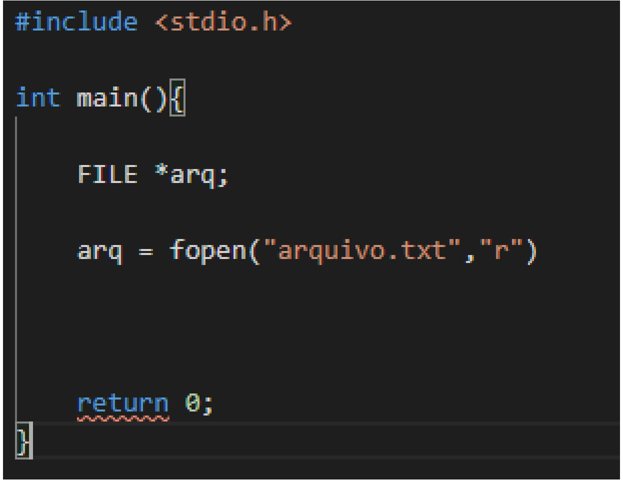
2-LEITURA DE ARQUIVOS

- Vamos aprender a ler arquivos.

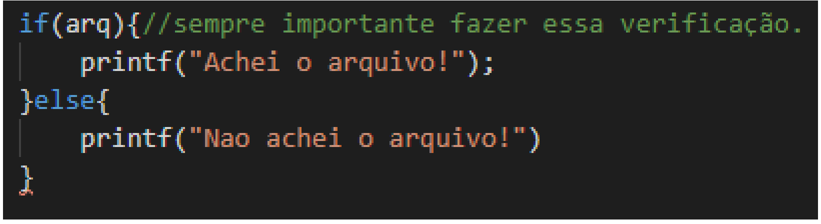


```c
#include <stdio.h>

int main(){

    FILE *arq;

    arq = fopen("arquivo.txt","r")



    return 0;
}
```

- Sempre que utilizamos o fopen, independente do modo de abertura (arq) pode vir nulo ou não.

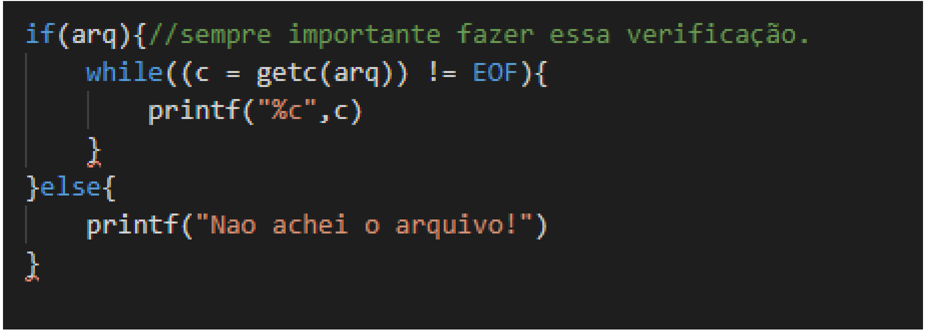
- Por isso temos que fazer uma checagem desse arquivo, pois so podemos continuar o progama ou fazer algo com esse arquivo se ele n for nulo.

    - A variavel poderia ser nula, caso por exemplo ele não exista.



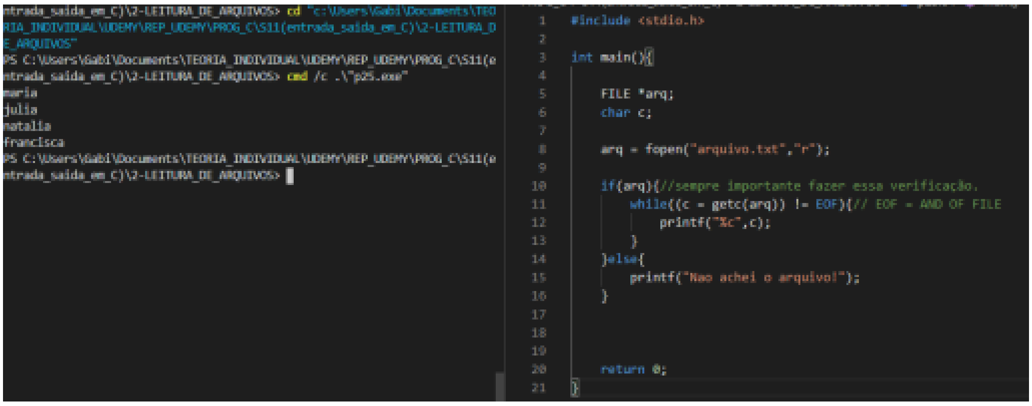
```c
if(arq){//sempre importante fazer essa verificação.
    printf("Achei o arquivo!");
}else{
    printf("Nao achei o arquivo!")
}
```

- Queremos fazer a leitura de um arquivo e imprimir o que esta escrito.



```c
if(arq){//sempre importante fazer essa verificação.
    while((c = getc(arq)) != EOF){
        printf("%c",c)
    }
}else{
    printf("Nao achei o arquivo!")
}
```

- WHILE = Estamos utilizando a função getc() = get caractere arquivo . Vai pegar cada um dos caracteres do arquivo, 1 por 1 sucessivamente, ate o EOF = and of file e imprimir.

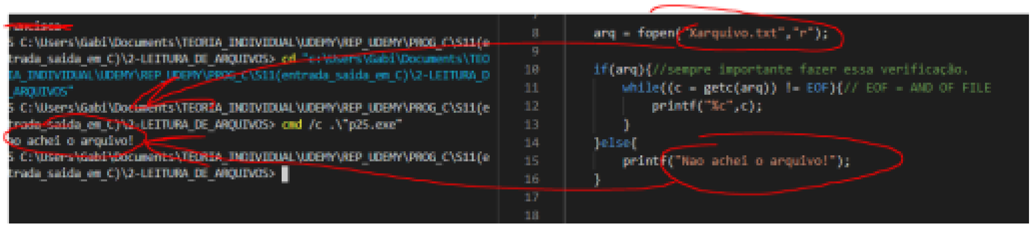


```
ntrada_saida_em_C)\2-LEITURA_DE_ARQUIVOS> cd "c:\Users\Gabi\Documents\TEO
RIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_D
E_ARQUIVOS"
PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(e
ntrada_saida_em_C)\2-LEITURA_DE_ARQUIVOS> cmd /c .\"p25.exe"
maria
julia
natalia
francisca
PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(e
ntrada_saida_em_C)\2-LEITURA_DE_ARQUIVOS> 
```

```c
#include <stdio.h>

int main(){

    FILE *arq;
    char c;

    arq = fopen("arquivo.txt","r");

    if(arq){//sempre importante fazer essa verificação.
        while((c = getc(arq)) != EOF){// EOF = AND OF FILE
            printf("%c",c);
        }
    }else{
        printf("Nao achei o arquivo!");
    }



    return 0;
}
```

- Repare que pegamos caractere por caractere.

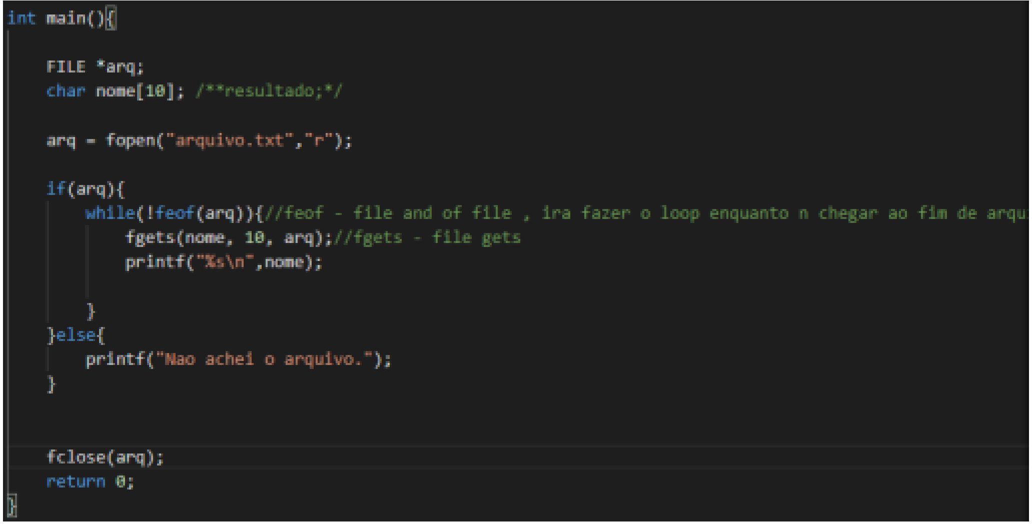
- Lembre-se que o fopen na opção r, so ira funcionar se o arquivo existir.



```
~~rancisca~~
S C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(e
trada_saida_em_C)\2-LEITURA_DE_ARQUIVOS> cd "c:\Users\Gabi\Documents\TEO
IA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_D
_ARQUIVOS"
S C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(e
trada_saida_em_C)\2-LEITURA_DE_ARQUIVOS> cmd /c .\"p25.exe"
ao achei o arquivo!
S C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(e
trada_saida_em_C)\2-LEITURA_DE_ARQUIVOS> 
```

```c
    arq = fopen("Xarquivo.txt","r");

    if(arq){//sempre importante fazer essa verificação.
        while((c = getc(arq)) != EOF){// EOF = AND OF FILE
            printf("%c",c);
        }
    }else{
        printf("Nao achei o arquivo!");
    }


```

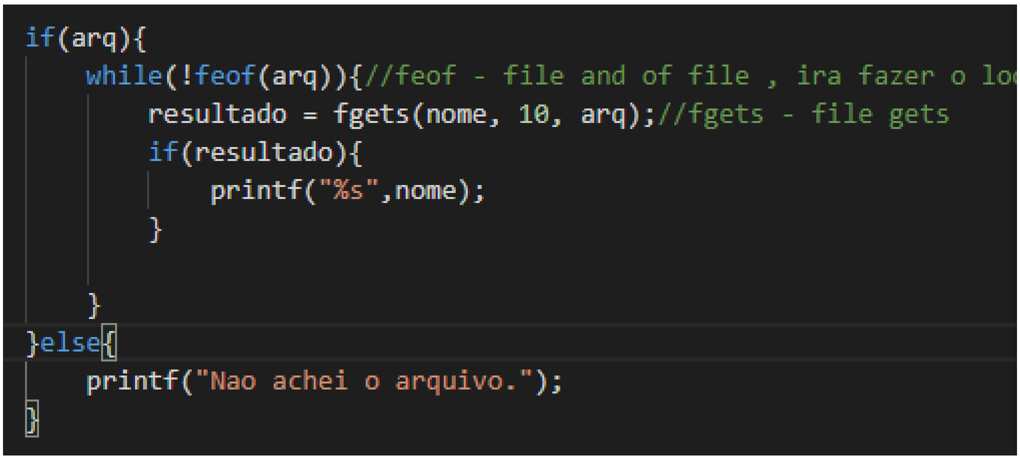
- Queremos agora pegar linha a linha, em vez de caractere por caractere.



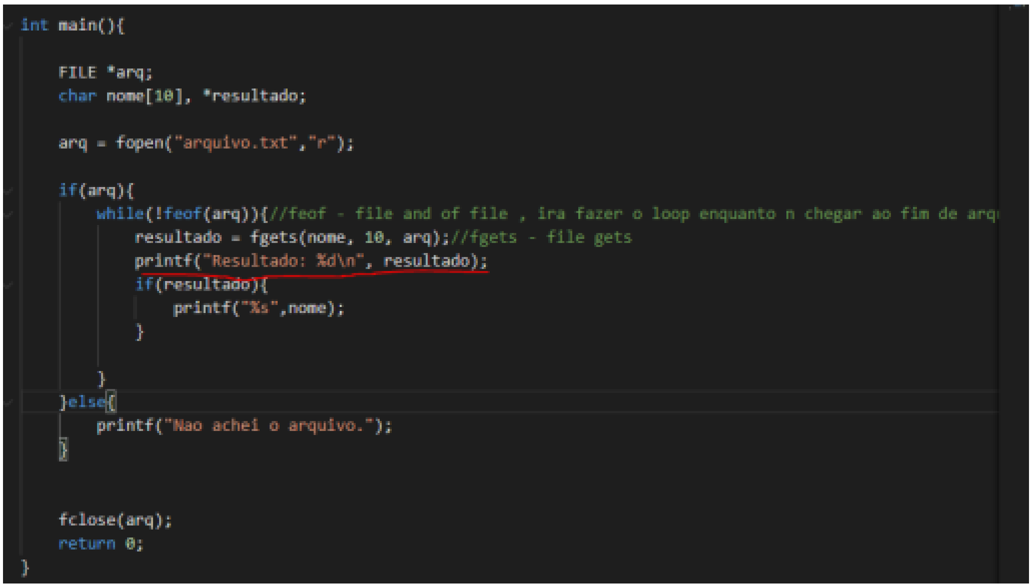
```c
int main(){

    FILE *arq;
    char nome[10]; /**resultado;*/

    arq = fopen("arquivo.txt","r");

    if(arq){
        while(!feof(arq)){//feof - file and of file , ira fazer o loop enquanto n chegar ao fim de arqu
            fgets(nome, 10, arq);//fgets - file gets
            printf("%s\n",nome);

        }
    }else{
        printf("Nao achei o arquivo.");
    }


    fclose(arq);
    return 0;
}
```

- **<u>fgets(nome-da-variavel-que-ira-colocar, numero_de_caracteres_pegos, nome_arquivo);</u>**
- Queremos pegar os 10 primeiros caracteres e colocar na variavel que criamos nome[10] e depois vamos imprimir.
- Gerando uma linha a mais que não da para ver.

- Para ter certeza de o que esta sendo impresso(pego) é o que nos queremos, vamos utilizar a variavel resultado para receber o valor que fgets pega. E so vamos imprimir se o resultado for verdadeiro. Ou seja, vamos fazer uma verificação primeiro desse resultado.



```c
if(arq){
    while(!feof(arq)){//feof - file and of file , ira fazer o loo
        resultado = fgets(nome, 10, arq);//fgets - file gets
        if(resultado){
            printf("%s",nome);
        }

    }
}else{
    printf("Nao achei o arquivo.");
}
```

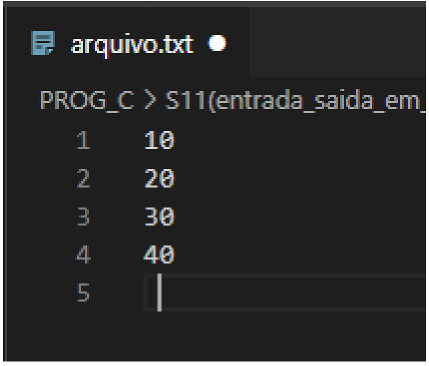


```c
int main(){

    FILE *arq;
    char nome[10], *resultado;

    arq = fopen("arquivo.txt","r");

    if(arq){
        while(!feof(arq)){//feof - file and of file , ira fazer o loop enquanto n chegar ao fim de arq
            resultado = fgets(nome, 10, arq);//fgets - file gets
            printf("Resultado: %d\n", resultado);
            if(resultado){
                printf("%s",nome);
            }

        }
    }else{
        printf("Nao achei o arquivo.");
    }


    fclose(arq);
    return 0;
}
```

- Esse 0 é o que estava gerando a linha a mais, como estamos fazendo uma checagem do resuldado (if2) o zero não entra mais .

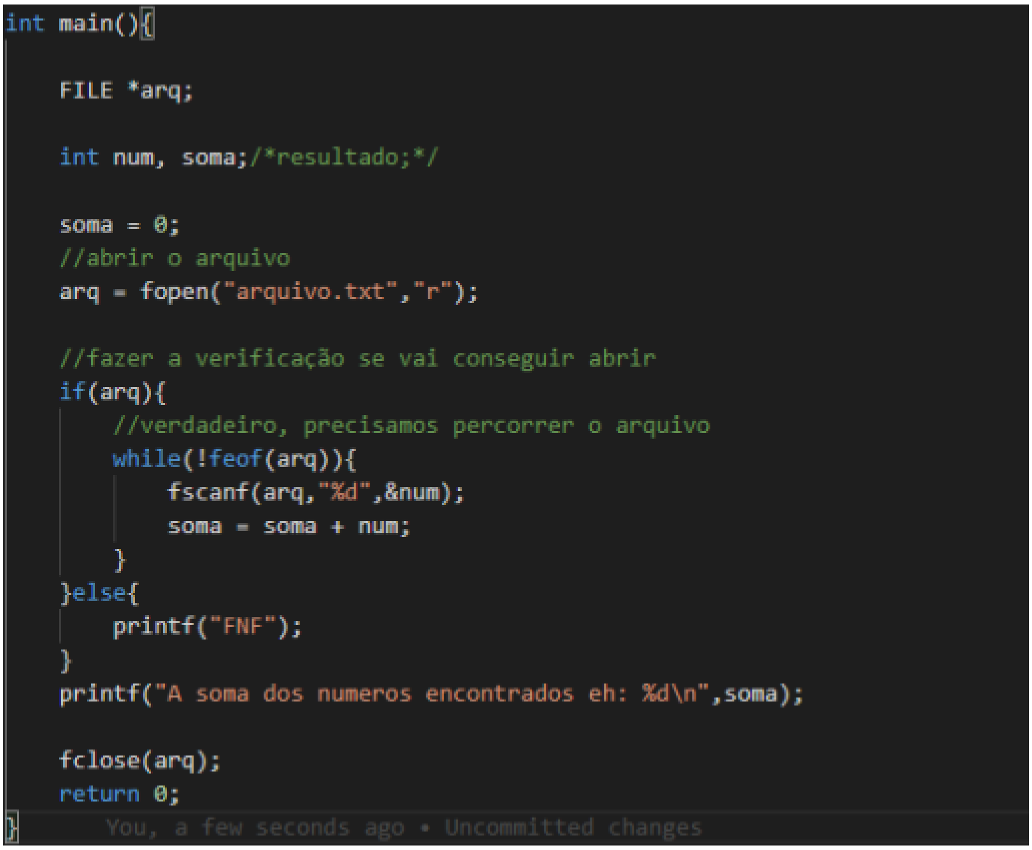
OUTRA SITUAÇÃO

- Imagine que tenhamos um arquivo somente com numero, e precisamos fazer alguma operação com eles..
- Por exemplo somar.



```
10
20
30
40

```

- Codigo

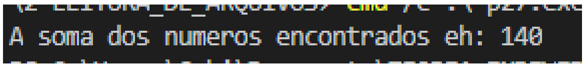


```c
int main(){

    FILE *arq;

    int num, soma;/*resultado;*/

    soma = 0;
    //abrir o arquivo
    arq = fopen("arquivo.txt","r");

    //fazer a verificação se vai conseguir abrir
    if(arq){
        //verdadeiro, precisamos percorrer o arquivo
        while(!feof(arq)){
            fscanf(arq,"%d",&num);
            soma = soma + num;
        }
    }else{
        printf("FNF");
    }
    printf("A soma dos numeros encontrados eh: %d\n",soma);

    fclose(arq);
    return 0;
}
```

-if{}else{} - checagem de arquivo
- while(){} - percorrer o arquivo.
- Descrição:

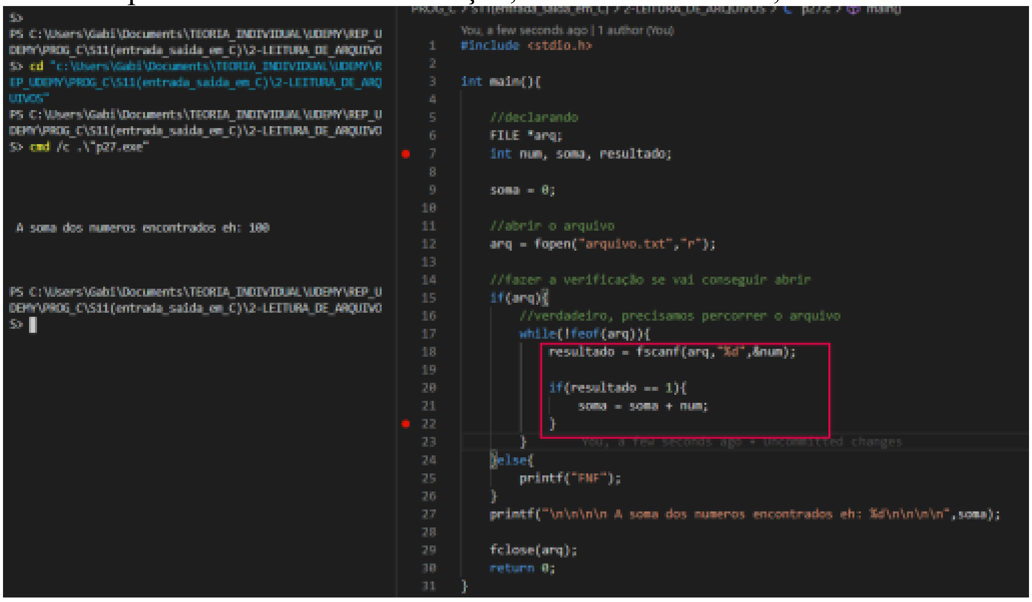
Na linha 1 do arquivo.txt temos o valor 10(num=10), A variavel soma(0) será somada por ela mesma mais a variavel num(10) , guardando o novo valor soma(10). Essa sequencia se repetirá ate o EOF - end of file.
Depois ira imprimir o valor da soma.

- Resultado



```
A soma dos numeros encontrados eh: 140
```

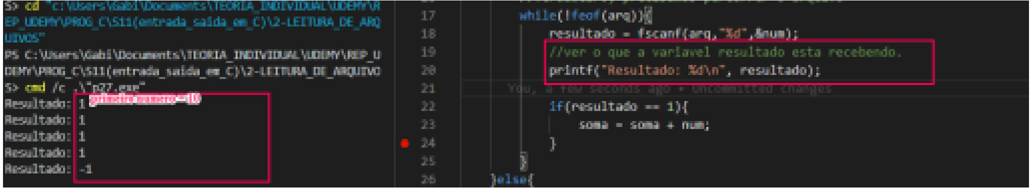
- O total seria 100, esta dando errado, pois a ultima linha esta sendo impressa, ou seja, antes do EOF existe uma linha vazia que não eh vista, temos que fazer uma verificação para que a ultima linha não apareça com a ajuda da variavel RESULTADO.
- Resultado ira receber o que o scanf conseguir.
- E depois fazemos uma verificação, se resultado for = 1, faz a soma.



```
S>
PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_U
DEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_DE_ARQUIVO
S> cd "c:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\R
EP_UDEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_DE_ARQ
UIVOS"
PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_U
DEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_DE_ARQUIVO
S> cmd /c .\"p27.exe"




 A soma dos numeros encontrados eh: 100




PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_U
DEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_DE_ARQUIVO
S> 
```

```c
#include <stdio.h>

int main(){

    //declarando
    FILE *arq;
    int num, soma, resultado;

    soma = 0;

    //abrir o arquivo
    arq = fopen("arquivo.txt","r");

    //fazer a verificação se vai conseguir abrir
    if(arq){
        //verdadeiro, precisamos percorrer o arquivo
        while(!feof(arq)){
            resultado = fscanf(arq,"%d",&num);

            if(resultado == 1){
                soma = soma + num;
            }
        }
    }else{
        printf("FNF");
    }
    printf("\n\n\n\n A soma dos numeros encontrados eh: %d\n\n\n\n",soma);

    fclose(arq);
    return 0;
}
```

- Vamos entender porque RESULTADO ==1;



```
S> cd "c:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_DE_ARQUIVOS"
PS C:\Users\Gabi\Documents\TEORIA_INDIVIDUAL\UDEMY\REP_UDEMY\PROG_C\S11(entrada_saida_em_C)\2-LEITURA_DE_ARQUIVO
S> cmd /c .\"p27.exe"
Resultado: 1
Resultado: 1
Resultado: 1
Resultado: 1
Resultado: -1
```

```c
        while(!feof(arq)){
            resultado = fscanf(arq,"%d",&num);
            //ver o que a variavel resultado esta recebendo.
            printf("Resultado: %d\n", resultado);

            if(resultado == 1){
                soma = soma + num;
            }
        }
    }else{
```

- 1 = significa que ele encontrou 1 numero naquele local.
- -1 = significa nenhum numero encontrado.
- Por isso precisamos fazer a verificação de efetuar a soma se o resultado for igual a 1. Poque assim se ele não encotrar um numero, a variavel não será substituida, ou seja, mantem o valor final.