INF 213 - Roteiro da Aula Pratica 1

Considerando o diagrama de classes do sistema *Agenda* dado no link *https://drive.google.com/open?id=1pgJNRmjGPV0Ne9v6_xgr1OR_r3c0A9Nk* , implemente a estrutura de classes em C++ correspondente a este diagrama, isto é, você deve implementar as interfaces das classes e as declarações dos atributos, o que deve ser feito criando os arquivos **.h* referentes a cada uma das classes e, nos arquivos *.cpp correspondentes, você deve implementar os métodos de cada classe.

Para facilitar o reaproveitamento de código, cada classe deve ser declarada e definida em arquivos separados, sendo a declaração feita em um arquivo *<nome_da_classe>.h* e a definição (dos métodos) no arquivo *<nome_da_classe>.cpp.* Assim, vocês devem criar 6 arquivos *Data.h, Data.cpp, Horario.h, Horario.cpp, Agenda.h* e *Agenda.cpp*. O arquivo *Agenda.h* deve conter a declaração das classes *ItemAgenda* e *Agenda* e a definição (implementação) dos métodos destas duas classes deve ser feita no arquivo *Agenda.cpp*. Observe que a inclusão destas duas classes em um mesmo arquivo não viola o conceito de separação de código porque a classe *Agenda* é uma composição de um ou mais objetos da classe *ItemAgenda* (no caso, a "parte") que formam a classe *Agenda* (no caso, o "todo").

Pergunta (discuta a resposta com o professor): em que Composicao (losango preto no diagrama de classes) e Agregacao (losango branco) afeta o desenvolvimento desse programa (que diferença faz)?

Após implementadas as classes, teste a sua implementação executando o programa *TestaAgenda.cpp* (*https://drive.google.com/open?id=1Wf7E49AnR5AMKyE1Wh2e8N1eyUXDFUfc*). Ao compilar suas classes com esse programa, e' possível utilizar arquivos de teste para testar as funcionalidades de insercao e consulta de items na agenda.

Considere, por simplicidade, que os dados não serao formatados (exemplo: imprimir a data 12/02/2018 na tela deveria gerar a saida 12 2 2018 em vez de 12/02/2018).

Exemplo de entrada/saida:

Entrada: 2 1 Estudar portugues 25 5 2020 2 47 59 Estudar C++ 25 5 2020 10 47 59 25 5 2020	Saida esperada: 25 5 2020 2 47 59 Estudar portugues 10 47 59 Estudar C++

Submissao da aula pratica:
A solucao deve ser submetida ate as 18 horas da proxima Segunda-Feira utilizando o sistema submitty (submitty.dpi.ufv.br).

Obs 1: a presenca nas aulas praticas e' obrigatoria (alunos ausentes que enviarem trabalho terao nota 0 na pratica correspondente)

Obs 2: O arquivo TestaAgenda.cpp NAO deverá ser modificado pelos alunos.

Obs 3: A versao corrigida dos trabalhos/praticas é sempre a ultima submetida (a não ser que você altere qual versao é a ativa).

Obs 4: As notas de aulas (praticas) e trabalhos praticos serao disponibilizadas no sistema Submitty.

OUTRAS OBSERVAÇÕES:

1. Você pode supor que os itens da agenda serão armazenados em um vetor de tamanho 1000. Isto é, suponha que a agenda é um vetor de tamanho 1000. Note que além de declarar esse vetor na classe *Agenda,* você deve incluir também um contador para indicar quantos itens existem efetivamente nesse vetor.

2. Evite codigo duplicado. Onde (nesta aula) você pode evitar duplicação de código?

3. Marque como const os metodos que não modificam o estado dos objetos (exemplo: métodos *get*). Caso contrario, tais métodos não poderao ser chamados em objetos constantes (e nem por outros métodos constantes).

4. Use referencias para evitar copias desnecessarias de objetos (o uso correto de const e referencias e' muito importante e sera cobrado em provas e trabalhos). Referencias não sao obrigatorias (na verdade, devem ser evitadas nesses casos) para a passagem de tipos "pequenos" (int, float, double, char, bool, ponteiros -- mesmo ponteiros para objetos grandes), visto que a copia de tais tipos e' uma operacao muito eficiente. **Peça ao professor para conferir o uso de const/referencia em seu codigo antes de submete-lo.**

5. Ha um limite de 20 submissoes para o servidor (por estudante), sendo que apenas a submissao ativa (normalmente a ultima) sera avaliada. Cada submissao adicional acarreta uma penalidade de 10% na nota final do laboratorio/trabalho → teste bastante os trabalhos/aulas praticas no seu computador antes de submete-lo pelo sistema.