CS319: Scientific Computing (with C++)

Niall Madden (Niall.Madden@NUIGalway.ie)

Week 12: Review

9am, 4 May, and 4pm, 5 May, 2021

1 Module review
- THE END!!!!!!!!

The assessment for CS319 is based on

1. 50% based on lab assignments:
   - ▶ 10% for each of Lab 2 and Lab 3;
   - ▶ 15% for Lab 4+5 and
   - ▶ 15% for Lab 7 (~~due today 212pm~~).

2. 50% based on your project work:
   - (i) Initial Project Plan/Discussion [**5 Marks**] ✓
   - (ii) 250 word Project Proposal [**5 Marks**], detailing ✓
     - (a) An external data source, so that you can show your expertise in read from and/or writing to files.
     - (b) A `class` (or set of classes) that you design yourself
     - (c) An algorithm that preforms some type of useful calculation
   - (iii) 3 page project report [**15 Marks**] ✓
   - (iv) Project code [**25 Marks**] ✓

{ Proposed deadline for report and code is 5pm, Friday **14** May. OK?? }

Usual reminders...

	Mon	Tue	Wed	Thu	Fri
9 − 10		Recorded Class			
10 − 11		Project ✔			
11 − 12	•				
12 − 1					
1 − 2		Project ✔			
2 − 3					
3 − 4					
4 − 5			Project		

1. Just one, short recorded class this week (and this is it).

2. Three hours of lab times, to support your project work: **Tuesday 10.00-10:50 and 13.00-13.50** (with Niall and Róisín) and **Wednesday 16.00–16.50** (Just Niall)

Module review

The topics we have covered (not necessarily in order) are:

(a) Basic I/O (`cin`, `cout`), and manipulators (`endl`, `setw`, ...);
   " input + output

(b) Flow of control and looping (`if`, `for`, `while`, etc.)
   if / else

(c) Fundamental data-types (`int`, `float`, `double`, `char`, `bool`, ...). Arrays.

(d) `string`, and C-style strings.
   # include <string>   —— arrays of char.

(e) Computer representation of numbers (underflow, overflow, machine epsilon, ...)   (Eg, largest / smallest Numbers ...)

(f) Bit-wise operators —— Eg   & , | (not && , ||).

(g) Dynamic memory allocation, including of multidimensional arrays
   \ new + delete

Module review

(h) Functions: default parameter values, overloading, functions as arguments to other functions, recursion, pass by reference/value.

(later : const, static).

(i) Classes, including the `private`, `public` and `friend` access specifiers.

(j) Classes: constructors and destructors, including copy constructors.

(k) `template`s.

*Eg Extend the definition of
+, *, /, etc, for objects.*

(l) Function and operator overloading (syntax, precedence, implicit/explicit arguments, the `this` pointer, unary/binary operators, assignment operators, ...).

 # define   #ifdef   #ifndef

(m) The C++ preprocessor

 # include

(n) The Standard Template Library (STL), including containers (especially `set`, `multiset` and `vector`), iterators, algorithms (but not functors); range-based loops.

Module review

(o) Optimisation and bisection;

(p) Sorting algorithms, and their complexities;

(q) Stacks, and their applications; — Eg, Solving Maze.

(r) Solving linear systems by the Jacobi and Gauss-Seidel methods;

(s) Solving diagonal and triangular systems (back-substitution);

(t) Sparse matrix representation, especially triplet and CCS; Matrix-vector multiplication

(u) Representation of graphs as adjacency matrices;

(v) PageRank and the Power Method;

Module review

(w) Reading to and writing from files; ~~Comma Separated Values (CSV) files;~~

(x) Static variables, and class variables.

(y) `const`

(z) Inheritance (but not `virtual`).

I hope you have enjoyed CS319, and have learned something: I have!

Thank you for your participation, your forbearance, your willing to help each other, your enthusiasm, and your talent: it has been a pleasure to teach this course.