03-60-340-01
2013 Winter, Tues. Jan. 29, 2013 in BB 113
University of Windsor, School of Computer Science

Midterm 1 Examination

Mr. Paul Preney

Student ID:	
FIRST Name:	
LAST Name:	

"I have neither given nor received unauthorized help with this examination. Any suspicion of cheating will automatically void my mark on this examination."

____

Signature
Unsigned examination booklets will not be graded.
Signature implies agreement with the above statement in quotes.

INSTRUCTIONS

1. You have **45 minutes** maximum to complete this examination. Pace yourself accordingly.
2. Write your answers in the space provided. No additional space will be provided.
3. Do **not** remove any papers from this booklet or add new ones.
4. You may **not** use any reference material(s) **except** what has been provided within this examination booklet and the book *C++ In A Nutshell*.
5. **You may not use the C Standard Library unless given explicit permission to do so.** This is a course on C++ --not C. C++ coding techniques and the C++ Standard Library without the C Standard Library subset must always be used. If you have any questions concerning this, then ask for clarification.
6. **Document your code where appropriate.** Unclear code may not receive partial marks without documentation. Ensure any written English uses proper spelling, grammar, and can be understood. Answers must be neat and legible to receive marks.
7. **Be sure** that you have printed your name and student number on all pages of this examination.
8. Ensure that you have all **8 pages** of this examination (including this page) before starting to write this exam. If you don't, bring this to the attention of the instructor immediately.
9. Ensure the proper case, spelling, syntax, grammar, and punctuation marks are correctly used in all answers involving code.

EXAMINATION MARK: ______________

MAXIMUM MARK: 54

Part I: Multiple Choice and Short Answer Questions (49 marks)

For each question in this section, neatly and plainly **circle or underline** the **single** response which most correctly completes/answers the statement/question given for multiple choice or True/False questions, otherwise, write in the appropriate answer(s) in the space provided. Read carefully! Unintelligible or ambiguous responses will receive a mark of zero (0) for that question, so ensure that your answer is clear.

Q1) The C++ programming language was created by _____ (full name).

Answer: ____ **[1 mark]**

Q2) C++ was originally called _____.

Answer: ____ **[1 mark]**

Q3) Briefly explain what the ISO C++'s committee "zero overhead rule" means in terms of design. **[1 marks]**

____

____

Q4) Explain the key differences between (i) modular and object-based, and, (ii) object-based and object-oriented programming. **[4 marks]**

(i) ____

____

____

(ii) ____

____

____

Q5) C++ is a multi-_____ programming language.

Answer: ____ **[1 mark]**

Q6) Briefly explain **why** the C programming language can **only** be said to **truly** support pass-by-value. **[2 marks]**

____

____

____

____

Q7) C++11 supports two types of references. What are they called?

Answer A: ____ **[1 mark]**

Answer B: ____ **[1 mark]**

Q8) Clearly explain what the differences, if any, are between `T *const` and `T const*`. **[2 marks]**

____

____

____

____

Q9) If the reference to some type, `T`, where to be written as `T&`, what would the <u>semantically equivalent pointer type declaration</u> to that reference be written as?

Answer: ____ **[2 marks]**

Q10) Using big-O, little-o, and omega complexity symbols, explain what the **cost of moving data** is in terms of **copying data** and **copying pointers to the data**. Also, is the cost of **moving data** ever zero? **[3 marks]**

____

____

____

____

____

____

Q11) One should view a _____ [1] operation as an optimized ____ [2] operation. (Hint: Q10.)

Answer 1: ____ **[1 mark]**

Answer 2: ____ **[1 mark]**

Q12) Write a C++11 lambda function that accepts an `int` as an argument and returns twice its value (as an `int`). **[2 marks]**

Q13) The mathematician that designed key portions of the STL is _____ (full name).

Answer: ____ **[1 mark]**

Q14) Briefly describe what a C++ Standard Library container represents. **[1 mark]**

____

____

____

Q15) Briefly describe what a C++ Standard Library iterator represents. **[1 mark]**

____

____

____

Q16) C++ Standard Library's iterators were modeled upon which C language construct?

Answer: ____ **[1 mark]**

Q17) In C++, object-oriented programming provides run-time polymorphism whereas its _____ programming provides compile-time polymorphism.

Answer: ____ **[1 mark]**

Q18) Briefly explain what is meant by a predicate in the C++ Standard Library. **[1 mark]**

____

____

Q19) True or false: In C++ a Standard Library function requiring a predicate allows the predicate to be stateful. **[1 mark]**

(a) True (b) False

Q20) In the C++ Standard Library, all sorting operations rely on a _____ (3 words) to sort things. The default overloaded operator to perform such is _____.

3-word answer: ____ **[1 mark]**

Operator answer: ____ **[1 mark]**

Q21) Why is C++'s use of the three-word answer with only a single operator (i.e., in Q20) better to use for sorting than using two operators. Briefly explain. **[2 marks]**

Q22) Why is C++'s use of the three-word answer with only a single operator (i.e., in Q20) enable one to do that cannot be done with a total order? Briefly explain. **[2 marks]**

Q23) Briefly explain what the differences between a forward iterator and a bidirectional iterator are in terms of their permitted operations. **[1 mark]**

Q24) Clearly describe what the C++ compiler adds to a C++ class/struct when one declares a virtual member function in it. Briefly what allows the correct virtual member function to be correctly invoked from all derived classes. **[4 marks]**

Q25) For a user-defined struct/class type `T`, write the member prototypes for the following: **[8 marks]**

Member	**Prototype**
Default constructor	T();
Copy constructor	T(T const&) = default;
Copy assignment operator	T& operator=(T const&) = default;
Move constructor	T(T&&) = default;
Move assignment operator	T& operator(T&&) = default;
Destructor	virtual ~T() = default;

Q26) Briefly explain how a programmer would (conceptually and) properly use the Resource Acquisition Is Initialization (RAII) design pattern. **[2 marks]**

ensures constructor allocated any needed resources, and the destructor
frees resources at the end of its lifecycle

Part II: General Questions (5 Marks)

Answer all parts of each question in the space provided below each question. The number of marks assigned to each question is indicated at the end of each question. You are expected to answer questions using complete sentences and proper grammar. If the answer is program code, simply write the code fragment that answers the question **unless you are explicitly asked to write a full-and-complete program**.

NOTE: *Unless you are asked to write a full-and-complete program, assume* `using namespace std;` *is at the* ***top*** *of the code fragment you are writing. If you are writing a code fragment within a function, assuming the proper* `#include` *files have been included elsewhere. You may use C++11 or C++98 code in your answers unless otherwise prohibited.*

Q20) Write the full-and-complete program that would be placed inside `main()` (include any variable declarations needed) to read in from standard input an **unknown** number of `int`s and output to standard output their sum. You are not allowed to use any containers. You are only allowed to declare **at most two** variables and can only use **one loop** construct. Your program must work correct even in the presence of input failures and errors (i.e., stop summing on the first failure, error, or EOF). You do not need to worry about integer overflow. **[5 marks]**

```
#include <iostream>

int main{
 using namespace std;

  int sum = 0;

  for(int i; cin >> i; sum+= i){
    ;
   }

      cout << sum << '\n';

}
```

```
int sum = 0;
int i;

while(cin >> i) {
   sum += i;
}

cout << sum << '\n'

return 0;
```

This page was intentionally left blank.

You may use it for rough work, or, if you've run out of space for a question.