

Maratona de Programação

16 de outubro de 2019

Caderno de Problemas

Elaboração:

Hitalo Joseferson Batista Nascimento
Francisco Edvan Chaves

Informações Gerais

A competição de programação é um subevento da Feira Tecnológica da UNIGRANDE e se destina a alunos do curso de graduação em Ciências da Computação, Análise de Sistemas e Sistemas para Internet e Engenharia de Computação da mesma instituição. A competição promove nos alunos a criatividade, a busca de novas soluções de software e a habilidade de resolver problemas.

Os alunos podem participar em duplas (ou trio). A competição é composta de 5 problemas. A duração será de 4 horas. Os alunos terão à sua disposição apenas um computador e material de pesquisa digital (livros, documentações, manuais, dentre outros que serão entregues pela comissão organizadora). Para vencer a competição, os competidores de cada time devem colaborar para descobrir os problemas mais fáceis, projetar os testes, e construir as soluções que sejam aprovadas pelos juízes da competição. Alguns problemas requerem apenas compreensão, outros conhecimentos de técnicas mais sofisticadas, e alguns podem ser realmente muito difíceis de serem resolvidos.

Os competidores receberão um caderno impresso com os problemas que devem ser resolvidos. Nos enunciados dos problemas constam exemplos, mas eles não têm acesso às instâncias testadas pelos juízes. O time que conseguir resolver o maior número de problemas é declarado o vencedor.

Linguagens: C, C++ e Java.

Softwares instalados: Dev-C++, Codeblocks, Visual Studio, Netbeans, Eclipse.

Forma de avaliação

Cada questão será avaliada através de diferentes inputs que serão desconhecidos dos alunos. O algoritmo desenvolvido por cada aluno deve ser capaz de lidar com as diferentes entradas testadas e fazer o tratamento de erros e exceções. Assim, ao testar 10 entradas, se o programa responder a saída como esperado em todas as vezes, o aluno recebe o valor de 10 pontos para a questão. Ao final, será atribuída ao aluno uma nota que será o número de pontos obtidos. Para casos de desempate será avaliada a qualidade do código fonte e também implementações parciais. Persistindo o empate, será considerado vencedor aquele que tiver maior número de créditos cursados.

Cada problema resolvido deve ser enviado para edvan@unigrande.edu.br e hitalo@unigrande.edu.br

Problema 1 – Área do Círculo

A fórmula para calcular a área de uma circunferência é: **area** = π . **raio**². Considerando para este problema que π = 3.14159:

- Efetue o cálculo da área, elevando o valor de **raio** ao quadrado e multiplicando por π .

Entrada

A entrada contém um valor de ponto flutuante (dupla precisão), no caso, a variável **raio**.

Saída

Apresentar a mensagem "A=" seguido pelo valor da variável **area**, conforme exemplo abaixo, com 4 casas após o ponto decimal. Utilize variáveis de dupla precisão (double). Como todos os problemas, não esqueça de imprimir o fim de linha após o resultado, caso contrário, você receberá "Presentation Error".

Exemplos de Entrada	Exemplos de Saída
2.00	A=12.5664
100.64	A=31819.3103
150.00	A=70685.7750

Problema 2 – Salário com Bônus

Faça um programa que leia o nome de um vendedor, o seu salário fixo e o total de vendas efetuadas por ele no mês (em dinheiro). Sabendo que este vendedor ganha 15% de comissão sobre suas vendas efetuadas, informar o total a receber no final do mês, com duas casas decimais.

Entrada

O arquivo de entrada contém um texto (primeiro nome do vendedor) e 2 valores de dupla precisão (double) com duas casas decimais, representando o salário fixo do vendedor e montante total das vendas efetuadas por este vendedor, respectivamente.

Saída

Imprima o total que o funcionário deverá receber, conforme exemplo fornecido.

Exemplos de Entrada	Exemplos de Saída
JOAO 500.00 1230.30	TOTAL = R\$ 684.54
PEDRO 700.00 0.00	TOTAL = R\$ 700.00
MANGOJATA 1700.00 1230.50	TOTAL = R\$ 1884.58

Problema 3 – De Dentro para Fora

A sua impressora foi infectada por um vírus e está imprimindo de forma incorreta. Depois de olhar para várias páginas impressas por um tempo, você percebe que ele está imprimindo cada linha de dentro para fora. Em outras palavras, a metade esquerda de cada linha está sendo impressa a partir do meio da página até a margem esquerda. Do mesmo modo, a metade direita de cada linha está sendo impressa à partir da margem direita e prosseguindo em direção ao centro da página.

Por exemplo a linha:

THIS LINE IS GIBBERISH

está sendo impressa como:

I ENIL SIHTHSIREBBIG S

Da mesma forma, a linha " MANGOS " está sendo impressa incorretamente como "NAM SOG". Sua tarefa é desembaralhar (decifrar) a string a partir da forma como ela foi impressa para a sua forma original. Você pode assumir que cada linha conterá um número par de caracteres.

Entrada

A entrada contém vários casos de teste. A primeira linha de entrada contém um inteiro **N** que indica a quantidade de casos de teste. Seguem **N** linhas, cada uma com uma frase com no mínimo 2 e no máximo 100 caracteres de letras maiúsculas e espaços que deverá ser desembaralhada (decifrada) à partir da forma impressa para a sua forma original, conforme especificação acima.

Saída

Para cada linha de entrada deverá ser impressa uma linha de saída com a frase decifrada, conforme a especificação acima.

Exemplo de Entrada	Exemplo de Saída
5 I ENIL SIHTHSIREBBIG S LEVELKAYAK H YPPAHSYADILO ABCDEFGHIJKLMNOPQRSTUVWXYZ VOD OWT SNEH HCNERF EGDITRAP A DNA SE	THIS LINE IS GIBBERISH LEVELKAYAK HAPPY HOLIDAYS MLKJIHGFEDCBAZYXWVUTSRQPON FRENCH HENS TWO DOVES AND A PARTRIDGE

Problema 4 – Consumo de combustível

Os dados ao lado, representam a relação entre tempo (em meses) e o consumo de combustível de determina do veículo, após procedimentos de testes de revisão desse veículo em pontos autorizados;

1	15.1
2	15.9
3	14.8
4	13.6
5	12.0
6	11.5
7	9.0
8	7.2
9	8.1
10	7.7
11	7.5
12	7.3

•Escreva um programa que leia esses dados a partir de um arquivo txt ou csv e apresente como saída uma estimativa para o consumo quando o usuário informar o mês de interesse, por exemplo 14 (1 ano e 2 meses).

Problema 5 – CorrectHTML

Crie um programa que leia um arquivo contendo um código em HTML e identifique linhas com tags aninhadas incorretamente.

- Exemplo de tags aninhadas corretamente:

```
<H1>This is my <span>heading</span></H1>
```

- Exemplo de tags aninhadas de forma incorreta:

```
<H1>This is my <span>heading</H1></span>
```

Problema 6– Acampamento de Férias

Nas férias de Julho, várias escolas de uma mesma região resolveram se organizar e levaram uma parte de seus alunos para um acampamento de férias por uma semana. Nestes acampamentos os alunos são divididos em chalés coletivos por gênero e idade, sempre com um supervisor ou supervisora que, além de dormirem com o grupo no chalé, também são responsáveis por criar e executar várias atividades interessantes e animadas, para todas as idades. Dentre as diversas atividades podem-se citar jogos, excursões, Gincana Musical, Gincanas Noturnas, etc. No primeiro dia de acampamento, devido à forte chuva, as atividades recreativas ficaram limitadas e as crianças foram levadas para o ginásio de esportes. Foi realizada uma gincana e uma das atividades da mesma consistiu em agrupar as crianças em um círculo (organizado no sentido anti-horário) do qual seriam retiradas uma a uma até que sobrasse apenas uma criança, que seria a vencedora.

No momento em que entra no círculo, cada criança recebe uma pequena ficha que contém um valor de 1 a 500. Depois que o círculo é formado, conta-se, iniciando na criança que está ao lado da primeira que entrou no círculo, o número correspondente à ficha que a primeira detém. A criança onde o número contado cair, deve ser retirada do grupo, e a contagem inicia novamente segundo a ficha da criança que acabou de ser eliminada. Para ficar mais interessante, quando o valor que consta na ficha é par, a contagem é feita no sentido horário e quando o valor que consta na ficha é ímpar, a contagem é feita no sentido anti-horário.

A brincadeira fez muito sucesso e o administrador do acampamento pediu para que sua equipe desenvolva um programa para que no próximo evento ele saiba previamente qual criança irá ser a vencedora de cada grupo, com base nas informações fornecidas.

Entrada

A entrada contém vários casos de teste. A primeira linha de cada caso de teste contém um inteiro **N** ($1 \leq N \leq 100$), indicando a quantidade de crianças que farão parte de cada círculo e participarão da brincadeira. Em seguida, as **N** linhas de cada caso de teste conterão duas informações, o **Nome** e o **Valor** ($1 \leq \text{Valor} \leq 500$) que consta na ficha de cada criança, separados por um espaço, na ordem de entrada na formação do círculo inicial.

Obs: O **Nome** de cada criança não deverá ultrapassar 30 caracteres e contém apenas letras maiúsculas e minúsculas, sem acentos, e o caractere “_”. O final da entrada é indicado pelo número zero.

Saída

Para cada caso de teste, deve-se apresentar a mensagem Vencedor(a): xxxxxx, com um espaço após o sinal ":" indicando qual é a criança do grupo que venceu a brincadeira.

Exemplo de Entrada	Exemplo de Saída
--------------------	------------------

Exemplo de Entrada	Exemplo de Saída
3	Vencedor (a) : Fernanda
Fernanda	7 Vencedor (a) : Pedro
Fernando	9 Vencedor (a) : Pedro
Gustavo	11
5	
Maria	7
Pedro	9
Joao_Vitor	5
Isabel	12
Laura	8
3	
Maria	4
Pedro	3
Joao	2
0	

Problema 7– Corrida de Rua

A corrida de rua vem se tornando uma das atividades esportivas mais praticadas em nosso país. Diante desse cenário, a assessoria esportiva *Corredores Programadores* deseja elaborar um programa que informe os números de inscrições dos atletas que não completaram uma determinada prova de corrida. Consideremos que uma competição possa ter até 100 inscritos e que os números de inscrição dos corredores são sequenciais de 1 até o número de inscritos.

Entrada

Seu programa deverá ler o número de candidatos inscritos N , o número de candidatos que concluíram a prova P e os números de inscrição P_i ($1 \leq i \leq P$) dos competidores que concluíram a prova. Para facilitar a entrada de dados, recomenda-se implementar leitura de dados de um arquivo texto no seguinte leiaute; porém, pode ser utilizada a entrada de dados padrão.

N P

$P_1 P_2 P_3 P_4 P_5 \dots P_p$

Exemplo:

10 7

8 2 4 5 1 6 7

Saída

Seu programa deve informar os números de inscrição dos competidores que não concluíram. Caso todos os competidores tenha completado a prova, será informado 0 (zero).

Exemplos

Entrada	Saída
6 4 3 2 4 5	1 6
10 7 9 8 5 3 1 2 6	4 7 10
5 5 2 4 5 3 1	0