



OBI2010

Caderno de Tarefas

Modalidade Programação • Nível Júnior, Fase 2

8 de maio de 2010

A PROVA TEM DURAÇÃO DE **3 HORAS**

Promoção:



Sociedade Brasileira de Computação

Patrocínio:



Fundação Carlos Chagas

Instruções

LEIA ATENTAMENTE ESTAS INSTRUÇÕES ANTES DE INICIAR A PROVA

- Este caderno de tarefas é composto por 7 páginas (não contando esta folha de rosto), numeradas de 1 a 7. Verifique se o caderno está completo.
- A prova deve ser feita individualmente.
- É proibido consultar a Internet, livros, anotações ou qualquer outro material durante a prova. É permitida a consulta ao *help* do ambiente de programação se este estiver disponível.
- As tarefas têm o mesmo valor na correção.
- A correção é automatizada, portanto siga atentamente as exigências da tarefa quanto ao formato da entrada e saída de seu programa.
- Não implemente nenhum recurso gráfico nas suas soluções (janelas, menus, etc.), nem utilize qualquer rotina para limpar a tela ou posicionar o cursor.
- As tarefas **não** estão ordenadas, neste caderno, por ordem de dificuldade; procure resolver primeiro as questões mais fáceis.
- Preste muita atenção no nome dos arquivos fonte indicados nas tarefas. Soluções na linguagem C devem ser arquivos com sufixo *.c*; soluções na linguagem C++ devem ser arquivos com sufixo *.cc* ou *.cpp*; soluções na linguagem Pascal devem ser arquivos com sufixo *.pas*. Para problemas diferentes você pode escolher trabalhar com linguagens diferentes, mas apenas uma solução, em uma única linguagem, deve ser submetida para cada problema.
- Ao final da prova, para cada solução que você queira submeter para correção, copie o arquivo fonte para o seu diretório de trabalho ou disquete, conforme especificado pelo seu professor.
- Não utilize arquivos para entrada ou saída. Todos os dados devem ser lidos da entrada padrão (normalmente é o teclado) e escritos na saída padrão (normalmente é a tela). Utilize as funções padrão para entrada e saída de dados:
 - em Pascal: *readln*, *read*, *writeln*, *write*;
 - em C: *scanf*, *getchar*, *printf*, *putchar*;
 - em C++: as mesmas de C ou os objetos *cout* e *cin*.
- Procure resolver o problema de maneira eficiente. Na correção, eficiência também será levada em conta. As soluções serão testadas com outras entradas além das apresentadas como exemplo nas tarefas.

Escada Rolante

Nome do arquivo fonte: escada.c, escada.cpp, ou escada.pas

O Shopping Boas Compras – SBC, através de sua política ambiental, está preocupado com o consumo de energia e, resolveu trocar todas as escadas rolantes por modelos mais modernos, que se desligam caso ninguém esteja utilizando, poupando energia.

A nova escada rolante possui um sensor no início. Toda vez que ela está vazia e alguém passa pelo sensor, a escada começa a funcionar, parando de funcionar novamente após 10 segundos se ninguém mais passar pelo sensor. Estes 10 segundos representam o tempo suficiente para levar alguém de um nível ao outro.

Preocupados em saber exatamente quanto de energia o shopping está economizando, o gerente pediu sua ajuda. Como eles sabem qual era o consumo da escada rolante antiga, eles te pediram para calcular o tempo que a nova escada ficou funcionando.

Tarefa

Dados os instantes, em segundos, em que passaram pessoas pela escada rolante, você deve calcular quantos segundos ela ficou ligada.

Entrada

A primeira linha da entrada contém um inteiro N que indica o número de pessoas que o sensor detectou ($1 \leq N \leq 1.000$). As N linhas seguintes representam o instante em que a i -ésima pessoa passou pelo sensor e contém um inteiro T ($0 \leq T \leq 10.000$). Os tempos estão em ordem crescente, sem repetições.

Saída

Seu programa deve imprimir uma única linha, contendo o tempo que a escada rolante ficou ligada.

Informações sobre a pontuação

- Em um conjunto de casos de teste que totaliza 30 pontos, a primeira pessoa passa pelo sensor no tempo 0.

Exemplos

Entrada	Saída
3 0 10 20	30

Entrada	Saída
5	35
5	
10	
17	
20	
30	

Entrada	Saída
3	12
1	
2	
3	

Tacógrafo

Nome do arquivo fonte: `tacografo.c`, `tacografo.cpp`, ou `tacografo.pas`

Tacógrafos são dispositivos instalados em determinados tipos de veículos, que registram a velocidade, tempo e distância percorrida por tal veículo. É utilizada principalmente em veículos de transporte coletivo e de transporte de cargas, assim ajudando a evitar abusos de velocidade por parte dos motoristas.

A empresa SBC (Sociedade Brasileira dos Caminhoneiros) decidiu encomendar uma versão um pouco mais básica (e barata) para seus associados não precisarem gastar tanto na instalação desses aparelhos. Essas versões modificadas registram apenas os intervalos de tempo e as velocidades médias do caminhão naqueles intervalos.

Tarefa

Apesar das restrições dos aparelhos novos, a SBC quer poder saber qual foi a distância percorrida pelos caminhões. Você deverá escrever um programa que recebe uma série de intervalos de tempo com suas respectivas velocidades médias e calcula qual foi a distância total percorrida pelo caminhão de acordo com o tacógrafo.

Entrada

A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 1000$) representando a quantidade de intervalos de tempo registrados no tacógrafo. As N linhas seguintes descrevem os intervalos de tempo. Cada uma dessas linhas possui dois inteiros T e V ($1 \leq T \leq 100$, $0 \leq V \leq 120$), que representam, respectivamente o tempo decorrido (em horas) e a velocidade média (em quilômetros por hora) no intervalo de tempo.

Saída

Seu programa deve imprimir uma única linha, contendo um único número inteiro representando a distância total percorrida, em quilômetros.

Informações sobre a pontuação

- Em um conjunto de casos de teste que totaliza 20 pontos, $N \leq 10$.

Exemplos

Entrada	Saída
3 10 0 55 12 75 120	9660

Entrada	Saída
10 45 46 46 101 7 2 95 104 12 107 78 29 10 26 52 86 13 79 1 107	26022

Entrada	Saída
8 37 24 68 69 28 26 79 8 36 0 50 71 13 68 87 113	21205

Dentista

Nome do arquivo fonte: `dentista.c`, `dentista.cpp`, ou `dentista.pas`

Os dentistas são extremamente meticolosos em seu trabalho, tendo que agir com muita precisão em todas as suas atividades. Pedro é um dentista meticoloso como todos os outros. Infelizmente sua secretária não é muito organizada e, com o intuito de ajudar sempre os pacientes, aceita que eles marquem consultas no horário que quiserem, sem se preocupar com os demais horários marcados, ocasionando vários conflitos de horários que muito incomodaram Pedro e os pacientes. Por exemplo, se uma consulta começar às 9 horas e durar 2 horas, nenhuma outra consulta deveria ser marcada para iniciar as 10 horas.

Ao perceber que sua agenda tinha conflito de horários, Pedro pediu sua ajuda para descobrir a maior quantidade de consultas que podem ser atendidas sem sobreposição.

Tarefa

Você deve escrever um programa que, dados os horários de início e término das consultas agendadas pela secretária, responda a quantidade máxima de consultas que podem ser atendidas sem sobreposição.

Entrada

A primeira linha da entrada contém um inteiro N ($1 \leq N \leq 10.000$) que indica quantas consultas a secretária marcou. Cada uma das N linhas seguintes contém um par de inteiros X e Y separados por um espaço em branco ($0 \leq X < Y \leq 1.000.000$) que representam, respectivamente, o horário de início e de término da consulta. Considere que se uma consulta inicia no exato instante em que outra termina não há sobreposição. Os horários de início são fornecidos em ordem, podendo haver mais de uma consulta que inicie no mesmo horário.

Saída

Seu programa deve imprimir uma única linha, contendo um inteiro que representa a quantidade máxima de consultas que podem ser atendidas sem que haja qualquer sobreposição.

Informações sobre a pontuação

- Em um conjunto de casos de teste que totaliza 40 pontos, ($0 \leq X < Y \leq 1.000$).

Exemplos

Entrada	Saída
3 10 100 40 130 120 200	2

Entrada	Saída
4 10 20 20 30 30 40 40 50	4

Entrada	Saída
5 10 30 20 40 30 60 40 80 60 100	3