SIM UOW CSCI251

ASSIGNMENT 2

Last updated: 29 January 2024

Timothy Mah

10258663

Lecture Group L01

Contents

[Introduction 2](#_Toc157412886)

[Usage 2](#_Toc157412887)

[Program flow 2](#_Toc157412888)

[Extra features 3](#_Toc157412889)

[Dynamic border character and grid spacing 4](#_Toc157412890)

[Input validation 4](#_Toc157412891)

[Main menu 4](#_Toc157412892)

[File not found 4](#_Toc157412893)

[Config file must be processed 4](#_Toc157412894)

[Config file ignores white spaces outside of the 5 lines 4](#_Toc157412895)

[Data visualisation 4](#_Toc157412896)

[Limitations 5](#_Toc157412897)

[Learnings 5](#_Toc157412898)

[How to update elements in vector with range type for loop 5](#_Toc157412899)

[Some functions want C style string, not regular string 6](#_Toc157412900)

[Map is just like a python dictionary with key:value 6](#_Toc157412901)

[Unordered\_map is map but unordered and slower 6](#_Toc157412902)

[Shortcut class constructor syntax 6](#_Toc157412903)

[Switch-case vs if-else 7](#_Toc157412904)

[Regex worked on Windows but not assignment Ubuntu 7](#_Toc157412905)

[Find() 7](#_Toc157412906)

[Thoughts and reflections 7](#_Toc157412907)

[Am I alone? 7](#_Toc157412908)

[Impressive teacher 8](#_Toc157412909)

[Assignment difficulty 8](#_Toc157412910)

[C++ is like C but with dynamic memory allocation 8](#_Toc157412911)

# Introduction

Hello this is my documentation for SIM UOW CSCI251 ASSIGNMENT 1.

# Usage

The final program file can only be run in Ubuntu because of the makefile. However, the main file can be compiled to support Windows by compiling as .exe.

To run,

./ csci251\_a1.app

To compile,

make

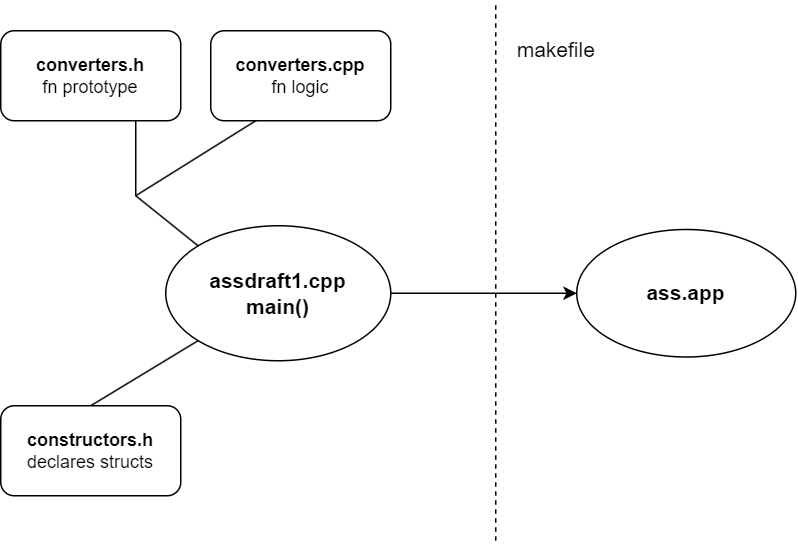
In same directory as makefile. Makefile should be in same directory as main file and the other helper files.

# Program flow

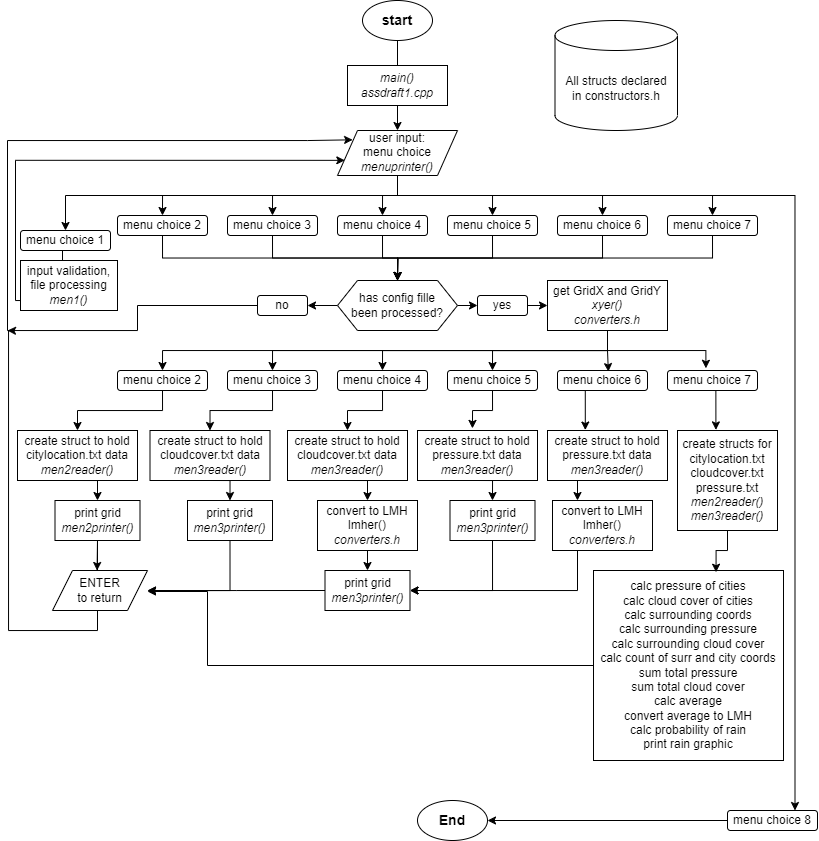
This is my program structure. assdraft.cpp is the main program file while converters.h and converters.cpp declare functions that transform values. All structs are declared in constructors.h.

**In the following diagrams, csci251\_a1.cpp is renamed to assdraft1.cpp. csci251\_a1.app is renamed to ass.app**

*I created these diagrams before renaming my files to the assignment’s required naming convention*



This is my program flowchart



There are primarily 5 functions in main(), and the rest are either my converter functions that are all declared in converters.h, or other miscellaneous functions such as menuprinter(). As mentioned earlier, all structs are declared in constructors.h. The 5 primary functions are, men1(), men2reader(), men2printer(), men3reader(), and men3printer(). Initially I had wanted to create an individual function for each menu option however I realised that many of the same logic and code is actually reused for options 2 to 6. I could have reused men2reader() and men2printer() for options 3 to 6 as well however men2reader() returns a different struct than what’s needed for options 3 to 6 so I had to create similar functions but with different return type.

I suppose I could have use the same struct for CityData (stores city name, city ID, and X Y) and CloudData(stores pressure or cloud cover and X Y). But then the struct that holds pressure and cloud cover data would have to hold city name and city ID. This method would mean that I have less functions created but structs take up more memory. I could have tested both styles to see which one performed better but this assignment program is so small that any difference would have been miniscule and may not even be noticed significantly. Regardless, my current code works.

# Extra features

## Input validation

## Data visualisation

# Limitations

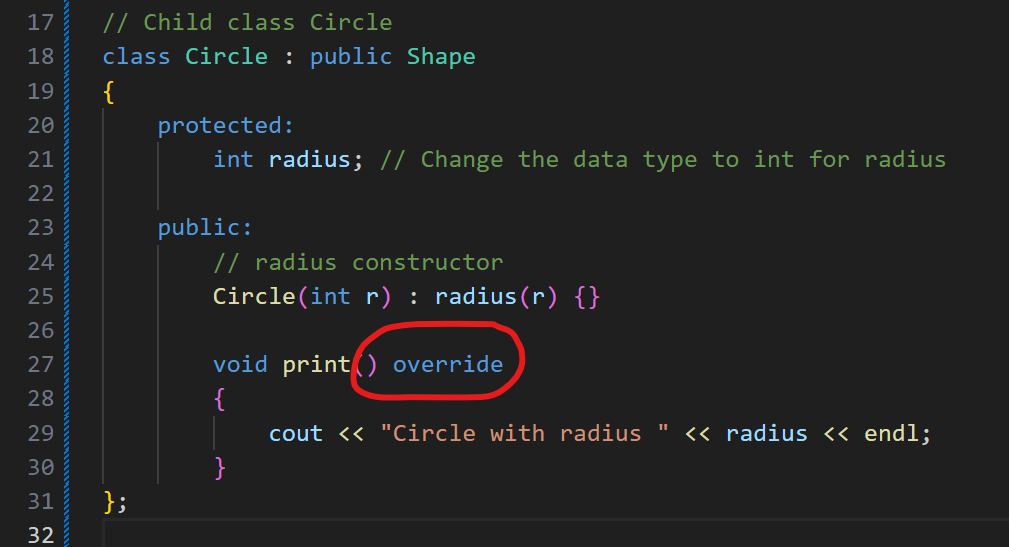
1. Regex string may be system agnostic. However, in today’s context where many programs run over the web, server side processing would mitigate this issue.

# Learnings

Some cool stuff I learnt

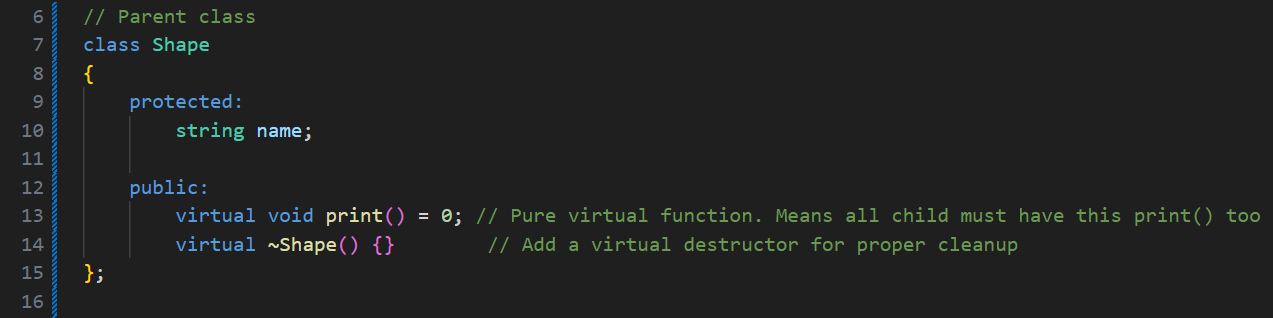
## Override

I understood from the lectures what is function overriding. Essentially function call of same name can invoke different behaviour based on the object type. However, I don’t remember this Override syntax being used. Maybe it’s a new C++ feature. Regardless, it’s something I had learnt while doing this assignment. This override keyword explicit signals that the specified function is being overridden. The code would work as per normal even if the override keyword is not used. It’s more for ease of understanding.



## Virtual functions

I think I did not understand fully in class but after writing out the code and taking reference from online sources, I have a better understanding of what virtual functions are. In the following context, basically print() in parent class shape does nothing, but declaring this virtual function means that all children classes must have a print() function. This also enables polymorphism because if there’s no print() in parent class Shape, then shape -> print() would not work.



## Abstract functions

Abstract function means that I cannot create an instance of it. In my context it means I cannot create an object of ShapeTwoD type directly.

# Thoughts and reflections

## Purpose?

What is the purpose of function overriding and polymorphism? Perhaps for optimising games code for more fps? This feature of OOP seems so specific and niche I can’t foresee myself using this feature often unless I have a specific problem that requires it.