

MC102W - Algoritmos e Programação de Computadores

Lab01: Mundo Invertido

Prazo: 05 de Abril de 2020

Peso da Atividade: 1

O ano é 1983. Você está andando por uma rua deserta durante a noite e encontra um monstro gigante. O monstro te captura e aplica uma substância que o faz dormir instantaneamente. Ele então leva você para um lugar desconhecido e o abandona. Ao acordar, você percebe estar em um mundo estranhamente familiar e ao mesmo tempo invertido, uma realidade alternativa. Como todos sabem, trocas de dimensões são extremamente exaustivas, mesmo quando se está desacordado. Por esse motivo, você está faminto. Você decide procurar alimento na cidade invertida e encontra algo similar à um mercado, o Tramlaw. No anúncio do mercado diz “Tramlaw, qualquer produto custa menos que 100 laer’s”. Apesar de não conhecer tal moeda, você decide entrar para comprar algo. Por sorte, sua mãe havia lhe dado 100 reais na manhã anterior e ainda estavam na sua carteira. Você então escolhe um pacote de waffles Oge que custa 1,99 laer’s. No caixa, você entrega uma nota de 5 reais, o caixa rapidamente converte o valor do produto para real e lhe informa que não é o suficiente para pagar o produto. Então, você entrega todo seu dinheiro e deixa que o caixa decida quanto deve devolver de troco. Você sente um alívio ao perceber que suas notas valem no tal mercado, porém o caixa retorna apenas 90 centavos de troco. Você fica revoltado, mas já está acostumado com a desvalorização do real. Enquanto ainda está no caixa, você observa um outro consumidor, nativo desta realidade paralela, comprando um produto de 50 laer’s e pagando apenas 5 centavos. Neste momento, você entende a lógica dos preços no mundo invertido: os valores devem ser lidos ao contrário. Para facilitar suas próximas compras, afinal você agora tem apenas 90 centavos, você decide implementar um programa que faz a conversão automática dos preços.

Tarefa

Nesta tarefa, você deverá ler o valor do produto no formato AB.CD e realizar a inversão dos dígitos, gerando como saída o valor convertido R\$ DC.BA.

O programa deve ser em python e deve abranger os seguintes tópicos:

- ☐ Variáveis e tipos básicos;
- ☐ Operações aritméticas;
- ☐ Entrada e saídas de dados.

Entrada

A entrada consiste de um único número com **no máximo** 4 dígitos AB.CD (lembre-se que no Tramlaw os produtos custam no máximo 100 laer's) onde:

A: dezena, B: unidade, C: décimo, D: centésimo.

Os dígitos não informados devem ser considerados nulos.

Por exemplo, para 2.3 A=0, B=2, C=3, D=0.

Saída

O seu programa deve gerar como saída um número com **exatamente** 4 dígitos R\$ DC.BA, representando o valor do produto em reais. Os dígitos nulos também devem ser mostrados na saída.

Exemplos

A grafia da saída abaixo deve ser seguida rigorosamente por seu programa, inclusive a impressão do símbolo monetário e o espaço entre ele e o valor. Note também que não há espaço após o número (isso é importante para o SuSy aceitar a sua resposta como correta).

Exemplo 1:

Entrada

12.34

Saída

R\$ 43.21

Exemplo 2:

Entrada

1.99

Saída

R\$ 99.10

Exemplo 3:

Entrada

50

Saída

R\$ 00.05

Nota

Por causa da representação conforme normas do IEEE, os números reais do tipo *float* geram imprecisões, como pode ser observado no trecho de código abaixo:

Código

```
a = 0.1
print("{:.55f}".format(a))
```

Saída

0.10000000000000000055511151231257827021181583404541015625

Observe que a impressão do *float* 0.1 gerou uma sequência não esperada nos dígitos menos significativos. Em algumas aplicações, esse comportamento não afeta o resultado. Porém, para a tarefa atual a imprecisão pode gerar uma saída diferente da esperada. Para uma representação mais precisa do número real, basta utilizar o tipo *Decimal*. Sendo assim, utilize:

1. Importação única - Você deve adicionar o seguinte comando na primeira linha do seu programa para permitir a utilização do tipo *Decimal*.
`from decimal import Decimal`
2. Definição de variáveis reais - Para a definição do número real 0.1, por exemplo, utilizamos o comando a seguir. Note que, o número deve estar entre aspas simples ou duplas; *Decimal('0.1')* é diferente de *Decimal(0.1)*.
`numero = Decimal('0.1')`

A entrada também pode ser convertida para *Decimal* utilizando o seguinte comando.

```
numero = Decimal(input())
```

3. Operações - As operações para números do tipo *Decimal* são as mesmas utilizadas para o tipo *float*. No código abaixo, é feita a soma considerando os dois tipos de números reais.

Código

No decorrer do semestre haverá 3 tipos de tarefas no SuSy (descritas logo abaixo). As tarefas possuirão os mesmos casos de testes abertos e fechados, no entanto o número de submissões permitidas e prazos são diferentes. As seguintes tarefas estão disponíveis no SuSy:

- ❑ **Lab01-AmbienteDeTeste:** Esta tarefa serve para testar seu programa no SuSy antes de submeter a versão final. Nessa tarefa, tanto o prazo quanto o número de submissões são ilimitados, porém os arquivos submetidos aqui **não serão corrigidos**.
- ❑ **Lab01-Entrega:** Esta tarefa tem limite de uma **única** submissão e serve para entregar a **versão final** dentro do prazo estabelecido para o laboratório. Não use essa tarefa para testar o seu programa e submeta aqui apenas quando não for mais fazer alterações no seu programa.
- ❑ **Lab01-ForaDoPrazo:** Esta tarefa tem limite de uma **única** submissão e serve para entregar a versão final fora prazo estabelecido para o laboratório. Esta tarefa irá substituir a nota obtida na tarefa **Lab01-Entrega** apenas se o aluno tiver realizado as correções sugeridas no *feedback* ou caso não tenha enviado anteriormente na tarefa **Lab01-Entrega**.

Avaliação

Este laboratório será avaliado da seguinte maneira: a nota será proporcional ao número de casos **fechados** para os quais o seu programa gerou a resposta correta, **desde que os critérios indicados neste enunciado tenham sido atendidos**. Se o programa apresentou resposta correta para todos os casos, a nota será 10; caso contrário será $p \cdot 10$, onde p é o percentual de respostas corretas.

Testando seu programa

Para testar se a solução do seu programa está correta, basta seguir o exemplo abaixo no terminal do Linux.

```
python lab01.py < arq01.in > arq01.out
diff arq01.out arq01.res
```

O `arq01.in` é a entrada e `arq01.res` é a saída esperada, ambos disponíveis no SuSy. O `arq01.out` é a saída gerada pelo seu programa. Após o prazo, os casos de teste fechados serão liberados e podem ser baixados e testados da mesma forma que os testes abertos.