

Primeiro Exercício-Programa

Prof. Luciano Antonio Digiampietri

Prazo máximo para a entrega: 23/05/2021

1 Caixa Eletrônico

Os caixas eletrônicos das Ilhas Weblands operam com todos os tipos de notas disponíveis, mantendo um estoque de cédulas para cada valor (B\$ 50,00, B\$20,00, B\$5,00 e B\$1,00). Os clientes do banco utilizam os caixas eletrônicos para efetuar retiradas de um certo número inteiro de Bits.

Sua tarefa é escrever uma função que, dado o valor de Bits desejado pelo cliente (passado por parâmetro), determine o número de cada uma das notas necessário para totalizar esse valor, de modo a minimizar a quantidade de cédulas entregues. Por exemplo, se o cliente deseja retirar B\$50,00, basta entregar uma única nota de cinquenta Bits. Se o cliente deseja retirar B\$72,00, é necessário entregar uma nota de B\$50,00, uma de B\$20,00 e duas de B\$1,00.

Sua função deve necessariamente ter a seguinte assinatura:

```
void fazRetirada(int valor)
```

Para esse exercício, não é necessário controlar o estoque de notas. Ou seja, você pode supor que o caixa sempre tem notas suficientes para efetuar o pagamento ótimo.

1.1 Entrada

A entrada é composta pelo parâmetro da função, que recebe um número inteiro positivo **valor** (≥ 0), indicando o valor solicitado pelo cliente.

1.2 Saída

Como saída, sua função deve abastecer quatro variáveis inteiras com escopo global (visíveis dentro de todo o programa), **n50**, **n20**, **n5** e **n1** indicando, respectivamente, o número de cédulas de B\$50,00, de B\$20,00, de B\$5,00 e de B\$1,00. Por exemplo, se a retirada for de B\$27,00, **n50** conterà zero, **n20** conterà 1, **n5** conterà 1 e **n1** conterà 2.

Para que você teste o funcionamento da função **fazRetirada()**, você pode implementar testes a partir da função **main()** e, dentro da função **main()**, escrever na tela o valor e a quantidade de notas dispensadas, como ilustrado abaixo (isso apenas para que você teste. Não use **printf()** diretamente em **fazRetirada()**):

Valor: 138
Notas de 50: 2
Notas de 20: 1
Notas de 5: 3
Notas de 1: 3

Em caso de erro (ou seja, de *valor* ≤ 0), cada uma das variáveis `n50`, `n10`, `n5` e `n1` deve ser abastecida com -1, como ilustrado abaixo:

Valor: -21
Notas de 50: -1
Notas de 20: -1
Notas de 5: -1
Notas de 1: -1

1.3 Material a Ser Entregue

Um arquivo, denominado *NUSP.c* (sendo NUSP o seu número USP, por exemplo: 123456789.c), contendo a função `fazRetirada` e qualquer outra função que ache necessário. Para sua conveniência, `completeERenomeie.c` será fornecido, cabendo a você então completá-lo e renomeá-lo para a submissão.

Atenção!

1. Não modifique a assinatura de `fazRetirada()` ou o nome das variáveis de escopo global (visíveis em todo o programa)!
2. Para avaliação, apenas a função `fazRetirada()` será invocada diretamente. Em especial, qualquer código dentro da função `main()` será ignorado. Então certifique-se de que o problema seja resolvido chamando-se diretamente somente essa função.

2 Entrega

A entrega será feita única e exclusivamente via sistema e-Disciplinas, até a data final marcada. Deverá ser postado no sistema um arquivo `c`, tendo como nome seu número USP:

`seuNumeroUSP.c` (por exemplo, `12345678.c`)

Não esqueça de preencher o cabeçalho constante do arquivo `.c`, com seu nome, número USP e turma etc.

A responsabilidade de postagem é exclusivamente sua. Por isso, submeta e certifique-se de que o arquivo submetido é o correto (fazendo seu download, por exemplo). Problemas referentes ao uso do sistema devem ser resolvidos com antecedência.

3 Avaliação

A nota atribuída ao EP terá como foco principal a funcionalidade solicitada, porém não esqueça de se atentar aos seguintes aspectos:

1. Documentação: se há comentários explicando o que se faz nos passos mais importantes e para que serve o programa (Tanto a função quanto o programa em que está inserido);
2. Apresentação visual: se o código está legível, indentado etc;
3. Corretude: se o programa funciona.

Além disso, algumas observações pertinentes ao trabalho, que influem em sua nota, são:

- Este exercício-programa deve ser elaborado individualmente;
- Não será tolerado plágio, em hipótese alguma;
- Exercícios com erro de sintaxe (ou seja, erros de compilação), receberão nota ZERO.