



# OBI2007

## Caderno de Tarefas

Modalidade Programação • Seletiva, Teste 3

A PROVA TEM DURAÇÃO DE DUAS HORAS

LEIA ATENTAMENTE ESTAS INSTRUÇÕES ANTES DE INICIAR A PROVA

- Este caderno de tarefas é composto por 1 páginas (não contando esta folha de rosto), numeradas de 1 a 1. 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.

Sociedade Brasileira de Computação

[www.sbc.org.br](http://www.sbc.org.br)

# Paralelogramos

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

A família de Alberto é muito pobre, e por isso muitas vezes os pais de Alberto não tem condições de comprar um brinquedo para ele no Natal. Mesmo assim, Alberto inventa suas próprias brincadeiras e monta os seus próprios brinquedos: seu pai joga fora pedaços de madeira com vários pregos ainda presos, e Alberto brinca de colocar elásticos nos pregos, formando vários tipos de polígono.

Alberto reparou que, dependendo da disposição dos pregos, pode ser possível formar vários paralelogramos (quadriláteros de lados paralelos) com vértices nos pregos. Apesar de tentar várias possibilidades, ele não tem certeza de ter encontrado todos os paralelogramos possíveis e pediu a sua ajuda.

## Tarefa

Escreva um programa que, dadas as posições dos pregos, calcula quantos paralelogramos distintos podem ser formados com vértices nestes pregos.

## 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 um inteiro  $N$ , indicando o número de pontos ( $4 \leq N \leq 1000$ ). As  $N$  linhas seguintes contém cada uma dois inteiros  $X, Y$ , indicando as coordenadas de um dos pontos ( $-1000 \leq X, Y \leq 1000$ ). Todos os pontos dados são distintos.

## Saída

Seu programa deve imprimir, na *saída padrão*, A saída deve ser uma única linha, contendo o número de paralelogramos com vértices nos pontos dados.

<b>Entrada</b>	<b>Entrada</b>	<b>Entrada</b>
5	5	8
0 0	0 0	0 0
0 2	0 2	0 1
1 1	1 1	0 2
2 0	2 0	0 3
2 2	3 3	-1 1
		-1 2
<b>Saída</b>	<b>Saída</b>	1 1
1	0	1 2
		<b>Saída</b>
		13



