

	LISTA DE EXERCÍCIOS II DE LTP/ LPI - 1º Bimestre Estruturas de Controle do Fluxo de Repetição		
	Profº. Marden Santos		Data: 21/03/2018
	Aluno:	Nº:	Turmas: 1AI; 1BI; 1CI

1. Elabore um algoritmo que leia 10 números quaisquer e imprima o número lido, o valor de seu respectivo “quadrado” e sua respectiva raiz quadrada.
2. Implementar um algoritmo que imprima na saída todos os múltiplos de 5, no intervalo de 1 até 100.
3. Desenvolva um algoritmo que leia 10 números e imprima a soma dos positivos e a quantidade de números negativos.
4. Elabore um algoritmo que leia e imprima os números inteiros no intervalo entre 10 e 100 de modo que estes não terminem em 0(zero).
5. Criar um algoritmo que leia um número e imprima todos os números de 1 até o número lido e o resultado do produto da sequência de números. Ex: 3 Saída: 1x2x3 = 6.
6. Criar um algoritmo que leia a quantidade de números que se deseja digitar para que possa ser impresso o maior e o menor número digitado. Trate inclusive os números negativos.
7. Desenvolva um algoritmo que leia um número inteiro a partir da unidade padrão de entrada e imprima na saída todos os divisores deste número.
8. Implemente um algoritmo que imprima na tela todas as tabuadas de multiplicar de 1 até 5.
9. Chico tem 1,50m e cresce 2 centímetros por ano, enquanto Juca tem 1,10m e cresce 3 centímetros por ano. Construir um algoritmo que calcule e imprima quantos anos serão necessários para que Juca seja maior que Chico.
10. Num frigorífico existem vários bois. Cada boi traz preso no seu pescoço um cartão contendo um número de identificação e seu peso. Implementar um algoritmo que leia as informações sobre os bois e escreva o número e o peso do boi mais gordo e do boi mais magro. O algoritmo acaba quando se digita um número menor ou igual a zero para a identificação.
11. Fazer um algoritmo que calcule e escreva o valor de S:

$$S = \frac{1}{1} + \frac{3}{2} + \frac{5}{3} + \frac{7}{4} + \dots + \frac{99}{50}$$

12. Fazer um algoritmo que calcule e escreva o valor de S onde:

$$S = \frac{1}{1} + \frac{2}{4} + \frac{3}{9} + \frac{4}{16} + \frac{5}{25} + \frac{6}{36} + \dots + \frac{-10}{100}$$

13. Fazer um algoritmo que calcule e escreva a seguinte soma:

$$S = \frac{2^1}{50} + \frac{2^2}{49} + \frac{2^3}{48} + \dots + \frac{2^{50}}{1}$$

14. Fazer um algoritmo que calcule e escreva a seguinte soma:

$$S = \frac{37 \times 38}{1} + \frac{36 \times 37}{2} + \frac{35 \times 36}{3} + \dots + \frac{1 \times 2}{37}$$

15. Fazer um algoritmo que calcule e escreva a soma dos 30 primeiros termos da série:

$$S = \frac{480}{10} - \frac{475}{11} + \frac{470}{12} - \frac{465}{13} + \dots$$

16. Escreva um algoritmo que gere uma série de **Fibonacci** até o vigésimo termo.

(1,1,2,3,5,8,13,21,...)

17. Elabore um algoritmo para calcular N! (**Fatorial** de N) de um N fornecido pelo usuário.

Sabe-se que $0! = 1$ e $N! = 1 \times 2 \times 3 \times \dots \times (N-1) \times N$

18. Número Triangular: Um número é triangular quando é resultado do produto de três números consecutivos. Exemplo $24 = 2 \times 3 \times 4$. Elabore um algoritmo que dado um número verifica se ele é ou não um número triangular. Utilize mensagens como saídas.

19. Número Primo: Faça um algoritmo que leia um número inteiro positivo (entrada deve ser validada) e determine se este é primo ou não. Outra opção deste algoritmo será implementada da seguinte maneira: dado um intervalo fechado a critério do usuário, determine todos os números primos pertencentes a este intervalo.

20. Número Perfeito: Número perfeito é aquele cuja soma de seus divisores, exceto ele próprio, é igual ao próprio número.

Ex: 6 é perfeito porque $1 + 2 + 3 = 6$.

28 é perfeito porque $1 + 2 + 4 + 7 + 14 = 28$

Elaborar um algoritmo que lido um número da unidade padrão de entrada, verifique e escreva se este número n fornecido é perfeito ou não (use mensagens). Adote $n = 0$ como flag de saída.

21. Multiplicação Sucessivas: Fazer um algoritmo que leia um número N qualquer maior ou igual a 1 e menor ou igual a 50 apresente o valor obtido na multiplicação sucessiva de N por 2, enquanto o produto for menor que 250. O valor N deve ser checado quanto a sua validade, caso o usuário informe um valor fora da faixa. O algoritmo deve informar por meio de uma mensagem que o valor digitado não é válido.

22. Fazer um algoritmo que calcule e escreva a soma dos 50 primeiros termos da série:

$$\frac{1!}{1} - \frac{2!}{3} + \frac{3!}{7} - \frac{4!}{15} + \frac{5!}{31} - \dots$$

23. Certa firma fez uma pesquisa de Mercado para saber se as pessoas gostaram ou não de um novo produto lançado no mercado. Para isso forneceu o sexo do entrevistado e sua resposta (sim ou não). Sabendo-se que foram entrevistadas 20 pessoas, fazer um algoritmo que calcule e escreva:

- a) A quantidade de pessoas que responderam sim;
- b) A quantidade de pessoas que responderam não;
- c) A porcentagem de pessoas do sexo feminino que responderam sim;
- d) A porcentagem de pessoas do sexo masculino que responderam não.

24. O sistema de avaliação de uma determinada disciplina obedece aos seguintes critérios:

Durante o semestre são dadas duas notas e anotada a quantidade de comparecimento as aulas;

A nota final é obtida pela média aritmética das notas dadas durante o curso. É considerado aprovado o aluno que obtiver a nota final superior ou igual a 6 e que tiver comparecido a um mínimo de 65 aulas.

Fazer um algoritmo que leia um conjunto de dados contendo o número de matrícula, as três notas e a frequência (número de aulas frequentadas) de 10 alunos.

Calcule:

- a) A nota final de cada aluno;
- b) A maior e menor nota da turma;
- c) A nota média da turma;
- d) O total de alunos reprovados;
- e) A porcentagem de alunos reprovados por faltas.

Escreva:

- a) Para cada aluno, o número de matrícula, a frequência, a nota final e o código (aprovado ou reprovado);
- b) O que foi calculado no item b (2, 3, 4 e 5).

25. Numa fábrica trabalham homens e mulheres divididos em três classes:

A – os que fazem até 30 peças por mês;

B – os que fazem de 31 a 35 peças por mês;

C – os que fazem mais de 35 peças por mês.

A classe A recebe salário mínimo. A Classe B recebe salário mínimo e mais 3% do salário mínimo por peça, acima das 30 iniciais. A classe C recebe salário mínimo e mais 5% do salário mínimo por peça acima das 30 iniciais. Fazer um algoritmo que leia várias linhas, contendo cada uma:

- ✓ O número do operário;
- ✓ O número de peças fabricadas por mês;
- ✓ O sexo do operário.

Calcule e escreva:

- a) O salário de cada operário;
- b) O total da folha mensal de pagamento da fábrica;
- c) O número total de peças fabricadas por mês;
- d) A média de peças fabricadas pelos homens em cada classe;
- e) A média de peças fabricadas pelas mulheres em cada classe;
- f) O número de operário ou operária de maior salário (não existe empate).

Observação: A última linha, que servirá de *flag*, terá o número de operário igual à zero.

26. Deseja-se fazer uma pesquisa a respeito do consumo mensal de energia elétrica em uma determinada cidade. Para isso, são fornecidos os seguintes dados de entrada:

- ✓ Preço do kWh consumido;
- ✓ Número do consumidor;
- ✓ Quantidade de kWh consumidos durante o mês;
- ✓ Código do tipo de consumidor (R - residencial, C - comercial, I - industrial).

O número do consumidor igual à zero deve ser usado como *flag*. Fazer um algoritmo que:

Calcule:

- a) Para cada consumidor, o total a pagar;
- b) O maior consume verificado;
- c) O menor consume verificado;
- d) O total do consume para cada um dos três tipos de consumidores;
- e) A média geral de consumo.

Escreva:

- a) Para cada consumidor o seu número e o total a pagar;
- b) O que foi calculado nos itens **b**, **c**, **d**, **e** acima especificados.

27. Ilhas Weblands

As Ilhas Weblands formam um reino independente nos mares do Pacífico. Como é um reino recente, a sociedade é muito influenciada pela informática. A moeda oficial é o Bit; existem notas de B\$ 50,00, B\$10,00, B\$5,00 e B\$1,00. Você foi contratado (a) para ajudar na programação dos caixas automáticos de um grande banco das Ilhas Weblands. 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\$10,00, 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. Escreva um programa que, dado o valor de Bits desejado pelo cliente, 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, duas de B\$10,00 e duas de B\$1,00.

Entrada

A entrada é composta de vários conjuntos de teste. Cada conjunto de teste é composto por uma única linha, que contém um número inteiro positivo V , que indica o valor solicitado pelo cliente. O final da entrada é indicado por $V = 0$.

Exemplo de Entrada

```
1
72
0
```

Saída

Para cada conjunto de teste da entrada seu algoritmo deve produzir três linhas na saída. A primeira linha deve conter um identificador do conjunto de teste, no formato “Teste n ”, onde n é numerado a partir de 1. Na segunda linha devem aparecer quatro inteiros I , J , K e L que representam o resultado encontrado pelo seu programa: I indica o número de cédulas de B\$50,00, J indica o número de cédulas de B\$10,00, K indica o número de cédulas de B\$5,00 e L indica o número de cédulas de B\$1,00. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

Exemplo de Saída

```
Teste 1
0 0 0 1
```

```
Teste 2
1 2 0 2
```

(esta saída corresponde ao exemplo de entrada acima)

Restrições

$0 \leq V \leq 1000$ ($V = 0$ apenas para indicar o fim da entrada)

28.OBI

O principal prêmio da Olimpíada Brasileira de Informática é o convite para os cursos de programação oferecidos no Instituto de Computação da Unicamp, com todas as despesas pagas pela Fundação Carlos Chagas, patrocinadora da OBI. São convidados apenas os competidores que atingem um certo número mínimo de pontos, consideradas as duas fases de provas. Você foi contratado pela Coordenação da OBI para fazer um programa que, dados os números de pontos obtidos por cada competidor em cada uma das fases, e o número mínimo de pontos para ser convidado, determine quantos competidores serão convidados para o curso na Unicamp. Você deve considerar que:

- todos os competidores participaram das duas fases;
- o total de pontos de um competidor é a soma dos pontos obtidos nas duas fases;

Por exemplo, se a pontuação mínima para ser convidado é 435 pontos, um competidor que tenha obtido 200 pontos na primeira fase e 235 pontos na segunda fase será convidado para o curso na Unicamp. Já um competidor que tenha obtido 200 pontos na primeira fase e 234 pontos na segunda fase não será convidado.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do dispositivo de entrada padrão (normalmente o teclado). A primeira linha da entrada contém dois números inteiros N e P , representando respectivamente o número de competidores ($1 \leq N \leq 1000$) e o número mínimo de pontos para ser convidado ($1 \leq P \leq 1000$). Cada uma das N linhas seguintes contém dois números inteiros X e Y indicando a pontuação de um competidor em cada uma das fases ($0 \leq X \leq 400$) e ($0 \leq Y \leq 400$).

Saída

Seu algoritmo deve imprimir na saída padrão uma única linha contendo um único inteiro, indicando o número de competidores que serão convidados a participar do curso na Unicamp.

Exemplo de entrada 3 100 50 50 100 0 49 50	Exemplo de saída 2
Exemplo de entrada 4 235 100 134 0 0 200 200 150 150	Exemplo de saída 2

29. Insensibilidade

O planeta Bizz fica a 133 upals de distância do planeta Terra (onde “upals” é uma unidade de medida dada por “um monte de anos-luz”), e parece ser o único planeta com vida for além do nosso. Este planeta é muito interessante, pois, em cada país, seus habitantes têm uma característica diferente. Um desses países é a Cegônia, que tem como característica o fato de que todos os seus habitantes são cegos. Em compensação, todos possuem um “sexto sentido” acentuado, podendo perceber o que está a sua volta mesmo sem enxergar. Este ano, o governo da Cegônia fará um censo, e dentre os dados de seus habitantes, quer saber o quanto de insensibilidade cada pessoa possui. A insensibilidade indica quão ruim é a capacidade das pessoas de perceber os objetos à sua volta sem precisar enxergar.

Tal teste é feito da seguinte maneira: a pessoa é colocada em uma sala onde encontram-se vários objetos em posições pré-determinadas. A pessoa deve, então, dizer quais são as coordenadas de cada objeto dentro da sala. Para cada objeto, calcula-se o quadrado da

distância entre a posição adivinhada pela pessoa e a posição real do objeto; esse valor é chamado de D. O nível de insensibilidade da pessoa é dado pela soma de todos os D. Por exemplo, suponha que na sala existam 4 objetos, nas coordenadas (1, 1), (3, 4), (5, 7) e (10, 10). Se a pessoa então disser que os objetos estão, respectivamente, nas posições (1, 2), (5, 4), (5, 7) e (19, 10), o valor de D para cada objeto será 1, 4, 0 e 81 e, portanto, o nível de insensibilidade da pessoa é $1 + 4 + 0 + 81 = 86$.

Você precisa fazer um programa que, dadas as coordenadas verdadeiras dos objetos e as coordenadas indicadas por uma pessoa, diga qual é o nível de insensibilidade dessa pessoa.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do dispositivo de entrada padrão (normalmente o teclado). A primeira linha da entrada contém um único inteiro N ($1 \leq N \leq 1.000$), indicando quantos objetos estão no quarto. As N linhas seguintes contêm cada uma quatro inteiros X1, Y1, X2, Y2 ($0 \leq X_i \leq 1000$). Cada linha representa um objeto: a posição real do objeto é (X1, Y1), e a posição onde a pessoa disse estar tal objeto é (X2, Y2).

Saída

Seu algoritmo deve imprimir, na saída padrão, uma única linha, contendo um único inteiro, indicando o nível de insensibilidade da pessoa estudada.

Exemplo de entrada	Exemplo de saída
<pre>4 1 1 1 2 3 4 5 4 5 7 5 7 10 10 19 10</pre>	<pre>86</pre>

Exemplo de entrada	Exemplo de saída
<pre>5 0 0 0 0 1 3 1 3 4 10 11 10 2 2 3 3 0 1 0 1</pre>	<pre>51</pre>

30. Letras

Considere as definições abaixo:

- Uma palavra é uma sequência de letras consecutivas.
- Um texto é um conjunto de palavras separadas pelo caractere espaço em branco.

Você foi contratado pela empresa Booble para escrever um programa que, dados uma letra e um texto, determina a porcentagem de palavras do texto que contém a letra dada.

Entrada

A primeira linha da entrada contém um único caractere, a letra de interesse na pesquisa. A segunda linha contém um texto, como definido acima.

Saída

Seu programa deve produzir uma única linha, contendo um único número real, a porcentagem de palavras do texto que contém a letra dada, com precisão de uma casa decimal.

Restrições

- O texto é composto apenas por letras minúsculas e o caractere espaço em branco.
- O texto é formado por no mínimo um caractere, e no máximo 1000 caracteres.
- O texto não contém dois espaços em branco consecutivos.

Exemplos

Entrada p papagaio	Saída 100.0
Entrada o no meio do caminho tinha uma pedra tinha uma pedra no meio do caminho	Saída 57.1
Entrada b nunca me esquecerei que no meio do caminho tinha uma pedra	Saída 0.0