

## Clones das sombras



Em outro dia comum na aldeia da folha, nosso querido naruto estava tranquilo comendo seu lámen quando foi surpreendido pela tentativa de invasão de dezenas de ninjas da aldeia da areia.

Para enfrentar tantos inimigos, a melhor arma do naruto é o jutsu clone das sombras. Porém, o lámen não caiu muito bem e naruto não está com sua força total. Então, **ele só consegue fazer um clone por vez, e cada clone também só consegue invocar mais um único clone**, ou seja, **a quantidade de narutos só pode ser dobrada a cada jutsu**.

A sua tarefa é fazer um programa que seja capaz de dizer **se será possível ter um número de narutos que seja exatamente igual ao número de inimigos**.

### Entrada

A primeira linha da entrada consiste de um inteiro ‘N’ ( $1 \leq N \leq 5000$ ), que representa **a quantidade de ninjas inimigos**.

### Saída

A saída consiste em uma linha contendo “**Dattebayo**”, caso seja possível ter o **número de narutos exatamente igual ao número de inimigos**, ou “**Tururuuu**”, caso não seja possível.

Entrada	Saída
32	Dattebayo
50	Tururuuu
16	Dattebayo
3	Tururuuu
4	Dattebayo
1	Dattebayo

# Clones

Autor: Marcos Fernandes

Durante a Quarta Guerra Ninja, Madara Uchiha utilizava a rede social Sharigram para enviar mensagens criptografadas para sua tropa e com medo que as ordens caíssem nas mãos da Aliança Shinobi, enviava ondas com uma quantidade de Zetsus Brancos para destruir a Vila Oculta da Computação. Naruto a fim de proteger a vila, enviou uma quantidade inicial de clones das sombras para proteção e toda vez que Madara enviava mais Zetsus, Naruto enviava também mais clones para ajudar.



Saiba que os clones de Naruto duram toda a guerra e que cada um consegue enfrentar apenas 1 Zetsu Branco, se aniquilando mutuamente. Sua tarefa será descobrir se Naruto conseguiu proteger a Vila Oculta da Computação. Madara vence se em algum momento o número de Zetsu Brancos for maior do que a quantidade de clones de Naruto na vila.

## Entrada

A primeira linha contém um inteiro “**N**” ( $1 \leq N \leq 100$ ), que representa a quantidade de ondas de Zetsus Brancos que Madara irá enviar, e um inteiro “**C**” ( $0 \leq C \leq 100$ ), que será a quantidade de clones do Naruto para a proteção inicial. Cada uma das próximas “**N**” linhas representa uma onda, que será composta por um inteiro “**Z**” ( $0 \leq Z \leq 100$ ), representando a quantidade de Zetsus brancos da onda e um inteiro “**S**” ( $0 \leq S \leq 100$ ), que será a quantidade de clones enviados por Naruto naquela onda.

## Saída

A saída será “**Naruto defendeu a Vila**”, caso os clones de Naruto tenham contido os avanços dos Zetsus Brancos ou “**Madara venceu**”, caso a vila tenha sido destruída.

## Exemplos

Entrada	Saída
3 20 30 20 60 40 50 75	Madara venceu
5 100 80 60 100 30 50 50 10 30 40 45	Naruto defendeu a Vila

# Defesa de Ba Sing Se

Água, Terra, Fogo, Ar. Há muito tempo, as quatro nações viviam em harmonia, até que tudo mudou quando a nação do fogo decidiu atacar. Atualmente, a grande cidade de Ba Sing Se é a última grande metrópole que resiste à nação do fogo. Porém, com o retorno do avatar, a nação do fogo decidiu começar uma campanha contra a grande cidade.

Sabendo da campanha da nação do fogo e conhecendo seus poderes de oráculo, o avatar pediu sua ajuda para saber se a muralha da cidade resistirá aos ataques até a sua chegada, ou se ela cairá.



## Entrada

A primeira linha contém dois inteiros '**D**' ( $1 \leq D \leq 100$ ), a quantidade de dias que o avatar irá levar para chegar, e '**E**' ( $100 \leq E \leq 10000$ ), a energia inicial da muralha da cidade. As próximas '**D**' linhas terão cada uma um inteiro '**A**' ( $1 \leq A \leq 100$ ) que representa o dano que a nação do fogo fez à energia da muralha em cada dia '**D**'. Se a energia da muralha chegar a zero, a cidade cairá.

## Saída

A saída consiste em uma única linha. Caso a energia da muralha seja maior que zero no último dia de ataque antes da chegada do Avatar, você deverá imprimir a mensagem "**Resistiu**", ou, caso contrário, você deverá imprimir o dia em que a cidade caiu, ou seja, o dia em que a energia da muralha chegou a zero.

Entrada	Saída
4 400 (4 dias e a energia inicial da muralha) 100 (energia – 100 = 300) (primeiro dia) 100 (energia – 100 = 200) (segundo dia) 100 (energia – 100 = 100) (terceiro dia) 99 (energia – 99 = 1) (quarto e último dia)	Resistiu
3 111 (3 dias e energia inicial = 111) 10 (energia – 10 = 101) 100 (energia – 100 = 01) 1 (energia – 1 = 0) (terceiro dia)	3
3 100 50 60 (segundo dia) 30	2

Obs.: O que está escrito em vermelho é apenas explicação, não faz parte da entrada.

## Desafio Tático

Em um jogo de estratégia, os jogadores possuem um exército de soldados numerados de 1 a N. Cada soldado possui um valor de ataque e um valor de defesa. Durante uma batalha, os jogadores podem escolher uma formação de soldados para enfrentar o exército inimigo.

Escreva um programa que, dado o número de jogadores, a quantidade de soldados que cada jogador possui e as informações de ataque e defesa dos soldados, calcule a soma dos valores de ataque e defesa para cada jogador.

### Entrada

Um número inteiro '**P**', representando o número de jogadores.

Um número inteiro '**S**', representando a quantidade de soldados que cada jogador possui.

'**P**' blocos de '**S**' linhas cada, onde cada linha contém dois números inteiros separados por espaço. O primeiro número representa o valor de ataque e o segundo número representa o valor de defesa de um soldado. Os valores de ataque e defesa estão na faixa de 1 a 100.

### Saída

'**N**' linhas com dois números inteiros separados por espaço, representando a soma dos valores de ataque e defesa para cada jogador.

Entrada	Saída
3	45 23
4	32 21
10 5	37 21
15 8	
8 3	
12 7	
5 2	
9 6	
7 4	
11 9	
6 1	
13 10	
10 4	
8 6	

# Desenhista

Autor: Jefferson J. Raimon



Mário cansou de pular em cogumelos, tubulações e realizar saltos em pirâmides. A idade chegou e agora ele está aposentado, e no seu tempo livre, ele adora desenhar.

Dessa vez ele está praticando desenhar pirâmides, lembrando dos velhos tempos. Ajude-o imprimindo os blocos dessa pirâmide.

## Entrada:

A entrada consiste em um número inteiro '**P**' ( $1 \leq P \leq 20$ ), que representa a altura que a pirâmide terá.

## Saída:

Na saída você deverá imprimir a pirâmide com o caractere '**#**', conforme a quantidade de linhas solicitadas, e nos locais vagos, usar o caractere '**>**', como no exemplo abaixo.

Entrada	Saída
7	>>>>>># >>>>>## >>>>### >>>#### >>##### >##### #####

# Detectando óleo no oceano

Autora: Nicole Silva

Alguns alunos da UFBA estão trabalhando em um projeto de Machine Learning (Aprendizado de Máquina), com o objetivo de detectar manchas de óleo no oceano. Para isso é necessário criar um programa que, ao receber imagens de um satélite, identifique padrões que indicam a presença desse resíduo no oceano. Para o trabalho não ficar pesado os alunos pediram a sua ajuda.

Dado uma sequência de números 0 ou 1, cada um representando uma determinada área, você deve identificar o grau de exposição ao óleo que a biodiversidade daquele local está sofrendo. O valor 0 indica a ausência de manchas de óleo naquela área e o valor 1 indica a presença.



## Os graus de exposição são:

Baixo - Quantidade total de manchas de óleo menor que 30%.

Médio - Quantidade total de manchas de óleo entre 30% e 50%.

Alto - Quantidade total de manchas de óleo maior que 50%.

(Graus e dados fictícios)

## Entrada

Na primeira linha será dado um inteiro “N” ( $1 \leq N \leq 1000$ ), representando o número de áreas analisadas. A seguir teremos “N” linhas, sendo que em cada uma será dado um inteiro M (0 ou 1), indicando a presença ou ausência de manchas de óleo na imagem daquela área.

## Saída

A saída consiste em uma das frases abaixo:

- “**Regiao segura**”, caso o grau de exposição seja baixo.
- “**Regiao em estado de alerta**”, caso o grau de exposição seja médio.
- “**Regiao com alto indice de perda de biodiversidade**”, caso o grau de exposição seja alto.

## Exemplos

Entrada	Saída
5 0 1 1 0 1	Regiao com alto indice de perda de biodiversidade
4 0 0 1 0	Regiao segura



## Escolha do Campeão



League of Legends está cheio de campeões de vários tipos, de mentes malignas a monstros épicos. Diferentes campeões assumem diferentes papéis e usam diferentes estratégias. Lucas sempre joga LOL e gosta de variar na sua escolha de campeão, **dessa vez ele quer jogar com o campeão de maior nível de poder** dentre os que ele mais gosta.

**Dado um número ‘N’, que representa a quantidade de campeões favoritos de Lucas, ajude ele a decidir qual deles tem o maior nível.**

### Entrada

A entrada consiste na primeira linha de um número inteiro ‘N’ ( $1 < N < 100$ ) que representa **o número de campeões favoritos de Lucas**. Nas próximas ‘N’ linhas será fornecido em cada uma, um número inteiro ‘P’ ( $0 \leq P \leq 10000$ ) que representa **o nível de poder de cada um dos ‘N’ campeões**.

### Saída

A saída deverá **conter apenas o nível de poder do campeão mais forte**.

Entrada :	Saída :
3 1500 3600 500	3600
7 300 5200 540 729 3567 480 4000	5200



# Pirâmide de limonadas

## Autor: Vinicius Martins



Sanji é um cozinheiro dedicado, que sempre busca atender os desejos da sua tripulação e hoje ele decidiu fazer algo diferente. Sanji decidiu organizar taças de limonada em uma pilha em forma de pirâmide, onde a altura é de '**N**' níveis e cada nível da pirâmide terá ' **$J * 2 - 1$** ' ( **$1 \leq J \leq N$** ) taças, sendo '**J**' o nível na pirâmide, sendo **o topo como 1 e a base como 'N'**. Note que as taças de cada nível devem ser representadas pelo número (algarismo numérico) que indica seu nível '**J**' na pirâmide, ou seja, as taças da base são representadas pelo algarismo '**N**' e a taça do topo pelo algarismo '**1**'. Vamos ajudar Sanji a impressionar seus companheiros?

### Entrada

A entrada possui um número inteiro '**N**' ( **$1 \leq N \leq 9$** ) representando o número de níveis da pilha piramidal.

### Saída

A saída consiste em '**N**' linhas que representam os níveis da pirâmide **em ordem crescente**, ou seja, **nível com mais taças por último**. Cada nível **começa com '**N - J**'** espaços em branco, seguidos pelos ' **$J * 2 - 1$** ' algarismos que **representam as taças** daquela fileira.

Entrada	Saída
2	1 222
7	1 222 33333 4444444 55555555 6666666666 7777777777777
1	1

# Vulcão

Autor: Gabriel Dahia

Você é o responsável pelo próximo turno na supervisão do vulcão Kilauea. Seu trabalho consiste em monitorar '**P**', a pressão interna do vulcão, e, caso ela ultrapasse '**T**', o limite para erupção, você deve soar o alarme.

## Entrada

A primeira linha da entrada consiste de '**T**' ( $1 \leq T \leq 100000$ ), um inteiro correspondente ao **limite de pressão para erupção**. As próximas linhas contêm inteiros '**P**' ( $1 \leq P \leq 100000$ ), denotando **as medições da pressão interna do vulcão**, até que **a entrada termine com 0 (zero)**.



## Saída

Seu programa deve imprimir "**ALARME**", assim que ocorrer ' $P > T$ ', e "**O Havai pode dormir tranquilo**", caso não haja medições acima de '**T**'. Portanto, seu programa deve imprimir uma das frases somente uma vez, de acordo com as condições dadas.

Entrada	Saída
4 2 3 4 5 0	ALARME
100 5 10 15 20 99 50 0	O Havai pode dormir tranquilo

# Bolinhas de Gude

O natal está chegando e Yuri já deixou avisado a seus familiares que gostaria de receber bolinhas de gude como presente este ano. Yuri sabe que seus familiares têm um hábito peculiar quando o assunto é presentear com bolinhas de gude, pois sempre que um familiar lhe presenteia com uma quantidade de bolinhas de gude o próximo familiar presenteia com o dobro do anterior. Como Yuri sabe que você é um ótimo programador, ele pediu para que você desenvolva um programa que dada a quantidade de familiares que irá presentear-lo com bolinhas de gude e a quantidade de bolinhas de gude que o primeiro familiar lhe dará, informe a quantidade total de bolinhas de gude que ele irá receber de presente de natal.



## Entrada

A entrada é composta por um número inteiro **N** ( $1 \leq N \leq 50$ ), que representa a quantidade de familiares que irão presentear-lo com bolinhas de gude e um inteiro **Q** ( $1 \leq Q \leq 50$ ), que representa a quantidade de bolinhas que o primeiro familiar dará de presente.

## Saída

A saída consiste em imprimir a quantidade total de bolinhas de gude que Yuri irá receber.

Entrada	Saída
10 2	2046
5 15	465