



Escrevendo arquivos com fprintf()

🧠 O que é fprintf() ?

`fprintf(FILE *arquivo, const char *formato, ...)` funciona como `printf`, mas escreve os dados formatados diretamente em um arquivo em vez de na tela.

📌 Sintaxe básica:

```
FILE *f = fopen("saida.txt", "w"); // "w" para escrita
fprintf(f, "Olá, mundo!\n");
fclose(f);
```

🔧 Modificadores mais comuns:

Código	Descrição
%d	íntero
%f	número decimal (float/double)
%s	string
%c	caractere

%lf

double (separadamente de float)

Exemplos práticos de como escrever em arquivos com fprintf()

✓ 1. Escrever nome, idade e bairro separados por ;

Código:

```
char nome[] = "João";
int idade = 25;
char bairro[] = "Centro";

FILE *f = fopen("saida1.txt", "w");
fprintf(f, "%s;%d;%s\n", nome, idade, bairro);
fclose(f);
```

Saída no arquivo:

```
João;25;Centro
```

✓ 2. Escrever uma lista de inteiros separados por espaço

Código:

```
int numeros[] = {10, 20, 30, 40, 50};
int i;

FILE *f = fopen("saida2.txt", "w");
for (i = 0; i < 5; i++) {
    fprintf(f, "%d ", numeros[i]);
}
fprintf(f, "\n");
fclose(f);
```

Saída no arquivo:

10 20 30 40 50

✓ 3. Escrever uma frase por linha (com espaços e pontuação)

Código:

```
char frases[2][100] = {
    "Hoje o tempo está bom.",
    "Amanhã pode chover."
};

FILE *f = fopen("saida3.txt", "w");
for (int i = 0; i < 2; i++) {
    fprintf(f, "%s\n", frases[i]);
}
fclose(f);
```

Saída:

```
Hoje o tempo está bom.
Amanhã pode chover.
```

✓ 4. Escrever nome e notas separados por vírgulas

Código:

```
char nome[] = "Fernanda";
float n1 = 8.5, n2 = 7.0, n3 = 9.0;

FILE *f = fopen("saida4.txt", "w");
fprintf(f, "%s,%f,%f,%f\n", nome, n1, n2, n3);
fclose(f);
```

Saída:

Fernanda,8.5,7.0,9.0

✓ 5. Escrever datas no formato **dd/mm/aaaa**

Código:

```
int dia = 13, mes = 5, ano = 2025;  
  
FILE *f = fopen("datas.txt", "w");  
fprintf(f, "%02d/%02d/%04d\n", dia, mes, ano);  
fclose(f);
```

Saída:

13/05/2025

🔍 **%02d** → garante que terá dois dígitos (ex: **01**, **09**, etc.)

🔍 **%04d** → garante quatro dígitos para o ano

✓ 6. Escrever uma mensagem caractere por caractere (como em criptografia)

Código:

```
char mensagem[] = "Hello, world!";  
FILE *f = fopen("saida6.txt", "w");  
  
for (int i = 0; mensagem[i] != '\0'; i++) {  
    fprintf(f, "%c", mensagem[i]);  
}  
  
fclose(f);
```

Saída:

Hello, world!

✓ 7. Escrever uma matriz (por exemplo: 3x3)

Código:

```
int matriz[3][3] = {  
    {1, 2, 3},  
    {4, 5, 6},  
    {7, 8, 9}  
};  
  
FILE *f = fopen("saida7.txt", "w");  
  
for (int i = 0; i < 3; i++) {  
    for (int j = 0; j < 3; j++) {  
        fprintf(f, "%d ", matriz[i][j]);  
    }  
    fprintf(f, "\n");  
}  
  
fclose(f);
```

Saída:

1 2 3
4 5 6
7 8 9

Dicas importantes sobre `fprintf`

Dica	Explicação
Sempre use <code>fclose(f)</code>	Para garantir que os dados sejam salvos corretamente
Use <code>"\n"</code> no final da linha	Para quebrar linhas manualmente

Combine <code>for</code> com <code>fprintf</code>	Ideal para gravar listas, vetores ou matrizes
Use <code>"%02d"</code> ou <code>"%04d"</code>	Para formatar números com zeros à esquerda (útil em datas)
Evite <code>fprintf</code> para texto grande com espaços se <code>fputs</code> for mais simples	Para strings puras, <code>fputs</code> também funciona bem