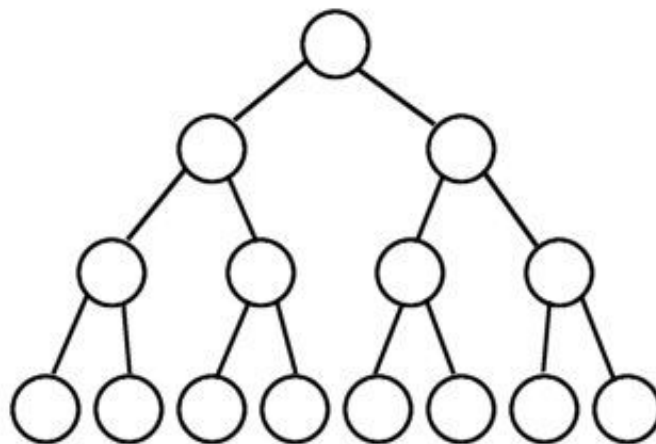


Problema A

Árvore Binária

Uma árvore é um conjunto de nós conectados de tal forma que haja exatamente um único caminho entre dois nós distintos. Para cada par de nós conectados, é dito que um é pai de outro. Toda árvore possui um único nó sem pai. Este nó é chamado de raiz. Os nós de uma árvore que não possuem filhos são chamados de folhas.

Árvore binária é aquela que cada nó possui no máximo dois filhos. Uma árvore binária é dita cheia se todos os nós que não forem folhas tiverem exatamente dois filhos. A altura de uma árvore binária é quantidade de nós entre a folha mais distante da raiz e a raiz mais um (em uma árvore binária cheia, todas as folhas têm a mesma quantidade de nós até a raiz). A altura da árvore binária abaixo é 3.



Dada a altura de uma árvore binária cheia, sua tarefa é determinar o número de nós dessa árvore.

Entrada

A primeira linha da entrada contém um inteiro **N** ($1 \leq N \leq 10$) indicando a quantidade de casos de teste. Cada caso de teste é composto por um inteiro **H** ($0 \leq H \leq 50$) indicando a altura da árvore binária cheia.

Saída

Para cada caso de teste, imprima uma linha contendo a quantidade de nós da árvore binária cheia.

Exemplo de Entrada		Exemplo de Saída	
3		3	
1		7	
2		15	
3			

Problema B

Binário para Decimal

Em sua mais recente aula para a turma de redes de computadores do IFPB, Thiago dissertou a respeito da representação binária de números inteiros e da conversão do sistema binário para o decimal.

Dada uma sequência de bits (0s e 1s), o bit mais à direita está na posição 0 e partindo para a esquerda, todos os bits têm a posição incrementada em relação ao que está imediatamente à sua direita. Cada posição está relacionada a uma potência de 2. O número na base decimal é obtido através do somatório do produto de cada bit com sua respectiva potência de 2.

Por exemplo, o número binário 1010 é equivalente a 10 na base decimal, pois seguindo o processo descrito acima, temos: $1*2^3+0*2^2+1*2^1+0*2^0 = 8+0+2+0 = 10$.

Para testar o conhecimento da turma e evitar filas, Thiago selecionou uma sequência de bits para cada aluno e pediu que esta sequência fosse convertida para a base decimal. Como Thiago anda muito ocupado com o doutorado, ele solicitou a você, mestre da programação, que criasse um programa que corrigisse a atividade dos alunos automaticamente.

Entrada

A entrada é composta por um inteiro **N** ($1 \leq N \leq 40$) indicando a quantidade de alunos da turma. Para cada aluno, haverá 2 linhas. A primeira contém uma sequência **S** de no máximo 20 bits. A segunda possui um inteiro **R** ($0 \leq R \leq 2^{30}$) indicando a resposta do aluno.

Saída

Para cada aluno, imprima "Acertou, miseravi!", caso a resposta do aluno esteja correta. Imprima "Tais lascado", caso contrário.

Exemplo de Entrada	Exemplo de Saída
3 11 2 1010 10 1111 16	Tais lascado Acertou, miseravi! Tais lascado

Problema C

Coeficientes

Herbert, mais conhecido como Vector, gosta muito de matemática, especialmente de polinômios. Estudando o assunto, ele cansou de calcular as raízes dos polinômios e resolveu tentar calcular os coeficientes a partir das raízes.

No entanto, ele também é bastante preguiçoso e não quer implementar um programa para isso. Então você terá de fazer isso. Afinal, para quê servem essas competições se não for para pedir que façam o que você não quer?

Vector não é tão cruel e vai lhe deixar uma dica. Considere que as raízes do polinômio sejam a_1, a_2, \dots, a_N . Então podemos escrever o polinômio como

$$(x - a_1)(x - a_2) \dots (x - a_N).$$

Como exemplo, se as raízes são 1, 2 e -1, o polinômio é dado por $(x - 1)(x - 2)(x + 1)$, que pode ser desenvolvido para $(x^2 - 3x + 2)(x + 1)$ e para $(x^3 - 2x^2 - x + 2)$. Neste caso, os coeficientes seriam 1, -2, -1 e 2. Note que um polinômio de 3 raízes tem 4 coeficientes.

Entrada

A primeira linha da entrada contém um inteiro **N** ($1 \leq N \leq 50$) representando a quantidade de raízes do polinômio. A segunda linha possuirá **N** inteiros **a_i** ($-1000 \leq a_i \leq 1000$), cada um representando uma raiz.

Saída

A saída deve constar de uma única linha com os **$N + 1$** coeficientes do polinômio, do de maior grau para o de menor, separados por espaços simples. É garantido que nenhum coeficiente da saída será maior que $2 \cdot 10^9$, em módulo.

Exemplo de Entrada	Exemplo de Saída
3 1 2 -1	1 -2 -1 2

Exemplo de Entrada	Exemplo de Saída
4 1 -1 1 -1	1 0 -2 0 1

Problema D

Douglas

Douglas é o rei de Bayeux. Sendo um rei, ele possui um nome deveras extenso. De fato, seu nome completo é Douglas Maciel Carvalho Andrade Fernandes Soares Albuquerque Gouveia Juarez Castro Pinto Carneiro Maciel Novamente Santos Jesus Maria e José Maciel de Novo Meira Melo Theotônio Monteiro Agora Está Certo Nevrek.

Douglas é extremamente irritado com este nome e gostaria de ser chamado por um único nome (como Kerven, por exemplo), mas ele sabe que dessa maneira é extremamente difícil que não haja repetições nos nomes dos cidadãos de Bayeux.

Sendo Douglas rei, ele é o chefe maior do poder executivo, legislativo e judiciário de Bayeux. Assim, Douglas pretende renomear todos os cidadãos de Bayeux de forma que todos os nomes possuam a mesma quantidade de caracteres. Contudo, para garantir que não haja repetições, Douglas está disposto a alterar o tamanho do alfabeto de Bayeux.

Como Douglas não sabe programar, ele solicitou a você a criação de um programa que dada a quantidade de caracteres que, segundo Douglas, cada nome deve ter e a atual população de Bayeux, retorne a quantidade de mínima de caracteres que o alfabeto deve possuir.

Entrada

A entrada é composta por vários casos de teste, pois Douglas se encontra extremamente indeciso sobre a quantidade de caracteres e a população de Bayeux tem um crescimento populacional aleatório. Cada linha contém 2 inteiros **P** ($1 \leq P \leq 2^{50}$) e **N** ($1 \leq N \leq 10$), indicando, respectivamente o tamanho da população de Bayeux e a quantidade de caracteres desejada por Douglas.

A entrada é terminada com **EOF**.

Saída

A saída é composta por um inteiro indicando o tamanho mínimo do novo alfabeto de Bayeux.

Exemplo de Entrada	Exemplo de Saída
10 1	10
96550 6	7
1379000000 10	10

Problema E

Estendendo Roupas

Mateus acha a atividade de estender roupas extremamente terapêutica. O que ele não acha nem um pouco terapêutico é quando a quantidade de pregadores é insuficiente para estender todas as roupas.

Quando isto acontece, ele tem que partir para o nível 2. O nível 2 consiste utilizar um pregador como interseção entre duas roupas consecutivas. Desta forma, 2 roupas podem ser estendidas com 3 pregadores, por exemplo.

Existem situações que até mesmo o nível 2 falha. Quando isto ocorre, Mateus utiliza o nível 3, que consiste em não estender as roupas.

A fim de evitar frustrações e estresses, ajude Mateus escrevendo um programa que dado o número de roupas e o número de pregadores disponíveis, indique qual nível ele deve usar.

Entrada

A entrada é composta por vários casos de teste. Cada caso de teste é composto por 2 inteiros **P** ($1 \leq P \leq 80$) e **R** ($1 \leq R \leq 70$), indicando, respectivamente o número de pregadores e roupas a serem estendidas.

A entrada é terminada com $P = R = 0$.

Saída

Para cada caso de teste, imprima “Nível x”, onde **x** corresponde ao nível a ser utilizado por Mateus.

Exemplo de Entrada	Exemplo de Saída
1 3 16 8 12 8 0 0	Nível 3 Nível 1 Nível 2

Problema F

Facilmente Palíndromo

Quem conhece Wesley sabe que seu cérebro possui um núcleo de processador de dedicação exclusiva a assuntos relacionados a palíndromos. O que poucos sabiam é que o cérebro de Wesley só possui 2 núcleos (o segundo está dedicado exclusivamente a assuntos de pão-pizza).

Como é de comum conhecimento, uma palavra palíndroma é uma palavra que pode ser lida de trás para frente e mantém o seu significado, então isso não gastava muito do núcleo do seu cérebro, até que uma ideia o fez passar noites acordados: Há palavras palíndromas em outros sistemas de escrita?

Wesley partiu então para estudar Morse, e viu que há palavras que são palíndromas quando são escritas em Morse, como é o caso de `pulp` e ainda mais, viu que as palavras podem ser palíndromas quando são escritas em inglês e em Morse também, como é o caso de `rotor`. Ele ficou apaixonado por esse tipo de conhecimento e nomeou esse tipo de palavra de facilmente palíndromas.

Como Wesley quer descansar o núcleo de palíndromos para guardar energia para os pães-pizza, ele pediu para que você fizesse um programa que, dada uma lista de palavras, classifique-as em palíndroma, facilmente palíndroma ou nada (o que o entristece).

Para facilitar seu trabalho, segue uma lista da tradução de cada letra para o seu respectivo código em Morse:

A	.-	N	-.
B	-...	O	---
C	-.-.	P	.--.
D	-..	Q	--.-
E	.	R	.-.
F	..-.	S	...
G	--.	T	-
H	U	..-
I	..	V	...-
J	.---	W	.---
K	-.-	X	-..-
L	.-..	Y	-.-.-
M	--	Z	--..

Entrada

A primeira linha da entrada contém um inteiro **N** ($1 \leq N \leq 100$), indicando a quantidade de palavras que serão analisadas. Cada uma das N linhas que se seguem contém uma palavra **S** ($1 \leq |S| \leq 100$). As palavras são compostas apenas por letras minúsculas.

Saída

Para cada caso de teste, imprima a categoria adequada, de acordo com os exemplos abaixo.

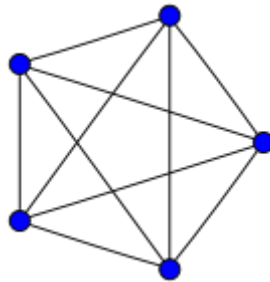
Exemplo de Entrada	Exemplo de Saída
4 pulp rotor arara opa	palindroma facilmente palindroma palindroma nada :(

Problema G

Grafos

Um grafo, na explicação de um grande mestre, é um conjunto de bolinhas (vértices) interligados por linhas (arestas). Um vértice é dito adjacente a outro se há um aresta que os interligue.

Um grafo é dito completo se todos os vértices são adjacentes entre si. A figura abaixo apresenta um grafo completo com 5 vértices e 10 arestas.



Dada a quantidade de vértices de um grafo completo, é possível determinar a quantidade de arestas. O inverso também é verdadeiro. Nesta questão, lhe é solicitado apenas que indique a quantidade de arestas de um grafo completo.

Entrada

A primeira linha da entrada contém um inteiro **N** ($1 \leq N \leq 10$), indicando a quantidade de grafos completos que serão analisados. Cada uma das N linhas que se seguem contém um inteiro **V** ($1 \leq V \leq 1000$), indicando a quantidade de vértices do grafo.

Saída

Para cada caso de teste, imprima a quantidade de arestas que o grafo completo possui.

Exemplo de Entrada		Exemplo de Saída	
3		0	
1		1	
2		10	
5			