Due before:	11:59 PM (before Midnight) on Thursday June 13 2019

Introduction to Systems Programming (Systems I)

Homework #3

Due: Thursday June 13 2019 before 11:59 PM

Email-based help Cutoff: 5:00 PM on Wednesday, June 12 2019

Maximum Points: 50

Submission Instructions

This part of the homework assignment must be turned-in electronically via Canvas. Ensure you name this document *MUid_homework3*.docx, where *MUid* is your Miami University unique ID. Complete the method shown for each problem. For each method, you can develop and test them in NetBeans and just copy-paste your solutions into this document.

Once you have completed answering the questions save this document as a PDF file (don't just rename the document; that is not the correct way to save as PDF) and upload it to Canvas

General Note: Upload each file associated with homework (or lab exercises) individually to Canvas. Do not upload archive file formats such as zip/tar/gz/7zip/rar etc.

Copy pasting from online resources is plagiarism. Instead you should understand concepts and explain them using your own words!

Name: Ce Zhang

The objective of this PART A of the homework is to review and recollect the necessary background information and terminology from CSE-174 and CSE-271 (the prerequisite courses). The terminology will be often used in the course.

PART A

1. **What is source code (1 or 2 sentences)?**

Source code is the set of instructions which is written by a programmer using a computer programming language, and the source code is readable by everyone.

2. **What is the difference between source code, pseudo code, and an algorithm (answer in 3 separate sentences)**

Algorithm: Systematic logical approach which is a kind of pattern to let the computer to figure out the problem.
Pseudo code: It is a simpler version of a programming code in plain English which is implantation the algorithm and it is look like actually programmed
And it cannot be compiled.
Source code is the actual code which can be compiled also.

3. **What is a syntax error? How do you detect them? How do you fix syntax errors? (Answer in 3 separate sentences)**

Syntax error in computer science is an error in the syntax of a coding or programming language and it will be run wrong when you want to compile it.

The program will find by software named compiler

The programmer should find the way to fix it and let it compiler and then run.

4. **What is a semantic error? How do you detect them? How do you fix semantic errors? (Answer in 3 separate sentences)**

Semantic is kind of error which is invalid program logic that produces incorrect results when the instructions are executed.

And this error cannot find by the compile, we can find it only we test it. For example, we can make some output we need, and test the program. If the output from the program is not ours. We see this as the semantic problem.

We fix it by ourselves. We need try a lot of times and make the output match what we need, this is a logic way to fix it.

5. **What is a debugger? What is a breakpoint? (answer in 2 separate sentences)**

A debugger is a software program which can help us to find the bug in the program. And debugger is also called debugging tool.

A breakpoint is a marker that you can set to a specific place when execution should pause

6. **Briefly (1 to 2 sentences each) state the 4 key principles of object-oriented programming**

Encapsulation: it is the mechanism of hiding of data implementation which is kind of restricting access to public methods.

Abstract: it means a concept or an idea which is not with the any instance.

Inheritances: it is that one object is kind of another object

Polymorphism: It means one name can be many forms.

7. **Briefly (2 to 3 sentences each) discuss pass-by-value versus pass-by-reference mechanisms for passing parameters to methods. Which approach is preferred based on data type of parameters in C++?**

When a parameter is passed by reference, the caller and the caller use the same variable for the parameter. If the caller change the parameter variable, the caller can see the parameter change.

When a parameter is passed by value, the caller and caller will have two independent variables with the same value. If the caller change the parameter variable, caller cannot see it.

8. **List at least 3 unique properties that can be inferred from data type of a variable?**

1. Local variable: A variable which is declared in the method. And we can see this variable as local variable.

2. instance variable: A variable declared in the class. And we can called it as instance variable outside of method.

3. Static variable: Any variable which is declared as static is called as static variable. And we cannot see as a local variable.

PART B

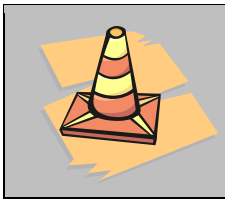
Objective
The objective of this PART B of the homework is to: • Understand working with `std::vector`. • Practice answering exam style questions

Required reading

Prior to answering the questions in this homework briefly review the following chapters from the E-book titled "C++ How to Program" (all students have free access to the electronic book):

Due before:	11:59 PM (before Midnight) on Thursday June 13 2019

- Chapter 7.10 (`std::vector`)
- Chapter 14.1 – 14.6 (File I/O)



Although the Safari E-books are available to all students there are only a limited number of concurrent licenses to access the books. Consequently, do not procrastinate working on this homework or you may not be able to access the E-books due to other users using them.

1. What is quoted text and how do you read quoted text in C++? Explain with a suitable example (other than the one shown in Chapter 14)

```
#include <iostream>
using namespace std;
int main()
{
  string s;
  cout<<"enter quoted text: ";
  getline(cin,x);
  cout<<"You eneterd: "<<x<<endl;
  return 0;
}
```

2. Assume you have a method called `processLines(std::istream& is, std::ostream& os)` that process line-by-line. Complete the `main` method below to call `processLines` method to process the 3 lines: `"Line #1"`, `"Second Line"`, and `"Last line"`. The output should be written to standard console output stream. (Hint: Use a `std::istringstream`)

```
// Prototype declaration
void processLines(std::istream& is, std::ostream& os);

int main() {
  istringstream a("Line #1");
processLines(a, cout);

istringstream b("Second Line");
processLines(b, cout);

istringstream c("Last Line");
processLines(c, cout);
```

```
}
```

3. Complete the following method that returns a vector with only even values in the `src` vector. If the `src` vector has values {2, -4, 7, 9, 3, 8} this method should return a vector with values {2, -4, 8}.

```
using IntVec = std::vector<int>;

IntVec evens(const IntVec& src) {

std::vector<IntVec> newvector (10);
int j = 0;

int i;

    for (i=0; i<src.size(); i++)
         if(src.at(i) % 2 == 0)
{

             newvector.at(j) = src.at(i);
}

     return newvector;









}
```

4. Complete the following method that returns a vector that contains a reverse of the words. For example if `src` is {`"one"`, `"two"`, `"three"`} the method should return a vector with strings {`"three"`, `"two"`, `"one"`}

```
using StrVec = std::vector<std::string>;

StrVec reverse(const StrVec& src) {

  StrVec rev;
    for (int i = 0; i < src.size(); ++i) {
       rev.push_back(src[src.size()-i-1]);
    }
    return rev;

}
```

5. Complete the following method that returns a vector with the first `n` prime numbers. For example, if `n == 7`, this method should return a vector with values {1, 2, 3, 5, 7, 11, 13}.

```
using IntVec = std::vector<int>;
IntVec getPrimes(int n) {


vector<int> primeNumbers;

   int count = 0;

   for(int i=2; count < 7; i++){

       if(isPrimeNumber(i)){
           primeNumbers.push_back(i);
           count++;
       }
   }

   return primeNumbers;

}

bool isPrimeNumber(int number){

for(int i=2; i<=number/2; i++){
if(number % i == 0){
return false;
}
}
return true;
```

```
}
```

Part C: Submission

- No late assignments will be accepted!
- This work is to be done individually
- The submission file will be saved with the name ***HW3_yourMUID*.cpp***
- Assignment is due Thursday June 13, 2019 before Midnight
- On or before the due time, drop the *electronic copy* of your work in the *canvas*

Don't forget to Turn in the files! HW3_yourMUID.pdf & HW3_yourMUID*.cpp