

TÉCNICAS DE DESENVOLVIMENTO DE ALGORITMOS

(Jogo da Forca)

Versão do documento: 1.0

Equipe: Luiz Eduardo / Matrícula: 30200059

William Rafael / Matrícula:30565022

Vinicius Santana / Matrícula:30339235

EliabeRodrigues/Matrícula:30378362

LuanGonçalves/Matrícula:31076556

- Introdução

Jogo da forca desenvolvido em C.

- **Descrição do jogo**

O jogo da forca é um jogo em que o jogador tem que acertar qual é a palavra proposta, tendo como dica o número de letras ligado à palavra.

- **Regras**

A cada letra errada, é desenhado uma parte do corpo do enforcado, caso acerte o jogo segue sem desenhar uma parte do corpo do enforcado.

- Resultados

informações gerais!!

- Descrição geral do jogo com exemplificação dos códigos fontes

O jogo foi feito com a utilização de funções, estruturas de repetição, estrutura de condição e tratamentos.

Funções:

int letraexiste(char letra) {

```

    for(j = 0; j < strlen(palavrasecreta); j++) {
        if(letra == palavrasecreta[j]) {
            return 1;
        }
    }

```

```

    return 0;
}

```

Estruturas de repetição:

```

for( i = 0; i <= randomico; i++) {
    fscanf(f, "%s", palavrasecreta);
}

```

Estrutura de condições e tratamentos:

```

printf("Voce deseja adicionar uma nova palavra no jogo (S/N)?");
scanf(" %c", &quer);
quer = toupper(quer);
if(quer != 'N' && quer != 'S'){
    printf("#####\n");
    printf("#          -Opcao Invalida-          #\n");
    printf("#          -Digite uma opcao valida-    #\n");
}

```

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

- Dificuldades encontradas e soluções implementadas

Dificuldades:

No início do projeto estávamos com dificuldades de implementar as perguntas para selecionar S/N, quando o usuário digitava "n" minúsculo ou qualquer coisa diferente o código dava erro porém não exibia o erro em questão.

soluções:

A solução encontrada para resolver foi a utilização do método "**toupper**" que é geralmente usado para converter uma cadeia de caracteres em letras maiúsculas para que ela possa ser usada em uma comparação.

- Demonstrativo das funcionalidades implementadas

Digite o texto a partir daqui (colocar imagens no demonstrativo – de 2 a 4 imagens pelo menos) ...

- Funcionalidade do método toupper:

```
void adicionapalavra() {  
    char quer;  
    do{  
        printf("Voce deseja adicionar uma nova palavra no jogo (S/N)?");  
        scanf(" %c", &quer);  
        quer = toupper(quer);  
        if(quer != 'N' && quer != 'S'){  
            printf("#####\n");  
            printf("-Opcao Invalida-\n");  
            printf("#####\n");  
        }  
        if(quer == 'S') {  
            char novapalavra[TAMANHO_PALAVRA];  
  
            printf("Digite a nova palavra, em letras MAIUSCULAS: ");  
            scanf("%s", novapalavra);  
  
            FILE* f;  
  
            f = fopen("palavras.txt", "a+");  
            if(f == 0) {  
                printf("Banco de dados de palavras nao disponivel\n\n");  
                exit(1);  
            }  
        }  
    }  
}
```

```
1 - *jogar*
2 - *Adicionar Palavra*
3 - *Sair*
2
Voce deseja adicionar uma nova palavra no jogo (S/N)?s
Digite a nova palavra, em letras MAIUSCULAS: LEOLOLZERO
Voce deseja adicionar uma nova palavra no jogo (S/N)?n

Process returned 78 (0x4E)   execution time : 19.668 s
Press any key to continue.
```

```
1 - *jogar*
2 - *Adicionar Palavra*
3 - *Sair*
2
Voce deseja adicionar uma nova palavra no jogo (S/N)?z
#####
-Opcao Invalida-
#####
Voce deseja adicionar uma nova palavra no jogo (S/N)?t
#####
-Opcao Invalida-
#####
Voce deseja adicionar uma nova palavra no jogo (S/N)?n
```

- Apêndice:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include "forca.h"
#include <ctype.h>
```

```
char palavrasecreta[TAMANHO_PALAVRA];
char chutes[26];
int chutesdados = 0;
```

```
int letraexiste(char letra) {

    for(int j = 0; j < strlen(palavrasecreta); j++) {
        if(letra == palavrasecreta[j]) {
            return 1;
        }
    }

    return 0;
}
```

```
int chuteserrados() {
    int erros = 0;

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

        if(!letraexiste(chutes[i])) {
            erros++;
        }
    }
}
```



```

    }

    return erros;
}

int enforcou() {
    return chuteserrados() >= 5;
}

int ganhou() {
    for(int i = 0; i < strlen(palavrasecreta); i++) {
        if(!jachutou(palavrasecreta[i])) {
            return 0;
        }
    }

    return 1;
}

void chuta() {
    char chute;
    printf("Qual letra? ");
    scanf(" %c", &chute);
    chute = toupper(chute);
    if(letraexiste(chute)) {
        printf("\n\n\n\n\n\n\n\nVoce acertou: a palavra tem a letra %c\n",
chute);
    } else {
        printf("\n\n\n\n\n\n\n\n\nVoce errou: a palavra nao tem a letra
%c\n", chute);
    }
}

```

```
    chutes[chutesdados] = chute;
    chutesdados++;
}
```

```
int jachutou(char letra) {
    int achou = 0;
    for(int j = 0; j < chutesdados; j++) {
        if(chutes[j] == letra) {
            achou = 1;
            break;
        }
    }

    return achou;
}
```

```
void desenhaforca() {

    int erros = chuteserrados();

    printf(" ____ \n");
    printf(" |/   | \n");
    printf(" |   %c%c%c \n", (erros>=1?'(':' '), (erros>=1?'_':' '), (erros>=1?')':' '));
    printf(" |   %c%c%c \n", (erros>=3?'\\':' '), (erros>=2?'|':' '), (erros>=3?'/':' '
));

    printf(" |   %c \n", (erros>=2?'|':' '));
    printf(" |   %c %c \n", (erros>=4?'/':' '), (erros>=4?'\\':' '));
    printf(" |       \n");
    printf(" |__   \n");
    printf("\n\n");

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

```

        if(jachutou(palavrasecreta[i])) {
            printf("%c ", palavrasecreta[i]);
        } else {
            printf("_ ");
        }

    }

    printf("\n");

}

void escolhepalavra() {
    FILE* f;

    f = fopen("palavras.txt", "r");
    if(f == 0) {
        printf("#####\n");
        printf("# -Banco de dados de palavras nao disponivel- #\n");
        printf("# -Arquivo palavras.txt inexistente- #\n");
        printf("#####\n");
        exit(1);
    }

    int qtddepalavras;
    fscanf(f, "%d", &qtddepalavras);

    srand(time(0));
    int randomico = rand() % qtddepalavras;

    for(int i = 0; i <= randomico; i++) {
        fscanf(f, "%s", palavrasecreta);
    }
}

```

```

        fclose(f);
    }

void adicionapalavra() {
    char quer;
    do{
        printf("Voce deseja adicionar uma nova palavra no jogo (S/N)?");
        scanf(" %c", &quer);
        quer = toupper(quer);
        if(quer != 'N' && quer != 'S'){
            printf("#####\n");
            printf("#    -Opcao Invalida-    #\n");
            printf("#  -Digite uma opcao valida-  #\n");
            printf("#####\n");
        }
        if(quer == 'S') {
            char novapalavra[TAMANHO_PALAVRA];

            printf("Digite a nova palavra, em letras MAIUSCULAS: ");
            scanf("%s", novapalavra);

            FILE* f;

            f = fopen("palavras.txt", "a+");
            if(f == 0) {

                printf("#####\n");
                printf("# -Banco de dados de palavras nao disponivel-  #\n");
                printf("#    -Arquivo palavras.txt inexistente-    #\n");
            }
        }
    } while(quer != 'S' && quer != 'N');
}

```

```

printf("#####\n");
        exit(1);
    }

    int qtd;
    fscanf(f, "%d", &qtd);
    qtd++;
    fseek(f, 0, SEEK_SET);
    fprintf(f, "%d", qtd);

    fseek(f, 0, SEEK_END);
    fprintf(f, "\n%s", novapalavra);

    fclose(f);

    }
} while(quer != 'N');

}

int main() {
    int op = 0;
    char quer2;

    printf("\n##### MENU PRINCIPAL #####");
    printf("\n#");
    printf("\n# Digite um comando para prosseguir: #");
    printf("\n#");
    printf("\n# 1 - Jogar #");
    printf("\n# 2 - Adicionar Palavras #");
    printf("\n# 0 - Sair #");

```

```

printf("\n#                               #");

printf("\n#####\n");
scanf("%d", &op);
switch(op){
    case 1:{
        do {
            escolhepalavra();
            desenhaforca();
            chuta();
        } while (!ganhou() && !enforcou());

        if(ganhou()) {
            printf("\nParabens, voce acertou!\n\n");
            printf("A palavra sorteada foi: %s*\n\n", palavrasecreta);

            printf("  _____  \n");
            printf("  '.====='  \n");
            printf("  .-\\:    /-. \n");
            printf("  | |:..  | | \n");
            printf("  '-|:..  |-' \n");
            printf("  \\::.  /   \n");
            printf("  '::. .'   \n");
            printf("      )(    \n");
            printf("  .' '.    \n");
            printf("  '-----'  \n\n");

        } else {
            printf("\nPoxa, voce foi enforcado!\n");
            printf("A palavra era %s*\n\n", palavrasecreta);

            printf("  _____  \n");
            printf("  /          \\ \n");

```

```

printf(" /      \\  \n");
printf("//      \\\\ \n");
printf("\\\\| XXXX  XXXX | / \n");
printf(" | XXXX  XXXX | / \n");
printf(" | XXX   XXX  |  \n");
printf(" |           |  \n");
printf(" \\__   XXX   _/  \n");
printf("  \\   XXX   /|   \n");
printf("  ||       ||   \n");
printf(" | | | | | | |   \n");
printf(" | | | | | | |   \n");
printf("  \\         _/   \n");
printf("   \\       _/    \n");
printf("    \\__/_/      \n");

```

```

    }

```

```

break;

```

```

}

```

```

case 2:{

```

```

    adicionapalavra();

```

```

break;

```

```

}

```

```

}

```

https://www.teses.usp.br/teses/disponiveis/55/55134/tde-18052005-163302/publico/10-Apendice_novo.pdf