Problem 1: Implement the complex step derivation explained in the third week lecture video Week3Video1IEEE.mp4.
Provided files:
hw3.1.h, an include file that
- Defines the virtual base clase Derivable

Derivable has one pure virtual function: eval that takes as argument *coord*, the *complex* coordinate where the function has to be evaluated and returns the complex value of the function at *coord*

```
class Derivable
{
public:
virtual complex<double>eval(const std::complex<double>coord)=0;
};
```

- Declares the function that implements the complex step derivative. The signature of this function has two arguments is

```
double firstDerivative(Derivable&, const double where);
```

First Parameter: Derivable& - a reference to an object of class derivable
Second Parameter: const double where - the coordinate where the derivative is required
hw3.1_main.cpp, the main program that computes the required derivatives.

Tasks

- Define and implement a derived class, **MyPotential** that implements the eval function for the following potential

$$(\frac{1}{r^2} - \frac{1}{r^6})$$

- Define and implement a derived class, **MyCosine** that implements the eval function for the cosine function

- Implement the function **firstDerivative** that evaluates the first derivative of the function evaluated by a *Derivable* object at the coordinate *where*. This function should estimate the stepsize h that computes the derivative with an error of the order of 1 ULP. (Refer to the IEEE lecture). Compare the accurcy of the derivative using the complex step to the accuracy obtained by the standard numerical compution of derivatives.

Deliverables

- An include file **MyPotential.h** that declares the class **MyPotential**
- An include file **MyCosine.h** that declares the class **MyCosine**
- An implementation file **hw3.1_impl.cpp** that contains the implementaion of the eval functions for both classes and the implementation of the firstDerivative function

HINT: compilation (assuming all files in the same directory)

```
g++ -c -I. hw3.1_impl.cpp
g++ -c -I. hw3.1_main.cpp
g++ -o hw3.1 hw3.1_impl.o hw3.1_main.o
```

HINT: how to compute derivatives
First Derivative - Classical Approach

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h)$$

First Derivative - Complex Step

$$f'(x) = \frac{img[f(x+ih)]}{h} + \mathcal{O}(h^2)$$

where $i = \sqrt{-1}$

Problem 2: Given a number x, print a list of all non-increasing sequences of numbers that sum up to x. The program gets the value of x from the command line. For example:
./hw3.2 4
Output Example: for $x = 4$ the program should generates the list

```
[4]
[3 1]
[2 2]
[2 1 1]
[1 1 1 1]
```

Print each list when it is generated. Do not store the lists and print all of them at the end.