Prova Final de Algoritmos e Estruturas de Dados I
6/dezembro/2008

Perguntas comuns e suas respostas:

- P: Tenho uma dúvida na questão tal.
  R: A compreensão do enunciado faz parte da prova.
- P: Se eu consultar algum material próprio ou de algum colega, o que acontecerá comigo?
  R: A prova é individual e sem consulta. Qualquer tentativa de fraude acarretará abertura de processo administrativo na UFPR.
- P: Posso fazer a prova a lápis?
  A prova é um documento, portanto deve ser feita a caneta.
- P: O que será corrigido?
  R: A lógica, a utilização de **estrutura em módulos**, a criatividade, a sintaxe, o uso correto dos comandos, a correta declaração dos tipos, os nomes das variáveis, a indentação, e, evidentemente, a clareza. Em particular, será considerado o correto uso de funções e procedimentos, bem como o bom uso de variáveis locais e globais. Enfim, será avaliada sua capacidade de modularização do código.

Questões (Valores entre parênteses. Total 100 pontos):

1. (60 pontos) Nesta questão você terá que providenciar ligações par-a-par entre diversos pontos distribuídos ao longo de uma rota qualquer. A entrada de dados consiste de uma conjunto de pares $(x, y), 1 \leq x, y \leq MAX$, sendo que o último par a ser lido é o (0,0), que não deve ser processado, apenas indicando final da entrada de dados.

   Para cada par (x, y) dado como entrada, você deve providenciar uma conexão física entre eles. As linhas de uma matriz podem representar a "altura" das linhas de conexão, enquanto que as colunas da matriz podem representar os pontos (x, y) sendo conectados. Um símbolo de "+" pode ser usado para se representar alteração na direção de uma conexão. O símbolo "|" pode ser usado para representar um trecho de conexão na vertical. Finalmente o símbolo "-" pode ser usado para se representar um trecho de conexão na direção horizontal. Quando um cruzamento de linhas for inevitável, deve-se usar o símbolo "x" para representá-lo. Considere que não existem trechos de conexões na diagonal.

   Por exemplo, suponha que a entrada é dada pelos seguintes pares:

```
3 5
2 9
0 0
```

   Uma possível saída para seu programa seria a impressão da seguinte matriz:

```
4
3
2   +-------------+
1   | +---+       |
  1 2 3 4 5 6 7 8 9
```

Outra possível matriz solução para este problema seria esta:

```
4
3
2    +---+
1  +-x---x-------+
 1 2 3 4 5 6 7 8 9
```

Note que nesta última versão foi preciso inserir dois cruzamentos.

Ainda como exemplo, se o par (6,8) também fosse dado como entrada no exemplo anterior, a saída do programa poderia ser assim exibida:

```
4
3          +---+
2  +-------x---x-+
1  | +---+ |   | |
 1 2 3 4 5 6 7 8 9
```

Você deve implementar um programa em *Free Pascal* que seja capaz de ler uma seqüência de pares terminada em $(0, 0)$ (como no exemplo acima) e que imprima o desenho das conexões como saída, também conforme o diagrama acima.

2. (40 pontos) Nesta questão o objetivo é minizar o número de cruzamentos da matriz gerada como solução do problema anterior. Assim, a matriz ideal para ser dada como resposta do último exemplo seria a seguinte:

```
4
3
2  +-------------+
1  | +---+ +---+ |
 1 2 3 4 5 6 7 8 9
```

Você deve deixar indicado quais as alterações no código do exercício 1 teriam que ser implementadas para que o programa minimize os cruzamentos. Quanto menor o número de alterações que você implementar, melhor.