



Instruções:

- LEIA TODO O DOCUMENTO ANTES DE RESPONDER À AVALIAÇÃO.
- A avaliação ocorre em modelo de atividade assíncrona.
- O prazo para o término da avaliação é de 200 (duzentos) minutos, ou 3 (três) horas e 20 (vinte) minutos. Haverá uma pequena tolerância de 10 minutos para a entrega. Respostas não entregues dentro deste prazo serão desconsideradas.
- As respostas devem ser entregues pelo aluno em um único arquivo programado e compilável, intitulado **"ESCREVA_O_SEU_NOME.cpp"**. A não entrega em arquivo ".cpp" ou a não compilação do arquivo acarretará na não correção por parte do professor e, consequentemente, na atribuição de nota 0 (zero).
- Após a entrega das respostas por parte do aluno, o professor tem a prerrogativa de arguir o aluno a respeito da solução encontrada para as suas respostas, em modelo de atividade síncrona.
- Não serão aceitos comandos com funções nem operadores ainda não estudados no Curso.

Descrição:

Nesta avaliação, vamos supor que você precisa escrever um programa a ser compartilhado por uma equipe de programadores na linguagem C++ e que seja capaz de manipular datas. Como não existe um tipo de dado nativo nessa linguagem de programação para representar datas, utilize três inteiros para constituírem o dia, o mês e o ano de uma data.

Diante da necessidade de compartilhamento do código, você precisa escrever cada função com o seu tipo de retorno, com o seu identificador e com os tipos dos seus parâmetros exatamente conforme pedido a você. Para cada nova função, você pode – e deve – chamar outra ou outras funções já programadas anteriormente nesta avaliação. De tal maneira, escreva:

- 1) **(5,0 pontos)** Uma função para receber duas datas. Esta função deve retornar:

- -1, se a primeira data for anterior à segunda data;
- 0, se ambas as datas forem iguais;
- +1, se a primeira data for maior que a segunda data.

Tipo de retorno: int
Nome da função: comparaDatas
Tipo do primeiro parâmetro: int, para representar o **dia** da **primeira** data
Tipo do segundo parâmetro: int, para representar o **mês** da **primeira** data
Tipo do terceiro parâmetro: int, para representar o **ano** da **primeira** data
Tipo do quarto parâmetro: int, para representar o **dia** da **segunda** data
Tipo do quinto parâmetro: int, para representar o **mês** da **segunda** data
Tipo do sexto parâmetro: int, para representar o **ano** da **segunda** data

- 2) **(5,0 pontos)** Outra função para receber duas datas. A primeira data representa o nascimento de alguma pessoa. A segunda data representa a data atual. Essa função deve retornar a idade, em anos, da pessoa nascida na data representada pelo primeiro parâmetro da função. Se, por acaso, a primeira data for posterior à segunda data, a função deve retornar -1, indicando que a pessoa ainda não nasceu.

Tipo de retorno: int
Nome da função: idade
Tipo do primeiro parâmetro: int, para representar o **dia** do **nascimento** da pessoa
Tipo do segundo parâmetro: int, para representar o **mês** do **nascimento** da pessoa
Tipo do terceiro parâmetro: int, para representar o **ano** do **nascimento** da pessoa
Tipo do quarto parâmetro: int, para representar o **dia** da **data atual**
Tipo do quinto parâmetro: int, para representar o **mês** da **data atual**
Tipo do sexto parâmetro: int, para representar o **ano** da **data atual**

Atenção com os testes que você deve fazer para verificar se sua função está escrita corretamente. O ano da primeira data pode ser maior, igual ou menor em comparação ao ano da segunda data. São 3 (três) possibilidades. O mês da primeira data pode ser maior, igual ou menor em comparação ao mês da segunda data. São mais 3 (três) possibilidades. Finalmente,



englobando mais uma vez, outras 3 (três) possibilidades, o dia da primeira data pode ser maior, igual ou menor em comparação ao dia da segunda data.

Portanto, é possível utilizar 27 (vinte e sete) combinações de datas diferentes para testar cada uma das funções a serem programadas para responder à essa avaliação.

Você não precisa entregar a função main programada, mas é altamente recomendado que você a programe para testar as funções implementadas. Seu código será avaliado da seguinte maneira: suas funções serão usadas por um programa já existente. Mais uma vez, é preciso que suas funções sejam compiladas. Para cada uma das 27 (vinte e sete) combinações possíveis de datas diferentes para as quais a sua função executar corretamente, será acrescentada a respectiva pontuação à nota da sua avaliação. Segue imagem de uma execução do programa teste:

PRIMEIRA DATA		SEGUNDA DATA		COMPARACAO	IDADE
13/8/2020	X	14/9/2021	-->	-1	1 ano(s).
14/8/2020	X	14/9/2021	-->	-1	1 ano(s).
15/8/2020	X	14/9/2021	-->	-1	1 ano(s).
13/9/2020	X	14/9/2021	-->	-1	1 ano(s).
14/9/2020	X	14/9/2021	-->	-1	1 ano(s).
15/9/2020	X	14/9/2021	-->	-1	0 ano(s).
13/10/2020	X	14/9/2021	-->	-1	0 ano(s).
14/10/2020	X	14/9/2021	-->	-1	0 ano(s).
15/10/2020	X	14/9/2021	-->	-1	0 ano(s).
13/8/2021	X	14/9/2021	-->	-1	0 ano(s).
14/8/2021	X	14/9/2021	-->	-1	0 ano(s).
15/8/2021	X	14/9/2021	-->	-1	0 ano(s).
13/9/2021	X	14/9/2021	-->	-1	0 ano(s).
14/9/2021	X	14/9/2021	-->	0	0 ano(s).
15/9/2021	X	14/9/2021	-->	1	-1 ano(s).
13/10/2021	X	14/9/2021	-->	1	-1 ano(s).
14/10/2021	X	14/9/2021	-->	1	-1 ano(s).
15/10/2021	X	14/9/2021	-->	1	-1 ano(s).
13/8/2022	X	14/9/2021	-->	1	-1 ano(s).
14/8/2022	X	14/9/2021	-->	1	-1 ano(s).
15/8/2022	X	14/9/2021	-->	1	-1 ano(s).
13/9/2022	X	14/9/2021	-->	1	-1 ano(s).
14/9/2022	X	14/9/2021	-->	1	-1 ano(s).
15/9/2022	X	14/9/2021	-->	1	-1 ano(s).
13/10/2022	X	14/9/2021	-->	1	-1 ano(s).
14/10/2022	X	14/9/2021	-->	1	-1 ano(s).
15/10/2022	X	14/9/2021	-->	1	-1 ano(s).

Boa avaliação.