

## Trabalho de Programação I

Prof. Flávio Miguel Varejão

### I. Descrição do Problema

O reitor de uma universidade é uma pessoa muito ocupada, tendo frequentemente de participar de reuniões e seminários. Quando surge a necessidade de participar em um compromisso é necessário verificar as possíveis disponibilidades em sua agenda anual para que ele possa escolher qual período é mais adequado para atender a essa demanda. O objetivo deste trabalho é fazer um programa para gerenciar a agenda anual de compromissos do reitor.

O reitor da universidade é uma pessoa bem sistemática com seus horários de trabalho. Seus compromissos podem ser agendados nos dias de semana (segunda a sexta) em horário comercial (turno da manhã, de 8 as 12 horas, ou turno da tarde, de 14 as 18 horas). Ele não trabalha nos finais de semana e feriados nem fora do horário comercial. Além disso, seus compromissos sempre se iniciam em um horário inteiro (por exemplo, é possível marcar compromissos para as 8 horas, mas não é possível marcar compromissos as 8:15 horas ou 8:30 horas) e a duração dos mesmos é sempre um múltiplo de 1 hora (por exemplo podem durar 2 horas, mas não podem durar 2,5 horas). Se um compromisso se estende além do final de um turno (por exemplo, se o compromisso começa as 11 horas e tem duração de 3 horas, ele ocupará uma hora do turno da manhã e duas horas do turno da tarde, se encerrando, portanto, às 16 horas).

Gerenciar a agenda do reitor consiste em realizar operações de consulta, inserção, cancelamento e reagendamento de compromissos. Este trabalho será dividido em duas etapas. A primeira etapa compreende a implementação de funções para a realização das operações e a segunda etapa envolve a criação de um programa interativo que permita usar essas funções para gerenciar a agenda. Este enunciado se refere apenas a primeira etapa do trabalho.

### II. Especificação das Estruturas de Dados a serem utilizadas

Um compromisso deve ser representado por um tupla com dois inteiros. O primeiro inteiro indica o horário de início do compromisso e o segundo inteiro representa a sua duração. Por exemplo, a tupla (9, 2) indica um compromisso que começa as 9 horas e tem 2 horas de duração.

Os compromissos de um dia são representados por uma tupla contendo o número do dia e uma lista dos compromissos agendados naquele dia. Por exemplo, a tupla (12, [(9,2), (14, 1), (16,2)]) indica que no dia 12 estão agendados os seguintes compromissos: início às 9 horas com 2 horas de duração, início às 14 horas com 1 hora de duração e início às 16 horas com 2 horas de duração.

Os compromissos de um mês são representados por uma tupla composta pelo número do mês e uma lista dos compromissos agendados nos dias daquele mês. Por exemplo, a tupla (3, [(2, [(8,1), (10, 1), (15,2)]), (5, [(9,2), (14, 1)])]) representa que no mês 3 (março) tem três compromissos marcados no dia 2 (começando as 8, 10 e 15 horas com durações respectivas de 1 hora, 1 hora e 2 horas) e dois compromissos no dia 5,

um começando as 9 com 2 horas de duração e outro começando as 14 horas com 1 hora de duração.

A agenda anual consiste de uma lista dos compromissos de cada mês. É importante destacar que os compromissos se encontram ordenados por mês, dia e hora de início. Assim, na agenda anual são colocados primeiro os compromissos de janeiro, depois os de fevereiro e assim por diante. Analogamente, os compromissos de um mesmo mês são ordenados pela data. Dessa forma, os compromissos do dia 1 vem antes dos do dia 2 e assim por diante. O mesmo ocorre com relação aos compromissos de um mesmo dia. Primeiro são colocados os compromissos que começam às 8 horas, depois os que começam às 9 e assim por diante. Outro fato a se destacar é que NUNCA haverá interseção de horários de compromissos. Se um compromisso ocorre em um determinado horário, NENHUM outro compromisso poderá estar agendado ocupando este mesmo horário.

Outra estrutura de dados que deve ser utilizada traz informações úteis do calendário do ano. Ela indica se o ano é bissexto (neste caso, o mês de fevereiro tem 29 dias e não 28 dias) e as datas de dias não úteis (finais de semana e feriado). Essa estrutura é uma tupla composta por um booleano (com valor True se o ano é bissexto e False, caso contrário) e por uma lista de tuplas indicando o mês e dia dos dias que NÃO são úteis. Por exemplo, a tupla (True, [(1,1), (1,6), (1,7), (2,3), (2,4)]) indica que o ano é bissexto e que os dias 1, 6 e 7 de janeiro e 3 e 4 de fevereiro não são dias úteis. Cabe ressaltar que a lista de dias não úteis também estará ordenada, isto é, primeiro são listados os dias não úteis do mês de janeiro, depois os de fevereiro e assim por diante. Do mesmo modo, os dias não úteis de um mesmo mês também estão dispostos na lista em ordem crescente.

### **III. Especificação das Operações a serem implementadas**

#### **1. Inserção de compromisso**

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês, Dia, Horário de início do compromisso, Duração do compromisso.

*Retorno:* Agenda atualizada.

*Observação:* Se o compromisso a ser inserido conflitar com compromisso existente na agenda, ela deve ser retornada sem modificação.

#### **2. Cancelamento de compromisso**

*Parâmetros de Entrada:* Agenda anual, Mês, Dia e Horário de início do compromisso.

*Retorno:* Agenda atualizada

*Observação:* Se o compromisso a ser cancelado não existir na agenda, ela deve ser retornada sem modificação.

#### **3. Reagendamento de compromisso**

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês, Dia, Horário de início do compromisso a ser reagendado, Novo Mês, Novo Dia, Novo Horário e Nova Duração do compromisso.

*Retorno:* Agenda atualizada.

*Observação:* Se o compromisso não existir na agenda, ele deve ser agendado no ponto solicitado. Se o compromisso reagendado conflitar com outro compromisso existente na agenda, ela deve ser retornada sem modificação.

#### 4. Verificação de disponibilidade

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês, Dia e Horário de início, Duração.

*Retorno:* Booleano indicando True se há disponibilidade e False em caso contrário.

*Observação:* Estar disponível implica não haver nenhum outro compromisso agendado ao longo do período verificado. Só horários em dias úteis podem estar disponíveis.

#### 5. Horários disponíveis em um dia

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês e Dia.

*Retorno:* Lista de horários disponíveis no dia especificado.

*Observação:* A lista de horários disponíveis deve conter as horas que não estão ocupadas por algum compromisso e deve estar ordenada. Caso não seja dia útil deve retornar uma lista vazia.

#### 6. Inserção de compromisso o mais breve possível

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês, Dia e Duração do compromisso.

*Retorno:* Agenda atualizada.

*Observação:* O compromisso deve ser inserido no primeiro horário no qual cabe sua duração a partir do dia especificado (inclusive o próprio dia). Caso não haja horário disponível, a agenda deve ser retornada inalterada.

#### 7. Inserção de compromisso no intervalo mínimo

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês e Duração do compromisso.

*Retorno:* Agenda atualizada.

*Observação:* O compromisso deve ser inserido no intervalo mínimo (menor intervalo) disponível que cabe sua duração no mês especificado. Caso haja mais de um intervalo mínimo, o compromisso deve ser alocado no primeiro horário disponível. Se não houver horário disponível, a agenda deve ser retornada inalterada.

#### 8. Inserção de compromisso no intervalo máximo

*Parâmetros de Entrada:* Agenda anual, Informações do calendário, Mês e Duração do compromisso.

*Retorno:* Agenda atualizada.

*Observação:* O compromisso deve ser inserido no intervalo máximo (maior intervalo) disponível que cabe sua duração no mês especificado. Caso haja mais de um intervalo máximo, o compromisso deve ser alocado no primeiro horário disponível. Se não houver horário disponível, a agenda deve ser retornada inalterada.

### **IV. Segunda Etapa do Trabalho**

A segunda etapa do trabalho consiste na criação de um programa interativo que possibilite a um usuário gerenciar a agenda do reitor. Além das operações listadas na seção III, o usuário deve poder recuperar e gravar a agenda do reitor em um arquivo chamado “agenda.txt”. Note que, toda vez que o programa for executado, antes mesmo do usuário escolher uma opção, o primeiro passo será ler o estado atual da agenda no arquivo “agenda.txt”. Isto quer dizer que o usuário não precisa selecionar a operação de recuperar agenda logo que se inicia o programa, uma vez que a agenda já

foi lida automaticamente no início da execução. É importante destacar que o arquivo “agenda.txt” pode estar vazio indicando que não existe nenhum compromisso na agenda. Atenção: não é preciso tratar a situação de inexistência do arquivo!

As opções do programa devem ser:

- 0 – Sair
- 1 - Recuperar agenda
- 2 – Verificar disponibilidade de horário
- 3 – Verificar disponibilidade no dia
- 4 – Inserir compromisso no horário
- 5 – Inserir compromisso mais breve
- 6 – Inserir compromisso no intervalo mínimo
- 7 – Inserir compromisso no intervalo máximo
- 8 – Cancelar compromisso
- 9 – Reagendar compromisso
- 10 – Gravar agenda

O formato do arquivo “agenda.txt” deve ser organizado conforme a seguinte especificação:

```
<mês>
<dia>
<lista de compromissos no dia>
<dia>
<lista de compromissos no dia>
...
```

```
<mes>
<dia>
<lista de compromissos no dia>
<dia>
<lista de compromissos no dia>
...
```

Cada item da especificação deve ser posicionado em uma única linha. Os dados de um mês são separados dos dados do mês seguinte por uma linha em branco. Os compromissos devem estar armazenados no arquivo em ordem cronológica. A seguir é apresentado um exemplo do arquivo “agenda.txt”

```
1
5
[(8, 4), (14:4)]
15
[(9, 2), (11,1), (17,1)]

2
7
[(9, 2)]
```

6

1

[(9, 1), (10,2), (14,1)]

A estrutura de dados com informações úteis do calendário anual deve ser lida no início do programa do arquivo “calendario.txt”. O formato do arquivo “calendario.txt” deve ser organizado conforme a seguinte especificação:

<booleano indicando se o ano é bissexto (True) ou não (False)>

<dias não úteis do primeiro mês separados por espaço>

<dias não úteis do segundo mês separados por espaço>

...

<dias não úteis do décimo segundo mês separados por espaço>

A seguir é apresentado um exemplo do arquivo “calendario.txt”

False

1 6 7 13 14 20 21 27 28

3 4 5 6 10 11 17 18 24 25

3 4 10 11 17 18 24 25

7 8 14 15 21 22 28 29

1 6 7 13 14 20 21 27 28 31

3 4 10 11 15 16 17 18 24 25

1 2 8 9 15 16 22 23 29 30

5 6 12 13 14 15 19 20 26 27

2 3 7 8 9 10 16 17 23 24

7 8 12 13 14 15 21 22 28 29

2 3 4 5 11 12 18 19 25 26

2 3 9 10 16 17 23 24 25 31

## V. Condições de Entrega

O trabalho deve ser feito individualmente e submetido por e-mail até as 23:59 horas de 20 de maio de 2018 para o endereço [josiasalexandre@gmail.com](mailto:josiasalexandre@gmail.com) com o subject PROG1\_TRABALHO\_X\_Nome\_Completo\_Aluno. O e-mail deve conter um arquivo .zip com o mesmo nome do subject do e-mail enviado contendo apenas o código fonte do trabalho. O arquivo principal (o que contém o main do trabalho) obrigatoriamente deve estar com o nome “main.hs”. Note que a data limite já leva em conta um dia adicional de tolerância para o caso de problemas de submissão via rede. Isso significa que o aluno deve submeter seu trabalho até no máximo um dia antes da data limite. Se o aluno resolver submeter o trabalho na data limite, estará fazendo isso assumindo o risco do trabalho ser cadastrado no sistema após o prazo. Em caso de recebimento do trabalho após a data limite, o trabalho não será avaliado e a nota será ZERO. Aluno que receber zero por este motivo e vier pedir para o professor considerar o trabalho estará cometendo um ato de DESRESPEITO ao professor e estará sujeito a perda adicional de pontos na média.

## VI. Avaliação

Os trabalhos terão nota zero se:

1. A data de entrega for fora do prazo estabelecido;

2. For detectada a ocorrência de plágio.

**Observação importante**

**Caso haja algum erro neste documento, serão publicadas novas versões e divulgadas erratas em sala de aula. É responsabilidade do aluno manter-se informado, freqüentando as aulas ou acompanhando as novidades na página da disciplina na Internet.**