

Sumário

1. Introdução:	2
2. Implementação:	2
3. Testes	3
4. Conclusão	4
Referências	4
Anexos	4
Atv1.c	
DadosEntrada.csv	
SituacaoFinal.csv	
	4

1. Introdução:

O problema a ser tratado é sobre o cálculo de média de notas das provas dos alunos e mostrar se a situação de reprovado ou aprovado de acordo com a média da instituição que é nota 7, aqueles que obtiverem nota inferior a essa estarão reprovados e aqueles com nota igual ou superior a ela estarão aprovados. O objetivo é fazer um sistema onde se possa colocar as notas dos alunos e o sistema por si próprio calcular a média e mostrar em um documento a média e a situação do aluno se está reprovado ou aprovado.

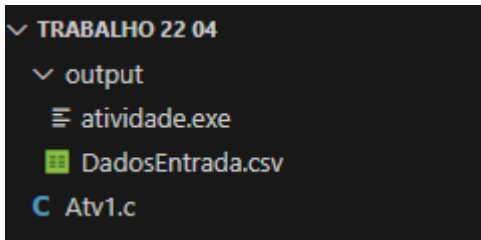
GitHub:

<https://github.com/ltees/EstruturaDeDados.git>

2. Implementação:

Este programa em C foi desenvolvido para processar dados de alunos a partir de um arquivo CSV. Utiliza uma estrutura de dados chamada Aluno, que armazena informações como nome, telefone, curso, notas e situação (aprovado/reprovado). As funções lerDados, calcularSituacao e escreverSituacao são responsáveis, respectivamente, por ler os dados dos alunos do arquivo CSV, calcular suas médias e determinar sua situação, e escrever essas informações em outro arquivo CSV. O programa principal exibe um menu com opções para processar os dados dos alunos ou sair, continuando a solicitar opções até que o usuário decida sair. Decisões de implementação incluem a verificação de abertura de arquivo bem-sucedida e a existência de alunos a serem processados antes de realizar operações. Utilizamos o GCC como compilador. Para executá-lo, compilamos o código-fonte e execute o executável gerado, garantindo que exista um arquivo "DadosEntrada.csv" com os dados dos alunos e permissões de escrita para gerar o arquivo de saída "SituacaoFinal.csv".

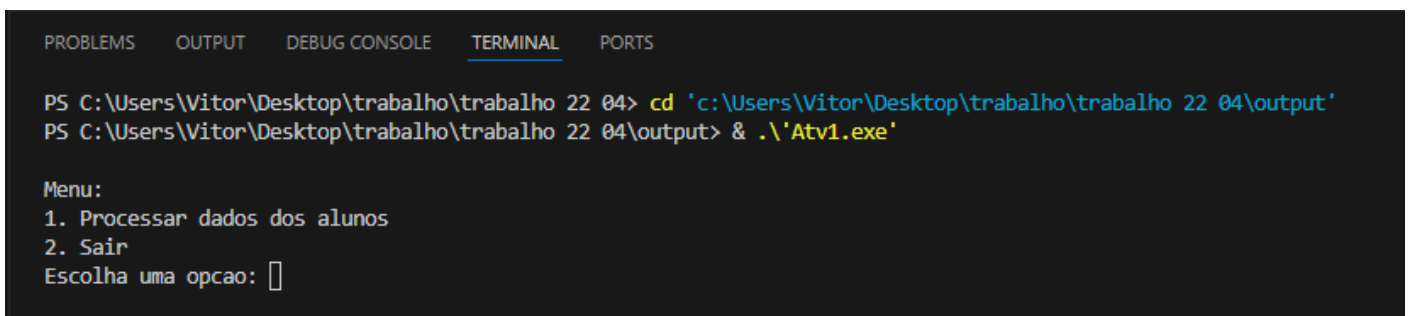
3. Testes



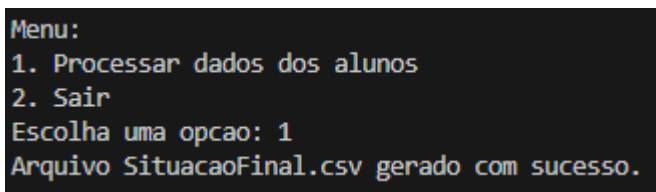
Repositório antes da execução do nosso programa.



Passamos por alguns problemas, porém resolvemos após a correção de alguns códigos escritos incorretamente.



Após corrigir, realizamos a compilação e criamos um menu para ficar mais intuitivo.



Após a compilação do .exe, o programa criou o arquivo SituacaoFinal.csv.

4. Conclusão

Encontramos em nosso projeto o resultado desejado por nós e o que foi pedido pela a atividade, alguns problemas apareceram durante a construção do código e nas primeiras execuções pois não estava imprimindo os resultados, mas ao final o grupo soube resolver o problema. Em possíveis melhorias que poderiam ser implementadas caso houvesse uma nova versão do trabalho, o grupo pensou adicionar novas informações para o cadastro dos alunos e calcular a nota de todas as atividades feitas por eles e somar uma média final.

Referências

<https://gist.github.com/wrogati/70feba528cd825640819f233f97341ad>

<https://linguagemc.com.br/calcular-media-em-c/>

Anexos

DadosEntrada.csv

SituacaoFinal.csv

Atv1.c

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

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

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

```
#define MAX_NOME 50
```

```
#define MAX_TELEFONE 15
```

```
#define MAX_CURSO 50
```

```
#define MAX_NOTAS 5
```

```
#define MAX_ALUNOS 100
```

```
typedef struct {  
    char nome[MAX_NOME];  
    char telefone[MAX_TELEFONE];  
    char curso[MAX_CURSO];  
    float notas[MAX_NOTAS];  
    float media;  
    char situacao[10];  
} Aluno;  
  
// funcao para ler os dados dos alunos a partir de um arquivo CSV  
  
int lerDados(Aluno alunos[]) {  
    FILE *entrada;  
    char nome_arquivo_entrada[] = "DadosEntrada.csv";  
    int num_alunos = 0;  
    entrada = fopen(nome_arquivo_entrada, "r");  
    if (entrada == NULL) {  
        printf("Erro ao abrir o arquivo de entrada.\n");  
        return -1;  
    }  
  
    while (fscanf(entrada, "%[^,],%[^,],%[^,],%f,%f\n", alunos[num_alunos].nome,  
        alunos[num_alunos].telefone, alunos[num_alunos].curso,  
        &alunos[num_alunos].notas[0], &alunos[num_alunos].notas[1]) != EOF) {  
        num_alunos++;  
    }  
  
    fclose(entrada);  
  
    return num_alunos;  
}
```

```
// funcao para calcular a situacao (aprovado/reprovado) dos alunos com base nas notas
```

```
void calcularSituacao(Aluno alunos[], int num_alunos) {  
  
    for (int i = 0; i < num_alunos; i++) {  
  
        alunos[i].media = (alunos[i].notas[0] + alunos[i].notas[1]) / 2;  
  
        if (alunos[i].media >= 7.0)  
  
            strcpy(alunos[i].situacao, "APROVADO");  
  
        else  
  
            strcpy(alunos[i].situacao, "REPROVADO");  
  
    }  
}
```

```
// funcao para escrever a situacao final dos alunos em um arquivo CSV
```

```
void escreverSituacao(Aluno alunos[], int num_alunos) {  
  
    FILE *saida;  
  
    char nome_arquivo_saida[] = "SituacaoFinal.csv";  
  
    saida = fopen(nome_arquivo_saida, "w");  
  
    if (saida == NULL) {  
  
        printf("Erro ao abrir o arquivo de saida.\n");  
  
        return;  
  
    }  
  
    for (int i = 0; i < num_alunos; i++) {  
  
        fprintf(saida, "%s, %.2f, %s\n", alunos[i].nome, alunos[i].media, alunos[i].situacao);  
  
    }  
  
    fclose(saida);  
  
    printf("Arquivo %s gerado com sucesso.\n", nome_arquivo_saida);  
}  
  
int main() {  
  
    Aluno alunos[MAX_ALUNOS];
```

```
int num_alunos;

int opcao;

do {

    printf("\nMenu:\n");

    printf("1. Processar dados dos alunos\n");

    printf("2. Sair\n");

    printf("Escolha uma opcao: ");

    scanf("%d", &opcao);

    switch (opcao) {

        case 1:

            num_alunos = lerDados(alunos);

            if (num_alunos <= 0) {

                printf("Nenhum aluno encontrado.\n");

                break;

            }

            calcularSituacao(alunos, num_alunos);

            escreverSituacao(alunos, num_alunos);

            break;

        case 2:

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

            break;

        default:

            printf("Opcao invalida. Tente novamente.\n");

    }

} while (opcao != 2);

return 0;

}
```

