Sample Exam Week 06
CSE 232 (Introduction to Programming II)


VERSION A

Full Name: ....

Student Number: ....

Instructions:
- DO NOT START/OPEN THE EXAM UNTIL TOLD TO DO SO.
- You may however write and bubble in your name, student number and exam **VERSION/FORM NUMBER** (with a #2 pencil) on the front of the printed exam and bubble sheet prior to the exam start. This exam is Version A. Your section doesn't matter and can be ignored.
- Present your MSU ID (or other photo ID) when returning your bubble sheet and printed exam.
- Only choose one option for each question. Please mark the chosen option in both this printed exam and the bubble sheet.
- Assume any needed `#includes` and `using std::...;` namespace declarations are performed for the code samples.
- Every question is worth the same amount of points. There are 55 questions, but you only need 50 questions correct for a perfect score.
- No electronics are allowed to be used or worn during the exam. This means smart-watches, phones and headphones need to be placed away in your bag.
- The exam is open note, meaning that any paper material (notes, slides, prior exams, assignments, books, etc.) are all allowed. Please place all such material on your desk prior to the start of the exam, (so you won't need to rummage in your bag during the exam).
- If you have any questions during the exam or finish the exam early, please raise your hand and a proctor will attend you.

http://xkcd.com/499/

---
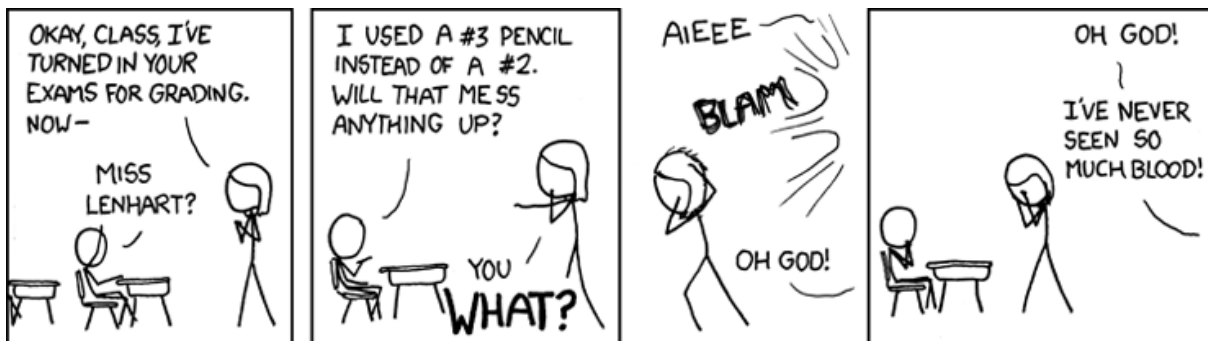
1. What happens if an exception is thrown in a program, but it is never caught?

   (a) It causes a compile time error, but only if a debugger is being used.
   (b) It causes a compile time error, but only if a debugger is NOT being used.
   (c) It causes a runtime error.
   (d) The main function will catch it and prompt the user for direction.
   (e) It generates undefined behavior.
   (f) None of the above.

2. What key can you press to auto-complete a partially typed filename at the command line?

   (a) Return
   (b) Control-A
   (c) Control-D
   (d) Control-Q
   (e) Tab
   (f) None of the above.

3. Which techniques are ways for a function to communicate information?

   (a) Changing a non-const reference argument
   (b) Writing to standard output
   (c) Assigning to a dereferenced pointer argument
   (d) Returning a value
   (e) Throwing an exception
   (f) All of the above

4. What happens if code in a `try` raises an exception that isn't the one a `catch` block specifies?

   (a) The `catch` block executes, but the same exception is raised at the end.
   (b) The `try` block ceases executing at the point of the raised exception.
   (c) The `try` block finishes, then the exception is raised again.
   (d) The `what` member function on the exception is called.
   (e) None of the above.

5. Why should data members generally be made private?

   (a) To prevent exceptions from being thrown in the class constructor.
   (b) To allow for separate compilation of multiple implementation files.
   (c) To improve the performance (speed) of the class.
   (d) To improve the readability of the class definition.
   (e) To protect the data members from being changed without validation by the class.

6. Why do we recommend static casting the result of the size method on containers?

   (a) Because implicit casting results in run-time exceptions.
   (b) Because working with unsigned ints can cause a number of common mistakes.
   (c) Because calling size on an empty container is undefined behavior.
   (d) Because the size of a container can change if elements are added or removed.
   (e) Because the type returned is compiler dependent.
   (f) None of the above.

7. What happens when an assertion fails?

   (a) The program is immediately halted.
   (b) An entry is added to the log (which can be accessed via `gdb` when the `-g` is used.
   (c) An exception is thrown that can be caught by a matching catch statement.
   (d) Assertions cant fail, so the question is nonsensical.
   (e) An error message is displayed and the opportunity is given to resume the program from that point.
   (f) A warning is presented at compile-time, provided `-Wall` is used.

8. If you expect a function your code will call may raise an exception, which keywords are needed to handle the exception?

   (a) `try`
   (b) `finally`
   (c) `except`
   (d) `throw`
   (e) `catch`
   (f) None of the above.
   (g) 2 of the above.
   (h) 3 of the above.
   (i) All of the above.

9. In order to use the assert function, you need to include the `cassert` header. What does the `c` in the name mean?

   (a) That the function runs at compile time.
   (b) That the header is from the C language.
   (c) That the function requires a compiler pre-processor.
   (d) None of the above.

10. When can you use a `catch` block without a *try* block?

    (a) When it is unclear where an exception can be thrown.
    (b) When you want the code in the block to always run.
    (c) When any exception can be caught.
    (d) When the statement immediately before it may raise an exception.
    (e) None of the above.

11. You are trying to rename a file, named "main.cpp", to be called "test.cpp". You type "`mv mai<TAB>`", that is you type the first few letters of the filename and press the TAB key. Nothing happens. What does this mean?

    (a) That the "test.cpp" file is interfering with the copy.
    (b) That there is no file in the working directory that starts with "mai".
    (c) That "main.cpp" is not part of the PATH.
    (d) Of course nothing happens, filenames can't contain whitespace.
    (e) That you need a "-r" to move read-only files.
    (f) That you misspelled the "mv" command.

12. When should a function throw an exception?

    (a) When the function attempts to access a private data member.
    (b) When the function needs a default parameter.
    (c) When the function catches an exception.
    (d) When the function encounters an error that the caller should handle.
    (e) When the function attempts to perform a complicated calculation.
    (f) When the function attempts to read from a stream.

13. Which of the following is true about assertions?

    (a) They are part of the `<assert>` library.
    (b) A true argument results in an error.
    (c) They should be used to indicate failures to the user (not the programmer).
    (d) They can generate compile-time errors.
    (e) They aren't used in modern C++ code.
    (f) They can generate run-time errors.
    (g) None of the above are true.

14. What key is needed to exit the `less` pager?

    (a) Escape
    (b) e
    (c) Control-C
    (d) Control-D
    (e) q
    (f) None of the above are correct.

15. `less` is a program that is used to display large amounts of text in the command line. What is the name for this type of program?

    (a) All of the above
    (b) a display
    (c) an editor
    (d) a pager
    (e) a viewer

This page intentionally left blank.