

Especificação do Trabalho Prático

O trabalho prático da disciplina consiste em desenvolver o mesmo sistema computacional para solução do problema descrito abaixo nas duas linguagens de programação apresentadas durante o curso: Java e C++.

O trabalho Java é dividido em 4 etapas para que o grupo possa receber feedback do professor nas etapas intermediárias e melhorar o código. O trabalho C++ deve ser entregue em uma única vez, considerando o que foi entregue na última etapa do trabalho Java.

Os trabalhos podem ser feitos em times de até 2 estudantes (i.e., pode ser feito sozinho ou em dupla). Recomenda-se que o mesmo time realize ambos os trabalhos, porém podemos nos adequar a situações diversas.

Novidades em relação a versões anteriores da especificação estão marcadas em amarelo.

1. Descrição geral do problema

Professores do Departamento de Informática da Ufes lecionam disciplinas para os cursos de graduação do Centro Tecnológico e para o Programa de Pós-Graduação em Informática. Em cada disciplina, é necessário efetuar avaliação dos estudantes, que pode ser por meio de provas ou trabalhos práticos em grupo. Por exemplo, o prof. Vítor em 2023/1 irá avaliar os alunos de Programação Web com 3 trabalhos práticos em grupos, já a prof^a. Veruska vai avaliar os alunos de Programação I (Engenharia Elétrica) com 2 provas e 1 trabalho prático.

Ao longo do semestre, o(a) professor(a) registra informações sobre as disciplinas que leciona naquele semestre, os alunos matriculados em cada disciplina, as avaliações de cada disciplina, os grupos formados para cada trabalho e as notas de cada grupo/aluno em cada avaliação. Ao final do período, o professor gostaria de produzir uma série de relatórios que indicariam, por exemplo, se há alguma inconsistência no que foi cadastrado, a pauta com notas, médias, etc. de cada disciplina, estatísticas para cada disciplina e curso, etc.

Seu trabalho é construir um programa que permita o registro dos cursos, disciplinas, avaliações, alunos e notas, gerando relatórios úteis para os professores do departamento.

O trabalho é dividido em quatro etapas, descritas a seguir. Cada etapa tem um prazo de entrega, que deve ser observado na agenda da disciplina.

2. Etapa 1: cadastros básicos

Ao ser iniciado, seu programa deve exibir um menu de opções e permitir que o(a) usuário(a) escolha uma das funcionalidades disponíveis. Na primeira etapa, são esperadas as seguintes funcionalidades:

- Cadastrar curso;
- Cadastrar disciplina;
- Cadastrar prova;
- Cadastrar aluno(a);

- Matricular aluno(a) em disciplina;
- Registrar nota de aluno em prova;
- Imprimir dados;
- Sair do programa.

É importante observar que, nesta etapa, não há ainda controle de erros. Portanto, é esperado que o usuário do seu programa informe sempre entradas válidas. Esta preocupação será adicionada em etapa posterior.

2.1. Cadastrar curso

Dos cursos, deseja-se registrar um código numérico (inteiro) e um nome. Todos os dados são obrigatórios.

Exemplo:

- Código: 11
- Nome: Ciência da Computação

Em outros cadastros, um curso deve ser referenciado pelo seu código, não podendo haver dois cursos com o mesmo código.

2.2. Cadastrar disciplina

Das disciplinas, deseja-se registrar um código (alfanumérico) e o nome. Todos os dados são obrigatórios.

Exemplo:

- Código: INF15933
- Nome: Programação Orientada a Objetos

Em outros cadastros, uma disciplina deve ser referenciada por seu código, não podendo haver duas disciplinas com o mesmo código.

2.3. Cadastrar prova

Das provas, deseja-se registrar em qual disciplina ela será aplicada, código (alfanumérico), nome, peso no cálculo da média (número real) e data que a prova será aplicada (formato dd/mm/aaaa). Com exceção do nome, todos os dados são obrigatórios (no caso de uma prova sem nome, usa-se apenas seu código na hora de imprimir seus dados).

Note que, para cadastrar uma prova é preciso antes cadastrar uma disciplina, que deve ser referenciada (localizada dentre as já cadastradas) da forma descrita na Seção 2.2, ou seja, pelo seu código. Para os próximos cadastros, considere situações análogas.

Exemplo:

- Disciplina: INF15933
- Código: Prog00-P1
- Nome: Prova Parcial 1
- Peso: 1.0
- Data: 25/04/2023

Em outros cadastros, uma prova deve ser referenciada pelo seu código, não podendo haver duas provas com o mesmo código.

2.4. Cadastrar aluno(a)

De um(a) aluno(a), deseja-se registrar sua matrícula (número inteiro), nome e o curso de graduação que ele faz na UFES.

Exemplo:

- Matrícula: 2000000005
- Nome: Arya Stark
- Curso: 11

Em outros cadastros, um(a) aluno(a) deve ser referenciado(a) pela sua matrícula, não podendo haver dois alunos com a mesma matrícula.

2.5. Matricular aluno(a) em disciplina

Para matricular um(a) aluno(a) em uma disciplina, basta indicar qual o(a) aluno(a) e qual a disciplina.

Exemplo:

- Aluno(a): 2000000005
- Disciplina: INF15933

2.6. Registrar nota de aluno em prova

Para registrar a nota de um(a) aluno(a) em uma prova, basta indicar qual o(a) aluno(a), qual a prova e qual a nota recebida (número real).

Exemplo:

- Aluno(a): 2000000005
- Prova: Prog00-P1
- Nota: 9.9

2.7. Imprimir dados

Ao ser solicitada a impressão dos dados, seu programa deve exibir o seguinte:

- Uma listagem de disciplinas e os alunos matriculados;
- Uma listagem de provas e as notas recebidas por cada aluno(a);

Apesar dos códigos e matrículas terem sido usados na entrada de dados, nas listagens devem ser usados os nomes dos cursos, disciplinas, alunos, provas (exceto se a prova não tem um nome, então usa-se o código), etc. Não há um formato padrão para as listagens, mas segue uma sugestão abaixo com base nos exemplos dados anteriormente:

Disciplinas e alunos matriculados:

- Programação Orientada a Objetos (INF15933)
 - Arya Stark (Ciência da Computação)



Universidade Federal do Espírito Santo

Centro Tecnológico

Departamento de Informática

Programação OO

(INF 15933) – 2023/1

Prof. Vítor E. Silva Souza

Provas e notas recebidas:

- Programação Orientada a Objetos – Prova Parcial 1 (25/04/2023)
- Arya Stark: 9.9

Note que nesta etapa estamos considerando apenas alunos de graduação e provas. Nas próximas etapas, passaremos a considerar alunos de pós-graduação, trabalhos e demais requisitos do sistema, como verificação de erros nos dados, produção de relatórios, etc.

Demais etapas ainda a serem detalhadas!

3. Observações finais

Quando for feito o detalhamento das demais etapas do trabalho e caso haja algum erro neste documento, serão publicadas novas versões e divulgadas erratas em sala de aula. É responsabilidade do aluno manter-se informado, frequentando as aulas ou acompanhando as novidades na página da disciplina na Internet.