//

// Newton-Parallel.cpp

// Xcode

//

// Created by Vinod Myll Mylsamy on 5/1/15.

// Copyright (c) 2015 Vinod Myll Mylsamy. All rights reserved.

//#include <iostream>

#include <stdio.h>

#include <math.h>

#include <sys/time.h>

using namespace std;

int main(int argc, const char \* argv[])

{

printf("Hello, World!\n");

int n=4;

int i,j,k;

double hessian[n+1][n+1];

double f[n+1][n+1];

double fn[n+1];

double x[5]= {1,1,1,1,1};

double gradient[n+1];

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

{

if(i==0)

{

fn[i]= (100 \* ((x[i]\*x[i])- x[i+1]) \* ((x[i]\*x[i])- x[i+1]))+ ((x[i]-1) \* (x[i]-1));

}

else

{

fn[i] = fn[i-1] + (100 \* ((x[i]\*x[i])- x[i+1]) \* ((x[i]\*x[i])- x[i+1]))+ ((x[i]-1) \* (x[i]-1));

}

}

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

{

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

{

if(i==0 && j==0)

{

f[i][j] = (800 \* x[0]\*x[0])+ (400 \* ((x[0]\*x[0])-x[1]))+2;

}

else if(i== n && j== n)

{

f[i][j] = 200;

}

else if (i==j && i!=n)

{

f[i][j] = (400 \*((x[i]\*x[i])-x[i+1]))+ (800\* x[i]\*x[i])+202;

}

else if (i!=j)

{

f[i][j] = -400 \* x[i];

}

}

}

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

{

if(i==0)

{

gradient[0] = (400 \* x[0] \* ((x[0] \* x[0])-x[1])) + (2\* (x[0] -1 ));

}

else if (i== n)

{

gradient[n] = (-200 \* (( x[n-1] \* x[n-1])- x[n]));

}

else if ( i < n)

{

gradient[i] = (-200 \* ((x[i-1] \* x[i-1]) - x[i])) + ( 400 \* x[i] \* (( x[i] \* x[i])-x[i+1])) + ( 2\* (x[i] - 1));

}

}

printf("Printing the gradient vector \n");

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

{

printf("%f\n",gradient[i]);

}

printf("Printing the HessianMatrix\n");

for(i=0; i<=1; i++)

{

for(j=0;j<=1;j++)

{

hessian[i][j] = f[i][j];

printf("%f\t",hessian[i][j]);

}

printf("\n");

}

return 0;

}