

Hi, André Reis de Brito

andrebrito@dcc.ufjf.br

HOME

PERFIL

NEWS²²

INSTRUTORES

ACADEMIC

CONTESTS

FÓRUM

PROBLEMAS

SUBMISSÕES

RANKS

SAIR

ACADEMIC 1018

SUBMISSÕESDESCRIÇÃO

INICIANTE | NÍVEL 1 | BASE TIME LIMIT: 1 SECOND | MEMORY LIMIT: 200 MB | THIS PROBLEM WAS CREATED BY ANDRÉ REIS DE BRITO

URI Online Judge | 1018

Créditos no parque de diversões

Por André Brito, 🇧🇷 Brazil

Timelimit: 1

Um parque de diversões está lançando um cartão para auxiliar os seus clientes a gerenciarem os créditos para utilizar seus brinquedos. O cartão deve ser carregado antes de ser usado em um brinquedo, senão o cliente é barrado. Cada brinquedo tem um preço diferente de uso de tabela, conforme mostrado abaixo:

- Roda-gigante (roda): C\$3,50
- Bate-bate (bate): C\$2,50
- Xícara (xica): C\$3,00
- Trem fantasma (trem): C\$4,00
- Montanha-russa (mont): C\$5,00
- Carrossel (carr): C\$2,00

Cada cliente pode carregar o seu cartão apenas uma vez, sendo necessário trocá-lo ao final do uso. Para a carga do cartão estão disponíveis os valores C\$25, C\$50, C\$100 e C\$500. Caso o cliente carregue o seu cartão C\$100, ele ganha 10% de desconto em todos os brinquedos. Se o cliente carregar com C\$500, ele ganhará 25% de desconto em todos os brinquedos.

Sua tarefa é criar um programa que receba o saldo inicial do cartão do cliente, e uma sequência não definida dos códigos de 4 letras que representam os brinquedos, como "tiro" ou "mont". A cada brinquedo o programa deve verificar se o cartão possui saldo suficiente para utilizar o brinquedo. Caso não tenha, será exibida a mensagem **SALDO INSUFICIENTE** . Durante todo o programa, o cliente pode tentar usar até 3 vezes o cartão sem saldo suficiente para um brinquedo. No terceiro uso do cartão sem saldo, o cartão se tornará inválido e o programa deve ser encerrado sem exibir nenhuma mensagem adicional. Caso o programa receba um brinquedo cujo nome seja "FIM!", o programa saberá que o usuário está indo embora e também encerrará o programa, exibindo o saldo final do cartão após os usos.

Entrada

Saldo inicial do cartão, sequência de strings de 4 caracteres com os brinquedos utilizados pelo cliente.

Saída

Mensagem SALDO INSUFICIENTE, caso o usuário utilize um brinquedo sem ter saldo suficiente para uso, ou o valor do saldo restante no cartão.

Samples Input	Samples Output
500 bate mont roda trem carr FIM!	487.25

CUSTOM PROBLEM

1018

LINGUAGEM

C++17

SOURCE CODE

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6
7     /**
8      * Escreva a sua solução aqui
9      * Code your solution here
10     * Escriba su solución aquí
11     */
12
13     return 0;
14 }
```

CONSTRUA A SUA SOLUÇÃO E ENVIE!

ENVIAR

© 2011 - 2021 URI Online Judge

Cookies | Privacidade | Termos & Condições | FAQs | Status | Créditos | Contato

Version 6.0.0

https://www.urionlinejudge.com.br/judge/pt/custom-problems/view/1018

1/2

25	SALDO INSUFICIENTE	
bate	SALDO INSUFICIENTE	
bate	SALDO INSUFICIENTE	
bate		
bate		
bate		
bate		
bate		
bate		
bate		
bate		
bate		
bate		
bate		
bate		
100	77.50	
bate		
bate		
bate		
bate		
bate		
bate		
mont		
mont		
FIM!		

Hi, André Reis de Brito

andrebrito@dcc.ufjf.br

HOME

PERFIL

NEWS²²

INSTRUTORES

ACADEMIC

CONTESTS

FÓRUM

PROBLEMAS

SUBMISSÕES

RANKS

SAIR

ACADEMIC 1017

SUBMISSÕESDESCRIÇÃO

INICIANTE | NÍVEL 1 | BASE TIME LIMIT: 1 SECOND | MEMORY LIMIT: 200 MB | THIS PROBLEM WAS CREATED BY ANDRÉ REIS DE BRITO

URI Online Judge | 1017

Média de peso da academia

Por André Brto, Brazil

Timelimit: 1

Uma academia, que possui **N** alunos matriculados, deseja fazer um censo entre seus clientes para descobrir o mais alto, o mais baixo, o mais pesado e o mais leve. Para isto você deve fazer um programa que pergunte a cada um dos clientes da academia seu nome, sua altura e seu peso. O programa deve encerrar quando for digitado o nome FIM. Após o término deve ser informado os nomes e valores do cliente mais alto, mais baixo, mais gordo, média da altura e do peso dos clientes, e quantidade total de clientes consultados.

Entrada

Dados dos N clientes da academia: nome, altura (em centímetros) e peso (em kg).

Saída

Nome e peso do cliente mais pesado; nome a peso do cliente mais leve; nome e altura do cliente mais alto; nome e altura do cliente mais baixo; média da altura; média do peso e quantidade de clientes.

Exemplo de entrada 1:

Alberto
182
96.3
Rogerio
177
88.8
Maria
165
62.3
Carla
166
61.1
FIM

Exemplo de saída 1:

Maior Peso: Alberto - 96.3 kg
Menor Peso: Carla - 61.1 kg
Maior Altura: Alberto - 182 cm
Menor Altura: Maria - 165 cm
Media Peso: 77.1 kg
Media Altura: 172 cm
Num Clientes: 4

Exemplo de entrada 2:

Alberto

CUSTOM PROBLEM1017

LINGUAGEMC++17

SOURCE CODE

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6
7     /**
8      * Escreva a sua solução aqui
9      * Code your solution here
10     * Escriba su solución aquí
11     */
12
13     return 0;
14 }
```

CONSTRUA A SUA SOLUÇÃO E ENVIE!

ENVIAR

© 2011 - 2021 URI Online Judge

Cookies | Privacidade | Termos & Condições | FAQs | Status | Créditos | Contato

Version 6.0.0

https://www.urionlinejudge.com.br/judge/pt/custom-problems/view/1017

1/2

182

96.3

FIM

Exemplo de saída 2:

Maior Peso: Alberto - 96.3 kg

Menor Peso: Alberto - 96.3 kg

Maior Altura: Alberto - 182 cm

Menor Altura: Alberto - 182 cm

Media Peso: 96.3 kg

Media Altura: 182 cm

Num Clientes: 1

Samples Input	Samples Output
Alberto	Maior Peso: Alberto - 96.3 kg
182	Menor Peso: Carla - 61.1 kg
96.3	Maior Altura: Alberto - 182 cm
Rogério	Menor Altura: Maria - 165 cm
177	Media Peso: 77.1 kg
88.8	Media Altura: 172 cm
Maria	Num Clientes: 4
165	
62.3	
Carla	
166	
61.1	
FIM	
Alberto	Maior Peso: Alberto - 96.3 kg
182	Menor Peso: Alberto - 96.3 kg
96.3	Maior Altura: Alberto - 182 cm
FIM	Menor Altura: Alberto - 182 cm
	Media Peso: 96.3 kg
	Media Altura: 182 cm
	Num Clientes: 1

Hi, André Reis de Brito

andrebrito@dcc.ufrj.br

HOME

PERFIL

NEWS²²

INSTRUTORES

ACADEMIC

CONTESTS

FÓRUM

PROBLEMAS

SUBMISSÕES

RANKS

SAIR

ACADEMIC 1016

SUBMISSÕESDESCRIÇÃO

AD-HOC | NÍVEL 3 | BASE TIME LIMIT: 2 SECOND | MEMORY LIMIT: 200 MB | THIS PROBLEM WAS CREATED BY ANDRÉ REIS DE BRITO

URI Online Judge | 1016

Poltronas do cinema

Por André Brito, Brazil

Timelimit: 2

O cinema da cidade de Nerdlandia está com problemas para gerenciar os lugares vagos em suas salas de projeção: seus usuários não estão conseguindo descobrir se conseguirão sentar-se juntos em um determinado filme antes de entrar na sala. Para resolver este problema, o cinema precisa que você crie um programa que exiba a mensagem **OK** se houver a possibilidade do grupo se acomodar junto numa mesma fileira, ou **NOK** caso contrário. Para o programa, considere que a resposta é válida para o momento da realização da consulta, ou seja, como se nenhum outro ingresso fosse vendido para a sessão após a sua execução. Só é considerado que o grupo ficará junto se todos ocuparem a mesma fileira.

As salas de projeção de Nerdlândia possuem **F** fileiras ($0 < F < 14.481$) e **C** colunas ($0 < C < 14.481$). Para o programa, considere que as poltronas ocupadas serão fornecidas em pares sequenciais **X Y**, onde **X** ($0 \leq X \leq F-1$) representa a fileira da poltrona e **Y** ($0 \leq Y \leq C-1$) representa a coluna da poltrona, sendo exibida uma poltrona por linha. Caso seja informado uma fileira negativa, o programa sabe que terminou de ler as entradas.

Após a leitura das poltronas, será fornecido uma quantidade **N** ($0 < N < 500$) de pessoas que desejam sentar numa mesma fileira. Baseado neste valor será exibida a mensagem referente a possibilidade do grupo sentar junto ou não.

Entrada

Quantidade inteira **F** de fileiras ($0 < F < 14.481$); quantidade inteira **C** de colunas ($0 < C < 14.481$); sequência de até **F** x **C** poltronas ocupadas; quantidade inteira **N** ($0 < N < 500$) de poltronas em sequência desejadas.

Saída

Mensagem **OK** se a quantidade de poltronas desejadas numa mesma fileira for possível de ser obtida, e **NOK** caso contrário.

Samples Input	Samples Output
4	OK
4	
0 0	
0 1	
0 3	
1 1	
1 2	
1 3	
2 0	
2 2	
3 0	
3 1	
-1 0	
2	

CUSTOM PROBLEM1016

LINGUAGEMC++17

SOURCE CODE

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6
7     /**
8      * Escreva a sua solução aqui
9      * Code your solution here
10     * Escriba su solución aquí
11     */
12
13     return 0;
14 }
```

CONSTRUA A SUA SOLUÇÃO E ENVIE!ENVIAR

© 2011 - 2021 URI Online Judge

Cookies | Privacidade | Termos & Condições | FAQs | Status | Créditos | Contato

Version 6.0.0

https://www.urionlinejudge.com.br/judge/pt/custom-problems/view/1016

1/2

4	NOK	
4		
0 0		
0 1		
0 3		
1 1		
1 2		
1 3		
2 0		
2 2		
3 0		
3 1		
-1 0		
4		

Hi, André Reis de Brito

andrebrito@dcc.ufrj.br

HOME

PERFIL

NEWS²²

INSTRUTORES

ACADEMIC

CONTESTS

FÓRUM

PROBLEMAS

SUBMISSÕES

RANKS

SAIR

ACADEMIC 1015

SUBMISSÕESDESCRIÇÃO

INICIANTE | **NÍVEL 1** | BASE TIME LIMIT: 1 SECOND | MEMORY LIMIT: 200 MB | THIS PROBLEM WAS CREATED BY ANDRÉ REIS DE BRITO

URI Online Judge | 1015

Datas Britonianas

Por André Brito, 🇧🇷 Brazil

Timelimit: 1

Uma data Britoniana é uma data cuja soma do dia, mês e ano resulta em 2000. Por exemplo, a data 01/01/1998 é uma data Britoniana, assim como 03/04/1993. A coisa mais legal destas datas é que nós conseguimos descobrir a idade de uma pessoa nascida numa data Britoniana apenas somando o dia e o mês de nascimento ao ano que você deseja saber a idade da pessoa. Assim, podemos saber que em 2021 a pessoa que nasceu em 01/01/1998 terá 23 anos, já que teremos $1 + 1 + 21 = 23$.

Crie um programa que receba uma data no padrão DD/MM/AAAA e, caso seja uma data Britoniana, **diga qual a idade da pessoa para o ano informado** . Caso não seja uma data Britoniana, exiba o **valor -1**.

Caso seja uma data inválida, tipo 30/02, exiba o **valor -2**. Data bissextas podem ser desconsideradas para a execução deste programa.

Entrada

Um inteiro ANO (ANO > 2001) e uma data no formato DD/MM/AAAA.

Saída

A idade da pessoa no final do ano informado, caso a data seja uma data Britoniana válida; -1 caso não seja uma data Britoniana válida; ou -2, caso não seja uma data válida.

Exemplo de entrada 1:

2021

01/01/1998

Exemplo de saída 1:

23

Exemplo de entrada 2:

2003

20/10/1988

Exemplo de saída 2:

-1

Exemplo de entrada 3:

2015

31/09/1978

Exemplo de saída 3:

-2

Samples Input	Samples Output
2021 01/01/1998	23

CUSTOM PROBLEM

1015

LINGUAGEM

C++17

SOURCE CODE

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6
7     /**
8      * Escreva a sua solução aqui
9      * Code your solution here
10     * Escriba su solución aquí
11     */
12
13     return 0;
14 }
```

CONSTRUA A SUA SOLUÇÃO E ENVIE!

ENVIAR

© 2011 - 2021 URI Online Judge

Cookies | Privacidade | Termos & Condições | FAQs | Status | Créditos | Contato

Version 6.0.0

https://www.urionlinejudge.com.br/judge/pt/custom-problems/view/1015

1/2

2003 20/10/1988	-1
2015 31/09/1978	-2

ACADEMIC 1019

SUBMISSÕES

DESCRIÇÃO

INICIANTE | **NÍVEL 3** | BASE TIME LIMIT: 5 SECOND | MEMORY LIMIT: 200 MB | THIS PROBLEM WAS CREATED BY ANDRÉ REIS DE BRITO

URI Online Judge | 1019

Lista de notas

Por André Brito, 🇧🇷 Brazil

Timelimit: 5

Um professor atrapalhado acabou misturando os nomes dos alunos e suas notas após realizar uma correção. A única coisa que ele sabia era que o nome dos alunos estavam ordenados da maior para a menor nota. Sua tarefa é receber uma lista de N notas e ordená-las. Em seguida, seu programa receberá N nomes e exibirá na tela os nomes dos alunos e suas respectivas notas.

Entrada

Sequência de N (0 < N < 5000) notas de alunos e uma sequência de N nomes de alunos. O maior nome tem 10 letras.

Saída

Uma sequência, linha por linha, apresentando o nome do aluno e sua respectiva nota.

Exemplo de entrada:

8.28
5.00
7.39
4.71
9.26
7.20
8.40
8.14
4.85
3.03
-1
Lani
Rima
Fumiko
Lecia
Joella
Demetrius
Kami
Tera
Betty
Cythia

Exemplo de saída:

Lani: 9.26
Rima: 8.40
Fumiko: 8.28

CUSTOM PROBLEM

LINGUAGEM

1019

C++17

SOURCE CODE

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main() {
6
7     /**
8      * Escreva a sua solução aqui
9      * Code your solution here
10     * Escriba su solución aquí
11     */
12
13     return 0;
14 }
```

CONSTRUA A SUA SOLUÇÃO E ENVIE!

ENVIAR

Joella: 7.39
Demetrius: 7.20
Kami: 5.00
Tera: 4.85
Betty: 4.71
Cythia: 3.03

Samples Input	Samples Output
8.28	Lani: 9.26
5.00	Rima: 8.40
7.39	Fumiko: 8.28
4.71	Lecia: 8.14
9.26	Joella: 7.39
7.20	Demetrius: 7.20
8.40	Kami: 5.00
8.14	Tera: 4.85
4.85	Betty: 4.71
3.03	Cythia: 3.03
-1	
Lani	
Rima	
Fumiko	
Lecia	
Joella	
Demetrius	
Kami	
Tera	
Betty	
Cythia	