C++ programming

EXERCICES
Basic problems \textit{(for newbies)}

- Try to estimate the value of PI using a limited development

\[ \zeta(2) = \frac{\pi^2}{6} = \sum_{n=1}^{\infty} \frac{1}{n^2} = \frac{1}{1^2} + \frac{1}{2^2} + \frac{1}{3^2} + \frac{1}{4^2} + \ldots \]

  - Estimate the accuracy of the estimation for a given order

- Create a program which provide the roots of a second order polynomial: \( y = ax^2 + bx + c \)
  
  - User should enter the parameters \( a, b, c \)
  
  - Once the roots are given, ask the user if he wants to solve another function
  
  - NB: you can try to use \texttt{std::complex} variables
```cpp
#include <iostream>

int main(){
    float sum=0;
    float average=0;

    for(int i=0;i<=100;i++){
        for(int j=0;j<=50;j++){
            sum+=i*j;
        }
    }

    average=sum/(100*50);

    for(int i=0;i<=100;i++){
        for(int j=0;j<=50;j++){
            rms=(pow(i*j,2)-average*average)
        }
    }

    rms=sqrt(rms);

    cout<<'These are the results:'<<endl;
    cout<<' Sum = '"<<sum<<endl;
    cout<<' Average = '"<average<<endl;
    cout<<" RMS = "<<rms<<endl;
}
```
Numerical uncertainties

• Produce two curves of $f(x)$ for the 2 following expressions

$$P_1(x) = x^7 - 7x^6 + 21x^5 - 35x^4 + 35x^3 - 21x^2 + 7x - 1$$

$$P_2(x) = (x - 1)^7$$

• Compute the numerical derivative of several well known functions (poly, cos, exp, ..) and check how the uncertainties evolves with $h$

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

• You can proceed similarly for second order derivative
Evaluation of execution time

• Try to evaluate the following tasks
  – Time to print a message on the terminal
  – Time to read a line from a file
  – Number of elementary operations done per secondes (+,*,...)
  – Compare the operation a=a+b with a+=b
  – Evaluate the impact of using inline functions

• Several approach can be used to evaluate execution time
  – time ./prog
  – using ctime
  – using high resolution chrono in C++11
=> Comment the different options
Unexpected behaviours

• Print the value of a non-initialized double
  (run the program several times)

• Try to access to a non-initialized pointer

• Try to access to an element of an array out-of-range
Exercises

I/O
- Standard, files, error

Class
- Basics
- Inheritance
- Polymorphism
- Templates

STL
- Vector
- String
- ...

Preprocessor commands

Compilation
- Optimization
- Lib
- Makefile

Templates
- Preprocessor commands
- Documentation
- Advices

Main
- Variables
- Pointers (address, array)
- Operators
- Instructions
- Functions
- Exceptions
I/O: cin/cout/cerr

- **cin**
  - Write a small piece of code which read values entered by the user
  - Redirect the flux
    ./aout < log.in

- **cout/cerr**
  - Write a small piece of code in which you report both standard and error messages
  - Try to redirect separately each of them in the command line while running:
    .a.out > log.out 2> log.err
Formatting the output

• Try several option to format output results
  – Choose a number of (fixed/not fixed) digits
  – Choose a scientific notation
  – Use horizontal tabulation
  – Use vertical tabulation
  – ...

• Write a program that
  – writes our results in a latex format (tables)
  – compile the latex document
Input file and configuration

• Create a code that is able to read a configuration file as following

```plaintext
#nof_channels 12
#data_file file.dat
...
```

• Create a code that is able to read a data file as following

```plaintext
#channel 1 energy time
#channel 2 energy time
...
```
Functions and arguments

• Create functions taking a vector of integer as argument
  – A function that removes negative numbers
  – A function that returns a vector containing only the positive numbers
  – A function that returns the sum of all values

• Choose the best option to provide the argument between copy/pointer/reference
  – Compare also the performances when several options are possible
std::vector

- Get use to std::vector
- Try several options to fill it
- Try several options to loop over and print the value
- Try to remove a element
- Sort the element by increasing (decreasing) order
- Redo the same with more complex object (ex: structure) and more complex rule to order
std::string

- Get used to std::string
- Create the name of n files with the following structure
  MyFile_i.txt
- Create all the names and change them as following
  file-i.dat
- While reading the list of name, retrieve the value of « i » as an integer in order to use it for computation

- Create a string that correspond to a text.
- Parse it and compute the number of times where the letter a appear
- Go futher and count the number of occurrence for each letter and order the result in a table
Arguments of the main

• Read a program that reads the arguments of the main
• Send a « help » message if there is no argument
• Option –o output file: message are sent in a output file
• Option –i input file: specify input file name
• Other fancy options are possible ...
Exercises

Class
- Basics
- Inheritance
- Polymorphism

STL
- Vector
- String
- ...

I/O
- Standard, files, error

Main
- Variables
- Pointers (address, array)
- Operators
- Instructions
- Functions
- Exceptions

Preprocessor commands

Compilation
- Optimization
- Lib
- Makefile

Documentation

Advises
Preprocessor command

• Create a function that uses preprocessor variables and returns a « formatted » error message
  → Help to deal with error message
  → Use a string as parameter to customize your message

• Uses #define command

• Use conditional statements (#if, ...)
Manipulation of bits

- Convert information from a file/string ... into a series of binary info
- Do binary operation to sort the information
Random

• Generate random numbers following a gaussian distribution and produce a graphic with

• Use many options to do it:
  – Use rand() function and the Box-Muller transform
  – Use C++11 function normal_distribution
  – Use eventually ROOT functions
Basic class
Snooker

- **Class Snooker**
  - Dimension
  - Position & diameter of the pocket
  - CoeffFric5ion

- **Class Ball**
  - Diameter
  - Color
  - Current position (x,y)
  - Current speed (vx,vy)

- **Class GamePlay**
  - Collection of ball
  - Dynamic of the balls
ESIPAP

Promotion 2015

**Applications:**
- Compute mean, rms, min, max per course
- Sort the courses w.r.t mean
- Sort the student per course
- Sort the student w.r.t average

**Student**
- Name
- ... identity ...
- M2 or PhD
- List of marks

**Course**
- Name
- Teacher
- #hours
- Results (mean, rms, min, max)

**Mark**
- Link to a course
- Value