Week 00 Sample Exam

CSE 232 (Introduction to Programming II)

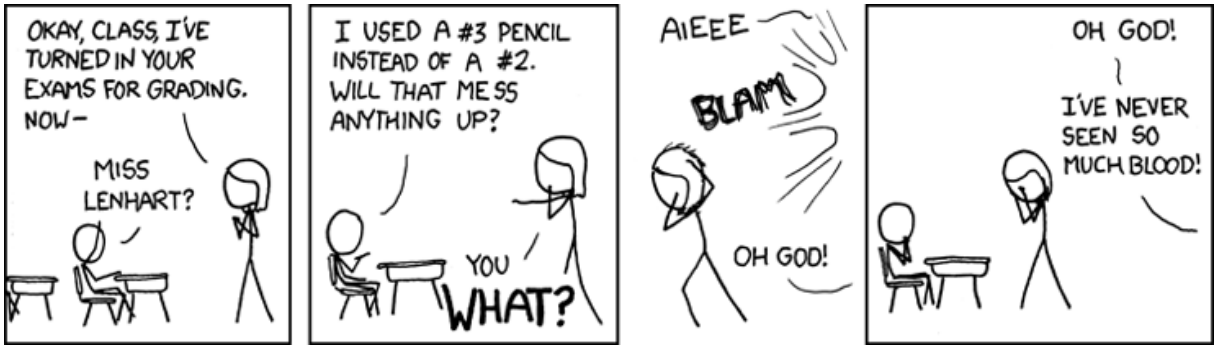
VERSION A

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 LETTER** (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 `#include`s 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). Please be sure to bring the required textbook!
- If you have any questions during the exam or when you finish the exam, please raise your hand and a proctor will attend you.



http://xkcd.com/499/ Date Accessed: August 20, 2024

1. If a student had the following scores, how many points from exams would contribute to their point total.

MCQ Exam 1	75
MCQ Exam 2	150
MCQ Exam 3	100
Coding Exam 1	60
Coding Exam 2	180
Coding Exam 3	30

   (a) 865
   (b) 505
   (c) 490
   (d) 330
   (e) 460
   (f) None of the above are the correct total

2. There is one source of extra credit in the class, how is it earned?

   (a) By having perfect attendance
   (b) By answering questions on Piazza
   (c) By requesting additional extra credit opportunities
   (d) By attending the honors section
   (e) By completing a project
   (f) All of the above

3. What distinguishes an interpreter from a compiler?

   (a) An interpreter only interprets the code, it doesn't run it
   (b) An interpreter can't use libraries
   (c) An interpreter can detect syntax errors
   (d) An interpreter can't work with multiple files
   (e) An interpreter directly executes the code
   (f) An interpreter can't manage overloaded functions
   (g) None of the above are true

4. What is CSE 232's policy with regard to using AI (e.g. GitHub Copilot, ChatGPT)?

   (a) They can be used, but only with attribution.
   (b) They can't be used on exams as they aren't allowed resources.
   (c) They can be used, but beware about using them in ways that deter learning.
   (d) All of the above are true.

5. On which of the following assessments are students **NOT** allowed to collaborate?

   (a) Lab assignments (asynchronous)
   (b) Exams
   (c) Homework assignments
   (d) Lab assignments (synchronous)
   (e) All of the above permit collaboration

6. The minimal C++ program in Section 1.2.1 has which of the following traits?

   (a) Variable declarations
   (b) Strings
   (c) Import directives
   (d) Include directives
   (e) Function definitions
   (f) Overloaded functions
   (g) It has none of the above

7. Given the following function declarations, which of the following functions are overloaded?

```
void Jump();
int Jump();
void After(double a);
void After(double);
int Nice(string);
int Nice(int);
string Bark(string);
string Bark();
```

   (a) `Jump` is overloaded
   (b) `After` is overloaded
   (c) `Nice` is overloaded
   (d) `Bark` is overloaded
   (e) (a) and (b) are both true
   (f) (c) and (d) are both true
   (g) All of the functions are overloaded
   (h) None of the functions are overloaded

8. Which of the following should always be cited if used as a source according to the syllabus?

   (a) The page on vectors on CppReference.com
   (b) Peers that you collaborated with
   (c) Section 1.2.1 from the required textbook
   (d) The lecture in Week 06
   (e) The example code from Week 03

9. When are parameter names NOT ignored by the compiler in function declarations?

   (a) When the function declaration is also a function definition
   (b) When the function declaration is ambiguous
   (c) When the function declaration has multiple parameters of the same type
   (d) When the function declaration is included after the main function
   (e) When the function declaration is overloaded

10. In the Preface of the required textbook, what does Stroustrup mean when he says the book isn't a layer cake?

    (a) He means that the book won't be focusing on how the language has evolved over time.
    (b) He means that his book isn't useful as a wedding dessert.
    (c) He means that each "slice" of the book will have many different concepts.
    (d) He means that the book can be read in many different ways.

11. What namespace contains `cout`? Hint: see Section 1.2.1.

    (a) `iostream`
    (b) `main`
    (c) `STL`
    (d) `C++`
    (e) `std`

12. Why was the function call, `print(0,0)`, in Section 1.3 ambiguous?

    (a) Because `(0,0)` could be interpreted as one number or two.
    (b) Because there were only declarations of single parameter overloads of `print`
    (c) Because `0` could represent an int or a double
    (d) Because `print` is a function provided in the standard library.
    (e) Because `print` hasn't been declared before the function call.

13. What is the return type for a function that doesn't return anything?

    (a) `null`
    (b) `void`
    (c) `int`
    (d) `None`
    (e) Nothing, the return type is omitted

14. What advice is given to reduce errors in Section 1.3?

    (a) Use short function names
    (b) Including all function declarations at the start of the file
    (c) Avoid overloading functions
    (d) Writing more functions and shorter functions
    (e) Write function declarations that are also function definitions

15. Why is it recommended to use the `-std=c++20` command line argument to `g++`?

    (a) To indicate to the compiler that C++20 style guides should be used.
    (b) To allow for C++20 features to be used in code.
    (c) To indicate that the code is C++, not C.
    (d) To allow for the fastest version of the compiler to be used.
    (e) To indicate to the compiler that C++20 warnings should be used.
    (f) To indicate that the STL should be made available to the code.
    (g) All of the above

16. What is the name of the C++ interpreter that we use in this course?

    (a) gdb
    (b) git
    (c) g++
    (d) gcc
    (e) None of the above

17. C++ was created to extend which prior programming language?

    (a) Bash
    (b) C+
    (c) Basic
    (d) Haskell
    (e) Fortran
    (f) Python
    (g) C
    (h) None of the above

18. In order for a function to be called, what must occur prior to that function call in the code?

    (a) The function's declaration
    (b) The function's description
    (c) The function's body
    (d) The function's definition
    (e) None of the above

19. What does `void` mean in `print_square`'s definition in Section 1.2.1 (page 3 of the required textbook)?

    (a) It means that the function doesn't return any value.
    (b) It means that the function's return type is the type named `void`.
    (c) It means that the function will empty any arguments that it is called with.
    (d) It means that the function's return type isn't specified.
    (e) None of the above

20. If you need a signed integer type of at least 64 bits in width, which of the following types should you use?

    (a) `int64_t`
    (b) `long double`
    (c) `long`
    (d) `int`
    (e) `double`

21. If a function is overloaded, it means that there is a another function with what quality?

    (a) The same return type
    (b) The same parameters
    (c) The same behavior
    (d) The same name
    (e) The same body
    (f) None of the above

22. A program that turns a source file into an executable program is called which of the following?

(a) A debugger
(b) An interpreter
(c) A compiler
(d) A converter
(e) A programmer
(f) None of the above