

ALUNO: Gabriel Henrique Correa Martins
DISCIPLINA: Algoritmos e Programação

Jogo Principal

```
// Define cores das letras e fundo.
HANDLE hConsole;
WORD corNormal;
CONSOLE_SCREEN_BUFFER_INFO consoleInfo;
// Define cores das letras e fundo.
int VIDA_TOTAL = 6;
char letraColetada;
void gotoxy(int X,int Y) { // Declaração do comando gotoxy.
    COORD c;
    c.X = X;
    c.Y = Y;
    SetConsoleCursorPosition(GetStdHandle(STD_OUTPUT_HANDLE),c);
}
void loopMais(char letra, int posicao) {
    int i;
    for(i=0; i<9; i++) {
        gotoxy(posicao,3+i);
        printf("%c", letra);
        Sleep(50);
        gotoxy(posicao,3+i);
        printf(" ");
    }
    gotoxy(posicao,12);
    printf("%c", letra);
}
void loopMenos(char letra, int posicao) {
    int i;
    for(i=0; i<9; i++) {
        gotoxy(posicao,20-i);
        printf("%c", letra);
        Sleep(30);
        gotoxy(posicao,20-i);
        printf(" ");
    }
    gotoxy(posicao,12);
    printf("%c", letra);
}
void bordaVitoria() {
    int i;
    for (i=0; i<43; i++) {
        gotoxy(25+i,10);
```

```
        printf("%c",205);
        Sleep(30);
        gotoxy(25+i,14);
        printf("%c",205);
    }
}

void desenhaVitoria(void) {
    loopMais(86,30); // V
    loopMenos(79,33); // O
    loopMais(67,36); // C
    loopMenos(69,39); // E
    loopMais(86,45); // V
    loopMenos(69,48); // E
    loopMais(78,51); // N
    loopMenos(67,54); // C
    loopMais(69,57); // E
    loopMenos(85,60); // U
    loopMais(33,63); // !
    bordaVitoria();
}

void desenhaDerrota(void) {
    loopMais(86,30); // V
    loopMenos(79,33); // O
    loopMais(67,36); // C
    loopMenos(69,39); // E
    loopMais(80,45); // P
    loopMenos(69,48); // E
    loopMais(82,51); // R
    loopMenos(68,54); // D
    loopMais(69,57); // E
    loopMenos(85,60); // U
    loopMais(33,63); // !
    bordaVitoria();
}

void lerStringSeguramente(char* palavra, int tamanho) {
    fflush(stdin);
    if(fgets(palavra, tamanho, stdin) != NULL) {
        /* Remove a nova linha (\n), caso ela tenha sido lida pelo fgets */
        int indiceUltimoCaractere = strlen(palavra) - 1;
        if(palavra[indiceUltimoCaractere] == '\n') {
            palavra[indiceUltimoCaractere] = '\0';
        }
        fflush(stdin);
    }
}

int contarLetras(char* palavra) {
```

```
int i, quantidadeLetras=0;
for(i=0; i<strlen(palavra); i++) {
    if(isalpha(palavra[i])) {
        quantidadeLetras++;
    }
}
return quantidadeLetras;
}

int ehLetraOuEspaco(char* palavra) {
    int i;
    for(i=0; i<strlen(palavra); i++) {
        if(palavra[i] != ' ' && !isalpha(palavra[i])) {
            return 0;
        }
    }
    return 1;
}

void bordaJogoPrincipal(void) {
    int contador;
    SetConsoleTextAttribute(hConsole, 0x08);
    for(contador = 2; contador < 23; contador++) { // Imprime as bordas laterais.
        gotoxy(16,contador);
        printf("%c",186);
        gotoxy(72,contador);
        printf("%c",186);
        gotoxy(72,1);
        printf("%c",187);
        gotoxy(16,1);
        printf("%c",201);
    }
    for(contador = 17; contador < 72; contador++) { // Imprime as bordas superiores e
inferiores.
        gotoxy(contador, 1);
        printf("%c",205);
        gotoxy(contador,23);
        printf("%c",205);
        gotoxy(72,23);
        printf("%c",188);
        gotoxy(16,23);
        printf("%c",200);
    }
    SetConsoleTextAttribute(hConsole, corNormal);
}

void imprimirCoracoes(int coracoes) {
    // Define cores das letras e fundo.
    GetConsoleScreenBufferInfo(hConsole, &consoleInfo);
```

```
corNormal = consoleInfo.wAttributes;
// Define cores das letras e fundo.
int i;
for(i=0; i<6*2; i+=2) {
    gotoxy(59 + i,3);
    printf(" ");
}
SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
for (i = 0; i < coracoes * 2; i+=2) {
    gotoxy(59 + i,3);
    printf("%c",3);
}
SetConsoleTextAttribute(hConsole, corNormal);
}

void vida(int vidasRestantes) {

    imprimirCoracoes(vidasRestantes);
    gotoxy(53,3);
    printf("VIDAS");
    switch (vidasRestantes) {
        case 5: {
            gotoxy(27,11);
            printf("%c",2);
            break;
        }
        case 4: {
            gotoxy(27,11);
            printf("%c",2);
            gotoxy(27,12);
            printf("%c",178);
            break;
        }
        case 3: {
            gotoxy(27,11);
            printf("%c",2);
            gotoxy(27,12);
            printf("%c",178);
            gotoxy(28,12);
            printf("%c",92);
            break;
        }
        case 2: {
            gotoxy(27,11);
            printf("%c",2);
            gotoxy(27,12);
```

```
        printf("%c",178);
        gotoxy(28,12);
        printf("%c",92);
        gotoxy(26,12);
        printf("%c",47);
        break;
    }
    case 1: {
        gotoxy(27,11);
        printf("%c",2);
        gotoxy(27,12);
        printf("%c",178);
        gotoxy(28,12);
        printf("%c",92);
        gotoxy(26,12);
        printf("%c",47);
        gotoxy(28,13);
        printf("%c",92);
        break;
    }
    case 0: {
        gotoxy(27,11);
        printf("%c",2);
        gotoxy(27,12);
        printf("%c",178);
        gotoxy(28,12);
        printf("%c",92);
        gotoxy(26,12);
        printf("%c",47);
        gotoxy(28,13);
        printf("%c",92);
        gotoxy(26,13);
        printf("%c",47);
        break;
    }
}

}

void desenhaForca(void) {
    int i;
    for (i=0; i<8; i++) {
        gotoxy(20 + i,18);
        printf("%c",205);
    }
    gotoxy(22,18);
    printf("%c",202);
    for (i=0; i<8; i++) {
        // Desenha a base da forca.
        // Desenha o centro da base.
        // Desenha corpo da forca.
```

```
        gotoxy(22,17-i);
        printf("%c",186);
    }
    gotoxy(22, 9);
    printf("%c",201);                // Desenha quina de cima.
    for (i=0; i<4; i++) {           // Desenha base de cima.
        gotoxy(23+i, 9);
        printf("%c",205);
    }
    gotoxy(27, 9);
    printf("%c",187);                // Desenha quina da forca.
    gotoxy(27, 10);
    printf("%c",179);                // Desenha o laço da forca.
}

void desenhaGrama(void) {
    int contador;
    for (contador = 0; contador < 55; contador++) {
        gotoxy(17+contador, 18);
        printf("%c",176);
        gotoxy(17+contador, 19);
        printf("%c",176);
        gotoxy(17+contador, 20);
        printf("%c",176);
        gotoxy(17+contador, 21);
        printf("%c",176);
        gotoxy(17+contador, 22);
        printf("%c",176);
    }
}

void desenhaLacuna(char* palavra) {
    int i;
    for (i = 0; i < strlen(palavra); i++) {
        gotoxy(29+i*2,16);
        printf("%c", palavra[i] == ' ' ? 255 : 95);
    }
}

void revelaLetraLacuna(char* palavra, char letraAcertada) {
    int i;
    for (i = 0; i < strlen(palavra); i++) {
        if (letraAcertada == palavra[i]) {
            gotoxy(29+i*2,16);
            printf("%c", letraAcertada);
        }
    }
}

int verificarAcertos(char* palavra, char* acertos) {
```

```
int i, j;
for (i=0; i<strlen(palavra); i++) {
    int encontrou = 0;
    for(j=0; j<strlen(acertos); j++) {
        if(palavra[i] == acertos[j]) {
            encontrou = 1;
        }
    }
    if(encontrou == 0) {
        return 0;
    }
}
return 1;
}

void paraMaiusculo(char* palavraParaMudar) {
    int i;
    for(i = 0; i < strlen(palavraParaMudar); i++) {
        palavraParaMudar[i] = toupper(palavraParaMudar[i]);
    }
}

void desenhaGrafico(char* palavra, int tamanhoPalavra, char* dica, char* nome) {
    bordaJogoPrincipal();
    imprimeCreditos();
    SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
    desenhaGrama();
    SetConsoleTextAttribute(hConsole, corNormal);
    desenhaForca();
    desenhaLacuna(palavra);
    paraMaiusculo(dica);
    gotoxy(40,6);
    printf("DICA:");
    gotoxy(46,6);
    printf("%s", dica);
    gotoxy(20,3);
    printf("%s", nome);
    imprimeCreditos();
    vida(VIDA_TOTAL);
}

int palavraTemLetra(char letra, char* palavra) {
    int i;
    for (i = 0 ; i < strlen(palavra); i++) {
        if (letra == palavra[i]) {
            return 1;
        }
    }
}

return 0;
```

```
}
void tentativas(char* palavra) {
    char historicoErros[6] = {0}, acertos[16] = {0};
    int quantidadeErros = 0, quantidadeAcertos = 0;
    do {
        int contaLetras = contarLetras(palavra);
        gotoxy(40,10);
        printf("Pressione uma letra:");
        gotoxy(61,10);
        letraColetada = getch();
        if (isalpha(letraColetada)) { // Verificar se o caracter é uma letra do alfabeto.
            gotoxy(40,12);
            printf("                ");
            letraColetada = toupper(letraColetada);
            gotoxy(61,10);
            printf("%c", letraColetada);
            //Verifica se a letra digitada ja foi errada.
            if(palavraTemLetra(letraColetada, historicoErros) == 1) { //Printa que a letra já
foi tentada
                gotoxy(40,12);
                printf("Esta letra j%c foi tentanda!", 160);
            } else if (palavraTemLetra(letraColetada, acertos) == 1) { //Printa que a letra já
foi tentada
                gotoxy(40,12);
                printf("Esta letra j%c foi tentanda!", 160);
            } else {
                //Verifica se a letra digitada é correta e revela a letra na lacuna.
                if(palavraTemLetra(letraColetada, palavra) == 1) {
                    acertos[quantidadeAcertos] = letraColetada;
                    quantidadeAcertos++;
                    gotoxy(40,12);
                    printf("                ");
                    SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
                    gotoxy(40,12);
                    printf("Voc%c acertou!", 136);
                    SetConsoleTextAttribute(hConsole, corNormal);
                    revelaLetraLacuna(palavra, letraColetada);
                } else {
                    //Salva letra errada e tira vida.
                    historicoErros[quantidadeErros] = letraColetada;
                    quantidadeErros++;
                    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
                    gotoxy(40,12);
                    printf("Voc%c errou!", 136);
                    gotoxy(39+ (quantidadeErros > 0 ? quantidadeErros*2 : 1), 14);
                    printf("%c", toupper(letraColetada));
                }
            }
        }
    } while (1);
}
```



```
        SetConsoleTextAttribute(hConsole, corNormal);
        VIDA_TOTAL -= 1;
        vida(VIDA_TOTAL);
    }
}
} else {
    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
    gotoxy(40,12);
    printf("Pressione somente letras!");
    SetConsoleTextAttribute(hConsole, corNormal);
}
} while(verificarAcertos(palavra, acertos) == 0 && VIDA_TOTAL > 0);
if(verificarAcertos(palavra, acertos) == 1) {
    gotoxy(40,12);
    printf("          ");
    SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
    desenhaVitoria();
    desenhaGrama();
    SetConsoleTextAttribute(hConsole, corNormal);
    desenhaForca();
    vida(VIDA_TOTAL);
} else if(!VIDA_TOTAL > 0) {
    gotoxy(40,12);
    printf("          ");
    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
    desenhaDerrota();
    SetConsoleTextAttribute(hConsole, corNormal);
    SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
    desenhaGrama();
    SetConsoleTextAttribute(hConsole, corNormal);
    desenhaForca();
    vida(VIDA_TOTAL);
}
}

void coletarPalavraSecreta(char* palavra) {
    int palavraSecretaValida;
    gotoxy(30,10);
    printf("Escreva uma palavra:");
    do {
        gotoxy(51,10);
        lerStringSeguramente(palavra, 18);
        if(ehLetraOuEspaco(palavra)) {
            if (contarLetras(palavra) > 15) {
                gotoxy(51,10);
                printf("          ");
                gotoxy(30,12);
            }
        }
    } while(palavraSecretaValida == 0);
}
```

```
        printf("                ");
        SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
        gotoxy(30,12);
        printf("Nao ultrapasse 15 letras!");
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 0;
    } else if(contarLetras(palavra) < 2) {
        gotoxy(51,10);
        printf("                ");
        gotoxy(30,12);
        printf("                ");
        SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
        gotoxy(30,12);
        printf("No m%cnimo duas letras!", 161);
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 0;
    } else {
        gotoxy(51,10);
        printf("                ");
        gotoxy(30,12);
        printf("                ");
        SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
        gotoxy(30,12);
        printf("Nome salvo com sucesso!");
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 1;
    }
} else {
    gotoxy(51,10);
    printf("                ");
    gotoxy(30,12);
    printf("                ");
    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
    gotoxy(30,12);
    printf("Digite somente letras!");
    SetConsoleTextAttribute(hConsole, corNormal);
    palavraSecretaValida = 0;
}
} while(!palavraSecretaValida);
paraMaiusculo(palavra);
}
char coletaDicaPalavraSecreta(char* dica) {
    int palavraSecretaValida;
    gotoxy(30,14);
    printf("Escreva uma dica:");
    do {
```

```
gotoxy(48,14);
lerStringSeguramente(dica, 12);
if(ehLetraOuEspaco(dica)) {
    if (contarLetras(dica) > 10) {
        gotoxy(48,14);
        printf("                ");
        gotoxy(30,16);
        printf("                ");
        SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
        gotoxy(30,16);
        printf("Nao ultrapasse 10 digitos!");
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 0;
    } else if(contarLetras(dica) < 2) {
        gotoxy(48,14);
        printf("                ");
        gotoxy(30,16);
        printf("                ");
        SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
        gotoxy(30,16);
        printf("No m%cnimo duas letras!", 161);
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 0;
    } else {
        gotoxy(48,14);
        printf("                ");
        gotoxy(30,16);
        printf("                ");
        SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
        gotoxy(30,16);
        printf("Nome salvo com sucesso!");
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 1;
    }
} else {
    gotoxy(48,14);
    printf("                ");
    gotoxy(30,16);
    printf("                ");
    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
    gotoxy(30,16);
    printf("Digite somente letras!");
    SetConsoleTextAttribute(hConsole, corNormal);
    palavraSecretaValida = 0;
}
} while(!palavraSecretaValida);
```

```
        paraMaiusculo(dica);
    }
    char coletaNomeJogador(char* nome) {
        int palavraSecretaValida, debug;
        gotoxy(30,6);
        printf("Digite o seu nome:");
        do {
            gotoxy(49,6);
            lerStringSeguramente(nome, 13);
            if(ehLetraOuEspaco(nome)) {
                if (contarLetras(nome) > 10) {
                    gotoxy(49,6);
                    printf("                ");
                    gotoxy(30,8);
                    printf("                ");
                    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
                    gotoxy(30,8);
                    printf("Nao ultrapasse 10 digitos!");
                    SetConsoleTextAttribute(hConsole, corNormal);
                    palavraSecretaValida = 0;
                } else if(contarLetras(nome) < 2) {
                    gotoxy(49,6);
                    printf("                ");
                    gotoxy(30,8);
                    printf("                ");
                    SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
                    gotoxy(30,8);
                    printf("No m%cnimo duas letras!", 161);
                    SetConsoleTextAttribute(hConsole, corNormal);
                    palavraSecretaValida = 0;
                } else {
                    gotoxy(49,6);
                    printf("                ");
                    gotoxy(30,8);
                    printf("                ");
                    SetConsoleTextAttribute(hConsole, FOREGROUND_GREEN);
                    gotoxy(30,8);
                    printf("Nome salvo com sucesso!");
                    SetConsoleTextAttribute(hConsole, corNormal);
                    palavraSecretaValida = 1;
                }
            } else {
                gotoxy(49,6);
                printf("                ");
                gotoxy(30,8);
                printf("                ");
            }
        } while (palavraSecretaValida == 0);
    }
```

```
        SetConsoleTextAttribute(hConsole, FOREGROUND_RED);
        gotoxy(30,8);
        printf("Digite somente letras!");
        SetConsoleTextAttribute(hConsole, corNormal);
        palavraSecretaValida = 0;
    }
} while(!palavraSecretaValida);
paraMaiusculo(nome);
}

void menuSelecionavel(void) {
    int TECLA_SETA_CIMA = 72, TECLA_SETA_BAIXO = 80, TECLA_ENTER = 13;
    int opcao = 0, teclaPressionada, i;
    int posicaoDaLinha = 7, posicaoDaColuna = 7, posicaoDaSeta = 7;
    SetConsoleTextAttribute(hConsole, 0x05);
    gotoxy(32,3);
    printf("M E N U   D O   J O G O S");
    gotoxy(33,7);
    printf("JOGAR CONTRA MAQUINA");
    gotoxy(33,10);
    printf("JOGAR CONTRA AMIGO");
    gotoxy(33,13);
    printf("VOLTAR");
    SetConsoleTextAttribute(hConsole, corNormal);
    imprimeCreditos();
    do {
        opcao = 0;
        gotoxy(28,posicaoDaLinha);
        printf("  -%c",16);
        // Desenha uma seta para selecionar as opções do menu.
        gotoxy(1,22);
        if(kbhit) {
            // Kbhit pega a tecla que o usuário pressionar e retorna o seu
            // valor em código (ASCII).
            teclaPressionada = getch();
        }
        if(teclaPressionada == TECLA_SETA_BAIXO) {
            posicaoDaColuna = posicaoDaLinha;
            posicaoDaLinha += 3 ;
            posicaoDaSeta += 3;
            if(posicaoDaSeta > 13) {
                // Se posição da seta para baixo ultrapassar "Sair" (posição 16),
                // retorna para a posição 7 (Jogar).
                posicaoDaSeta = 7;
                posicaoDaLinha = 7;
            }
        }
    }
}
```

```
if(teclaPressionada == TECLA_SETA_CIMA) {
    posicaoDaColuna = posicaoDaLinha;
    posicaoDaLinha -= 3;
    posicaoDaSeta -= 3;
    if(posicaoDaSeta < 7) {
        // Se a posição da seta para cima ultrapassar "Jogar" (posição
7), retorna para a posição 16 (Sair).
        posicaoDaSeta = 13;
        posicaoDaLinha = 13;
    }
}
if(posicaoDaLinha != posicaoDaColuna) {
    // Limpa as setas (quando movimentadas).
    gotoxy(27,posicaoDaColuna);
    printf("  ");
    posicaoDaColuna = posicaoDaLinha;
}
if(teclaPressionada == TECLA_ENTER) {
    opcao = posicaoDaSeta;
}
} while(opcao == 0);
switch (opcao) {
    case 7: {
        char nome[11];
        limpaTela();
        bordaJogoPrincipal();
        imprimeCreditos();
        coletaNomeJogador(nome);
        break;
    }
    case 10: {
        char nome[11], palavra[16], dica[16];
        int tamanhoPalavra;
        limpaTela();
        imprimeCreditos();
        bordaJogoPrincipal();
        gotoxy(35,3);
        printf("B E M - V I N D O");
        coletaNomeJogador(nome);
        coletarPalavraSecreta(palavra);
        coletaDicaPalavraSecreta(dica);
        limpaTela();
        tamanhoPalavra = strlen(palavra);
        desenhaGrafico(palavra, tamanhoPalavra, dica, nome);
        tentativas(palavra);
        break;
    }
}
```

```
        }
        case 13: {
            menuPrincipal();
            break;
        }
    }
    while(opcao != 3);
}
void menuDoJogo(void) {
    system("cls");
    bordaJogoPrincipal();
    menuSelecionavel();
    imprimeCreditos();
}
void jogoPrincipal(void) {
    // Define cores das letras e fundo.
    hConsole = GetStdHandle(STD_OUTPUT_HANDLE);
    GetConsoleScreenBufferInfo(hConsole, &consoleInfo);
    corNormal = consoleInfo.wAttributes;
    // Define cores das letras e fundo.
    //Menu do Jogo Principal
    menuDoJogo();
    //Menu do Jogo Principal
}
```