

Computer Science 35IB

Assignment(s) For February 11, 2002

Section J of the dossier is due at the beginning of class.

Finish all source code and internal documentation.

Ensure that all functions are separated into separate documentation (*.h) and implementation (*.cpp) files.

Ensure that the documentation of all functions includes the function's purpose, preconditions and postconditions.

Ensure that the beginning of **every file** of source code contains an identification header similar to the following.

```
/*=====
File:      artdriv.cpp
Purpose:   Demonstrates the construction of a menu that calls functions.
Author:    Ayebee Wannabe
School:    Sir Winston Churchill High School, Calgary, Alberta, Canada
Teacher:   Mr. Gerry Donaldson
Language:  Dev-C++ 4.01 using the MinGW (Minimalist GNU for Windows) compiler
Target:    Real Mode Application (DOS)
System:    AMD Athlon, 1100 MHz Running Under Windows 2000

This Revision:  June 7, 2000
=====*/
```

All modules (C++ functions) must be "small and clearly structured" (Syllabus 2000, p. 52)

All identifier names must be "meaningful". (Syllabus 2000, p. 52)

There must be a consistent indentations scheme. (Syllabus 2000, p. 52) This includes:

indenting about 2-3 spaces within blocks of code,

blank lines or a commented pattern between significant blocks of code, such as the ending of one function and the beginning of another, before the beginning and after the end of a loop, before the beginning and after the end of a switch statement, et cetera.

Use syntax highlighting. (Suggested but not required in Syllabus 2000, p. 52)

Use an IDE that numbers all lines. (Suggested but not required in Syllabus 2000, p. 52)

Print your source code. Both Borland C++ 5.02 and Dev-C++ 4.01 support the following requirements.

Set the font size at 10 points using a non-proportional font such as Courier New.

Insert header(s) and/or footer(s) that incorporate the file names, dates and page numbers.

Ensure that the source code is printed in software that automatically numbers the lines.

Set font properties to emphasize different parts of syntax.

Set printing to wrap lines rather than truncate them, but that is the second best option.

You should *manually* structure your code so that lines do not exceed the right margin.

A section entitled **Mastery Techniques** is due at the beginning of class. Do this in a wordprocessor.

Start each technique at the top of a new, separate page.

Use a subheading identifying the technique at the top of a page where a new technique begins.

Leave a blank line and then reference the name of the file, page and line numbers in the source code from which the technique source code will be copied.

Copy the actual source code illustrating the technique in your program listing from your text file editor and paste it onto the corresponding page in your wordprocessor.

Create a footer and use the automatic page number function of your wordprocessor to automatically number your pages in your footer. In WordPerfect 10, click on Format --> Page --> Insert Page Number...

Place this section as your second section (just after *Preface* and before *A: Analyzing the Problem*)