

Nome do arquivo: *obi.c, obi.cpp ou obi.py*

OBI

O principal prêmio da Olimpíada Brasileira de Informática é o convite para os cursos de programação oferecidos no Instituto de Computação da Unicamp, com todas as despesas pagas pela Fundação Carlos Chagas, patrocinadora da OBI. São convidados apenas os competidores que atingem um certo número mínimo de pontos, consideradas as duas fases de provas. Você foi contratado pela Coordenação da OBI para fazer um programa que, dados os números de pontos obtidos por cada competidor em cada uma das fases, e o número mínimo de pontos para ser convidado, determine quantos competidores serão convidados para o curso na Unicamp. Você deve considerar que

- todos os competidores participaram das duas fases;
- o total de pontos de um competidor é a soma dos pontos obtidos nas duas fases;

Por exemplo, se a pontuação mínima para ser convidado é de 435 pontos, um competidor que tenha obtido 200 pontos na primeira fase e 235 pontos na segunda fase será convidado para o curso na Unicamp. Já um competidor que tenha obtido 200 pontos na primeira fase e 234 pontos na segunda fase não será convidado.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do dispositivo de entrada padrão (normalmente o teclado). A primeira linha da entrada contém dois números inteiros N e P , representando respectivamente o número de competidores ($1 \leq N \leq 1000$) e o número mínimo de pontos para ser convidado ($1 \leq P \leq 1000$). Cada uma das N linhas seguintes contém dois números inteiros X e Y indicando a pontuação de um competidor em cada uma das fases ($0 \leq X \leq 400$) e ($0 \leq Y \leq 400$).

Saída

Seu programa deve imprimir na saída padrão uma única linha contendo um único inteiro, indicando o número de competidores que serão convidados a participar do curso na Unicamp.

Exemplos

Exemplo de entrada 1	Exemplo de saída 1
3 100 50 50 100 0 49 50	2

Exemplo de entrada 2

4 235
100 134
0 0
200 200
150 150

Exemplo de saída 2

2

Nome do arquivo: *bacterias.c, bacterias.cpp, bacterias.java, bacterias.js ou bacterias.py*

Bactérias

Tadeu foi contratado recentemente para trabalhar no mais novo laboratório de biologia de sua universidade. Sua primeira tarefa consiste em estudar a taxa de reprodução de um determinado tipo de bactérias.

Tadeu iniciou seu experimento com uma única bactéria e descobriu que as bactérias se reproduzem a cada dia que passa. Mais especificamente, a cada dia, cada bactéria no recipiente de Tadeu se transforma em P bactérias, onde P é um inteiro positivo que Tadeu denominou como fator de multiplicação das bactérias.

Tadeu gostaria de obter a maior quantidade de bactérias possível, mas ele não pode deixar que essa quantidade ultrapasse a capacidade de N bactérias do recipiente, pois isso invalidaria todo o seu experimento. Observe que a quantidade de bactérias pode ser exatamente igual a N e isso não invalida o estudo de Tadeu.

Por exemplo, em um experimento cuja bactéria tem fator de multiplicação $P = 2$ e a capacidade do recipiente é $N = 32$:

- no início do experimento há apenas 1 bactéria;
- após 1 dia haverá $1 \cdot P = 2$ bactérias;
- após 2 dias haverá $2 \cdot P = 4$ bactérias;
- após 3 dias haverá $4 \cdot P = 8$ bactérias;
- após 4 dias haverá $8 \cdot P = 16$ bactérias;
- após 5 dias haverá $16 \cdot P = 32$ bactérias, o que excede a capacidade $N = 30$ do recipiente

Nesse caso, Tadeu pode deixar as bactérias se reproduzindo por no máximo 4 dias.

Dados o fator de multiplicação P e a capacidade N do recipiente, ajude Tadeu a determinar durante quantos dias ele pode deixar as bactérias se reproduzindo sem que a quantidade de bactérias exceda a capacidade do recipiente.

Entrada

A entrada é composta de duas linhas, cada uma contendo um único inteiro. A primeira linha contém N , a capacidade do recipiente. A segunda linha contém P , o fator de multiplicação das bactérias.

Saída

Seu programa deverá imprimir uma única linha, contendo apenas um inteiro, a quantidade máxima de dias que Tadeu pode deixar as bactérias se reproduzindo sem invalidar seu experimento.

Restrições

- $2 \leq P \leq N \leq 30\,000$

Exemplos

Exemplo de entrada 1 30 2	Exemplo de saída 1 4
Exemplo de entrada 2 20000 5	Exemplo de saída 2 6
Exemplo de entrada 3 49 7	Exemplo de saída 3 2

Nome do arquivo: *epidemia.c, epidemia.cpp, epidemia.java, epidemia.js ou epidemia.py*

Epidemia

Uma nova pandemia é sempre possível (e temida), mas a experiência recente mostrou que atualmente a ciência é capaz de desenvolver vacinas eficazes em muito pouco tempo. Outra consequência da pandemia recente é que muito se estudou sobre epidemias em geral, e vários modelos matemáticos foram desenvolvidos. Neste problema vamos usar um modelo simples de epidemia:

- Quando uma pessoa é infectada, ela infecta outras R pessoas, mas apenas no dia seguinte à sua infecção (R é chamado de fator reprodutivo da infecção).
- Ninguém é infectado mais do que uma vez.

Por exemplo, se no dia 0 da epidemia 3 pessoas são infectadas e o fator reprodutivo R é igual a 2, então no dia 1 outras 6 pessoas são infectadas ($3 + 6 = 9$ pessoas no total), no dia 2 outras 12 pessoas são infectadas ($3 + 6 + 12 = 21$ pessoas no total), no dia 3 outras 24 pessoas infectadas ($3 + 6 + 12 + 24 = 45$ pessoas no total), e assim por diante. Dados o número inicial de pessoas infectadas no dia 0 e o fator reprodutivo R da epidemia, escreva um programa para determinar qual o número de dias necessários para a epidemia infectar P ou mais pessoas no total.

Entrada

A primeira linha contém um inteiro N , o número de pessoas infectadas no dia 0. A segunda linha contém o fator reprodutivo R da infecção. A terceira e última linha contém um inteiro P , o número alvo de pessoas infectadas.

Saída

Seu programa deve produzir uma única linha, contendo um único inteiro, o número de dias para P ou mais pessoas serem infectadas

Exemplos

Exemplo de entrada 1 1 5 156	Exemplo de saída 1 3
Exemplo de entrada 2 2 1 11	Exemplo de saída 2 5

Nome do arquivo: plano.x, onde x deve ser c, cpp, java, js ou py

Plano de Internet

João conseguiu contratar um ótimo plano de Internet para o seu telefone celular. O plano permite que João utilize uma quota de até X megabytes de dados por mês para navegar na Internet. Se João não usa toda a sua quota no mês, os megabytes que ele não usou são adicionados à quota do mês seguinte. Pelo contrato, João nunca pode usar mais megabytes do que a sua quota corrente. Por exemplo, se $X=200$ megabytes e João usou 150 no primeiro mês e 220 megabytes no segundo mês, então no terceiro mês João tem uma quota de 230 megabytes para usar (50 megabytes são transferidos do primeiro para o segundo mês, 30 megabytes são transferidos do segundo para o terceiro mês). Nesta tarefa são dados o valor da quota mensal X e quantos megabytes João usou em cada um dos primeiros N meses do plano. Você deve determinar quantos megabytes João tem para usar no mês N+1.

Entrada

A primeira linha da entrada contém um número inteiro X, o valor da quota mensal em megabytes. A segunda linha contém um inteiro N, o número de meses. Cada uma das linhas seguintes contém um número inteiro M_i , indicando a quantidade de megabytes que João usou em cada mês, do mês 1 até o mês N.

Saída

Seu programa deve produzir uma única linha, contendo um único número inteiro, a quantidade de megabytes que João tem para usar no mês N+1.

Restrições

- $1 \leq X \leq 100$
- $1 \leq N \leq 100$
- $0 \leq M_i \leq 10000$ para $1 \leq i \leq N$
- M_i nunca é maior do que a quantidade de megabytes que João tem para usar no mês.

Exemplos

Exemplo de entrada 1	Exemplo de saída 1
100 2 50 120	130

Exemplo de entrada 2 10 3 4 6 2	Exemplo de saída 2 28
Exemplo de entrada 3 100 2 100 100	Exemplo de saída 2 100

Nome do arquivo: "torneio.x", onde x deve ser c, cpp, pas, java, js ou py

Torneio de tênis

A prefeitura contratou um novo professor para ensinar as crianças do bairro a jogar tênis na quadra de tênis do parque municipal. O professor convidou todas as crianças do bairro interessadas em aprender a jogar tênis. Ao final do primeiro mês de aulas e treinamentos foi organizado um torneio em que cada participante disputou exatamente seis jogos. O professor vai usar o desempenho no torneio para separar as crianças em três grupos, de forma a ter grupos de treino em que os participantes tenham habilidades mais ou menos iguais, usando o seguinte critério:

- participantes que venceram 5 ou 6 jogos serão colocados no Grupo 1;
- participantes que venceram 3 ou 4 jogos serão colocados no Grupo 2;
- participantes que venceram 1 ou 2 jogos serão colocados no Grupo 3;
- participantes que não venceram nenhum jogo não serão convidados a continuar com os treinamentos.

Dada uma lista com o resultado dos jogos de um participante, escreva um programa para determinar em qual grupo ele será colocado.

Entrada

A entrada consiste de seis linhas, cada linha indicando o resultado de um jogo do participante. Cada linha contém um único caractere: V se o participante venceu o jogo, ou P se o jogador perdeu o jogo. Não há empates nos jogos.

Saída

Seu programa deve produzir uma única linha na saída, contendo um único inteiro, identificando o grupo em que o participante será colocado. Se o participante não for colocado em nenhum dos três grupos seu programa deve imprimir o valor -1 .

Exemplos

Exemplo de entrada 1 V V P P P V	Exemplo de saída 1 2
Exemplo de entrada 2 P	Exemplo de saída 2 -1

P P P P P	
-----------------------	--