



Projeto Olímpico de Programação **Seletiva POP 2019**

31 de agosto de 2019

(Este caderno contém 6 problemas)

A PROVA TERÁ DURAÇÃO DE **DUAS HORAS E MEIA**

LEIA ATENTAMENTE AS INSTRUÇÕES ABAIXO ANTES DE INICIAR A PROVA

- A prova deve ser realizada individualmente;
- Observem o nome do arquivo que deve ser enviado para cada problema;
- Cada questão tem um tempo limite para execução;
- É permitido consultar material impresso durante a prova;
- Não é permitida a consulta de qualquer material online;
- Este caderno de tarefas é composto de 8 páginas;
- Verifique se o caderno está completo.

Problema A – ASCII

Nome do Programa: ascii.(c|cpp|py)

Tempo: 2 segundos

Um assembler é um programa que pega um texto do código de um programa (escrito em assembly) e converte em linguagem binária. Uma das etapas desse processo de montagem é a conversão de caracteres em dados binários através de uma codificação, como a ASCII.

Sua tarefa é escrever um programa que dada uma sequência de caracteres minúsculos, imprima o código ASCII binário de cada uma delas, conforme a tabela seguinte.

Caractere	Código
a	01100001
b	01100010
c	01100011
d	01100100
e	01100101
f	01100110
g	01100111
h	01101000
i	01101001
j	01101010
k	01101011
l	01101100
m	01101101

Caractere	Código
n	01101110
o	01101111
p	01110000
q	01110001
r	01110010
s	01110011
t	01110100
u	01110101
v	01110110
w	01110111
x	01111000
y	01111001
z	01111010

Entrada

A entrada é composta por uma única palavra, formada apenas por caracteres minúsculos. A palavra pode ter até 1000 caracteres.

Saída

Para cada caractere na entrada, imprima uma linha contendo o código binário do caractere.

Casos de Teste

Entrada	Saída
pop	01110000 01101111 01110000

Entrada	Saída
abcxyz	01100001 01100010 01100011 01111000 01111001 01111010

Problema B - Bactérias

Nome do Programa: bacterias.(c|cpp|py)

Tempo: 2 segundos

Mateus é obcecado por bactérias. Para estudar o comportamento reprodutivo delas, ele criou um ambiente e colocou algumas bactérias e deseja saber as áreas contaminadas ao final do processo.

O ambiente é representado por uma matriz formada pelos caracteres X, O, B. Um campo com um X representa uma barreira que as bactérias não conseguem atravessar. Um campo com O representa um ambiente no qual uma bactéria consegue se reproduzir. Por fim, um B representa um ambiente já contaminado por bactérias.

Através do processo reprodutivo das bactérias, elas conseguem contaminar todos os campos propícios ao redor dela. Veja o exemplo seguinte:

```

OOXO   OBXO   BBXO
XBOX → XBBX → XBBX
XXXO   XXXO   XXXO

```

Observe que no terceiro estágio, não há mais lugar para as bactérias se reproduzirem, por causa barreiras que estão ao redor.

Entrada

A primeira linha da entrada contém dois inteiros **N** e **M** ($1 \leq N, M \leq 100$) indicando a quantidade de linhas e colunas da matriz. Cada uma das próximas **N** linhas contém **M** caracteres cada (cada caractere é uma letra B, O ou X, com significado de acordo com o descrito acima).

Saída

A saída deve conter a matriz ao final do processo de reprodução das bactérias.

Casos de Teste

Entrada	Saída
3 4 OOXO XBOX XXXO	BBXO XBBX XXXO

Problema C – Canos

Nome do Programa: canos.(c|cpp|py)

Tempo: 2 segundos

A Conductorora possui um sistema de encanamento muito complexo, projetado pelo digníssimo Wesley, um dos maiores encanadores da Ilustre Fluxo (IF).

O sistema de encanamento consiste em um único cano, que se ramifica por toda a empresa. O ramo principal se divide em A_1 ramos iguais. Cada um destes, por sua vez, se divide em A_2 ramos. E cada um destes, em A_3 ramos. E assim por diante, até obter-se os ramos finais, que não se dividem mais.

Sua tarefa é dizer quantos ramos finais existem na empresa, dada a sequência de divisões seguida pelo encanamento.

Entrada

A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 25$) indicando a quantidade de vezes que todos os canos se ramificam. A próxima linha contém N inteiros A_i ($1 \leq A_i \leq 1000$) representando em quantos ramos se dividem os canos na i -ésima divisão.

Saída

A saída deve conter um único inteiro X indicando a quantidade de ramos finais do encanamento da Conductorora. É garantido que $1 \leq X \leq 10^9$.

Casos de Teste

Entrada	Saída
4 2 3 1 4	24

Entrada	Saída
10 2 2 2 3 3 4 3 3 2 2	10368

Problema D - Difícil

Nome do Programa: `dificil.(c|cpp|py)`

Tempo: 2 segundos

Mateus está sempre certo e todo mundo sabe disso. Desta vez, ele entrou em uma discussão com a diretoria do POP sobre a dificuldade dos problemas desta seleção.

Alguns da diretoria insistem que existem problemas fáceis. Outros até afirmam que todos os problemas são fáceis. Mateus discorda. Para ele, todos os problemas são difíceis.

Sua tarefa é resolver esse impasse. Dada a letra ou o nome do problema, você deve informar se o problema é fácil, médio ou difícil.

Entrada

A entrada é composta por uma única linha contendo uma palavra que pode ser uma letra de A a F ou o nome de algum dos problemas desta prova, em maiúsculas e sem acentos.

Saída

Para cada entrada, sua saída deve informar se o problema é fácil, médio ou difícil, imprimindo FACIL, MEDIO, ou DIFICIL, respectivamente.

Casos de Teste

Entrada	Saída
D	DIFICIL

Entrada	Saída
DIFICIL	DIFICIL

Problema E – Explosão

Nome do Programa: explosao.(c|cpp|py)

Tempo: 2 segundos

Muito revoltado, Thiago resolveu explodir coisas. Ele comprou diversos explosivos e os ativou, um a um, exatamente no mesmo local. Depois de se acalmar, ele percebeu que tinha feito um estrago grande, mas queria estimar a área afetada.

Considere que cada explosivo atinge uma área circular ao seu redor do ponto no qual ele foi ativado. Dados os raios de alcance de cada explosivo, você consegue estimar a área total atingida? Como os cálculos são apenas estimativas, considere o valor de π como sendo 3.

Entrada

A primeira linha da entrada contém um inteiro **N** ($1 \leq N \leq 10^7$) representando a quantidade de explosivos ativados por Thiago. A segunda linha contém **N** inteiros **R_i** ($1 \leq R_i \leq 10^3$) indicando o raio de alcance (em metros) de cada explosivo **i**.

Saída

A saída deve conter um único inteiro representando a área atingida aproximada em metros quadrados.

Casos de Teste

Entrada	Saída
3 2 1 3	27
Entrada	Saída
4 2 1 9 10	300
Entrada	Saída
5 9 8 11 7 5	363

Problema F – Fácil

Nome do Programa: facil.(c|cpp|py)

Tempo: 2 segundos

Cansado de escrever questões para a seletiva do POP, Kerven resolveu ajudar os competidores e fazer uma questão muito fácil de se resolver.

O problema é simples: dado um valor que vai ser dividido igualmente para várias pessoas, diga quanto cada uma vai precisar pagar. Para facilitar ainda mais a vida, os valores serão dados em centavos, assim nem com vírgula é necessário se preocupar.

Note que não é possível alguém pagar menos de um centavo. Então caso a divisão não seja exata, arredonde para cima, mas informe o quanto de dinheiro irá sobrar no final. Por exemplo, se deve-se dividir 9999 centavos para 2 duas pessoas, cada uma irá pagar 5000 centavos e no final sobrá 1 centavo.

Entrada

A entrada é composta por uma única linha com dois inteiros V ($1 \leq V \leq 10^7$), o valor em centavos a ser dividido, e P ($1 \leq P \leq 10^4$), a quantidade de pessoas que dividirão o valor V .

Saída

Caso a divisão seja exata, a saída deve conter um único inteiro informando quanto cada pessoa deve pagar. Caso contrário, a saída deve conter o inteiro indicando quanto cada pessoa deve pagar e, na linha seguinte, a mensagem SOBRA X , onde X é o valor excedente ao juntar a contribuição de todas as pessoas.

Casos de Teste

Entrada	Saída
56815 11	5165

Entrada	Saída
9999 2	5000 SOBRA 1

Entrada	Saída
16484 78	212 SOBRA 52