

```
1) #include <stdio.h>
```

```
#include <stdlib.h>
```

```
#include <windows.h>
```

```
int main(){
```

```
    char operacao;
```

```
    float numero01, numero02, resultado;
```

```
    while (1){
```

```
        system("cls");
```

```
        printf("Escolha uma operacao: \n+ para adicao\n- para subtracao\n* para multiplicacao\n/ para divisao\nPara encerrar o programa, digite 0\nSua escolha: ");
```

```
        scanf("%c", &operacao);
```

```
        if(operacao == 0){return 0;}
```

```
        printf("Digite o primeiro numero: ");
```

```
        scanf("%f", &numero01);
```

```
        printf("Digite o segundo numero: ");
```

```
        scanf("%f", &numero02);
```

```
        switch(operacao){
```

```
            case '+':
```

```
                resultado = numero01 + numero02;
```

```
                break;
```

```
            case '-':
```

```
                resultado = numero01 - numero02;
```

```
                break;
```

```

        case '*':
            resultado = numero01 * numero02;
            break;
        case '/':
            resultado = numero01 / numero02;
            break;
        default:
            printf("Digite uma operacao valida!");
            break;
    }

    printf("O resultado e igual a: %.2f", resultado);

    Sleep(5000);
}
}

```

---

2)

```

#include <stdio.h>
#include <ctype.h>
#include <string.h>

```

```

void isValidInput(float real, float fatorConversao){
    int i, contador = 0;
    char str[20], digito_char;

    snprintf(str, sizeof(str), "%.2f", real);

    for (i = 0; str[i] != '\0'; i++){
        digito_char = str[i];

        if (digito_char == '.'){

```

```

        contador++;

        if(contador > 1){
            printf("Voce digitou varios '.");
            return;
        }

        continue;
    } else if(!isdigit(digito_char)){
        printf("Digite numero valido\n");
    }
}

```

```

float euro = real / fatorConversao;

```

```

    printf("Deu %.2f", euro);
}

```

```

int main(){
    float real, fatorConversao = 5.94573;
    printf("Digite o valor em reais para converter para euro: ");
    scanf("%f", &real);

    isValidInput(real, fatorConversao);

    return 0;
}

```

---

3)

```

#include <stdio.h>

```

```

int main(){
    char frase[50];

    FILE *arquivo = fopen("arquivo.txt", "a+");

```

```
    if (arquivo == NULL){
        perror("Erro ao abrir o arquivo");
        return 1;
    }

    printf("Digite a frase que sera gravada no arquivo: ");
    fgets(frase, sizeof(frase), stdin);

    fprintf(arquivo, "%s", frase);

    fclose(arquivo);

    return 0;
}
```

---

```
4) #include <stdio.h>
```

```
#include <string.h>
```

```
int main(){
    char nome[30], profissao[30], biografia[200];
    int idade;

    FILE *arquivo = fopen("index.html", "w");
    if(arquivo == NULL){
        perror("Erro ao abrir o arquivo");
        return 1;
    }

    printf("Digite seu nome: ");
    fflush(stdin);
    fgets(nome, sizeof(nome), stdin);
```

```

nome[strcspn(nome, "\n")] = 0;

printf("Digite sua profissao: ");
fflush(stdin);
fgets(profissao, sizeof(profissao), stdin);
profissao[strcspn(profissao, "\n")] = 0;

printf("Digite sua idade: ");
scanf("%d", &idade);

printf("Digite uma breve biografia sobre voce:\n");
fflush(stdin);
fgets(biografia, sizeof(biografia), stdin);
biografia[strcspn(biografia, "\n")] = 0;

fprintf(arquivo,
"<html>\n<head>\n<title>%s</title>\n</head>\n<body>\n  <p>%s</p>\n  <p>%s</p>\n  <p>%d</p>\n  <p>%s</p>\n</body>\n</html>", nome, nome, profissao, idade, biografia);

fclose(arquivo);

return 0;
}

```

---

5)

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <windows.h>
#include <ctype.h>

```

```
#define MAX_TENTATIVAS 6
```

```
#define MAX_PALAVRA 30
```

```
const char UNDERLINE = '_';
```

```
int i;
```

```
void adicionarPalavra(FILE *arquivo) {
```

```
    char palavra[MAX_PALAVRA];
```

```
    printf("Digite a palavra a ser adicionada: ");
```

```
    fflush(stdin);
```

```
    fgets(palavra, sizeof(palavra), stdin);
```

```
    // Remove newline character, if present
```

```
    size_t len = strlen(palavra);
```

```
    if (len > 0 && palavra[len-1] == '\n') {
```

```
        palavra[len-1] = '\0';
```

```
    }
```

```
    fprintf(arquivo, "%s\n", palavra);
```

```
    fclose(arquivo);
```

```
}
```

```
void exibirPalavra(char palavra[], int acertos[], int tamanho) {
```

```
    for (i = 0; i < tamanho; i++) {
```

```
        if (acertos[i]) {
```

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

```
        } else {
```

```
            printf("%c ", UNDERLINE);
```

```
        }
```

```
    }
```

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

```
int verificarLetra(char letra, char palavra[], int tamanho, int acertos[]) {  
    int acerto = 0;  
    for (i = 0; i < tamanho; i++) {  
        if (toupper(letra) == toupper(palavra[i]) && !acertos[i]) {  
            acertos[i] = 1;  
            acerto = 1;  
        }  
    }  
    return acerto;  
}
```

```
void jogar(FILE *arquivo) {  
    char palavra[MAX_PALAVRA];  
    char linha[MAX_PALAVRA];  
    int quantidadeLinhas = 0;  
    srand(time(NULL));  
  
    rewind(arquivo);  
    while (fgets(linha, sizeof(linha), arquivo) != NULL) {  
        quantidadeLinhas++;  
    }
```

```
    int linhaAleatoria = rand() % quantidadeLinhas;
```

```
    rewind(arquivo);  
    for (i = 0; i <= linhaAleatoria; i++) {  
        fgets(palavra, sizeof(palavra), arquivo);  
    }
```

```
// Remove newline character, if present
size_t len = strlen(palavra);
if (len > 0 && palavra[len-1] == '\n') {
    palavra[len-1] = '\0';
}

int tamanho = strlen(palavra);
int acertos[tamanho];
for (i = 0; i < tamanho; i++) {
    acertos[i] = 0;
}

int tentativas = 0;
char letra;
while (tentativas < MAX_TENTATIVAS) {
    exibirPalavra(palavra, acertos, tamanho);
    printf("Digite uma letra: ");
    scanf(" %c", &letra);

    if (!verificarLetra(toupper(letra), palavra, tamanho, acertos)) {
        tentativas++;
    }

    int completou = 1;
    for (i = 0; i < tamanho; i++) {
        if (!acertos[i]) {
            completou = 0;
            break;
        }
    }
}
```



```

        if (completou) {
            printf("Parabens! Voce adivinhou a palavra: %s\n", palavra);
            return; // Sai do jogo ao completar a palavra
        }
    }

    printf("Voce perdeu! A palavra era: %s\n", palavra);
}

int main() {
    int operacao;
    FILE *gabarito;

    while (1) {
        gabarito = fopen("gabarito.txt", "a+"); // Abertura do arquivo

        if (gabarito == NULL) {
            perror("Erro ao abrir o arquivo.\n");
            return 1;
        }

        printf("Escolha uma operacao:\n1. Para adicionar palavras\n2. Para jogar (Em loop)\n3. Para encerrar\nSua escolha: ");
        scanf("%d", &operacao);

        switch (operacao) {
            case 1:
                adicionarPalavra(gabarito);
                break;
            case 2:

```

```
jogar(gabarito);

break;

case 3:

    printf("Encerrando o programa!\n");

    fclose(gabarito);

    return 0;

default:

    printf("Operacao invalida! Tente Novamente!\n");

    break;

}


fclose(gabarito);

Sleep(5000);

system("cls");

}

}
```