# Logotipo Descrição gerada automaticamenteINSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ

**Especialização em Programação para Dispositivos Móveis Lógica de Programação**

# Ensino à Distância

Avaliação Final - Lógica de Programação

* As questões a seguir foram extraídas do *BeeCrowd* (<https://www.beecrowd.com.br/judge/pt/>).
* O aluno pode ficar à vontade para tentar resolver a questão diretamente no site, lembrando de usar a linguagem JavaScript.
* A avaliação possui 7 questões e os alunos podem escolher 5 questões para resolução, duas questões são extra e também podem ser resolvidas.
* A entrega dos códigos pode ser realizada por quaisquer meios em que seja possível copiar e executar os códigos, por exemplo: arquivos de texto, repositório como github, sites de compartilhamento como codepen, replit, JSFiddle, etc...
* Os códigos serão verificados no beecrowd
* Alguns tutoriais de uso do Beecrowd:
  + <https://youtu.be/7ECer9gnjpo>

1. **Distância (1016 -** https://[www.beecrowd.com.br/judge/pt/problems/view/1016)](http://www.beecrowd.com.br/judge/pt/problems/view/1016))

Dois carros (X e Y) partem em uma mesma direção. O carro X sai com velocidade constante de 60 Km/h e o carro Y sai com velocidade constante de 90 Km/h.

Em uma hora (60 minutos) o carro Y consegue se distanciar 30 quilômetros do carro X, ou seja, consegue se afastar um quilômetro a cada 2 minutos.

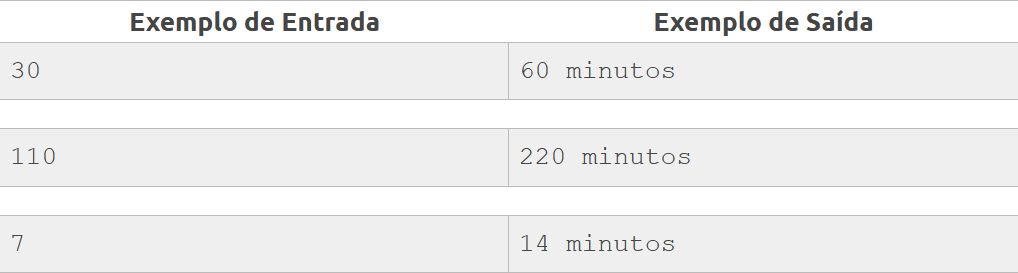
Leia a distância (em Km) e calcule quanto tempo leva (em minutos) para o carro Y tomar essa distância do outro carro.

# Entrada

O arquivo de entrada contém um número inteiro.

# Saída

Imprima o tempo necessário seguido da mensagem "minutos".



1. **Idade em Dias (1020 -** <https://www.beecrowd.com.br/judge/pt/problems/view/1020>)

Leia um valor inteiro correspondente à idade de uma pessoa em dias e informe-a em anos, meses e dias

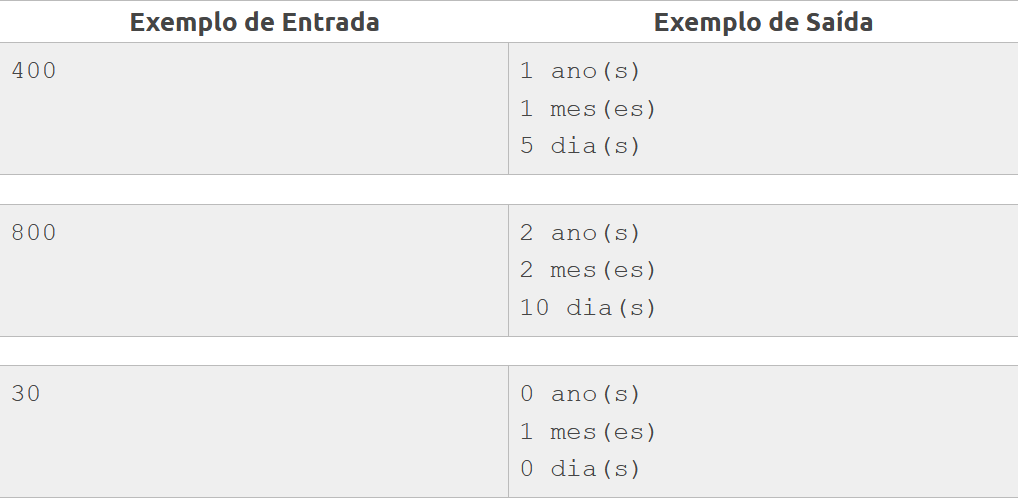
Obs.: apenas para facilitar o cálculo, considere todo ano com 365 dias e todo mês com 30 dias. Nos casos de teste nunca haverá uma situação que permite 12 meses e alguns dias, como 360, 363 ou 364. Este é apenas um exercício com objetivo de testar raciocínio matemático simples.

# Entrada

O arquivo de entrada contém um valor inteiro.

# Saída

Imprima a saída conforme exemplo fornecido.



1. **Seleção em Vetor I (1174 -** <https://www.beecrowd.com.br/judge/pt/problems/view/1174>)

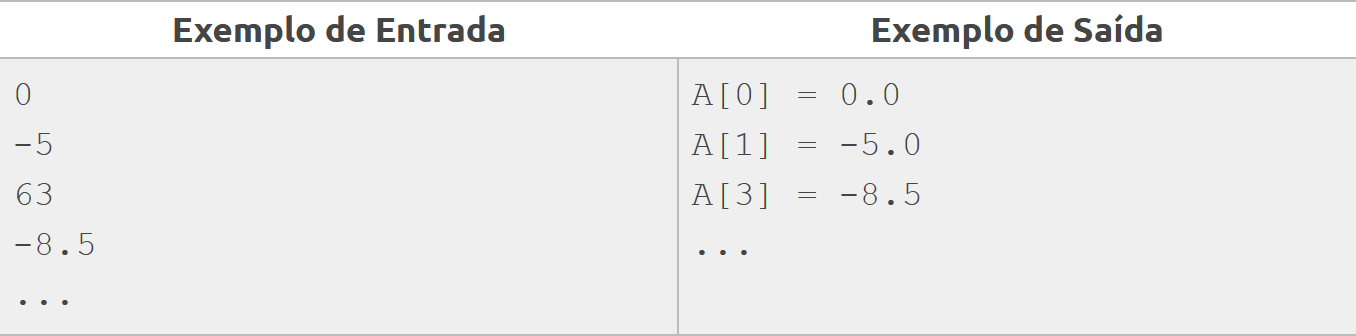
Faça um programa que leia um vetor A[100]. No final, mostre todas as posições do vetor que armazenam um valor menor ou igual a 10 e o valor armazenado em cada uma das posições. Dica: O valor de *length* do vetor *lines* no beecrowd é igual a 101.

# Entrada

A entrada contém 100 valores, podendo ser inteiros, reais, positivos ou negativos.

# Saída

Para cada valor do vetor menor ou igual a 10, escreva "A[i] = x", onde i é a posição do vetor e x é o valor armazenado na posição, com uma casa após o ponto decimal.



1. **Dividindo X por Y (1116 -** <https://www.beecrowd.com.br/judge/pt/problems/view/1116>)

Escreva um algoritmo que leia 2 números e imprima o resultado da divisão do primeiro pelo segundo. Caso não for possível mostre a mensagem “divisao impossivel” para os valores em questão.

# Entrada

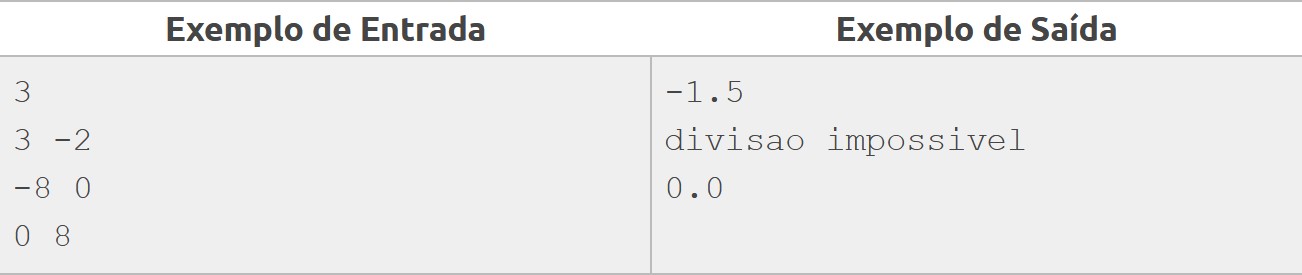
A entrada contém um número inteiro N. Este N será a quantidade de pares de valores inteiros (X e Y) que serão lidos em seguida.

# Saída

Para cada caso mostre o resultado da divisão com um dígito após o ponto decimal,

ou “divisao impossivel” caso não seja possível efetuar o cálculo.

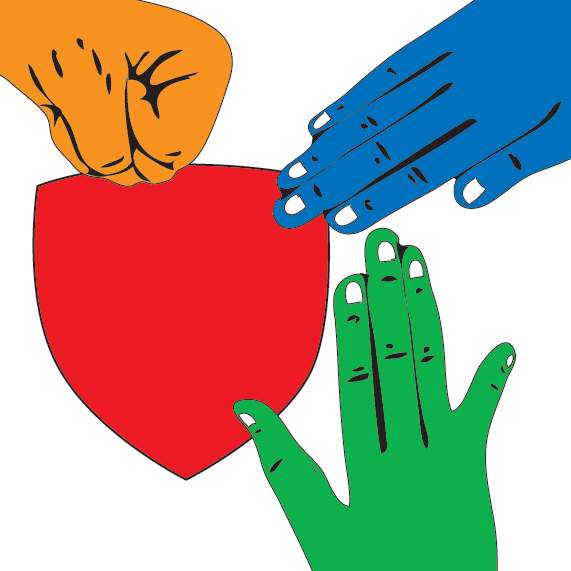
Obs.: Cuide que a divisão entre dois inteiros em algumas linguagens como o C e C++ gera outro inteiro. Utilize cast :)



1. **Pedra, Papel, Ataque Aéreo (2031 -** [https://tinyurl.com/3mx8w7fe](https://www.beecrowd.com.br/judge/pt/problems/view/2031))

Pedra, Papel, Ataque Aéreo é um jogo infantil muito popular, em que duas ou mais crianças formam um círculo e fazem gestos com a mão na tentativa de obter a vitória. As regras são surpreendentemente complexas para um jogo de crianças, mesmo assim é bastante popular por todo o mundo.

As partidas são muito simples. Os jogadores podem escolher entre o sinal de uma Pedra (o punho), o sinal de um Papel (a palma aberta), e o sinal para o Ataque Aéreo (igual ao do Papel, mas com apenas o polegar e o mindinho estendidos).



Uma partida com dois jogadores possui as seguintes regras para se definir um vencedor:

* + Ataque Aéreo vs. Pedra: Neste caso, o jogador com o Ataque Aéreo derrota o jogador com a Pedra, por razões óbvias.
  + Pedra vs. Papel: Neste caso, o jogador com a Pedra derrota o com Papel, porque a Pedra machuca muito mais.
  + Papel vs. Ataque Aéreo: Aqui o Ataque Aéreo ganha, porque Ataque Aéreo sempre ganha e o Papel é patético.
  + Papel vs. Papel: Nesta variação, ambos os jogadores ganham, porque o Papel é inútil e ninguém que enfrenta o Papel pode perder.
  + Pedra vs. Pedra: Para este caso não há ganhador, porque depende do que os jogadores decidem fazer com a Pedra e normalmente não fazem nada.
  + Ataque Aéreo vs. Ataque Aéreo: Quando isto acontece, todos os jogadores perdem, devido a Aniquilação Mútua.

Sua tarefa é escrever um programa que, dada as escolhas de dois jogadores, informe quem venceu o jogo.

# Entrada

A entrada consiste de N (1 ≤ N ≤ 1000) casos de teste. N deve ser lido na primeira linha da entrada. Cada caso de teste é composto por duas linhas, cada uma contendo uma string. A primeira string representa o sinal escolhido pelo jogador 1 e a segunda string representa o sinal escolhido pelo jogador 2. Essas strings podem ser:

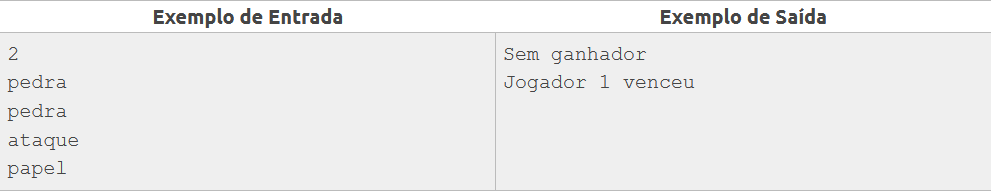
* + “ataque”: para representar o Ataque Aéreo
  + “pedra”: para representar a Pedra
  + “papel”: para representar o Papel

# Saída

A saída deve conter o seguinte:

* + “Jogador 1 venceu”: se o Jogador Um tiver vencido a partida
  + “Jogador 2 venceu”: se o Jogador Dois tiver vencido a partida
  + “Ambos venceram”: se os dois jogadores tiverem vencido a partida
  + “Sem ganhador”: se não houver ganhador
  + “Aniquilacao mutua”: se ocorrer Aniquilação Mútua

Cada saída de um caso de teste deve estar em uma linha.



1. **A Sociedade do Anel (2949 -** [https://tinyurl.com/cn7yu9vw](https://www.beecrowd.com.br/judge/pt/problems/view/2949))

Frodo era um pequeno hobbit (pessoinhas pequenas e de pés peludos) que vivia tranquilamente no Condado, tomando seus vários cafés da manhã recheados de muitos alimentos suculentos que a dieta de um bom hobbit proporciona. Certo dia, seu tio Bilbo lhe entrega seu famoso anel dourado, e Gandalf, um mago muito “bacanudo”, diz a Frodo que esse anel não era normal e que deveria ser jogado na Montanha da Perdição, para que um grande mal fosse evitado. Para essa jornada, foi formada uma comitiva, composta de anões, elfos, humanos, hobbits e magos.

Frodo deseja saber a quantidade de cada raça que irá com ele para a jornada. Dada uma lista das pessoas que se alistaram, faça um relatório para Frodo da comitiva.

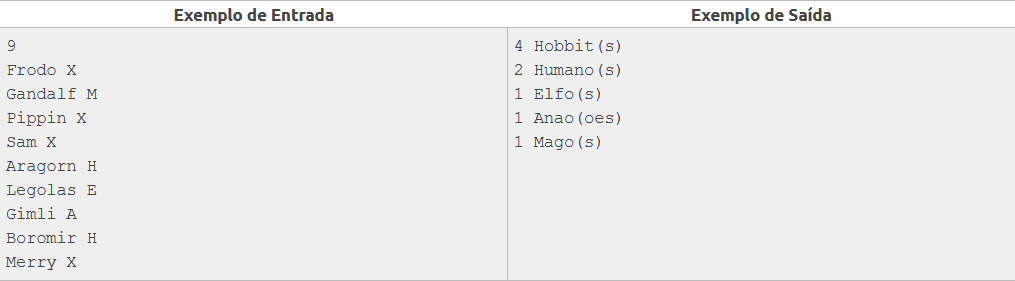
# Entrada

A primeira linha da entrada é composta por um inteiro N(0 < N <= 10), indicando o número de pessoas que se alistaram. Cada uma das próximas N linhas seguintes são compostas por uma cadeia de caracteres (sem espaços e de caracteres alfanuméricos apenas) e um caractere maiúsculo, indicando, respectivamente, o nome e o tipo da raça do respectivo ser. Este caractere poderá ser:

* A - Para anões;
* E - Para elfos;
* H - Para humanos;
* M - Para magos;
* X - Para hobbits (X, pois todo hobbit é uma incógnita para o mundo).

# Saída

Deve ser apresentado um relatório com a comitiva do Frodo, indicando em cada linha quantos seres de cada espécie estarão na jornada, seguindo a ordem: hobbits, humanos, elfos, anões e magos.



1. **System of a Download (2582 -** [https://tinyurl.com/5n7jcccm](https://www.beecrowd.com.br/judge/pt/problems/view/2582))

System of a Download é uma famosa banda de Hacker Metal! Certa vez, eles criaram um dispositivo, com seis botões, numerados de 0 a 5, e colocaram nesse dispositivo os seus 11 maiores sucessos. Para tocar uma destas músicas, é preciso pressionar dois botões. Com isso, os números destes dois botões são somados, e então toca-se a música correspondente ao número da soma, conforme a relação abaixo:

0 - PROXYCITY

1 - P.Y.N.G.

1. - DNSUEY!
2. - SERVERS
3. - HOST!
4. - CRIPTONIZE
5. - OFFLINE DAY
6. - SALT
7. - ANSWER!
8. - RAR?
9. - WIFI ANTENNAS

Por exemplo, se os botões pressionados forem 3 e 4, irá tocar a música 7 - SALT Escreva um programa que, dados os dois botões que forem pressionados, determine qual música irá tocar.

# Entrada

Um número inteiro C será informado, que será a quantidade de casos de teste. Cada caso tem dois valores inteiros, X e Y, representando quais botões foram pressionados.

# Saída

Para cada caso de teste, imprima o nome da música correspondente.

