

Laboratório de Programação

Trabalho em equipe

Regras para o Trabalho

- 1 Em caso de plágio, inclusive na Internet, as equipes envolvidas serão prejudicadas não importando quem realmente resolveu a questão.
- 2 O professor rodará um conjunto de testes diferente dos exemplos que constam no problema. Cada solução deve passar em todos os testes para ser considerada correta.
- 3 Cada aluno da equipe será interrogado pelo professor sobre as soluções da equipe, o que equivalerá a uma nota individual atribuída ao aluno.
- 4 O trabalho será avaliado em reunião da equipe com o professor, e o aluno ausente perderá todos os pontos individuais.
- 5 Perguntas referentes ao trabalho podem ser feitas através do Google Sala de Aula ou grupo de whatsapp.

Missão 01: Listas em Java

Lista é uma estrutura de dados na qual elementos de um mesmo tipo de dado estão organizados de maneira sequencial. Não necessariamente, estes elementos estão fisicamente em sequência, mas a ideia é que exista uma ordem lógica entre eles.

Em Java as Listas são definidas através da Interface List, a qual define os métodos comuns para manipulação deste tipo de estrutura. A Interface List é implementada por classes como: ArrayList e LinkedList.

- 1 Pesquise sobre as duas implementações de Listas em Java, ArrayList e LinkedList:
 - 1.a Como são implementadas?
 - 1.b Como o ArrayList se comporta quando sua capacidade é alcançada? Porque o LinkedList não apresenta o mesmo problema?
 - 1.c Quais os problemas de performance de cada uma das implementações?
- 2 Implemente uma versão do Crivo de Eratóstenes para cada uma das implementações de Listas em Java: ArrayList e LinkedList. Faça uma comparação de tempo de execução entre as duas implementações (para tomada de tempo utilize o método **`System.currentTimeMillis()`**). Para cada uma das implementações teste para os seguintes valores:
 - 2.a $N = 10.000$
 - 2.b $N = 100.000$
 - 2.c $N = 1.000.000$
- 3 Expliquem porque as diferenças de tempo entre a performance das duas implementações de listas em Java para o problema proposto.
- 4 O que vocês podem fazer para melhorar a performance do algoritmo com pior desempenho?

Crivo de Eratóstenes

O Crivo de Eratóstenes é um algoritmo, bem como um método simples e prático para encontrar números primos até um certo valor limite. Segundo a tradição, foi criado pelo matemático grego Eratóstenes (a.c. 285-194 a.C.), o terceiro bibliotecário-chefe da Biblioteca de Alexandria.

Dado um número inteiro N , você deve encontrar todos os números primos entre 2 e N ?

Informações sobre o algoritmo em: https://pt.wikipedia.org/wiki/Crivo_de_Eratóstenes

O MELHOR DIVISOR

Carla adora brincar e comparar números. Ela acha que se ela pegar dois números positivos diferentes, aqueles cujos dígitos somam um número maior é melhor que o outro. Se a soma dos dígitos for igual para ambos os números, então ela acha que o número menor é melhor. Por exemplo, Carla acha que 13 é melhor que 31 e que 12 é melhor que 11.

Dado um número inteiro N , você pode encontrar o divisor de N que Carla considerará ser o melhor?

Formato de entrada

Um único inteiro denotando N .

Formato de saída

Imprima um inteiro denotando o melhor divisor de N segundo Carla.

Exemplo de Entrada

12

Exemplo de Saída

6

Explicação

O conjunto de divisores de 12 pode ser expresso como [1, 2, 3, 4, 6, 12]. O divisor cujos dígitos somam o maior número é 6 (o qual tem apenas um dígito). Assim, imprimimos como nossa resposta.

FATORES PRIMOS DE LEONARDO

Leonardo ama números primos e criou C consultas onde cada consulta assume a forma de um inteiro, N. Para cada N, ele quer que você apresente todos fatores primos exclusivos de qualquer número no intervalo inclusivo [1, N] e imprima esse valor em uma nova linha.

Nota: Lembre-se de que um número primo só é divisível por 1 e por ele mesmo, e 1 não é um número primo.

Formato de entrada

A primeira linha contém um inteiro Q denotando o número de consultas.

Cada uma das linhas subsequentes contém um único inteiro N, a ser consultado.

Formato de saída

Para cada consulta, imprima os fatores primos exclusivos para qualquer número no intervalo inclusivo [1, N] em uma nova linha.

Exemplo de Entrada

```
6
1
2
3
500
5000
10000000000
```

Exemplo de Saída

```
[]
[2]
[3]
[ 2, 3, 5, 7]
[ 2, 3, 5, 7, 11]
[ 2, 3, 5, 7, 11, 13, 17, 19, 23, 29]
```

Explicação

O número máximo de fatores primos únicos de qualquer número na faixa inclusiva [1,1] é 0, porque 1 não é primo e seu único fator é ele mesmo.

O número máximo de fatores primos únicos de qualquer número na faixa inclusiva [1,2] é 1. Já sabemos que o número 1 tem 0 fatores primos, mas 2 tem 1 fator primo (ele mesmo).

O número máximo de fatores primos únicos de qualquer número na faixa inclusiva [1,3] é 1. O número 3 tem 1 fator primo (ele mesmo).

O número máximo de fatores primos únicos na faixa inclusiva [1, 500] é 4. O produto dos nossos primeiros quatro primos únicos é $2 \times 3 \times 5 \times 7 = 210$, e não há primos únicos adicionais que possamos multiplicar por esse número que resulte em um valor menor ou igual a 500.

MÚLTIPLO ESPECIAL

Você recebe um número inteiro N . Você pode encontrar o menor inteiro positivo X composto de apenas de 9 e 0, de tal forma que X é um múltiplo de N ? Onde X é composto de uma ou mais ocorrências de 9 e zero ou mais ocorrências de 0.

Formato de entrada

A primeira linha contém um inteiro T ($1 \leq T \leq 10^4$) que indica o número de casos de teste. Seguem T linhas, cada linha contém o inteiro N ($1 \leq N \leq 500$) para o qual a solução deve ser encontrada.

Formato de saída

Imprima a resposta X correspondente a cada caso de teste. A saída não deve conter zeros à esquerda.

Exemplo de Entrada

```
3
5
7
1
```

Exemplo de Saída

```
90
9009
9
```

Explicação

90 é o menor número composto de 9 e 0 divisíveis por 5. Da mesma forma, você pode derivar para outros casos.

FIBONACCI

Dados três números: A, B, e N, e tudo o que você precisa fazer é encontrar o número F_n onde:

$$F_0 = A$$

$$F_1 = B$$

$$F_i = F_{i-1} + F_{i-2}, \text{ para } i \geq 2$$

Formato de entrada

A primeira linha contém um único inteiro T (o número de testes). Seguem T linhas, cada uma contendo três inteiros: A, B, e N.

Formato de saída

Para cada caso de teste, imprima um único inteiro representando F_n .

Exemplo de Entrada

```
8
2 3 1
9 1 7
9 8 3
2 4 9
1 7 2
1 8 1
4 3 1
3 7 5
```

Exemplo de Saída

```
3
85
25
178
8
8
3
44
```

Explicação

O primeiro caso de teste é óbvio.

Vamos dar uma olhada no segundo:

$$F_0 = 9, F_1 = 1, F_2 = 1 + 9 = 10, F_3 = 10 + 1 = 11, F_4 = 11 + 10 = 21$$

$$F_5 = 21 + 11 = 32, F_6 = 32 + 21 = 53, F_7 = 53 + 32 = 85$$

OBS: PONTUAÇÃO EXTRA PARA QUEM FIZER UMA VERSÃO RECURSIVA.

Duro de Matar

Simon: Na fonte, deve haver 2 jarros – um de 5 litros e outro de 3 litros. Você os vê? Encha um dos jarros com exatamente 4 litros de água e coloque-o na balança e o cronômetro irá parar. Você deve ser preciso; um pouco a mais ou menos resultará em detonação. Se você ainda estiver vivo em 5 minutos, falaremos.

Bruce: Espere, espere um segundo. Eu não entendo. Você entendeu?

Samuel: Não.

Bruce: Pegue os jarros. Obviamente, não podemos encher o jarro de 3 litros com 4 litros de água.

Samuel: Obviamente.

Bruce: Tudo bem. Eu sei, aqui vamos nós. Nós enchamos o jarro de 3 litros exatamente até o topo, certo?

Samuel: Uh hum.

Bruce: Ok, agora nós colocamos esses 3 litros no jarro de 5, nos dando exatamente 3 litros no jarro de 5 litros, certo?

Samuel: Certo, então o que?

Bruce: Pegamos o jarro de 3 litros e enchemos um terço dele...

Samuel: NÃO! Ele disse: "Seja preciso". Exatamente 4 litros.

Bruce: DROGA! Todo policial dentro de 50 milhas está correndo e eu estou aqui jogando jogos infantis no parque.

Samuel: Ei, você quer se concentrar no problema em questão?

1) Dados dois jarros de capacidade A e B litros, e um suprimento infinito de água, você pode encher um dos jarros com exatamente N litros de água? Se sim apresente uma solução?

Formato de entrada

A entrada de dados consiste em 3 inteiros A, B e N. A e B indicam a capacidade dos dois jarros

respectivamente, e N denota a capacidade exata com a qual um dos jarros deve ser preenchido.

Formato de saída

Devem ser impressos os passos de uma possível solução para o problema, se um dos jarros não puder ser preenchido com exatamente N litros de água, então deve ser impresso "SEM SOLUÇÃO".

Exemplo de Entrada

2

3 6 4

5 3 4

Exemplo de Saída

3 6 4

SEM SOLUÇÃO

5 3 4

Enche A --> A = 5

Transfere de A para B --> A = 2, B = 3
Esvazia B --> B = 0
Transfere de A para B --> A = 0, B = 2
Enche A --> A = 5
Transfere de A para B --> A = 4, B = 3

2) Melhore o algoritmo para que o mesmo apresente a melhor solução possível, considerando que a melhor solução será aquela que apresentará o menor número de passos.