

Universidade Federal de Goiás

Instituto de Informática

Introdução à Programação - 2017-1

Lista de Exercícios 1 (L1)

Professores:

Gustavo Teodoro Laureano

Luciana de Oliveira Beretta





Thierson Couto Rosa

Tarefa L1-B

Introdução

A primeira lista de exercícios - Lista de exercícios 1 (L1) contém 30 exercícios. Essa lista está dividida em três tarefas (*assignments*) no sistema Sharif. Essa é a segunda tarefa (L1-B) e contém os 11 seguintes exercícios de L1. Todos os exercícios de L1-B devem ser resolvidos individualmente. Para cada exercício deve ser escrito um programa, o qual deve ser submetido para avaliação no sistema Sharif¹. Os exercícios de L1 são classificados em quatro níveis de dificuldade sob o ponto de vista de um aluno iniciante: fácil, intermediário, difícil e muito difícil. Cada grau de dificuldade é indicado por um emoticon distinto, colocado ao lado direito do título do exercício. O valor de cada nível é mostrado na tabela a seguir:

Tabela 1: Valor de cada nível de exercício

Nível	Emoticon	Valor
Fácil		0,3
Intermediário		0,5
Difícil		1,0
Muito Difícil		2,0

A lista L1 vale 10 pontos. O aluno pode conseguir somar 10 pontos de várias maneiras, resolvendo uma combinação de exercícios dos quatro níveis espalhados entre as três tarefas que formam a L1. Embora não seja necessário a resolução de todos os exercícios da lista L1, recomenda-se, principalmente aos iniciantes em programação que tentem resolver o maior número de exercícios possível. O treinamento é muito importante para desenvolver o raciocínio lógico na solução de problemas.

¹<http://sharif.inf.ufg.br/thierson/index.php/login>.

Aprovado ou Reprovado 😊

Fazer um algoritmo que calcule a média aritmética das três notas de um aluno e mostre, além do valor da média, uma mensagem de “APROVADO”, caso a média seja igual ou superior a seis, ou a mensagem “REPROVADO”, caso contrário.

Entrada

A entrada conterà uma linha com as três notas do aluno, separadas entre si por um caractere de espaço.

Saída

A saída deve conter duas linhas. A primeira linha deve conter uma frase com o seguinte formato: $MEDIA = x$, onde x é o valor da média entre as três notas do aluno, contendo duas casas decimais. A segunda linha contém uma das duas mensagens: APROVADO ou REPROVADO. Após o valor da média e após a mensagem, o programa deve imprimir o caractere de quebra de linha: ‘\n’.

Exemplos

A seguir são mostrados dois casos distintos de entrada, somente para efeito de ilustração, porém, esse problema contém apenas um caso de teste na entrada, formado apenas por uma linha de entrada contendo as três notas.

Entrada
7.4 6.2 3.7
5.1 9.9 7.2

Saída:
MEDIA = 5.77
REPROVADO
MEDIA = 7.40
APROVADO

Valor de y dado x 😊

Desenvolver um algoritmo para ler um número x , calcular e imprimir o valor de y de acordo com as condições abaixo:

$$y = \begin{cases} x, & \text{se } x < 1; \\ 0, & \text{se } x = 1; \\ x^2, & \text{se } x > 1; \end{cases}$$

Entrada

O programa deve ler uma linha contendo um único número inteiro correspondendo ao valor de x .

Saída

O programa deve imprimir $Y = y$, onde y é o valor computado de y dado x . Após o valor de y , o programa deve imprimir um caractere de quebra de linha: ‘\n’.

Exemplo

Entrada
3

Saída:
Y = 9

Divisível por Três e por Cinco 😊

Desenvolver um programa que leia um número inteiro e verifique se o número é divisível por três e também é divisível por cinco.

Entrada

O programa deve ler uma linha contendo um número inteiro na entrada.

Saída

O programa deve imprimir uma linha contendo a frase: O NUMERO E DIVISIVEL, se ele for divisível tanto por três quanto por cinco, ou a frase O NUMERO NAO E DIVISIVEL, em caso contrário. Após imprimir uma das frases, o programa deve imprimir um caractere de quebra de linha: ‘\n’.

Exemplo

Entrada
75

Saída
O NUMERO E DIVISIVEL

Locadora de Charretes 😊

Uma locadora de charretes cobra R\$ 10,00 de taxa para cada 3 horas de uso de uma charrete e R\$5,00 para cada 1 hora abaixo dessas 3 horas. Fazer um programa que leia a quantidade de horas que a charrete foi usada e que calcule e escreva quanto o cliente tem de pagar.

Entrada

O programa deve ler uma única linha na entrada, contendo o número de horas que o locatário utilizou a charrete.

Saída

O programa deve imprimir uma linha contendo a frase: O VALOR A PAGAR E = X , onde X é o valor do aluguel e deve conter no máximo 2 casas decimais. Após o valor do aluguel o programa deve imprimir um caractere de quebra de linha: ‘\n’.

Exemplo

Entrada
29

Saída:
O VALOR A PAGAR E = 100.00

Reajuste Salarial 😊

Fazer um algoritmo que calcule e imprima o salário reajustado de um funcionário de acordo com as seguintes regras:

- Salário de até R\$ 300,00, reajuste de 50%;
- Salário maior que R\$300,00 reajuste de 30%;

Entrada

A entrada conterá uma linha com o salário do funcionário.

Saída

A saída deve conter, numa linha com a frase: SALARIO COM REAJUSTE = x , onde x é um valor real com duas casas decimais e corresponde ao valor do salário reajustado. Logo em seguida ao valor de x , o programa devem imprimir o caractere de quebra de linha: ‘\n’.

Exemplos

A seguir são mostrados dois casos distintos de entrada, somente para efeito de ilustração, porém, esse problema contém apenas um caso de teste na entrada, formado por apenas uma linha de entrada.

Entrada	Saída:
755.00	SALARIO COM REAJUSTE = 981.50
265.32	SALARIO COM REAJUSTE = 397.98

Raízes de Equações de Segundo Grau 🤖

Desenvolver um programa que leia os coeficientes (a , b e c) de uma equação de segundo grau e calcule as raízes da equação. O programa deve mostrar a classificação das raízes, e, quando possível, o valor das raízes calculadas.

Considerações

Dada uma equação do segundo grau do tipo $ax^2 + bx + c$, Δ (delta) $= b^2 - 4ac$. Se $\Delta = 0$, a raiz da equação é ÚNICA. Se $\Delta < 0$. As raízes da equação são IMAGINÁRIAS. Se $\Delta > 0$, então há duas RAÍZES DISTINTAS para a equação. A fórmula geral para computar as raízes de uma equação do segundo grau é a fórmula de Báskara, dada por:

$$x = \frac{-b \pm \sqrt{\Delta}}{2a}$$

Entrada

O programa deve ler três valores reais na entrada. O primeiro valor corresponde ao valor do coeficiente a , o segundo, do coeficiente b e o terceiro, do coeficiente c , de uma equação de segundo grau. Os três valores ocorrem em uma única linha na entrada, separados entre si por um espaço.

Saída

O programa deve imprimir uma linha contendo uma das seguintes frases, conforme for o resultado do cálculo das raízes da equação: RAIZES DISTINTAS, ou RAIZ UNICA, ou RAIZES IMAGINARIAS. No primeiro caso o programa deve imprimir uma outra linha contendo a frase $X1 = x_1$, onde x_1 é o valor da menor raiz encontrada para a equação. Ainda no primeiro caso, o programa deve imprimir uma terceira linha com a frase $X2 = x_2$, onde x_2 corresponde ao valor da segunda raiz. No segundo caso, o programa deve imprimir uma frase $X1 = x_1$, onde x_1 é o valor da única raiz da equação. O terceiro caso não há o que imprimir pois as raízes são imaginárias.

Exemplos

A seguir são mostrados três exemplos distintos de entrada, e suas correspondentes saídas, entretanto, existe apenas uma linha de entrada para esse problema.

Entrada
2 12 10
2 12 18
15 17 89

Saída:
RAIZES DISTINTAS
X1 = -1.00
X2 = -5.00
RAIZ UNICA
X1 = -3.00
RAIZES IMAGINARIAS

Conta de Água 🤖

Desenvolver um programa para calcular a conta de água para uma empresa de saneamento. O custo da água varia dependendo se o consumidor é residencial, comercial ou industrial. A regra para calcular a conta é:

- Residencial: R\$ 5,00 de taxa mais R\$ 0,05 por metros cúbicos gastos;
- Comercial: R\$ 500,00 para os primeiros 80 metros cúbicos gastos mais R\$ 0,25 por metros cúbicos gastos;
- Industrial: R\$ 800,00 para os primeiros 100 metros cúbicos gastos mais R\$ 0,04 por metros cúbicos gastos;

O programa deverá ler a conta do cliente, o consumo de água por metros cúbicos e o tipo de consumidor (residencial, comercial e industrial). Como resultado, o programa deve imprimir a conta do cliente e o valor em Reais a ser pago pelo mesmo.

Entrada

O programa deverá ler uma linha na entrada contendo: a conta do cliente (um número inteiro), o consumo de água por metros cúbicos (float) e o tipo do consumidor (um caractere: 'C' - COMERCIAL, 'I' - INDUSTRIAL ou 'R' - RESIDENCIAL). Há um espaço entre os valores na linha de entrada

Saída

O programa deve imprimir duas linhas, contendo o seguinte:

- $CONTA = u$, onde u é o código inteiro identificador da conta;
- $VALOR DA CONTA = v$, onde v é o valor da conta com duas casas decimais, a ser pago pelo consumidor;

Após o valor impresso em cada linha, o programa deve imprimir o caractere de quebra de linha; *'backslashn'*. Os valores de v , x e w devem conter duas casas decimais.

Exemplos

Abaixo são mostrados dois exemplos de entrada e saída, mas há apenas um caso de entrada (uma linha) para esse programa.

Entrada	Saída:
39393939 230 C	CONTA = 39393939
888 3752 I	VALOR DA CONTA = 537.50
	CONTA = 888
	VALOR DA CONTA = 946.08

Observação

Sugestão para a leitura: `scanf("%d %f %c", &conta, &consumo, &tipo);`, onde `conta` é uma variável `int`, `consumo` é uma variável `float` e `tipo` é uma variável `char`.

Soma dos Três Menores 🟢

Fazer um programa para quatro valores inteiros e imprimir a soma dos três menores.

Entrada

O programa deve ler quatros valores inteiros na entrada. Cada valor ocupa uma linha na entrada.

Saída

O programa deve imprimir uma linha contendo o valor da soma dos três menores números. Após o valor da soma, o programa deve imprimir um caractere de quebra de linha: ‘\n’.

Exemplo

Entrada
9
4
2
12

Saída:
15

Triângulo ou Trapézio? 😊

Leia três valores reais (A , B e C) e verifique se eles formam ou não um triângulo. Em caso positivo, calcule o perímetro do triângulo e imprima a mensagem:

$$\text{Perimetro} = \text{XX.X}$$

Caso os valores não formem um triângulo, calcule a área do trapézio que tem A e B como base e C como altura, mostrando a mensagem:

$$\text{Area} = \text{XX.X}$$

Considerações

Para que os três valores: A , B e C formem um triângulo as três condições abaixo devem ser satisfeitas:

- $|b - c| < a < b + c$;
- $|a - c| < b < a + c$;
- $|a - b| < c < a + b$;

A área de um trapézio é computada como $\text{Área} = \frac{(A+B)*C}{2}$

Entrada

A entrada é formada por uma linha contendo três valores decimais separados um do outro por um espaço em branco.

Saída

A saída deve conter em uma única linha a frase apropriada. Observe nos exemplos acima que a saída deve conter apenas uma casa decimal. Os valores “X” que aparecem nos formatos são substituídos por dígitos que formam o valor de saída. Depois desses valores o programa deve imprimir o caractere de quebra de linha: ‘\n’.

Exemplos

Entrada:	Saída:
6.0 4.0 2.0	Area = 10.0
6.0 4.0 2.1	Perimetro = 12.1

Observações

Para imprimir um valor float com apenas uma casa decimal você deve usar a função **printf** com o seguinte formato: `printf("Frase apropriada = %.1f\n");`

Cálculo do Imposto de Renda 😊

Desenvolver um algoritmo que determine o imposto de renda cobrado de um funcionário pelo governo. Seu programa deverá ler a matrícula de um funcionário, o valor do salário mínimo, o número de dependentes, o salário do funcionário e a taxa de imposto normal que já foi paga pelo funcionário. O imposto bruto é:

- 20% do salário do funcionário, se o funcionário ganha mais de 12 salários mínimos;
- 8% do salário do funcionário, se o funcionário ganha mais de cinco salários mínimos;
- Zero % do salário do funcionário, se ele ganha cinco salários mínimos ou menos.

Determine o imposto líquido a ser pago pelo funcionário subtraindo R300,00 no imposto bruto, para cada dependente do funcionário. O programa calculará e imprimirá o imposto a ser pago ou devolvido, que é a diferença entre o imposto líquido e o imposto normal **descontado** do salário do funcionário. Se a diferença for negativa, o programa deve emitir a mensagem de “imposto a receber”. Se a diferença for um valor positivo o programa deve emitir a mensagem, “imposto a pagar”, e, se for igual a zero, deve emitir a mensagem “imposto quitado”.

Entrada

O programa deve ler uma linha contendo cinco valores na entrada, separados entre si por um espaço: a matrícula (um número inteiro), o valor do salário mínimo (float), o número de dependentes (inteiro), o salário do funcionário (float) e a taxa de imposto (float), nesta ordem.

Saída

O programa deve imprimir quatro linhas, contendo o seguinte:

- $MATRICULA = u$, onde u é o valor da matrícula do funcionário;
- $IMPOSTO\ BRUTO = v$, onde v é o valor do imposto bruto;
- $IMPOSTO\ LIQUIDO = x$, onde x é o valor do imposto líquido;
- $RESULTADO = w$, onde w é o valor da diferença entre o imposto bruto e o imposto líquido;
- A mensagem IMPOSTO A RECEBER, se o valor de w for negativo ou a mensagem IMPOSTO QUITADO, se w for igual a zero, ou a mensagem IMPOSTO A PAGAR, caso w for maior que zero.

Os valores de v, x e w devem conter duas casas decimais.

Exemplos

Abaixo são mostrados dois exemplos de entrada e saída, mas há apenas um caso de entrada (uma linha) para esse programa.

Entrada
510.0 3 1531.97 8.5
630.00 2 4567.01 56.7

Saída:
IMPOSTO BRUTO = 0.00
IMPOSTO LIQUIDO = -900.00
RESULTADO = -1030.22
IMPOSTO A RECEBER
IMPOSTO BRUTO = 379.98
IMPOSTO LIQUIDO = -220.02
RESULTADO = -2809.52
IMPOSTO A RECEBER

Ordem 🙄

Você receberá três valores inteiros e deve descobrir quais deles correspondem às variáveis a , b e c . Os números não serão dados em ordem exata, mas sabemos que o valor correspondente a a é menor do que o valor correspondente a b , e que o valor correspondente a b é menor do que o correspondente a c . Será informada a você a ordem em que os valores associados a cada variável devem ser impressos. Escreva um programa que imprima os valores na ordem requisitada.

Entrada

A entrada conterá duas linhas. A primeira, com três números inteiros positivos, separados entre si por um espaço. Todos os três números são inferiores ou iguais a 100. A segunda linha conterá três letras maiúsculas A , B e C (sem espaços entre elas) representando a ordem desejada de impressão dos valores das variáveis.

Saída

A saída deve conter, numa linha, os inteiros a , b e c na ordem desejada, separados por espaços simples. Após o último número da saída deve aparecer apenas o caractere de quebra de linha: `'\n'`.

Exemplos

A seguir são mostrados dois casos distintos de entrada, somente para efeito de ilustração, porém, esse problema contém apenas um caso de teste na entrada, formado pelas duas linhas de entrada descritas acima.

Entrada	Saída:
1 5 3	1 3 5
ABC	6 2 4
6 4 2	
CAB	

Observação

Após o último número na primeira linha da entrada, está no buffer de entrada o caractere `'\n'`. Com isso ao tentar ler o primeiro caractere (A , B , ou C) na segunda linha de entrada com `scanf("%d", &x);` será lido o caractere `'\n'` na variável x , ao invés de uma das letras na entrada (A , B , ou C). Para evitar isso, você pode fazer com que a leitura do último número na primeira linha consuma o caractere `'\n'` da primeira linha, colocando esse caractere na especificação de formato do `scanf()`. Por exemplo, suponha que você declarou as seguintes variáveis na entrada: `int a, b, c;` para armazenar os três números da primeira linha e `char x, y, z;` para armazenar as três letras que aparecem na segunda linha de entrada. A leitura dessas variáveis de entrada pode ser realizada assim: `scanf("%d %d %d\n", &a, &b, &c); scanf("%c%c%c", &x, &y, &z);` Repare o `'\n'` ao final da formatação do primeiro `scanf` e repare que não há espaços entre os `"%c"` na formatação do segundo `scanf`. O `'\n'` ao final da formatação do primeiro `scanf()` faz com que o caractere de quebra de linha seja consumido no buffer. Assim, no segundo `scanf()` será armazenada na variável x a primeira letra da segunda linha e não o `'\n'`, resolvendo o problema da leitura.