

## Problema I

### Boleto

*Arquivo fonte:* boleto.{ c | cpp | java | py }

*Autor:* Danilo Ruy Gomes (Fatec Itapetininga)

Desde o início de 2017, o Brasil vem mudando a forma de cobrança de boletos bancários. Anteriormente, qualquer pessoa podia emitir um boleto, desde que tivesse um CNPJ e uma conta em banco. Com as novas mudanças, além do CNPJ da empresa, os bancos vêm impondo que todos os boletos devem ser registrados, ou seja, com CPF, endereço e CEP válido. A NTG Soluções em Tecnologia trabalha com desenvolvimento de sistemas e vem sofrendo demais com essas novas mudanças, visto que antigamente ela utilizava uma API de terceiro para o desenvolvimento do *layout* de boletos e agora a própria NTG vem desenvolvendo todo o algoritmo para recebimento de boletos. Para a questão do intermédio entre as empresas e os bancos, utiliza-se um arquivo de texto num padrão conhecido como CNAB, com suas variantes. Dentro deste arquivo, há 240 caracteres em cada linha (CNAB 240) onde da coluna  $n$  até  $n+x$ , está uma informação, como número do boleto, data de vencimento, valor, etc. A tabela seguinte mostra um exemplo do padrão CNAB:

Tamanho	Coluna inicial	Coluna Final	Informação
6	5	10	Vencimento (DDMMYY)
6	11	16	Valor de Pagamento inteiro
2	17	18	Valor de Pagamento Decimal
4	19	22	Número Documento
6	23	28	Data de Pagamento

No momento, a empresa está desenvolvendo o processamento do arquivo de retorno, o qual contém as informações dos boletos pagos. Sua função será fazer um programa para extrair das cadeias de caracteres do arquivo texto, o somatório recebido com adimplência e inadimplência, baseando-se na data de pagamento.

### Entrada

A entrada consiste em um único caso de teste por vez, informando-se várias linhas, cada qual contendo uma cadeia com 30 caracteres com informações sobre o pagamento de um boleto.

### Saída

Seu programa deverá imprimir o valor total, com duas casas decimais, seguido do termo "ADIMPLENTE" para todo os boletos pagos em dia e, na outra linha, o valor total pago de todos os boletos com atraso seguido do termo "INADIMPLENTE". Esta última linha deve terminar com uma quebra de linha.

#### Exemplo de Entrada 1

006712101800023230012317051815	904.64-ADIMPLENTE
003808111800044000065217051816	794.40-INADIMPLENTE
008907061700023150235617051817	
006329051800023234672317051812	
003220041800056290341117051814	

#### Exemplo de Saída 1

**Exemplo de Entrada 2**

```
006731101800123230012310011815
003816111801213000065227051816
008917061700233150235614051817
006329051809923234672323071812
003230041800156290341107051814
```

**Exemplo de Saída 2**

```
13362.30-ADIMPLENTE
103126.74-INADIMPLENTE
```