



## Nota

Se você não estiver conseguindo submeter uma solução via CMS, por favor siga as instruções:

- Coloque suas submissões em um diretório com o nome `submit_<task name>` na área de trabalho (o diretório já deve existir) **antes que a prova termine**.
- Para problemas apenas-saída, seus arquivos devem ser nomeados `01.out`, `02.out`, ..., `10.out`.
- Para outros problemas, deve existir **exatamente um** arquivo neste diretório, contendo seu código fonte.
- Peça para o líder do seu time submeter um pedido de recurso.

Se você acredita que deve receber tempo extra, por favor siga as instruções:

- Envie uma requisição de esclarecimento (através do sistema da prova ou por papel) assim que possível.
- Não saia da sua mesa nem fale com outros competidores depois que a prova terminar.

---

Cada tarefa possui um pacote anexo que você pode baixar do sistema da prova.

Podem existir uma ou mais tarefas "apenas-saída", para as quais:

- O pacote anexo contém casos de teste de entrada e casos de teste de exemplo. Cada caso de teste é uma subtarefa separada.
- Você pode submeter múltiplos arquivos de saída como um arquivo zipado. Para esse propósito, seus arquivos de saída devem ser nomeados `?.out`, onde `??` é o número do caso de teste (e.g., `03.out`). Você pode zipar múltiplos arquivos usando o seguinte comando: `zip output.zip *.out`
- Você pode realizar até 100 submissões para tarefas apenas-saída. Em cada submissão, você pode submeter os arquivos de saída para qualquer subconjunto dos casos de teste.

Para outras tarefas:

- O pacote anexo contém corretores exemplo, exemplo de implementações, exemplos de casos de teste, e scripts de compilação.
- Você pode realizar até 50 submissões para cada tarefa, e você deve submeter exatamente um arquivo para cada submissão.
- O nome do arquivo que você deve submeter é dado no cabeçalho do enunciado da

tarefa. Ele deve implementar os procedimentos descritos no enunciado da tarefa usando as assinaturas providenciadas nos exemplos de implementação.

- Você é livre para implementar outros procedimentos.
- Suas submissões não devem ler da entrada padrão, escrever na saída padrão, ou interagir com qualquer outro arquivo. Contudo, elas podem escrever na saída padrão de erro.
- Suas submissões **não devem chamar `exit()` ou `System.exit()`.**
- Ao testar seus programas com os corretores exemplo, sua entrada deve seguir o formato e restrições do enunciado da tarefa, caso contrário comportamentos inesperados podem ocorrer.
- Nas entradas do corretor exemplo, todo par consecutivo de tokens em uma linha são separados por um único espaço, a menos que outro formato seja explicitamente especificado.
- Ao testar seu código na sua máquina local, nós recomendamos que você use os scripts no pacote anexo. Caso contrário, especialmente em C++, tenha certeza de adicionar a opção `-std=gnu++14` ao compilar.

## Convenções

Os enunciados das tarefas especificam assinaturas usando nomes de tipos genéricos `void`, `int`, `int64`, e `int[]` (array).

Em cada uma das linguagens de programação suportadas, os corretores exemplo usam tipos de dados apropriados ou implementações, como listado abaixo:

Linguagem	<code>void</code>	<code>int</code>	<code>int64</code>	<code>int[]</code>	comprimento do array <code>a</code>
C++	<code>void</code>	<code>int</code>	<code>long</code> <code>long</code>	<code>std::vector&lt;int&gt;</code>	<code>a.size()</code>
Java	<code>void</code>	<code>int</code>	<code>long</code>	<code>int[]</code>	<code>a.length</code>

## Limites

Tarefa	Tempo limite	Memória limite
line	output-only	output-only
vision	1 sec	1024 MB
walk	4 sec	1024 MB