PASS MOCK EXAM – FOR PRACTICE ONLY

Course: ECOR 1606 Facilitator: Alex Sereda

Dates and locations of mock exam take-up:

Tues, Dec 4th: 6-730pm, Library (ML) 402
Wed, Dec 5th: 6-730pm, Library (ML) 402

IMPORTANT:

It is **most beneficial** to you to write this mock final **UNDER EXAM CONDITIONS**. This means:

- Complete the midterm in _3_ hour(s).
- Work on your own.
- Keep your notes and textbook closed.
- Attempt every question.

After the time limit, go back over your work with a different colour or on a separate piece of paper and try to do the questions you are unsure of. Record your ideas in the margins to remind yourself of what you were thinking when you take it up at PASS.

The purpose of this mock exam is to give you practice answering questions in a timed setting and to help you to gauge which aspects of the course content you know well and which are in need of further development and review. Use this mock exam as a *learning tool* in preparing for the actual exam.

Please note:

- Come to the PASS session with your mock exam complete. There, you can work with other students to review your work.
- Often, there is not enough time to review the entire exam in the PASS session. Decide which questions you most want to review – the facilitator may ask students to vote on which questions they want to discuss.
- Facilitators do not bring copies of the mock exam to the session. Please print out and complete the exam before you attend.
- Facilitators do not produce or distribute an answer key for mock exams. Facilitators help students to work together to compare and assess the answers they have. If you are not able to attend the PASS session, you can work alone or with others in the class.

1) Discuss the differences between **while**, **do**-**while**, and **for** loops.

2) The following code evaluates potential factors of z, and outputs whether both x and y are factors, only one of them is a factor (does not determine which), or neither of them are a factor.

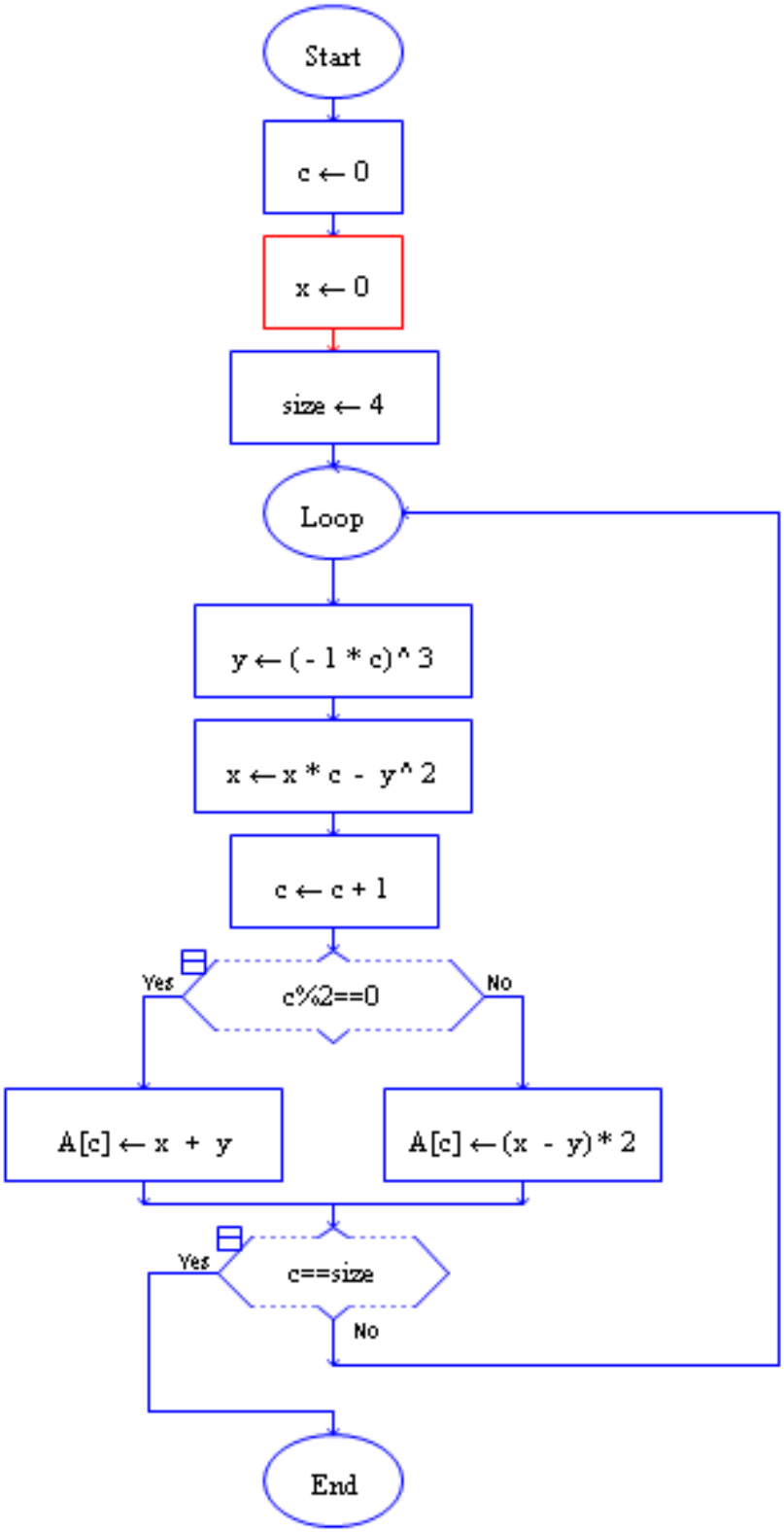
a) Fill in the two blank cout statements (with proper syntax).

```
if ( z % x == 0 && z % y == 0){
    cout << " x and y are both factors of z";
} else if ( z % y == 0 || z % x == 0  ) {
    cout
} else {
    cout
}
```

b) If x = 4, y = 5, and z = 124, what would the output be?

_____

3. Evaluate the following flowchart to determine what final values are stored in the array.



A[1] = A[3] =

A[2] = A[4] =

2. Convert the flowchart from question 1 into C++

3. Below is the main function of a C++ program.

a) Identify and correct the 8 errors in the code.

```
//     This program solicits input for the A, B, C cooefficients of a quadratic function as well as the endpoints
//    for an interval, and determines the minimum and maximum value of the function over the interval.

int main(){  //
        double starts, done, x, y, min, max, step

        int size == 3;
        double coeffs[];
        cout << input a, b, c << endl;

        for (int i = 0; i <= size ; i + 1) {
        cin >> coeffs[i];
        }

        cout << "input start and end point"  endl;
        cin >> starts >> done;
        min = calcfx(starts, coeffs)
        max = min;

        step = sqrt(pow((starts - done)/10,2));    //  pow is a built-in function of c++. Eg: Pow(x, 2) is x^2.

        x = starts;

        while (x <= done) {

                y = calcfx(x, coeffs);
                checkminmax(y, min, max);
                x = x + step;
                if (x > done) {
                    y = calcfx(done, coeffs);
                    checkminmax(y, min, max);
                }
        }

    cout << "Ultimate minimum over the interval: " << min << endl << "Ultimate maximum over the interval: "
<< max << endl;
    system ("PAUSE");
    return 0;
}
```

b) The previous code calls two separate functions. From the function calls in the code, write the corresponding functions (including each function header).

1ST FUNCTION

2ND FUNCTION

4. a) Draw a raptor flowchart that would solicit the user's birthday and current date and output how many days until the user's birthday. If the dates match, the program should output an appropriate message and terminate. If the input does not make sense, the program should output an appropriate message and terminate. Your flowchart must make use of at least one procedure.

b) convert your flowchart to C++.

5. In C++, implement the bisection approach to find the square root of a given number.