



Olimpíada Brasileira de Informática

OBI2026

Caderno de Tarefas

Modalidade Programação • Nível 1 • Fase 1 - Turno B

18 a 19 de Junho de 2026

A PROVA TEM DURAÇÃO DE 2 horas

Promoção:



Sociedade Brasileira de Computação

Apoio:



Coordenação:



Instruções

LEIA ATENTAMENTE ESTAS INSTRUÇÕES ANTES DE INICIAR A PROVA

- Este caderno de tarefas é composto por 7 páginas (não contando a 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; em particular, seu programa não deve escrever frases como “Digite o dado de entrada:” ou similares.
- 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 necessariamente 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 Java devem ser arquivos com sufixo *.java* e a classe principal deve ter o mesmo nome do arquivo fonte; soluções na linguagem Python 3 devem ser arquivos com sufixo *.py*; e soluções na linguagem Javascript devem ter arquivos com sufixo *.js*.
- Na linguagem Java, **não** use o comando *package*, e note que o nome de sua classe principal deve usar somente letras minúsculas (o mesmo nome do arquivo indicado nas tarefas).
- Você pode submeter até 50 soluções para cada tarefa. A pontuação total de cada tarefa é a melhor pontuação entre todas as submissões. Se a tarefa tem sub-tarefas, para cada sub-tarefa é considerada a melhor pontuação entre todas as submissões.
- 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 C: *scanf*, *getchar*, *printf*, *putchar*;
 - em C++: as mesmas de C ou os objetos *cout* e *cin*.
 - em Java: qualquer classe ou função padrão, como por exemplo *Scanner*, *BufferedReader*, *BufferedWriter* e *System.out.println*
 - em Python: *read*, *readline*, *readlines*, *input*, *print*, *write*
 - em Javascript: *scanf*, *printf*
- Procure resolver a tarefa 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.

Aula de Matemática

Nome do arquivo: `aula.c`, `aula.cpp`, `aula.pas`, `aula.java`, `aula.js` ou `aula.py`

Na aula da professora Márcia, os alunos aprenderam a somar e subtrair números. Depois de tirar dúvidas, ela ouviu um de seus alunos dizer a seguinte frase: "a soma é muito mais legal que a subtração, porque se eu tenho A reais, é melhor somar do que subtrair B reais".

Márcia gostou que as crianças demonstraram interesse na matéria e, para atizar a curiosidade de todos, respondeu que aquilo não era necessariamente verdade. Após uma grande revolta da sala, que tinha certeza de que a soma dois números A e B era sempre maior que a diferença, ela disse que existiam números especiais que ficavam maiores quando eram subtraídos, mas que os alunos só aprenderiam isso no futuro.

Assim, para provar o ponto da professora, dados dois números A e B determine qual é maior, a soma $A + B$ ou a subtração $A - B$.

Entrada

A primeira linha da entrada contém um único inteiro A .

A segunda linha da entrada contém um único inteiro B .

Saída

A saída deve conter um único inteiro, o maior valor entre $A + B$ e $A - B$. Não coloque nada além disso na saída, senão será considerado errado.

Restrições

- $-200 \leq A, B \leq 200$.

Informações sobre a pontuação

A tarefa vale 100 pontos. Estes pontos estão distribuídos em subtarefas, cada uma com suas **restrições adicionais** às definidas acima.

- **Subtarefa 1 (0 pontos):** Esta subtarefa é composta apenas pelos exemplos mostrados abaixo. Ela não vale pontos, serve apenas para que você verifique se o seu programa imprime o resultado correto para os exemplos.
- **Subtarefa 2 (100 pontos):** Sem restrições adicionais.

Exemplos

Exemplo de entrada 1	Exemplo de saída 1
10 5	15

Explicação do exemplo 1: Nesse caso, $A + B = 10 + 5 = 5$ e $A - B = 10 - 5 = 5$. Logo, a resposta é 15. Perceba que a saída deve conter apenas o número 15, nada além disso.

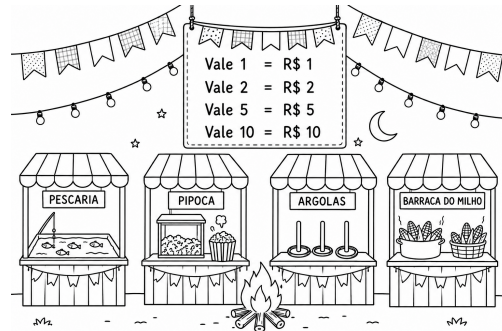
Exemplo de entrada 2	Exemplo de saída 2
-10 -5	-5

Explicação do exemplo 2: Nesse caso, $A + B = -10 + (-5) = -15$ e $A - B = -10 - (-5) = -5$. Logo, a resposta é -5 .

Exemplo de entrada 3	Exemplo de saída 3
1 0	1

Fichas

Nome do arquivo: `fichas.c`, `fichas.cpp`, `fichas.pas`, `fichas.java`, `fichas.js` ou `fichas.py`



Gustavo está muito animado para ir à festa junina de seu bairro. Para ele, essa é a melhor festa do ano, por causa das deliciosas comidas típicas, como milho, canjica e arroz-doce, e também das brincadeiras e atividades presentes na festa, como a pescaria e o jogo de argolas.

No dia da festa, a mãe do garoto lhe deu X reais para gastar no evento. No entanto, para organizar e facilitar os pagamentos internos, a festa junina do bairro de Gustavo usa um sistema de fichas, as quais podem ser de quatro valores: R\$1, R\$2, R\$5 e R\$10. Dessa forma, ao chegar na festa, ele precisa trocar seu dinheiro por fichas para pagar pelas comidas e brincadeiras.

Gustavo tem certeza que vai gastar tudo que recebeu, mas não quer carregar muitas fichas. Portanto, dada a quantidade X de reais que o garoto ganhou da mãe, ajude-o a calcular qual o mínimo de fichas pelas quais ele pode trocar o dinheiro.

Entrada

A entrada contém uma única linha com um inteiro X , o dinheiro de Gustavo.

Saída

A saída deve conter um único inteiro, o mínimo de fichas que Gustavo precisa para trocar todo seu dinheiro.

Restrições

- $1 \leq X \leq 200$.

Informações sobre a pontuação

A tarefa vale 100 pontos. Estes pontos estão distribuídos em subtarefas, cada uma com suas **restrições adicionais** às definidas acima.

- **Subtarefa 1 (0 pontos):** Esta subtarefa é composta apenas pelos exemplos mostrados abaixo. Ela não vale pontos, serve apenas para que você verifique se o seu programa imprime o resultado correto para os exemplos.
- **Subtarefa 2 (20 pontos):** $X \leq 9$.
- **Subtarefa 2 (20 pontos):** X é múltiplo de 5.
- **Subtarefa 2 (60 pontos):** Sem restrições adicionais.

Exemplos

Exemplo de entrada 1	Exemplo de saída 1
8	3

Explicação do exemplo 1: Nesse caso, são necessárias 3 fichas:

- 1 ficha de R\$1.
- 1 ficha de R\$2.
- 1 ficha de R\$5.

Note que esse exemplo satisfaz as restrições da Subtarefa 2.

Exemplo de entrada 2	Exemplo de saída 2
75	8

Explicação do exemplo 2: Nesse caso, são necessárias 8 fichas:

- 1 ficha de R\$5.
- 7 fichas de R\$10.

Note que esse exemplo satisfaz as restrições da Subtarefa 3.

Exemplo de entrada 3	Exemplo de saída 3
199	22

Explicação do exemplo 3: Nesse caso, são necessárias 22 fichas:

- 2 fichas de R\$2.
- 1 ficha de R\$5.
- 19 fichas de R\$10.

Transporte

Nome do arquivo: `transporte.c`, `transporte.cpp`, `transporte.pas`, `transporte.java`,
`transporte.js` ou `transporte.py`

A semana olímpica é muito aguardada e almejada por todos os competidores da OBI. Esse evento é muito proveitoso para os alunos, devido não só às aulas e aos simulados de alto nível, mas também às amizades formadas durante o evento. No entanto, como alunos de todo o país se reúnem na competição, alguns precisam de transporte do aeroporto para o hotel no dia da chegada.

Todo ano, alguns ônibus são contratados para realizar esse deslocamento, mas a logística desse processo é um pouco complicada. Isso porque cada um dos N competidores chega em um horário, sendo que o i -ésimo deles chega no minuto t_i do dia ($1 \leq i \leq N$). Além disso, para não preocupar os pais e alunos, a OBI garante que nenhum aluno precisará esperar mais de T minutos desde o momento do pouso até a saída de seu transporte.

Os ônibus contratados são grandes o suficiente para levarem até N passageiros, porém, como os alunos não devem esperar por muito tempo para serem levados, podem ser necessários muitos transportes para levar todos do aeroporto para o hotel. Assim, dados a quantidade N de competidores, o tempo máximo de espera T e os tempos de chegada $t_1, t_2, \dots, t_N$, ajude a OBI a calcular a quantidade mínima de ônibus que ela precisa contratar para transportar todos os competidores.

Entrada

A primeira linha da entrada contém dois inteiros, N e T , respectivamente o número de competidores e o tempo máximo de espera. A segunda linha da entrada contém N números inteiros $t_1, t_2, \dots, t_N$, os tempos de chegada dos competidores.

Saída

A saída deve conter um único inteiro, a quantidade mínima de ônibus que a OBI precisa contratar para transportar todos os competidores.

Restrições

- $3 \leq N \leq 100$.
- $0 \leq T \leq 1000$.
- $1 \leq t_1 \leq t_2 \leq \dots \leq t_N \leq 1000$.

Informações sobre a pontuação

A tarefa vale 100 pontos. Estes pontos estão distribuídos em subtarefas, cada uma com suas **restrições adicionais** às definidas acima.

- **Subtarefa 1 (0 pontos):** Esta subtarefa é composta apenas pelos exemplos mostrados abaixo. Ela não vale pontos, serve apenas para que você verifique se o seu programa imprime o resultado correto para os exemplos.
- **Subtarefa 2 (20 pontos):** $N = 3$.
- **Subtarefa 3 (30 pontos):** $T = 0$.
- **Subtarefa 4 (50 pontos):** Sem restrições adicionais.

Exemplos

Exemplo de entrada 1	Exemplo de saída 1
8 10 5 8 15 20 21 25 42 50	3

Explicação do exemplo 1: Nesse caso, são necessários 3 ônibus para levar todos os alunos:

- O primeiro pode sair no minuto 8, levando os alunos 1 (com tempo de espera de 3 minutos) e 2 (com tempo de espera de 0 minuto).
- O segundo, pode sair no minuto 25, levando os alunos 3, 4, 5 e 6, perceba que, em relação a este ônibus, o aluno com maior tempo de espera será o aluno 3 com 10 minutos de espera.
- Por fim, o terceiro pode sair no minuto 52, levando os alunos 7 (com tempo de espera de 10 minutos) e 8 (com tempo de espera de 2 minutos).

Perceba que, desta forma, nenhum aluno precisou esperar mais de $T = 10$ minutos.

Exemplo de entrada 2	Exemplo de saída 2
3 10 1 11 12	2

Explicação do exemplo 2: Perceba que este exemplo satisfaz as restrições da subtarefa 2.

Exemplo de entrada 3	Exemplo de saída 3
5 0 1 1 2 2 3	3

Explicação do exemplo 3: Perceba que este exemplo satisfaz as restrições da subtarefa 3.