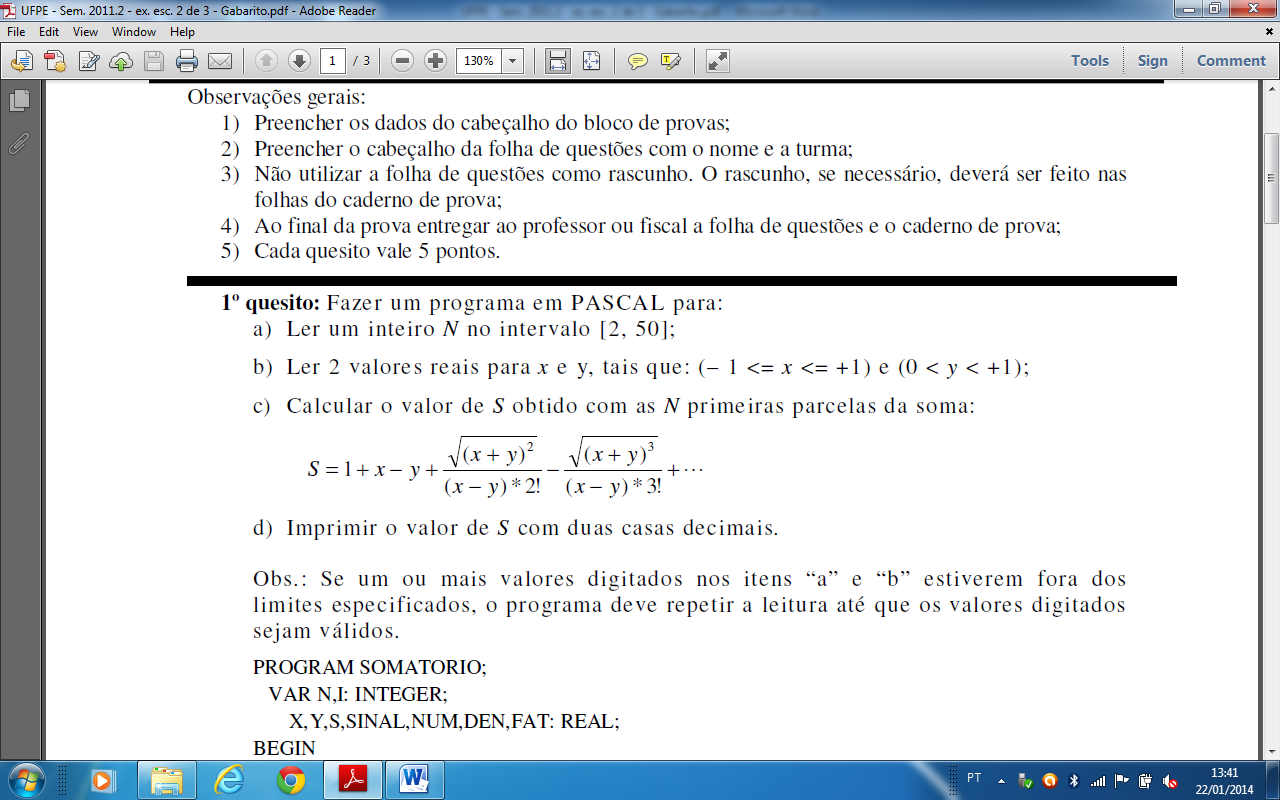
**2º EE – 2011.2**

**1º quesito:** Fazer um programa em PASCAL para:

a) Ler um inteiro *N* no intervalo [2, 50];

b) Ler 2 valores reais para *x* e y, tais que: (– 1 <= *x* <= +1) e (0 < *y* < +1);

c) Calcular o valor de *S* obtido com as *N* pr imeiras parcelas da soma:



d) Imprimir o valor de *S* com duas casas decimais.

**Obs.:** Se um ou mais valores digitados nos itens “a” e “b” estiverem fora dos limites especificados, o programa deve repetir a leitura até que os valores digitados sejam válidos.

#include <stdio.h>

int main ()

{

int n, i;

float x, y, s, sinal, num, den, fat;

do

{

printf (“Entre N:”);

scanf (“%d”, &n);

} while (n<2 || n>50);

do

{

printf (“X, Y = ?”);

scanf (“%f%f”, &x, &y);

} while (x<-1 || x>1 || y<=0 || y>=1);

if (n==2)

s = x+1;

else

s = 1+x+y;

sinal = 1;

num = x+y;

den = x-y;

fat = 1;

for (i=4; i<=n; i++)

{

num = num\*(x+y);

fat = fat\*(i-2);

s = s+(sinal\*(sqrt(num)/(den\*fat)));

sinal = sinal\*(-1);

}

printf (“Resultado: %f”, s);

return 0;

}

**2º quesito:** Faça um programa em Pascal que calcule a distância entre diferentes cidades satisfazendo as restrições abaixo:

**O programa deve ler do teclado:**

a) Um vetor de caracteres **Cidades** que armazene o nome de *m* diferentes cidades. O valor de *m* deve ser fornecido pelo usuário – **validar** para que o valor de *m* esteja no intervalo [2,10];

b) Uma matriz **Distancias** *m* x *m* correspondendo a distância entre *m* as cidades definidas no vetor **Cidades**.

**Obs.: -** Nesta matriz o valor da diagonal principal (representando a distância de uma cidade

para ela mesma) deve ser atribuído automaticamente como 0, ou seja, não deve ser permitida a digitação.

**-** Não é necessário supor que a distância entre cidade A e B deverá será igual à distância entre B e A. Por exemplo, na tabela abaixo a distância entre A e B é 90 Km e

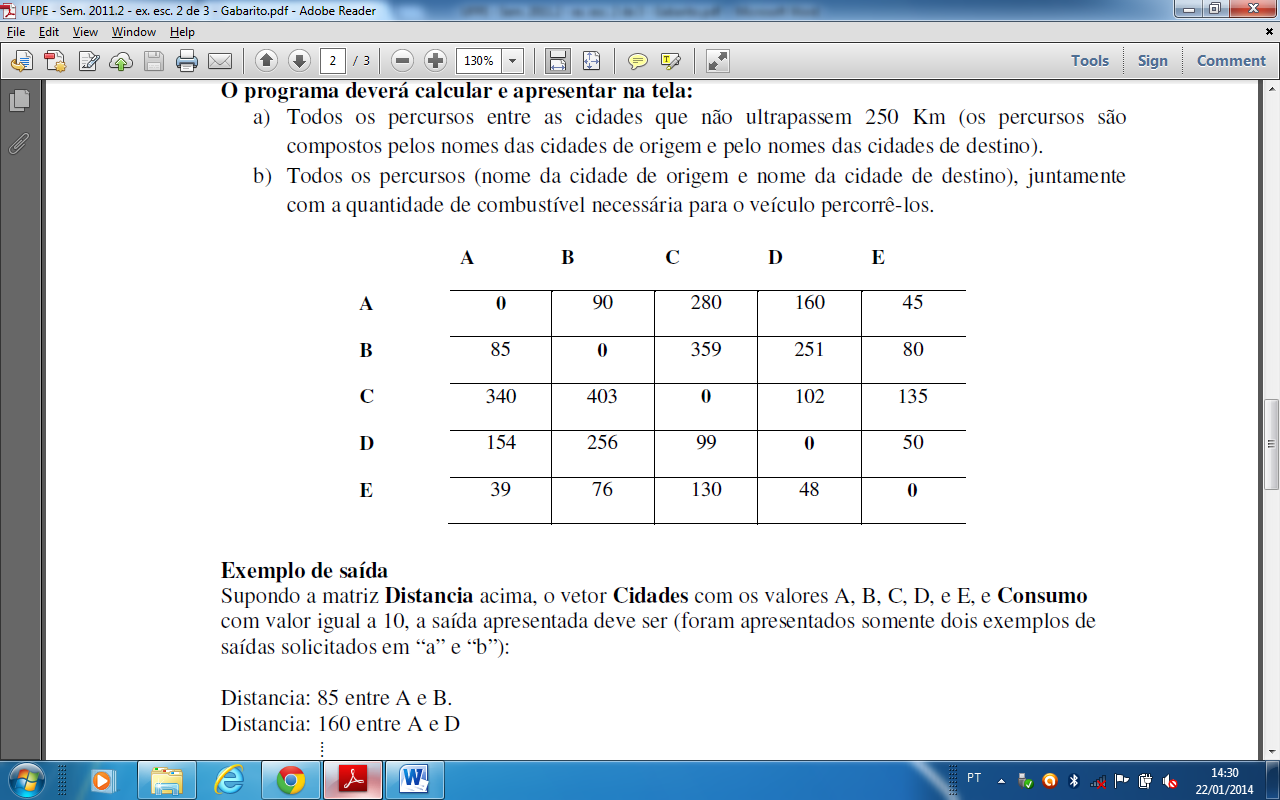
a distância entre B e A é de 80 Km.

c) O valor para a variável **Consumo**, correspondendo ao consumo de combustível médio do veículo utilizado, ou seja, quantos quilômetros este veículo percorre com um litro de combustível.

**O programa deverá calcular e apresentar na tela:**

a) Todos os percursos entre as cidades que não ultrapassem 250 Km (os percursos são compostos pelos nomes das cidades de origem e pelo nomes das cidades de destino).

b) Todos os percursos (nome da cidade de origem e nome da cidade de destino), juntamente com a quantidade de combustível necessária para o veículo percorrê-los.



**Exemplo de saída**

Supondo a matriz **Distancia** acima, o vetor **Cidades** com os valores A, B, C, D, e E, e **Consumo** com valor igual a 10, a saída apresentada deve ser (foram apresentados somente dois exemplos desaídas solicitados em “a” e “b”):

Distancia: 85 entre A e B.

Distancia: 160 entre A e D

⁞

Consumo entre A e B = 8.50

Consumo entre A e C = 28.00

⁞

#include <stdio.h>

int main ()

{

int m, i, j;

float consumo, qtde;

float distancia[10][10];

char cidade[10][30];

do

{

printf (“Digite o número de cidades:”);

scanf (“%d”, &m);

} while (m<2 || m>10);

for (i=0; i<m; i++)

{

printf (“Digite o nome da cidade %d”, i+1);

scanf (“%s”, cidade[i]);

}

printf (“Digite o consumo:”);

scanf (“%f”, &consumo);

for (i=0; i<m; i++)

for (j=0; j<m; j++)

if (i==j)

distancia[i][j]=0;

else

{

printf (“Digite a distancia entre %s e %s”, cidade[i], cidade[j]);

scanf (“%f”, distancia[i][j]);

}

for (i=0; i<m; i++)

for (j=0; j<m; j++)

if (distancia[i][j]<=250 && distancia[i][j]>0)

printf (“Distancia entre %s e %s é: %f”, cidade[i], cidade[j], distancia[i][j]);

for (i=0; i<m; i++)

for (j=0; j<m; j++)

if (i!=j)

{

qtde = distancia[i][j]/consumo;

printf (“O consumo entre %s e %s é igual a %f”, cidade[i], cidade[j], qtde);

}

return 0;

}