

Universidade Federal de Goiás
Instituto de Informática
Introdução à Programação 2017-2
Trabalho sobre Arquivos

Profa. Luciana de Oliveira Berretta

1 Especificação

O trabalho consiste em um pequeno sistema para cadastrar, consultar e alterar dados de alunos de turmas de Introdução à Programação. O sistema deve manter os seguintes dados para cada aluno:

- Matrícula - um número inteiro sem sinal (digitado pelo usuário).
- Nome - cadeia de caracteres (digitado pelo usuário).
- Notas de prova - um vetor com seis valores reais (digitadas pelo usuário).
- Notas de listas de exercício - um vetor com seis valores reais (digitadas pelo usuário).
- Nota de trabalho (NT) - um número real (digitado pelo usuário)
- Nota final - um valor real (computada pelo sistema).
- Número de ausências - um número inteiro sem sinal (digitado pelo usuário).
- Situação - cadeia de caracteres: “Aprovado”, “Reprovado por ausência”, “Reprovado por Nota”, “Reprovado por ausência e por nota” (Computada pelo sistema).

O sistema deve conter cinco módulos: Cadastro, Consulta, Alteração, Listagem, Principal. Esses módulos serão especificados nas subseções seguintes.

Módulo de Cadastro

O programa deve ter um módulo de cadastro que permite a digitação dos dados e gravação dos mesmos em um arquivo binário. O módulo deve perguntar ao usuário qual é o nome do arquivo correspondente à turma a ser cadastrada e seu caminho no disco onde o cadastro deve ser gravado. Após criar o arquivo, o módulo deve ter um comando de repetição que recebe do teclado diversos dados de alunos, calcula suas notas finais e as suas situações finais e grava os dados dos alunos no arquivo. O comando de repetição termina quando o usuário digitar um número de matrícula igual a zero. Esse número de matrícula não deve ser armazenado no arquivo e serve para informar ao módulo que o cadastro terminou.

A nota final de um aluno é composta pela média das notas de prova MP e pela média das notas de listas de exercícios de exercícios e pela nota de trabalho. A média das provas é computada conforme a equação 1.

$$MP = \frac{\sum_{i=1}^6 P_i}{6} \quad (1)$$

onde P_i é a nota da prova i , $1 \leq i \leq 6$. A média das notas de listas de exercícios é dada pela equação ??

$$MP = \frac{\sum_{i=1}^6 L_i}{6} \quad (2)$$

onde L_i é a nota da lista de exercícios i , $1 \leq i \leq 6$. A nota final NF do aluno é computada conforme a equação 3.

$$NF = NF * 0,8 + NL * 0,1 + NT * 0,1 \quad (3)$$

A situação do aluno é definida pelos seguintes critérios:

- Se $NF \geq 6$ e número de ausências menor que 25% de 128 a situação do aluno é “Aprovado”.
- Se $NF < 6$ e número de ausências menor que 25% de 128 a situação do aluno é “Reprovado por nota”.
- Se $NF \geq 6$ e número de ausências for maior ou igual a 25% de 128 a situação do aluno é “Reprovado por ausência”.
- Se o aluno não se enquadrar em nenhum dos critérios anteriores a sua situação final dele é “Reprovado por ausência e por nota”.

Módulo de Consulta

O módulo de consulta quando ativado, deve perguntar ao usuário qual o arquivo corresponde à turma sobre a qual a consulta deve ser feita. O módulo deve ficar mostrando a mensagem “Digite o nome de um arquivo de turma” e deve ficar lendo um nome digitado pelo usuário até que o usuário digite um nome de arquivo válido, ou digite “fim” para sair do módulo. Após o usuário digitar um nome válido, o programa deve entrar em um laço de repetição em que ele pede uma matrícula para o usuário, procura no arquivo a matrícula e mostra os dados do aluno na tela. Esse laço de repetição termina somente quando o usuário digita uma matricula igual a zero.

Módulo de alteração

Esse módulo também deve ter um laço de repetição que fica pedindo ao usuário um nome de arquivo enquanto o usuário não digitar um nome de arquivo que existe ou digitar “fim”. Após o usuário digitar um nome válido de arquivo, o programa deve abrir o arquivo, se ele já não estiver aberto, e deve entrar em um laço em que pede repetidamente ao usuário um número de matricula válido, até que um número válido seja digitado ou até que o usuário digite uma matricula igual a zero. No último caso, o módulo encerra sua execução. Após o

usuário digitar um número de matrícula válido, o programa deve mostrar os dados do aluno correspondente e pedir ao usuário que digite novos dados correspondentes aos campos da estrutura correspondente ao aluno. Apenas os dados que são digitados pelo usuário devem ser requisitados. Após o usuário digitar novos dados para o aluno, o módulo deve recalcular as médias, a nota final, a situação final do aluno e gravar os dados no arquivo. Em seguida, o módulo volta a pedir nova matrícula para o usuário. O usuário pode digitar uma matrícula válida ou zero para termina a execução do o módulo.

Módulo de Listagem

Esse módulo deve ter um laço de repetição que fica pedindo ao usuário um nome de arquivo enquanto o usuário não digitar um nome de arquivo que existe ou digitar “fim”. Se o usuário digitar um nome válido de arquivo, o módulo deve entrar em outro laço de repetição e mostrar o seguinte menu de opções ao usuário:

- “1 - listar os alunos por ordem crescente de matrícula”
- “2 - listar os alunos por ordem alfabética de nome”
- “3 - Listar os alunos por ordem crescente de nota final”
- “4 - terminar e voltar para processar nova turma”
- “5 - terminar a execução do módulo”

Ainda dentro do laço de repetição, o módulo deve ler a opção escolhida pelo usuário e apresentar a listagem correspondente ou terminar o módulo e voltar ao módulo principal.

Módulo Principal

O módulo principal corresponde à função `main()`. Esse módulo tem como função apenas ficar ativando os demais módulos para atenderem às requisições dos usuários. Ele contém um laço de repetição onde o menu de opções abaixo fica sendo mostrado repetidas vezes ao usuário, até que ele digite “4” (terminar a execução do sistema):

- “1 - Cadastrar uma turma”
- “2 - Consultar um aluno”
- “3 - Alterar dados de um aluno”
- “4 - Listar dados de uma turma”
- “5 - Terminar a execução”

2 Informações Sobre a Implementação e a Entrega

O trabalho é individual ou em grupo de dois alunos apenas. Se o trabalho for feito em dupla deve haver uma divisão de tarefas da seguinte forma:

- Um dos componentes fica responsável por implementar e explicar a implementação dos seguintes módulos: Cadastro e Alteração.

- O outro componente fica responsável por implementar e explicar a implementação dos seguintes módulos: Consulta e Listagem.
- Ambos ficam responsáveis pelo módulo Principal.

O trabalho deve ser entregue em um diretório compactado contendo os fontes dos arquivos e Makefile que permite compilar e ligar todos os executáveis. Para isso, deve-se colocar todos os arquivos fontes e o Makefile em um diretório (ex. Trabalho) e executar o seguinte comando: *tar -zcvf Trabalho.zip Trabalho*. O arquivo .zip gerado deve ser submetido na tarefa criada no Moodle para essa finalidade. O trabalho deve ser entregue até o dia 16/12/2017 no Moodle. A avaliação do trabalho será o resultado de duas avaliações. A primeira é uma apresentação do trabalho que deve ser feita na aula do dia 12/12/2017, mesmo que o trabalho não esteja totalmente concluído. Porém, os módulos devem estar em estágio avançado de implementação. A segunda avaliação será feita com base na entrega do trabalho.

Importante:

Cópias idênticas ou modificadas dos códigos ou de partes dos códigos são consideradas plágios. **Plágio é crime!** Alunos envolvidos em plágio (quem cedeu e quem copiou) receberão nota zero no trabalho, mesmo que o plágio seja apenas parcial.