

Computação para Informática - Prof. Adriano Joaquim de Oliveira Cruz
Arquivos text

1 Arquivo Texto

Exercício 1: Escreva um programa que grave na mesma linha de um arquivo chamado `notas.txt` dois números reais lidos do teclado. Abra este arquivo usando um programa do tipo `notepad` e verifique o resultado do programa. **O nome do arquivo deve ser lido do teclado.**

Exemplo de Formato de Entrada:

```
Entre com o nome do arquivo
notas.txt
Entre com o primeiro numero real:
10.0
Entre com o segundo numero real:
15.0
```

Exemplo do formato do arquivo

```
10.0 15.0
```

Exercício 2: Os alunos de uma turma fizeram duas provas. Escreva um programa que leia do teclado as notas destes alunos e as grave em um arquivo. O arquivo contém duas notas por linha. O número de alunos da turma é igual a 10. As notas devem ser gravadas com uma casa decimal e alinhadas a direita conforme o exemplo abaixo.

Abra este arquivo usando um programa do tipo `notepad` e verifique o resultado do programa. **O nome do arquivo deve ser pedido ao usuário.**

Exemplo de Formato de Entrada:

```
Entre com o nome do arquivo
notas.txt
Entre com as notas do aluno 1:
10.0 8.5
Entre com as notas do aluno 2:
5.0 8.5
...
Entre com as notas do aluno 10:
8.0 8.0
```

Exemplo do formato do arquivo

```
10.0 8.5
5.0 8.5
9.0 10.0
1.0 1.0
```

```
...  
8.0 8.0
```

Exercício 3: Escreva um programa que leia os nomes completos de 10 alunos e a sua média final e grave estas informações em um arquivo chamado `turma.txt`. Abra este arquivo usando um programa do tipo `notepad` e verifique o resultado do programa. Considere que os nomes dos alunos tem no máximo 20 caracteres.

Exemplo de Formato de Entrada:

```
Entre com o nome do arquivo  
turma.txt  
Nome do aluno 1:  
12345678901234567890  
Entre com a nota de 12345678901234567890:  
5.0  
Nome do aluno 2:  
Ze Sa  
Entre com a nota de Ze Sa:  
5.0  
Nome do aluno 3:  
Ola Ali  
Entre com a nota de Ola Ali:  
9.0  
...
```

Exemplo do formato do arquivo.

```
12345678901234567890 10.0  
Ze Sa 5.0  
Ola Ali 9.0  
Joao da Silva Xavier 6.8
```

Exercício 4: Use um programa tipo `notepad` para escrever em um arquivo texto um conjunto de números inteiros de tamanho desconhecido. Escreva um número por linha do arquivo. Escreva um programa leia estes números e os imprima.

Para ler um arquivo com tamanho desconhecido iremos usar a função `fEOF` que tem o seguinte protótipo

```
int fEOF(FILE *p);
```

Esta função avisa que o final do arquivo foi atingido e não há mais nada para ser lido. OBSERVE QUE A FUNÇÃO SÓ AVISA QUE O ARQUIVO ACABOU DEPOIS QUE UMA TENTATIVA DE LEITURA É REALIZADA. O trecho de programa mostrado na listagem 1 serve para ler um arquivo contendo inteiros quando não se sabe quantos serão lidos. Outra possibilidade é mostrada na listagem 2

Listagem 1: Leitura de arquivo de tamanho desconhecido.

```
int i;  
  
while (1)  
{  
    fscanf(p, "%d", &i);  
    if (fEOF(p))
```

```
    {
        puts("Acabou");
        break;
    }
    printf("%d\n", i);
}
```

Listagem 2: Leitura de arquivo de tamanho desconhecido.

```
int i;

fscanf(p, "%d", &i);
while (!feof(p))
{
    printf("%d\n", i);
    fscanf(p, "%d", &i);
}
puts("Acabou");
```

Exercício 5: Modifique o programa anterior de modo que ele imprima os números lidos na ordem inversa a que eles foram lidos.

Dica:

1. Leia os números lidos apenas para contar a quantidade de números.
2. Agora que sabe quantos são, reserve memória (`malloc`) para guardar os números em um vetor.
3. Execute a função `rewind(parq)` para retornar para o início do arquivo.
4. Agora leia guardando os números no vetor
5. Imprima os números na ordem inversa.

Exercício 6: Escreva um programa que adicione ao arquivo do exercício anterior dez números lidos do teclado. O novo arquivo e o original somente devem ser diferentes por estes dez novos números ao final do arquivo.

Dica: usar `fopen("NomeArquivo.txt", "a")`. Este "a" garante que tudo que você escrever no arquivo será automaticamente adicionado ao final do arquivo.

Exercício 7: Lendo caracteres:

Neste exercício iremos ler um arquivo texto com caracteres e não como números. O trecho de programa mostrado na listagem 3 serve para ler um arquivo caracter a caracter. Observe que mesmo que um arquivo contenha números o programa irá ler cada caracter do número individualmente.

Listagem 3: Leitura de arquivo como caracteres.

```
#include <stdio.h>

int main(int argc, char *argv[])
{
    FILE *pa;
    char *nome = "texto2.txt";
```

```
char c;

pa = fopen(nome, "r");
if (!pa)
{
    printf("Erro ao abrir %s\n", nome);
    return 1;
}

while (1)
{
    c = fgetc (pa);
    if (c == EOF) break;
    putchar(c);
}

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

Exercício 8: Escreva um programa que leia um arquivo texto contendo um conjunto de linhas. Em cada linha há um nome (somente nome) e um número. O programa deve gerar um outro arquivo em que cada linha apareça o número e em seguida o nome.

Exercício 9: Escreva um programa que leia um arquivo texto contendo um conjunto de linhas. Em cada linha há um nome, agora com um sobrenome, e um número. O programa deve gerar um outro arquivo em que cada linha apareça o número e em seguida o nome que pode conter vários sobrenomes.

2 Para os fortes

Exercício 10: Use o programa da listagem 3 para escrever um outro programa que faça uma cópia de um arquivo texto. Os nomes dos dois arquivos devem ser pedidos ao usuário e não podem ser iguais. Se os nomes forem iguais peça novo nome até que o usuário forneça um nome diferente.

Exercício 11: Escreva um programa que leia do arquivo gerado no exercício 1 as notas de 10 alunos. O seu programa deve calcular a média de cada aluno e gravar as três notas (**n1**, **n2** e a **media**) no mesmo arquivo usando o mesmo esquema de notas de um aluno por linha do arquivo.

Este programa não deve imprimir absolutamente nada. Abra este arquivo usando um programa do tipo **notepad** e verifique o resultado do programa.

Perguntas: Sabemos que não é fácil (quase impossível) escrever um dado diretamente no meio de arquivo texto. Sabemos? Por que não pode? Como inserir a média?

Dica

Passo 1: Abra o arquivo para leitura, leia os dados para vetores e feche o arquivo.

Passo 2: Em seguida abra o arquivo para escrita e grave os dados junto com a média.

Exercício 12: Use um programa tipo **notepad** para criar um arquivo texto contendo o nome (um único nome) e a quantidade de pontos de jogadores de um jogo qualquer. O total de jogadores armazenados é no máximo 5. Cada jogador ocupa uma linha do arquivo. Em cada linha, primeiro temos o nome e depois o total de pontos. Considere que os dados estão armazenados do melhor jogador para o menos melhor. Escreva um programa que leia os dados de um novo jogador (nome e pontos) e verifique se ele deve entrar na lista dos cinco melhores jogadores. Caso ele tenha pontos para entrar, o jogador com menos pontos deve ser retirado e o novo jogador colocado na sua posição do rank. Após rodar o programa verifique usando o notepad se o programa está correto. Teste para vários casos diferentes.