

Butler to the Dark Lord (B2DL)

Professor Caleb Fowler

June 7, 2020

Problem.

This problem is taken from a "find your path" book of the same name. The idea is you read a passage of text and then decide which of the presented options you will choose next. If you make good decisions, you will win (or in this case, survive). We can represent part of this book with a flowchart, see figure 1. Paragraphs are indicated with "A" + a number. These also appear at the end of each passage and direct the reader where to read next. You are going to modify that with a menu option. The reader will select a number and program execution will jump to that location. You will also notice words in ALLCAPS. These are boolean variables reflecting your status as you progress through the story. You will need to keep track of these.

Requirements.

- The text for each function is listed below. Feel free to copy and paste this text. Display the text, then display the possible choices.
- Re-prompt if incorrect data was entered. Keep re-prompting until the client gets it right.
- Don't forget to incorporate these elements from the style guide:
 - Comments
 - Source file header.
 - Program Greeting - this is NOT the same as the menu!
 - Constant variable style if applicable.
 - No global variables or global variable look a-likes.

Specification Bundles.

Code elements from the specification bundles to control the maximum potential grade you can get for this assignment. The more work you do, the better grade you can get. This is the starting point for your grade.

"C" Specification Bundle.

- ☐ // Specification C1 - A function
Use a function somewhere in your program. Don't forget the function prototype.
- ☐ // Specification C2 - Simple menu
Use a simple menu to call the next function. Just add this comment to the first one; you don't need to comment all of them. This

You must meet the specifications of the lowest group before I will count the specifications for the highest group. For example, you must meet the "D" specifications before I will count the "C" specifications. If you miss one element of a specification bundle, that is the grade you will get for the assignment - regardless of how much extra work you do. Note: sometimes you will get comments stacking on top of each other - that's OK even though you would never do that in the real world.

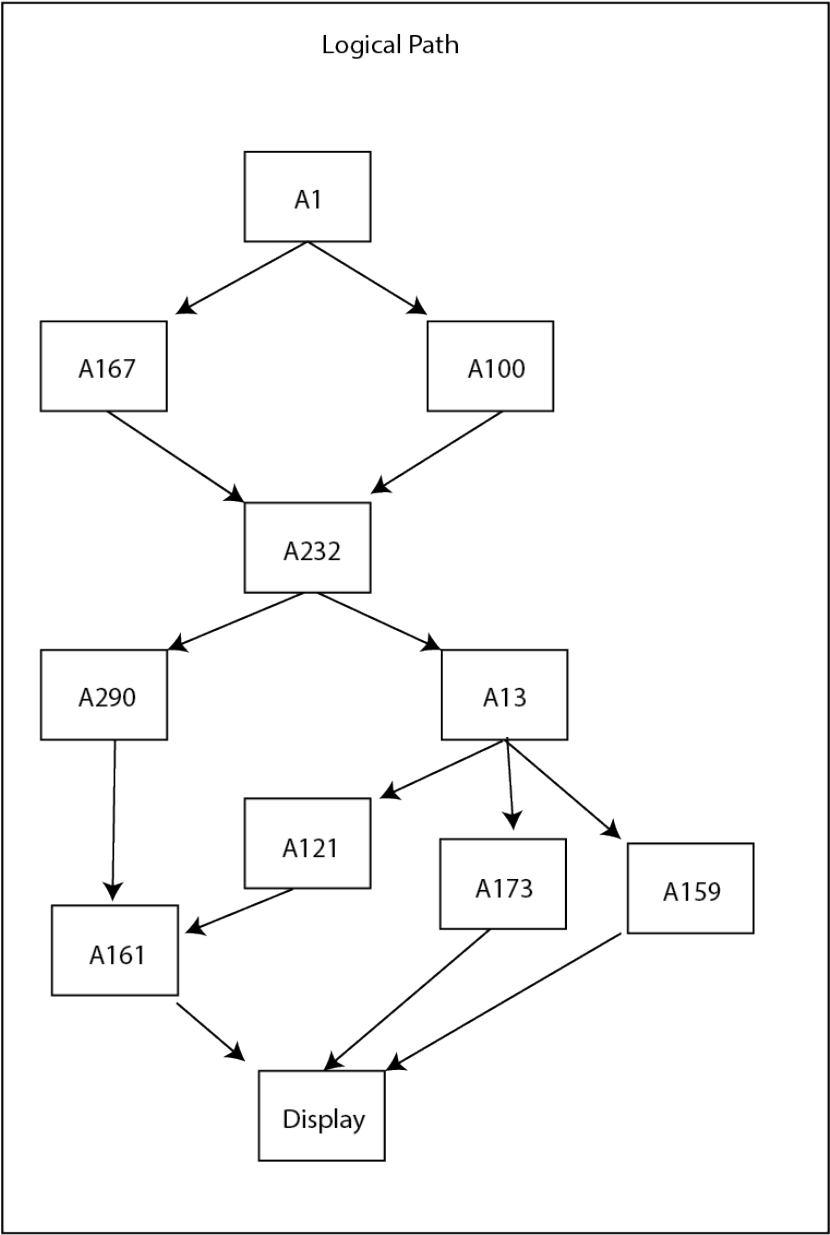


Figure 1: Partial path through B2DL.
The book is much bigger.

can be numeric.

□ // Specification C3 - Room Header

Each time you display a decision result, display it's room number and the decision you came from at the top, before you show the text of the new decision.

"B" Specification Bundle.

□ // Specification B1 - Booleans

When done, display the current state of all the BOOLEAN variables.

□ // Specification B2 - Individual Functions

Enclose each text passage (A1, A167, etc) is within it's own function. Call the functions to progress through the story. You don't need to comment all the functions with this specification comment, just the first one. Hint: Call the functions by the passage identifier - makes your code easier to read.

□ // Specification B3 - Valid choices only

In A13, there is one option which cannot be taken unless a boolean variable is set. If the variable is set, only offer this option, suppress the other 2.

"A" Specification Bundle.

□ // Specification A1 - Input Validation

Use input validation to allow only the correct menu values. Reprompt until proper input is entered.

□ // Specification A2 - Path

Display the path the client took through the story (use the A1, A167, etc. not the menu options) when the story is completed.

□ // Specification A3 - Game loop

Wrap the program in a loop which prompts you to go again when you finish.

Paragraph Descriptions.

A1.

Well, Mr Artanon, look at you. Who would have thought that the son of a lowly spider farmer could rise to such esteemed heights? Yet here you are, standing in the kitchens of Cayfax Castle, dressed in the finest livery onyx can buy. For you, sir, are butler to Malacandros, Dark Lord of all Nocturnia. The flames of fireplaces tickle suspended pots, which thicken the air with competing scents. Workers along stone benches busily chop and kill ingredients - nothing but the best for his lordship, of course. In the centre of the chamber a great oven blazes, its cracked chimney oozing smoke. 'Ah, Mr Artanon,' simpers Repugnis, the head cook. He's a fat goblin who samples too

many of his own creations, as evidenced by his bulging belly and the delta of stains down his apron. 'I'm roasting his Horrendousness a prime briar beast basted in snake oil. Mmm-mmm! The intestines are stuffed with rotten apple and breadcrumb, and I've shoved a carrot up its arse.' You arch an eyebrow, for Mal is already situated upstairs in the dining hall. With midday waning, he will be growing hungry. 'And just how long,' you ask, 'do you intend to keep his lordship staring at an empty plate?' Repugnis titters nervously, and glances towards the oven. 'Good sir, a roast takes as long as a roast takes. If I stoke the flames any hotter the skin will crisp to charcoal. At best I could aim for a rare cut?'

What is your reply?

1. 'It's you who'll get a rare cut across that corpulent neck of yours if you don't speed things along, you feckless blob of festering lard.'

(turn to A100)

2. 'Surely in such an industrious kitchen, there must be a starter or two ready to send along and sate His Abhorentness's appetite?' (turn to A167)

Source for all of the text paragraphs: Bowring, Sam. *Butler to the Dark Lord: A Grim Choices Gamebook* (Kindle Locations 36-49). Sam Bowring. Kindle Edition.

A100

Repugnis turns a paler shade of green, and gulps. "I beg you, Mr Artanon, stay your hand. I'm sure that something must be ready." The cook glances around and spots a bench at which several trays are being assembled. "Ah!" he says. "How about some pickled rat heads and carpaccio of slug?" You scowl, long enough to make Repugnis quiver, then turn and snap at some lurking footmen. "Get those trays up to his lordship without delay!" The footmen spring into action, seizing the trays and heading for the door. "Thank you, sir," says Repugnis. "The roast will not be far behind, I promise." "See that it isn't," you reply, "or the next item on the menu might be goblin stew." It seems you are an INTIMIDATOR.

1. Onwards, Mr Artanon. (turn to A232)

A167

Repugnis nods eagerly, which ripples his many chins. 'Yes sir, of course! Let me see now...' He glances around and spots a bench where several platters are being assembled. 'Ah! How about some pickled rat heads and delectable carpaccio of slug?' 'Very good,' you say. 'I am told that Captain Haiken is particularly partial to rat. But see that you hurry along now, Repugnis. I don't need to remind you that his lordship's patience is virtually non-existent.' 'Of course, sir, of course.' You click your fingers at some waiting footmen and they

leap to action, seizing the platters and heading for the door. It seems you are a DIPLOMAT.

1. Onwards, Mr Artanon. (turn to A232)

A232

You overtake the footmen along a corridor lit by stuttering candles, and lead them through a wide archway into the airy dining hall. Great windows stand open along one wall, looking out upon a pleasant view of circling lizard-birds cannibalizing their young. Down the length of the hall runs an immense marble table, at the head of which the Dark Lord Malacandros reclines in a chair of elaborately carved dragon bone. He is a gaunt man with darting eyes and sharp features, currently dressed in a very tasteful grey robe (which you helped him get into this morning), his black hair clasped in a ponytail down his back. To his right sits Captain Haiken, one of his favored brutes, a well muscled orc in polished steel. "Ah, Mr Artanon," says Mal, "I was beginning to worry we'd been forgotten." He gives you a thin smile, which warns you he is very close to becoming irked. You bow deeply. "My humble apologies, oh Intolerant One. Repugnis is putting the final touches on today's main course, and you know what a perfectionist he is. In the meantime," you whisk a platter from one of the footmen, "here are some starters to whet your appetites." As you move toward the head of the table, Haiken licks his lips. "Rat heads!" he exclaims. "My favorite."

Who will you serve first?

1. The enthusiastic Captain Haiken? (turn to A13)
2. Or your master, Malacandros? (turn to A290)

A13

Feeling dangerous, Mr Artanon? You move past Mal to Captain Haiken and, using a pair of tongs fashioned to look like crow's claws, start placing rat heads onto his plate. As you do, you glance surreptitiously at Mal, wondering if you have irritated him by serving his guest first - he is not exactly known for his good manners - but he does not seem to have noticed. At any rate, your first footman, Kalag - a young man who shows much promise - hops to action, to lay slivers of slug meat on Mal's plate. "Er..." says Haiken, his meaty hand skimming the tableware. You realize he is uncertain about what, amongst the array of cutlery laid out on either side of his plate, is the correct implement to use for rat heads.

1. Are you a DIPLOMAT? (turn to A212)
2. If not, do you simply inform him which is the proper rat-head-eating-fork? (turn to A159)
3. Or do you let him flounder? (turn to A173)

A290

As is only fitting, you serve your master first. As you place the largest and juiciest rat heads before him, your first footman Kalag - a young fellow who shows much promise - moves to Haiken and starts laying slug meat on his plate. "Er," says Haiken, "not too much slug for me, thank you - I have an allergy." Mal, who is in mid-chew, scoffs, and shoots a shard of half-chewed rat skull out of his nostril to ping off a wine glass. "Rut me, Haiken," he says. "I've seen you rip the throat from a charging horse, yet you can't stomach a little slug?" Haiken seems embarrassed. "They give me hives, my lord." "Ha! Well, better get over there with some of his precious rat heads, Mr Artanon." You do as you are bid, and place the remaining rat heads on Haiken's plate. They are a little on the smaller side, and you notice a wistful look in Haiken's eye as Mal lifts another plump one to his mouth. With your lords served, you move back from the table to remain in waiting.

1. Excellent posture, Mr Artanon. (turn to A161)

A159

You clear your throat and speak softly. "That little one there is your rat-head-eating-fork, my lord." Despite your attempt to be subtle, Mal registers the exchange and chortles. Haiken, embarrassed to be found ignorant, frowns. "I know what fork to use, you stupid butler! Begone before I stab you in the eye with it." With your lords served, you move back from the table to remain in waiting. Somehow you doubt Haiken's bedclothes will be properly turned.

1. Done

A161

You move away from Haiken, whose hand still hovers uncertainly over the cutlery, to serve Mal rat heads. Finally Haiken makes a decision and picks up a carapace-cracking-spork. Mal chuckles at his ignorance. "By Stygias, Captain - have you been in the field so long that you've forgotten the basic functions of tableware?" Haiken grimaces. "Pardon me, your Pettiness. I may have had a noble birth, yet life as a soldier puts me out of touch with certain niceties. We tend to eat with our hands." Mal laughs. "Well, let us not stand on ceremony, Captain! I am happy to do it your way!" Mal picks up a rat head with his hand and shovels it into his mouth. Haiken happily follows suit, munching and cracking skulls with his teeth. Inwardly, you give a sigh. No doubt with such messy fingers, Mal will stain the dragon bone. Why, you have often wondered, would anyone fashion a dining chair out of such easily blemished material as bleached porous bone? Just as you fear, Mal leans back to rest his grubby mitts upon the armrests. The CHAIR IS DIRTY. You move back to the wall while your lords eat, to remain in waiting. A butler's work is never

1. Done

A173

As they gnash away, Mal and Haiken discuss the Captain's recent raid on the town of Greenflower. "Honestly," says Haiken, "those wretched Brightlanders are so easily slaughtered, sometimes I almost feel bad for them." Mal laughs at this hilarious joke. "Ha! That's a good one, Captain. But tell me, what of your primary objective?" "I am pleased to report success, my lord." "You have brought me a virgin?" "She languishes in your dungeon even now - lovely, virtuous and ripe for sacrifice." "Excellent. My skywatchers predict a full moon in three nights time - perfect conditions for the ceremony." Footsteps sound at the dining hall entrance, and a goblin messenger, whom you recognize as Peedy, enters wearing a feathered hat and carrying a scroll. Mal's gaze slides to him. "What is it?" Peedy clears his throat. "Apologies for the interruption, Your Atrociousness. I bring you word from Vorganis Castle." "Oh?" Mal's tone is so instantly cold that the very temperature of the room seems to drop. "Er... yes, my lord. Lord Nefarian accepts your kind invitation to attend the Stygias Ceremony, and intends to arrive tomorrow morning." Mal's eyes narrow. You know he does not care for his cousin Nefarian, not least because the man is next in line to the throne. Nefarian's presence at the ceremony, however, must be tolerated, for he is a powerful lord in his own right, with many allies amongst Nocturnia's nobility - a fact which Mal does not enjoy being reminded of. "Why bring me such news at lunch?" he growls. "It has put a foul taste in my mouth and sullied my food." He drums his fingers on the dragon bone armrest, and fiery sparks flare at his fingertips. Peedy quails, for he knows that Mal is prone to taking out his anger on the bearers of bad news. It would be a shame, however (and a staffing issue), to lose a competent messenger to a moment of ill temper over something that really isn't his fault.

1. Do you attempt to intercede on Peedy's behalf? (Done)
2. Or abandon him to his miserable fate? (Done)

In either case, both paths lead to the end of the homework, too.

A212

You clear your throat very slightly, and Haiken glances up at you. As you move your hand back from his plate, you let your finger rest momentarily on the appropriate rat-head-eating fork. Haiken understands, and gives you a relieved look. Mal is too busy shoveling slug meat into his gullet to notice the exchange, so HAIKEN IS GRATEFUL. You add some rat heads to Mal's plate and move back from the table to take a place in waiting.

1. Nicely done, Mr Artanon. (turn to A161)

Display

Just indicate the game is finished.

Homework Checklist.

Check the following before you turn in your work:

- ☐ You coded your homework.
- ☐ Does it meet all the requirements?
- ☐ Test your code.
 - ☐ Does it compile?
 - ☐ Does it have any compiler warnings?
 - ☐ Does it run?
 - ☐ Does it produce correct output?
 - ☐ Did you use the grep trick to make sure I can see your work?
- ☐ Upload to Canvas.
- ☐ What's the plagiarism checker score?

Discussion.

The vast amount of text poses a problem for your coding - it hides your program and makes it hard for you see the whole thing at once. The solution is to code the framework of everything and leave the text out. This makes your code much smaller and makes it easier to get it working. Once you get it working the way you want, you can add the text and expand your source file.

Due Date.

This assignment is due by 11:59 PM on **Sunday** on the date on the calendar in Canvas. All the assignments are open the first day of class and you can begin working on them immediately. I encourage you to start sooner rather than later with your homework, these always seem to take longer than you think.

Late Work.

If you miss the due date and time specified above, your work is late. Canvas records the date and time your homework upload COMPLETES. Late work is still acceptable, but it suffers a 1 letter grade penalty. You may turn late work in up until MONDAY 11:59 PM AFTER THE ASSIGNMENT WAS DUE. That is, you have 1 day to turn your work in - after that the Canvas drop box closes. Once Canvas closes I will not accept an assignment.

Pro-Tip: Get a bare bones copy of your code running and turn it in. Then go ahead and modify it, fix it and whatnot. Upload it with

the same name when you finish. That way, if something unexpected happens, you have some working code turned in. Risk management, class, risk management.¹

¹ If you really want to go pro, get some sort of version control system running (like Git).

How to Turn in your Homework.

I ONLY accept homework through the Canvas Dropbox. Do not add it to the submission comments or email it to me - I will not accept it. Turn homework in by uploading to the appropriate Canvas Dropbox folder. Save your homework as a .cpp file. Don't zip or otherwise compress your files. Do NOT split your file up into multiple files. I know that is a standard industry practice, but it just gets in the way for this class.

Create a file with the following naming format: W12345678.cpp (your w number). This allows me to sort the class in alphabetical order - don't stand out here! If you are having trouble submitting the assignment, email me immediately. Don't change your filename if you make multiple submissions - Canvas will keep track of them and download the latest one by default.

Style Guide.

All programs you write MUST have the following code and/or comments. Again, I look for these elements with my scripts, you want me to find them.

Comments.

Use white space and comments to make your code more readable. I run a program called cloc (count lines of code) which actually looks for this stuff.

End of line comments are only permitted with variable declarations. Full line comments are used everywhere else.

Specification Comments.

Specifications are bundled into groups: "A", "B", "C". You must meet the specifications of the lowest group before I will count the specifications for the highest group. For example, you must meet the "B" specifications before I will count the "A" specifications. If you miss one element of a specification bundle, that is the grade you will get for the assignment - regardless of how much extra work you do.

Use whole line comments for Specifications. Put the comment on the line above the start of the code implementing the Specification. If the same Specification code appears in more than 1 place, only comment the first place that Specification code appears. Number your Specifications according to the specification bundle and the specific specification you are using, also provide a very short description. DO

NOT BUNCH ALL YOUR SPECIFICATIONS AT THE TOP OF THE SOURCE FILE. Example specification comment:

```
// Specification A2 - Display variables
Your code to do this starts here;
```

It's very important to get the specifications down correctly. **If your specification code isn't commented, it doesn't count.** I use the grep trick to find your specification code. Proper documentation is part of the solution, just like actually coding the solution is.

Compiler Warnings.

Compiler warnings are a potential problem. They are not tolerated in the production environment. In CISP 360 you can have them. I will deduct a small number of points. CISP 400 - I will deduct lots of points if compiler warnings appear. Make sure you compile with -Wall option. This is how you spot them.

C++ Libraries.

We are coding in C++, not C. Therefore, you must use the C++ libraries. The only time you can use the C libraries is if they haven't been ported to C++ (very, very rare).

Non-Standard Language Extensions.

Some compilers support unapproved extensions to the C++ syntax. These extensions are **unacceptable**. Unsupported extensions are compiler specific and non-portable. Do not use them in your programs.

Program Greeting.

Display a program greeting as soon as the program runs. This is a simple description of what the program does. Example:

```
// Program Greeting
cout << "Simple program description text here." << endl;
```

Source File Header.

Start your source file with a program header. This includes the program name, your name, date and this class. I use the grep trick for .cpp (see below) to look for this. I focus on that homework name and display the next 3 lines. Example:

```
// drake.cpp
// Pat Jones, CISP 413
// 12/34/56
```

Specifications and Specification Bundles.

You document specifications like this: // Specification C1 - Some stuff

You do not need to code them in order. You will probably want to because the specifications get harder as you move up in bundles (not THAT much harder). You also don't need to worry about the specification comments appearing in order in your code, either.

However, all of a specification bundle must be coded to reach that bundle grade (ie all C bundle to get a C). Partially completed bundles DO NOT COUNT. Say you code all specifications for a B bundle and only 1 for an A bundle (out of 5 for example). The highest grade you would get would be a B because that's the last bundle you've completed.

You can stop at any bundle you want, you just can't get a higher grade (ex, you code all specifications for bundle B - the best you can get for this homework is a B). This is designed to mirror the work word, the more features your code has, usually, the happier your clients are. This also gives you some control over your grade.

This style guide has more information on the specifics of these comments.

Variables.

Constant variables - anytime you have a value which is not supposed to change, that's a constant. We make it read only with the `const` keyword and signify it with the ALL CAPS style: `const PI = 3.14;` We prefer using constants because they make the code easier to read. There are a few situations where we do not usually use them, such as starting a loop at zero. However, if we have that loop end at, say, 33, then it's a magic number. What's 33? Who knows? If we use `const SIZE = 33;` we know what 33 is.

When we have numeric literals appearing in the program we call these magic numbers. We don't know what they are, but if we change them, the program breaks. hence, magic. Magic numbers are generally frowned upon.

Crep Trick.

Always run your code immediately before your turn it in. I can't tell you how many times students make 'one small change' and turn in broken code. It kills me whenever I see this. Don't kill me.

You can check to see if I will find your specification and feature comments by executing the following command from the command line. If you see your comments on the terminal, then I will see them. If not, I will NOT see them and you will NOT get credit for them. The following will check to see if you have commented your specifications:

```
grep -i 'specification' homework.cpp
```

This will generate the following output. Notice the specifications are numbered to match the specification number in the assignment. This is what I would expect to see for a 'C' Drake assignment. Note the `cd Desktop` changes the file location to the desktop - which is where the source file is located.

```
calebfowler@ubuntu:~$ cd Desktop
calebfowler@ubuntu:~/Desktop$ grep -i 'specification' cDrake.cpp
// Specification C2 - Declare Variables
// Specification C3 - Separate calculation
// Specification C1 - Program Output
calebfowler@ubuntu:~/Desktop$
```

This is what I would expect to see for an 'A' level Drake assignment.

```
calebfowler@ubuntu:~/Desktop$ grep -i 'specification' aDrake.cpp
{ // Specification C2 - Declare Variables
  // Specification C3 - Separate calculation
  // Specification B1 - Calculation
  // Specification C1 - Program Output
  // Specification B 2 - double and half
  // Specification A1 - Output Headers
  // Specification A2 - Display variables
calebfowler@ubuntu:~/Desktop$
```

We can also look at the line(s) after the `grep` statement. I do this to pay attention to code segments.

```
grep -i -C 1 'specification' aDrake.cpp
```

```
calebfowler@ubuntu:~/Desktop$ grep -i -C 1 'specification' aDrake.cpp
int main()
{ // Specification C2 - Declare Variables
  int r_starcreation = 7; // rate of star creation
  --
  // Specification C3 - Separate calculation
  float drake = 0; // initialize to 0
  // Specification B1 - Calculation
  drake = r_starcreation * perc_starswithplanets * ave_numberofplanetlife *
perc_devlife * perc_devintlfe * perc_comm * exp_lifetime;
  // Specification C1 - Program Output
  cout << "The estimated number of potential alien civilizations in the univer
se is ";
  --
  // Specification B 2 - double and half
  cout << "Half this value: " << drake * .5 << endl;
  --
  // Specification A1 - Output Headers
  cout << endl;
  --
  // Specification A2 - Display variables
  cout << "Variables:" << endl;
calebfowler@ubuntu:~/Desktop$
```

We can also use this to look for other sections of your code. The `grep` command searches for anything withing the single quotes "", and the `-i` option makes it case insensitive. This is how I will look for your program greeting:

```

calebfowler@ubuntu:~/Desktop$ grep -i -C 1 'greeting' aDrake.cpp

// Program Greeting
cout << "This program calculates and displays the number of potential";
calebfowler@ubuntu:~/Desktop$

```

The grep trick is extremely powerful. Use it often, especially right before you turn in your code. This is the best way I can think of for you to be sure you met all the requirements of the assignment.

Client System.

Your code must compile and run on the client's system. That will be Ubuntu Desktop Linux, version 18.04. Remember, sourcefile.cpp is YOUR program's name. I will type the following command to compile your code:

```
g++ -std=c++14 -g -Wall sourcefile.cpp
```

If you do not follow this standard it is likely I will detect errors you miss - and grade accordingly. If you choose to develop on another system there is a high likelihood your program will **fail to compile**. You have been warned.

Using the Work of Others.

This is an individual assignment, you may use the Internet and your text to research it, but I expect you to work alone. You **may** discuss code and the assignment. Copying code from someone else and turning it in as your own is plagiarism. I also consider isomorphic homework to be plagiarism. You are ultimately responsible for your homework, regardless of who may have helped you on it.

Canvas has a built in plagiarism detector. You should strive to generate a green color box. If you submit it and the score is too high, delete it, change your code and resubmit. You are still subject to the due date, however. This does not apply if I have already graded your homework.

Often, you will not be able to change the code to lower the score. In this case, include as a comment with your homework, what you did and why you thought it was ineffective in lowering your score. This shows me something very important - you are paying attention to what you are doing and you are mindful of your plagiarism score.

ProTip: Get a bare bones copy of your code running and turn it in. Then go ahead and modify it with bonuses and whatnot. Upload it with the same name so it replaces your previous homework. This way, if something comes up or you can't finish your homework for some reason, you still have something turned in. A "C" is better than a zero. Risk management class, risk management.