



Disciplina Algoritmos e Estruturas de Dados II	Curso	Turmas	Período 2º
Professores Erickson Nascimento, William Schwartz			

Trabalho Prático 1 - Grade de Horários

Data de entrega: 16/04/2013

Valor: 10 pontos

O que deve ser feito

Um problema frequente na definição de uma grade de horário de aula é a marcação de aulas simultâneas (mesmo horário) para o mesmo professor, a alocação conflitante de espaço físico (mesma sala de aula para mais de uma disciplina no mesmo horário) e a divulgação da grade de horários. Sendo assim a secretaria acadêmica resolveu desenvolver um sistema informatizado para gerenciamento de grade de horários, e, para tanto, você foi designado para implementá-lo. A especificação do sistema segue abaixo:

Tipos Abstratos de Dados

Neste trabalho prático o aluno deverá implementar PELO MENOS os TADs definidos a seguir:

Horário

Para facilitar a comunicação e o lançamento dos horários de aula na grade, a secretaria acadêmica propôs uma codificação do horário de aulas a partir de um número inteiro de 3 dígitos, o qual utiliza uma organização interna que segue o formato DTO, a saber:

- D representa o dia da semana (1 = Domingo, 2 = Segunda-feira, ..., 7 = Sábado)
- T representa o turno (1 = Manhã, 2 = Tarde, 3 = Noite)
- O representa a ordem cronológica do horário da aula

Ex:

Horário 433 = quarta-feira, turno da noite, terceira aula.

Horário 225 = segunda-feira, turno da tarde, quinta aula.

Horário 711 = sábado, turno da manhã, primeira aula.

O número de ordem da aula será sempre um número inteiro positivo entre 1 e 9 (maior que 0 e menor que 10).

Considere a utilização de tipos enumeráveis (*enum*) para representar o atributo dia da semana e o turno.

Classe

Uma classe representa uma disciplina na qual o aluno se matricula e possui as seguintes informações:

- **período:** grau de ensino (período do curso) no qual a disciplina está sendo ofertada. Representado por um número inteiro entre 1 e 10 (maior que 0 e menor que 11).

- **disciplina:** nome da disciplina sendo ofertada. Representado por uma sigla de no mínimo 2 e no máximo 7 caracteres alfa-numéricos, onde o primeiro caractere é sempre um caractere literal entre A-Z. Além disso, todo caractere literal é representado em caixa alta (maiúsculo).
- **professor:** nome do professor responsável pela classe.
- **sala:** número do ambiente de aprendizagem (sala de aula, laboratório, gabinete, etc) onde as aulas são ministradas. Representado por um número inteiro entre 1000 e 9999 (sempre de 4 dígitos).
- **horários:** LISTA dos horários em que as aulas são ministradas.

O número de horários de aula é arbitrário, não havendo um limite máximo estabelecido. Entretanto, toda disciplina deverá possuir no mínimo 1 horário de aula previsto. Este é um atributo multivalorado, o qual deverá utilizar uma lista encadeada para armazenar seus diversos valores.

Grade

Todas as classes ofertadas são mantidas em uma grade de horários, a qual utiliza um vetor de tamanho `GRADE.INCREASE` para armazenar as classes ofertadas. À medida que não houver mais espaço disponível para armazenar novas classes, o vetor deve ser expandido em `GRADE.INCREASE` posições.

Ex: Suponha `GRADE.INCREASE=5` e que estão sendo ofertadas 7 classes. Como definido anteriormente, ao ser instanciada, a grade deve criar um vetor para armazenar 5 (`GRADE.INCREASE`) classes. Ao adicionar a 6ª classe na grade, o sistema deve expandir a capacidade do vetor de classes em `GRADE.INCREASE` posições, resultado em um vetor de 10 posições. Nessa grade, os elementos são sempre inseridos no final do vetor.

Entrada

A entrada do programa será feita através de um arquivo de entrada, passado como parâmetro durante a execução utilizando a flag abaixo:

$-[i|I]^1$: arquivo de entrada com a grade de horários.

O arquivo de entrada será organizado no seguinte formato:

`<periodo> <disciplina> <sala> "<professor>" [<horario>]+`

Ex:

```
1 AEDS1 2021 "Silvio Santos" 211 212 313 314 513 514
1 MD 2021 "Joao Bosco" 213 214 411 412
1 GAAL 2021 "Vinicius de Moraes" 311 312 511 512
1 CD2 2021 "Joao Gilberto" 413 414
1 ICC 2021 "Gilberto Gil" 611 612
1 CDI1 2021 "Caetano Veloso" 613 614
1 EF 2021 "Tom Jobin" 713 714
1 IFE 2021 "Toquinho" 624 625 626
2 FM 3012 "Wilson Simonal" 221 222 241 242
2 CDI2 3012 "Caetano Veloso" 223 224 423 424
2 ISL 3012 "Oscar Castro Neves" 225 226 425 426
2 ALGA 3012 "Roberto Menescal" 321 322 521 522
2 AEDS2 3012 "William Schwartz" 325 326 525 526
```

¹Os parâmetros podem ser informados com letras maiúsculas ou minúsculas

Saída

A saída do programa será feita através de um arquivo de saída, passado como parâmetro durante a execução utilizando a flag abaixo:

–[o|O]: arquivo de saída contendo o quadro de horários e as inconsistências.

Como saída, o sistema deverá produzir o quadro de horário de aulas no seguinte formato:

1o Período

Turno: Manhã

Horario	2a-feira	3a-feira	4a-feira	5a-feira	6a-feira	Sabado
1	AEDS1	GAAL	MD	GALL	ICC	
2	AEDS1	GALL	MD	GALL	ICC	
3	MD	AEDS1	CD2	AEDS1	CDI1	EF
4	MD	AEDS1	CD2	AEDS1	CDI1	EF

Turno: Tarde

Horario	2a-feira	3a-feira	4a-feira	5a-feira	6a-feira	Sabado
4					IFE	
5					IFE	
6					IFE	

2o Período

Turno: Tarde

Horario	2a-feira	3a-feira	4a-feira	5a-feira	6a-feira	Sabado
1	FM	ALGA	FM	ALGA		
2	FM	ALGA	FM	ALGA		
3	CDI2		CDI2			
4	CDI2		CDI2			
5	ISL	AEDS2	ISL	AEDS2		
6	ISL	AEDS2	ISL	AEDS2		

COLISOES DE HORARIOS

<periodo> <horario> [<disciplina>]+

COLISOES DE ESPACO FISICO

<sala> [<horario da(s) colisão(ões)>]+

COLISOES DE DOCENTE

<professor> [<horario da(s) colisão(ões)>]+

Ao final da impressão da grade de horários, deverão ser apresentadas as inconsistências na alocação de horários de aula, de espaço físico e docentes, conforme apresentado anteriormente.

O sistema deverá verificar se existem colisões de horários de aula, quando mais de uma disciplina de um determinado período estiver alocada para o mesmo horário. Nesse caso, no horário em que ocorre a colisão, deverá ser exibida a mensagem ERRO no lugar do nome da disciplina.

ATENÇÃO: no quadro de horários e na apresentação das inconsistências, os períodos, os turnos, os números de ordem das aulas, os horários de aula, os números das salas e o nome dos professores deverão ser gerados em ordem crescente. Utilize espaços em branco para organizar as informações. Observe a linha em branco separando os períodos. Ao final de cada linha, não deve restar espaço em branco. Ao final do arquivo de saída, deixe uma linha em branco.

O que deve ser entregue

- Código fonte do programa em C (todos os arquivos .c e .h), devidamente comentado e documentado.

- Documentação sobre o trabalho (máximo de 4 folhas em formato pdf), incluindo as decisões tomadas relativas aos casos e detalhes de especificação que porventura estejam omissos no enunciado. Um exemplo de documentação pode ser encontrado no Moodle. Porém, ela deve conter, entre outras coisas:
 1. Introdução: descrição sucinta do problema a ser resolvido e visão geral sobre o funcionamento do programa.
 2. Implementação: descrição sobre a implementação do programa. Deve ser detalhada a estrutura de dados utilizada (de preferência com diagramas ilustrativos), o funcionamento das principais funções e procedimentos utilizados, o formato de entrada e saída de dados, compilador utilizado, bem como decisões tomadas relativas aos casos e detalhes de especificação que porventura estejam omissos no enunciado.
 3. Estudo de Complexidade: estudo da complexidade do tempo de execução dos procedimentos implementados e do programa como um todo (notação O), considerando conjuntos de tamanho n .
 4. Testes: descrição dos testes realizados e listagem da saída (não edite os resultados).
 5. Conclusão: comentários gerais sobre o trabalho e as principais dificuldades encontradas em sua implementação.
 6. Bibliografia: bibliografia utilizada para o desenvolvimento do trabalho, incluindo sites da Internet se for o caso.
- O trabalho será entregue através do sistema de submissão de trabalhos práticos: <http://aeds.dcc.ufmg.br>, em um arquivo compactado (.zip) contendo os arquivos fonte. O a documentação deve ser enviada em pdf, no link específico para sua submissão no sistema.

Comentários gerais

- Comece a fazer este trabalho logo, enquanto o prazo para terminá-lo está tão longe quanto jamais poderá estar.
- Todos os TADs deverão ser sempre declarados como ponteiros e instanciados (criados) dinamicamente.
- O aluno deverá implementar um método que remove todos os elementos da memória, o qual deverá ser chamado antes do encerramento do programa.
- Técnica, clareza, legibilidade, organização, coesão, indentação, comentários, nomes adequados a variáveis, atributos e funções também vão valer pontos.
- Trabalhos copiados serão penalizados com a nota zero.
- O trabalho é individual.
- Penalização por atraso: $(2d - 1)$ pontos, onde d é o número de dias de atraso.