

Fraser International College

Full Name: _____

A

Preferred/Nick Name: _____

FIC Student ID: _____

CMPT 130 Introduction to Computing Science and Programming I Midterm Exam

February 21

2016

Instructions

- Read each question carefully and write all your answers ON THE SPACE PROVIDED IN THIS QUESTION BOOKLET.
- Write legibly and with proper indentation. Coding style will be marked.
- Global variables outside functions are not allowed.
- This Midterm Exam is closed book. No reference material (printed, electronic or otherwise) is allowed.
- Use the back of each page for your answers if you run out of space and also for your scratch work.
- The mark assigned for each question is indicated on the question number.

Total Marks 100

Duration 90 min

Total Pages 9

For Instructor Use ONLY

Q 1	Q2	Q3	Q4	Q 5	Q 6	Q 7	Total

Question 1 (10 marks)

- I. **(6 marks)** In the following table, convert among 8-bit unsigned binary, decimal and hexadecimal representations of integers **a** and **b**:

	unsigned binary	decimal	hexadecimal
a	1111 1111		
b		43	

- II. **(4 marks)** In the following table, convert among decimal and 8-bit 2's complement representations of integers **a** and **b**:

	decimal	2's complement
a	-58	
b		1000 0000

Question 2 (30 marks in total, 3 marks each) Assuming each of the following code fragments is embedded in a correct and complete C++ program, what is the output from each of the code fragments? If an error results, write “error” and explain briefly what the error is (if you just write “error” without explanation, you will get zero).

	Code fragments	output
1	<pre>float d = 10; int n = static_cast<int> (d); cout << n / 4 << endl; cout << d / 4 << endl;</pre>	
2	<pre>float d = 10; static_cast<int> (d); cout << d / 4 << endl; cout << static_cast<int> (d / 4) << endl;</pre>	
3	<pre>int n; for(int i = 1; i < 10; i += 2) n = i; cout << n << endl;</pre>	
4	<pre>int n; for(int i = 4; i > 0; i -= 2) n = ++i; cout << n << endl;</pre>	
5	<pre>int i = 0; while (i < 5) i++; cout << i << endl;</pre>	
6	<pre>char ch = '5'; while (ch > '1') { ch--; cout << ch; cout << ch - '1' << endl; }</pre>	

7	<pre> int sth = 0; for (int i = 0; i < 3; ++i) { cout << i << " "; for (int j = 1; j < 4; ++j) sth += j; cout << sth << " "; } </pre>	
8	<pre> void myFunction(int n) { for (int i = 0; i < n; i++) { if (i % 2 == 0) cout << "Even "; else cout << "Odd "; } } int main() { myFunction (5); return 0; } </pre>	
9	<pre> int do_sth(int n) { for (int i = 1; i < n; i++) { if (i % 2 == 0) return i; else return -i; } } int main() { cout << do_sth(5); return 0; } </pre>	

10	<pre> void foo (int n) { n = 100; } int main() { int n = 10; foo (n); cout << n << endl; return 0; } </pre>	
----	----------------------------------------------------------------------------------------------------------------------------------------------	--

Question 3 (8 marks)

Convert the following code fragment to a logically equivalent code fragment that uses only while-loops (outer while and inner while) instead of the for-loops.

```

int sth = 0;
for (int i = 0; i < 3; ++i)
{
    cout << i << " ";
    for (int j = 1; j < 4; ++j)
        sth += j;
    cout << sth << " ";
}

```

Question 4 (6 marks)

The following is supposed to be a program that asks the user for three numbers and then prints the largest number. Now answer the question below the table.

```
int n1, n2, n3;
cout << "Enter three integers: ";
cin >> n1 >> n2 >> n3;
if (n1 > n2)
{
    if (n1 > n3)
        cout << n1 << " is the largest.";
    else
        cout << n3 << " is the largest.";
}
else if (n2 > n3)
    cout << n2 << " is the largest.";
```

There is a logical (semantic) error in the above program; that is for some input numbers, it doesn't work as expected. Provide such a test example (the three input numbers) for which the program does not work as expected.

Question 5 (12 marks)

Write a C++ function named **print_digits_backwards** that takes one integer argument and prints backwards all the digits in the given integer argument. Assume the argument is always non-negative.

Three function calls and their corresponding outputs are provided for your reference as follows:

function call

```
print_digits_backwards(3561);  
print_digits_backwards(123356);  
print_digits_backwards(0);
```

output

```
1 6 5 3  
6 5 3 3 2 1  
0
```

Question 6 (20 marks) A positive integer number n is prime if it is divisible by ONLY 1 and itself.

a. (5 marks) Write a C++ function named **isPrime** that takes a positive integer argument and returns true if the argument is prime and returns false otherwise. Assume the argument is greater than 1.

b. (15 marks) Write a C++ function named **sumOfPrimes** that takes a positive integer argument n and returns the sum of the first n prime numbers. For example, **sumOfPrimes(6)** must return 41, which is the sum of the first 6 prime numbers: $2+3+5+7+11+13 = 41$. You must use the **isPrime** function you defined in part (a) to define the function **sumOfPrimes**.

Question 7 (14 marks)

a. (4 marks) Write a function named **myRandomNumber** that takes no argument and returns a random integer in the range [1, 100].

b. (10 marks) Write a C++ main program that **repeatedly** generates and **prints** random integers in the range [1, 100]. The program must stop when the absolute value of the difference between successive printed values is less than 5.

You must make use of the **myRandomNumber** function you defined in part (a) when you write your main program.

You must provide all the required includes and namespaces when writing your main program.

Two sample outputs are given to help you understand the problem:

1. The program can possibly output this: **76 43 77 94 54 41 45**
2. The program can possibly output this: **42 40**