

# Exercícios de fixação 02 - Arquivos de dados estruturados

- Entrega 25 ago em 23:59
- Pontos 1
- Perguntas 4
- Limite de tempo Nenhum

## Instruções

Este questionário contém questões sobre os tipos de dados usados em arquivos, sobre codificação de caracteres e sobre registros de tamanho fixo e variável.

## Histórico de tentativas

	Tentativa	Tempo	Pontuação
MAIS RECENTE	<a href="#">Tentativa 1</a>	3 minutos	0,88 de 1
Pontuação deste teste: 0,88 de 1			
Enviado 25 ago em 21:07			
Esta tentativa levou 3 minutos.			
⋮			
Pergunta 1			
0,25 / 0,25 pts			
Qual das seguintes formas é a mais adequada para o armazenamento em arquivo de um atributo data-hora, isto é, de um atributo que armazene um momento específico de uma data específica (ex.: 20/08/2024 19:30). Considere, na sua resposta, que o sistema em que essa data será armazenada deverá ser capaz de indicar quanto tempo já se passou desde esse momento específico.			
<input type="radio"/> Cinco atributos <i>int</i> (para dia, mês, ano, horas e minutos).			
<input type="radio"/> Um atributo <i>int</i> (para os dias passados desde uma data inicial) e outro atributo <i>float</i> (para a fração de dia que representa as horas e minutos).			
Correto!			
<input checked="" type="radio"/> Um atributo <i>long</i> (representando os milissegundos passados desde uma data inicial).			
<input type="radio"/> Uma <i>string</i> em que a data e a hora serão armazenados na forma de caracteres.			
A melhor alternativa entre essas é aquela que usa apenas um atributo e que permite operações matemáticas, já que se espera determinar quanto tempo se passou desde a data. Assim, a melhor alternativa é converter todos os valores (datas e horas) para milissegundos passados desde uma data inicial (dia zero). Obviamente, serão necessárias funções de conversão entre esse tipo <i>long</i> e			

um tipo *Date* que esteja disponível na linguagem de programação. Essa conversão seria realizada nas interações com o usuário.



Pergunta 2

0,25 / 0,25 pts

Por que atributos como CPF e CNPJ devem ser armazenados em arquivos como *strings* ao invés de tipos numéricos?



Porque não há tipos numéricos que comportem todos os dígitos desses atributos e os valores acabariam sendo cortados.



Porque apenas ao serem armazenados como strings, seria possível ordená-los sequencialmente.



Porque esses atributos podem conter caracteres entre os dígitos, como pontos, traços e barras, que seriam perdidos em tipos numéricos.

Correto!



Porque esses atributos não são valores usados em operações matemáticas e os seus dígitos precisam ser separados para verificação.

Valores como CPF e CNPJ, apesar de serem representados apenas com dígitos, não devem ser armazenados como números, porque não são usados em operações matemáticas. Na verdade, seus dígitos devem ser separados para a única operação frequente: a verificação do(s) dígito(s) verificador(es). Para isso, é mais fácil extrair os dígitos de uma *string* do que de um tipo numérico.

Apesar de usarmos vários separadores nesses atributos, eles não precisam ser inseridos nos valores. Eles podem ser usados apenas nas máscaras (de edição ou de apresentação).

Também é importante lembrar que, nesses atributos, os zeros à esquerda são significativos.



Pergunta 3

0,13 / 0,25 pts

Indique quais são os tipos de dados adequados para cada um dos campos de um registro Cliente.

Você respondeu

Data de nascimento (sem horas)

long

▼

int

Você respondeu

Número de filhos

int

byte

Correto!

CEP

string

Correto!

Renda pessoal (até 100 mil reais)

double

Outras opções de respostas incorretas:

- long
- char

O atributo `Renda pessoal` é um valor monetário, com duas casas decimais. Assim, o tipo adequado para esse atributo é o `float`. O CEP é um campo baseado em dígitos, mas que não é usado em operações aritméticas. Assim, pode ser armazenado como `string`. O atributo número de dependentes conterá sempre um valor pequeno, menor que 128. Portanto, pode ser do tipo `byte`. A `Data de nascimento` pode ser armazenado como um `int`, permitindo que se calcule, com facilidade, a idade atual do cliente.



Pergunta 4

0,25 / 0,25 pts

Campos com múltiplos valores são adequados quando o número de valores do atributo é variável. Por exemplo, uma pessoa pode ter 1, 2, 3 ou mais números de telefone, emails, etc. Nesses casos, qual é a forma adequada para se registrar esses valores em um registro de tamanho variável?

☐ Usar também o delimitador de campos para separar cada valor desse campo específico.

Correto!

☒ Escrever um indicador da quantidade de valores no início do campo e, em seguida, cada um dos valores.



Pré-determinar a quantidade máxima de valores e escrever aqueles não usados como espaços em branco no arquivo.



Reservar uma quantidade de *bytes* no arquivo suficiente para conter vários valores e preencher os *bytes* que sobram com espaços em branco.

A solução correta para campos com múltiplos valores é escrever um único *byte* inicial que contenha a quantidade de valores do campo. Dessa forma, é possível se ler cada um dos valores desse campo, sem confundi-los com os valores de outros campos.

Pontuação do teste: 0,88 de 1