

Participação no Torneio TECLA

A edição de 2012 do Torneio Estudantil de Computação multi-Linguagem de Aveiro (TECLA) conta com cerca de 450 participantes oriundos de 36 escolas secundárias e profissionais dos distritos de Aveiro, Braga, Coimbra, Porto e Viseu.

Após uma fase de testes onde as cinco equipas participantes da escola puderam praticar (respondendo a questões de A a Z) e submeter os seus programas, chegou a fase preliminar, que decorreu no dia 8 de fevereiro, na sala 3.33 da nossa escola, tendo participado duas equipas da turma 12.º AV. Os alunos, que tiveram uma introdução à programação na disciplina de Aplicações Informáticas B, elaboraram programas recorrendo à linguagem de programação Pascal.



Ao acederem à plataforma, os alunos depararam-se com cinco desafios, o primeiro era o seguinte:

Barbeiro



Introdução

O António barbeiro tem a sua barbearia aberta de 2ª a 6ª-feira e cobra por cada corte 6€ de cabelo. No entanto, quando chega ao final da semana nunca sabe quanto ganhou.

Com a resolução deste problema vai ajudar o barbeiro a fazer as suas contas.

Problema

Dado o número de cabelos que o António barbeiro corta, ajude-o a saber quanto ganhou no final da semana.

Dados de entrada

O número de cabelos cortados em cada dia da semana. A cada coluna corresponde um dia da semana.

Dados de saída

Quanto o barbeiro ganhou no final da semana.

Limites de teste

Em cada dia o barbeiro corta no máximo 30 cabelos.

As equipas rapidamente submeteram com sucesso os seus programas:

Equipa 4 Rodrigo e Pagaimo

```
Program Barbeiro;
var cabelo1, cabelo2, cabelo3, cabelo4, cabelo5: integer;
soma: integer;
Begin
  Read (cabelo1);
  Read (cabelo2);
  Read (cabelo3);
  Read (cabelo4);
  Read (cabelo5);

  If (cabelo1 >= 30) then
    soma := 30
  else soma:= cabelo1;

  If (cabelo2 >= 30) then
    soma := soma + 30
  else soma:= soma + cabelo2;

  If (cabelo3 >= 30) then
```

```

soma := soma + 30
else soma:= soma + cabelo3;

If (cabelo4 >= 30) then
soma := soma + 30
else soma:= soma + cabelo4;

If (cabelo5 >= 30) then
soma := soma + 30
else soma:= soma + cabelo5;

Writeln (soma * 6)
End.

```



Equipa 5 Carlos e Amável:

```

Program A ;
var dia1,dia2,dia3,dia4,dia5,total : integer;
Begin
    read(dia1);

    if dia1 > 30 then
        begin;
            dia1:=30
        end;
    read(dia2);
    if dia2 > 30 then
        begin;
            dia2:=30
        end;

```

```
read(dia3);  
if dia3 > 30 then  
    begin;  
        dia3:=30  
    end;  
  
read(dia4);  
if dia4 > 30 then  
    begin;  
        dia4:=30  
    end;  
read(dia5);  
if dia5 > 30 then  
    begin;  
        dia5:=30  
    end;  
  
total:=(6*dia1)+(6*dia2)+(6*dia3)+(6*dia4)+(6*dia5);  
  
writeln(total);  
End.
```



Depois surgiu novo desafio:

Austeridade



Introdução

Uma das medidas de estabilização financeira do país incide sobre os vencimentos dos funcionários públicos correspondentes aos subsídios de férias e de Natal (cada subsídio tem valor igual a um salário mensal). No ano de 2011 o corte foi de 50% no valor do subsídio de Natal acima do salário mínimo nacional (485€). No ano de 2012 as regras não são iguais para todos os funcionários públicos. Os trabalhadores com salários iguais ou inferiores a 600€ não têm qualquer corte. Trabalhadores com salários acima de 1100€ não recebem qualquer subsídio. Todos os outros trabalhadores terão um corte em ambos os subsídios. O valor do corte de um subsídio é dado pela fórmula:

$$1320 - 1.2 \times \text{salário}$$

Problema

A sua tarefa é calcular o montante anual recebido, 12 salários mais os subsídios, nos anos 2010 (sem austeridade), 2011 e 2012.

Dados de entrada

Os dados de entrada são compostos por apenas uma linha que representa o vencimento mensal v do funcionário público.

Limites

- $0 \leq v \leq 5000, v \in \mathcal{R}$

Dados de saída

A aplicação a criar deverá ter como saída 3 linhas. A primeira linha corresponde ao total anual para um ano sem austeridade, a segunda linha corresponde ao total anual do ano 2011 e a terceira linha corresponde ao total anual do ano 2012. Todos os valores de saída são arredondados a 1 casa decimal.

Ao que as equipas responderam:

Equipa 4 Rodrigo e Pagaimo

```
Program Austeridade;
var salariomensual, salario2010, salario2011, salario2012: real;
Begin
  Readln (salariomensual);
  salario2010:= 14*salariomensual;

  If (salariomensual > 485) then
    salario2011:= 13.5*salariomensual
  else salario2011:= 14*salariomensual;

  If (salariomensual <= 600) then
    salario2012 := 14*salariomensual
  else
    If (salariomensual <= 1100) then
```

```

salario2012:= salariomensual-( 1320-1.2*salariomensual) + salariomensual-( 1320-
1.2*salariomensual) + 12*salariomensual
else salario2012:= 12*salariomensual;

Writeln ( salario2010:0:1);
Writeln ( salario2011:0:1);
Writeln ( salario2012:0:1);

End.

```

Equipa 5 Carlos e Amável:

```

Program B ;
var v:integer ;
    Salario1,Salario2,Salario3:real;

Begin
    read(v);
    if (v>=0)and(v<=5000) then
    begin

        //2010
        Salario1:=14*v;

        //2011

        if (v>485) then
            begin;
                Salario2:=13.5*v;
            End;
        if (v<=485) then
            begin;
                Salario2:=14*v;
            End;
        //2012

        if (v<=600) then
            begin;
                Salario2:=14.5*v;
            End;
        if (v>1100) then
            begin;
                Salario3:=12*v;
            End;
        if (v>600)and(v<=1100) then
            begin;
                Salario3:=v-(1320-1.2*v)+v-(1320-1.2*v)+12*v
            End;

        writeln( Salario1:5:1);
        writeln( Salario2:5:1);
        writeln( Salario3:5:1);
    end;

End.

```

Aqui a plataforma da universidade começou a dar problemas, ficou lenta e inacessível, os programas apesar de corretos não foram validados, por isso, e para não perder tempo, as equipas aventuraram-se no terceiro desafio, desafio este muito mais complexo e exigente que as equipas não conseguiram concluir.

Vito Deadstone



Introdução

O famoso gangster Vito Deadstone está a preparar a sua mudança para Nova York. Ele já tem uma grande família em Nova York, e todos vivem na mesma rua: Lamafia Avenue. Sabendo que ele gosta de manter contacto permanente com todos os seus parentes, ele pretende encontrar a localização ideal para a sua casa nessa mesma rua.

Problema

A tua tarefa é a de escrever um programa que encontre a localização que minimize a distância total para todos os parentes de Vito Deadstone.

Dados de entrada

Para cada caso de teste será dada a sequência de números de rua $s_1, s_2, \dots, s_i, \dots, s_r$. Note-se que vários parentes podem ter casa com o mesmo número.

Limites

- $2 \leq r \leq 50, r \in \mathcal{N}$
- $0 \leq s_r \leq 100, s_r \in \mathcal{N}$

Dados de saída

O seu programa deve devolver a localização ideal para a nova casa de Vito Deadstone, ou seja a que minimiza a soma de todas as distâncias para as casas dos seus familiares. Assuma que a distância entre as casas s_i e s_j é $d_{ij} = |s_i - s_j|$

Atendendo a que os alunos não têm nos seus currículos disciplinas específicas de programação, apenas tiveram contato com uma linguagem de programação e apenas puderam participar duas equipas, os resultados só podem ser considerados positivos, servindo este desafio para abrir novos horizontes num possível futuro profissional.

E assim se passou uma tarde à frente do computador a competir, programar, exercitar o cérebro e a resolver programas utilizando a linguagem Pascal.

Os professores da disciplina de Aplicações Informáticas B

Artur Freitas e Pedro Alves

Os restantes desafios:

Histogramas



Introdução

A empresa BorlaJanta Lda todos os meses oferece a um dos seus 4 empregados um jantar. É tirado à sorte o empregado que recebe o jantar grátis.

Problema

Pretende-se com este problema fazer um histograma do número de vezes que cada empregado recebe o jantar grátis ao longo de um ano.

Dados de entrada

As 4 primeiras linhas correspondem ao número do empregado e respetivos nomes. A 5ª linha contém 12 colunas, cada uma indica o número n do empregado premiado com o jantar.

Limites

- $1 \leq n \leq 4, n \in \mathcal{N}$

Dados de saída

A saída de dados deve ser criada de acordo com o seguinte conjunto de regras:

- Na primeira linha aparece o caractere '^' seguindo-se nove espaços em branco;
- As n linhas seguintes, onde n representa o maior número de entre o número de jantares oferecidos aos empregado durante o ano, têm a seguinte estrutura:
 - A primeira coluna contém o caractere '|';
 - As 2ª, 4ª, 6ª e 8ª colunas têm, de acordo com o número de jantares oferecidos aos empregados, '*' ou espaço em branco;
 - As 3ª, 5ª, 7ª, 9ª e 10ª coluna têm espaço em branco;
 - A linha $n+2$ é composta por '+----->';
 - A linha $n+3$ tem os números correspondentes aos empregados nas colunas 2, 4, 6 e 8 (as colunas 1, 3, 5 e 7 são espaços em brancos);
 - A linha $n+4$ é uma linha em branco;
 - As últimas 4 linhas têm a seguinte frase 'O <nome do empregado> teve direito a <k> jantares.', os empregados estão ordenados pelo número do empregado.



Introdução

Pretende-se construir um programa que ajude um pequeno rato a encontrar a saída de um labirinto onde se encontra perdido. Assuma que o labirinto possui uma e apenas uma saída e que apenas existe um único caminho possível.

Problema

Dados de entrada

A primeira linha contém três valores inteiros: dimensão N do labirinto (comprimento é igual à largura); a linha l correspondente à posição inicial (inicia em zero); e a coluna c correspondente à posição inicial (inicia em zero). As seguintes linhas N linhas contêm a configuração do labirinto, em que as paredes são representadas por '1' e os espaços vazios por '0'.

Limites

- $0 \leq N \leq 10, N \in \mathcal{N}$
- $0 \leq l \leq N - 1, l \in \mathcal{N}$
- $0 \leq c \leq N - 1, c \in \mathcal{N}$

Dados de saída

O seu programa deve devolver o conjunto de coordenadas pelas quais o rato deve passar e que correspondem ao caminho mínimo para sair do labirinto. Cada coordenada é composta por um número l que corresponde à linha (inicia em zero); e por um número c que corresponde à coluna (inicia em zero), separados por “,”. A primeira coordenada corresponde à posição inicial do rato e a última à última coordenada do labirinto. As coordenadas são separadas por “- >”.