



Questão 1.

Início

```
tipo v = vetor [1:50] inteiro;  
v: VET;  
inteiro: elemento, i, quantidade;  
Para i de 1 até 50 passo 1 faça  
  Leia (elemento);  
  VET[i] ← elemento;  
  Imprimir (elemento);  
  Se (elemento < 0)  
    então VET[i] ← 999;  
    quantidade ← quantidade + 1;  
  fim-se;  
fim-para;  
Para i de 1 até 50 passo 1 faça  
  Imprimir (VET[i]);  
fim-para;  
Imprimir (quantidade);
```

Fim.



Questão 2.

Início

```
tipo mat = matriz [1:6, 1:4] inteiro;
tipo vet = vetor [1:6] inteiro;
mat: MAT;
vet: VET;
inteiro: i, j, maior;
Para i de 1 até 6 passo 1 faça
    Para j de 1 até 4 passo 1 faça
        Leia(MAT[i,j]);
    fim-para;
fim-para;
Para i de 1 até 6 passo 1 faça
    maior ← MAT[i,1];
    Para j de 1 até 4 passo 1 faça
        Se (MAT[i,j] > maior)
            então maior ← MAT[i,j];
        fim-se;
    fim-para;
    VET[i] ← maior;
fim-para;
Para i de 1 até 6 passo 1 faça
    Para j de 1 até 4 passo 1 faça
        MAT[i,j] ← MAT[i,j] * VET[i];
        Imprimir(MAT[i,j]);
    fim-para;
fim-para;
```

Fim.



Questão 3.

Início

```
tipo mat = matriz [1:5, 1:5] inteiro;
tipo vet = vetor [1:25] inteiro;
mat: MAT;
vet: VET1, VET2, VET3;
{VET1 representa todos os elementos, VET2 representa os
elementos repetidos, VET3 representa a quantidade}
inteiro: i, j, pos;
booleano: repetidos;
Leia(MAT);
pos ← 0;
Para i de 1 até 5 passo 1 faça
    Para j de 1 até 5 passo 1 faça
        pos ← pos + 1;
        VET1[pos] ← MAT[i,j];
        VET2[pos] ← 0;
        VET3[pos] ← 0;
    fim-para;
fim-para;
pos ← 1;
Para i de 1 até 25 passo 1 faça
    Se (VET1[i] != -1)
        então VET2[pos] ← VET1[i];
        pos ← pos + 1;
    fim-se;
    Para j de 1 até 25 passo 1 faça
        Se (VET2[pos - 1] = VET1[j])
            então VET1[j] ← -1;
            VET3[pos-1] ← VET3[pos-1] + 1;
        fim-se;
    fim-para;
fim-para;
repetidos ← falso;
Para i de 1 até 25 passo 1 faça
    Se (VET3[i]>1)
        então Imprimir (VET2[i], VET3[i]);
        repetidos ← verdadeiro;
    fim-se;
fim-para;
Se (!repetidos)
    então Imprimir ("Não há elementos repetidos.");
fim-se;
```

Fim.