

(***) 1. Os incas construíam pirâmides de base quadrada nas quais a única forma de se atingir o topo era seguir em espiral pela borda, que acabava formando uma rampa em espiral*. Uma “matriz inca” é uma matriz quadrada NxN de números inteiros na qual, partindo do canto superior esquerdo, no sentido horário, em espiral, a posição seguinte na ordem é o inteiro consecutivo da posição anterior. Por exemplo, as matrizes abaixo são incas:

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 12 & 13 & 14 & 5 \\ 11 & 16 & 15 & 6 \\ 10 & 9 & 8 & 7 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 3 \\ 8 & 9 & 4 \\ 7 & 6 & 5 \end{bmatrix}$$

Escreva um programa que gera uma matriz quadrada e a preenche de forma a gerar uma “matriz inca”. O preenchimento deve ser realizado em uma função com o protótipo abaixo:

```
void geraMatrizInca (int matriz [N][N]);
```

*Esta informação não foi verificada, mas não afeta o exercício.

(***) 2. Escreva um programa no qual uma matriz é preenchida com caracteres (use a inicialização direta, com chaves). O programa deve mostrar na tela o padrão espelhado nos 2 eixos. Por exemplo, dado o padrão (com 5 linhas de 5 caracteres):

```
ABCDE
 BCDE
  CDE
   DE
    E
```

O programa mostra a seguinte saída:

```
ABCDEEDCBA
BCDEEDCB
CDEEDC
DEED
EE
EE
DEED
CDEEDC
BCDEEDCB
ABCDEEDCBA
```

Faça 2 versões do programa: em uma, a saída é inteiramente armazenada em uma matriz; na outra, apenas o padrão original deve ser armazenado.

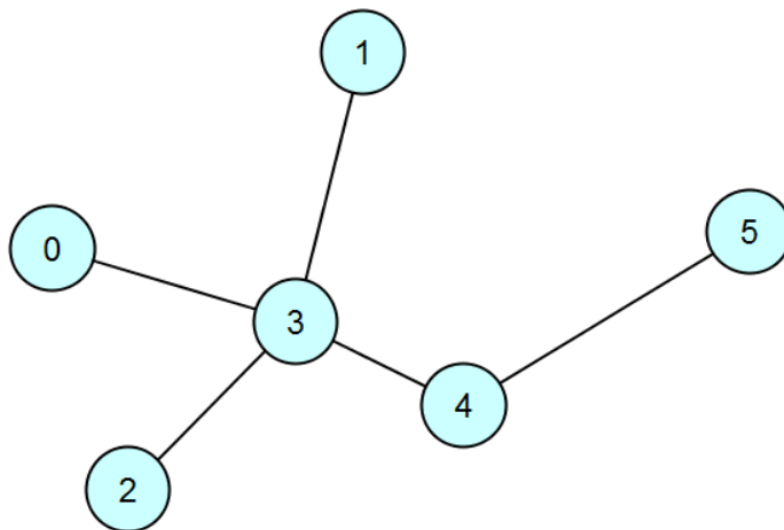
(***) 3. A crescente utilização do transporte aéreo preocupa os especialistas de Pindorama, que preveem que o congestionamento em aeroportos poderá se tornar um grande problema no futuro. Os números atuais já são alarmantes: relatórios oficiais demonstram que, em no mês passado, houve uma média de 7000 atrasos de voos por dia. Preocupada com a previsão dos seus especialistas em tráfego aéreo, a Associação de Transporte Aéreo Pindorâmiano (ATAPI) está começando um estudo para descobrir quais são os aeroportos onde o tráfego aéreo pode vir a ser mais problemático no futuro. Além disso, eles estão interessados em determinar trajetos possíveis entre aeroportos considerando escalas.

Considere que os dados sobre a malha aérea foram lidos a partir de um banco de dados, e estão em uma matriz $V \times 2$, onde V é o número de vôos existentes. Os aeroportos são identificados por inteiros de 0 a $A-1$, onde A é o número de aeroportos. Cada linha da matriz possui os dados de um vôo, dado por um par (i, j) , que indica que existe um voo do aeroporto i para o aeroporto j , e vice-versa - todo voo sempre existe nas duas direções (ida e volta). Para simplificar os testes, vamos supor que os números de aeroportos e vôos são constantes, e que a matriz descrevendo os vôos está declaradamente diretamente no código-fonte. Por exemplo:

```
#define N_AEROPORTOS 5
#define N_VOOS 5

int voos [N_VOOS][2] = {{0,3},{1,3},{2,3},{4,3},{4,5}};
```

Neste problema, a malha aérea é representada como um grafo – um conjunto de *vértices* (os aeroportos) conectados por *arestas* (os voos). Considere a figura abaixo, que corresponde à matriz do exemplo acima: o grafo indica que existe apenas um voo direto a partir do aeroporto 0, que o conecta ao aeroporto 3. A partir do aeroporto 3, por outro lado, só não existe voo direto até o aeroporto 5.



Para representar computacionalmente o grafo que modela a malha aérea, usaremos *matrizes de adjacência*: uma matriz quadrada $A \times A$ na qual a posição (i, j) tem valor 1 se existir uma conexão entre i e j , e zero do contrário. Por exemplo, para o grafo acima, a matriz de adjacência é:

$$M = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

Como programador recém contratado pela ATAPI você foi encarregado de:

1. Criar uma função que recebe a matriz de voos e uma matriz de saída, com conteúdo inicial indeterminado, que deve ser preenchida com os dados da matriz de adjacência correspondente aos voos. A matriz de saída já possui o tamanho correto.

2. Definir uma função que receba como entrada uma matriz de adjacência que representa as entradas e retorne o identificador do aeroporto com maior número de voos (considere tanto os que chegam quanto os que partem de cada aeroporto).

3 (☹☹☹). Criar um programa (defina quantas funções julgar adequadas) que permita procurar por voos entre dois aeroportos quaisquer respeitando algumas condições (parâmetros e opções que devem ser determinados pelo usuário):

- a) **Sem** conexões.
- b) Com **exatamente** N conexões.
- c) Com **até** N conexões.

Dica: crie funções para tarefas como mostrar, copiar e multiplicar matrizes.

Dica para a 3.b e 3.c: se M é a matriz de adjacência de um determinado grafo, seu quadrado, a matriz M^2 , representa, em cada uma de suas células, o número de ligações entre as cidades correspondentes com exatamente dois voos! Você consegue justificar o porquê? Como podemos explorar isto para o caso geral, com um número arbitrário de voos?

Obs: neste problema, não estamos preocupados com as distâncias ou custos de cada voo, apenas com a existência ou não de voos e com o número de escalas necessárias (ou possíveis) para voar entre duas localidades.

Obs: considere que idas e vindas são permitidas. Por exemplo, o caminho $A1 \rightarrow A2 \rightarrow A1 \rightarrow A2$ é um trajeto válido com 3 conexões entre $A1$ e $A2$! E $A1 \rightarrow A2 \rightarrow A1$ é um trajeto válido entre $A1$ e $A1$!!!