas; i++) {

if(!letraCorreta(tentativas[i])) {

erros++;

}

}

return erros;

}

// verifica se a letra está correta.

int chute(char letra) {

int correto = 0;

for(int i = 0; i < tentativasFeitas; i++) {

if(tentativas[i] == letra) {

correto = 1;

break;

}

}

return correto;

}

//limite de erros para a derrota

int derrota(){

return tentativasErradas() >=5;

}

// verifica se todas as letras foram adivinhadas

int vitoria(){

for(int i = 0; i < strlen(palavra); i++) {

if(!chute(palavra[i])) {

return 0;

}

}

return 1;

}

// pede o input de uma tentativa e testa se acertou ou não

void adivinha() {

char chute;

printf("digite uma letra para completar a palavra: ");

scanf(" %c", &chute);

if(letraCorreta(chute)) {

printf("Você acertou uma letra %c\n\n", chute);

} else {

printf("\nVocê errou a letra %c\n\n", chute);

}

tentativas[tentativasFeitas] = chute;

tentativasFeitas++;

}

// desenha a palavra escondida e as vidas restantes. a palavra vai sendo revelada com as tentativas corretas

void desenha() {

int erros = tentativasErradas();

for(int i = 0; i < strlen(palavra); i++) {

if(chute(palavra[i])) {

printf("%c ", palavra[i]);

}else {

printf("\_ ");

}

}

printf("\nerros %d/5\n", erros);

}

// faz um jogador escolher uma palavra para ser descobetra

void escolhePalavra(){

printf("\nDigite a palavra (sem acentos, letra maiúscula e espaço):");

scanf("%[^\n]s", palavra);

printf("\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n########################################################################\n\n");

}

// imprime o inicio do jogo

void bemVindo(){

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* JOGO DA FORCA \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n\n");

printf("Escolha alguém para escolher uma palavra,\nessa será a palavra misteriosa.\n\n");

}

int main() {

setlocale(0, "portuguese");

char continua;

bemVindo();

//do{

escolhePalavra();

do {

desenha();

adivinha();

} while (!vitoria() && !derrota());

if(vitoria()) {

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");

printf("\nVocê Ganhou!!!\n\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

} else {

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("Você Perdeu :(\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("A palavra era: %s\n\n", palavra);

}

// printf("deseja jogar novamente? S/N: ");

// scanf(" %c", &continua);

//} while (continua == 'S' || continua == 's'); "jogar novamente" desabilitado pq apresentava bug.

printf("jogo encerrado");

}