CSci 1113
Midterm 1

Name: __________________________________________

Student ID: _________________________________

Instructions: Please pick and answer any 6 of the 7 problems for a total of 60 points. If you answer more than 6 problems, only the first 6 will be graded. The time limit is 60 minutes. Please write your answers in the space provided. The exam is open book and notes. You may use electronic devices to ONLY look at either an e-book version or electronic notes. You may not use the internet, compiler or any other outside resources. (If you are typing on your keyboard/input device for anything other than ctrl-F to find words in the e-book or notes, this is probably not acceptable.)

Problem (1) [10 points] For both parts, show the outputs of the following loops:

(a)
```
int x=1;
for(int i=1 ;i < 100; i*=2)
{
    cout << i%x << " ";
x++;
}
```

(b)
```
int z = 4;
for(int i=0; i <= z; i++)
{
    for(int j=0; j <= z; j++)
    {
        if(i == j || i == z-j)
        {
            cout << "x";
        }
        else
        {
            cout << " ";
        }
    }
    cout << endl;
}
```
Grade breakdown:
part (a) 4 points
  1 point - first and second 0
  1 point - 3rd and 4th (1, 0)
  1 point - rest of sequence correct
  1 point - stopping at the right spot
part (b) 6 points
  2 points - first row correct
  1 point - second row correct
  2 points - diagonal correct
  1 point - fully correct

Solution:
(a)
0 0 1 0 1 2 1

(b)
x  x
x  x
x
x  x
x  x
Problem (2) [10 points] Ask the user how many times they want to loop (assume at least once), read in the requested amount of numbers (no restrictions on what they can enter), then display the largest number that they entered (i.e. the maximum).

Grade breakdown:
1 point - read how many numbers
2 points - declare and initialize max (max as double, initialize value reasonable)
4 points - 1 point each for: using any loop, correct initial/end (i=1; i <= amount, is okay), i++
3 points - correctly finding the max

Solution:

```cpp
int amount; cin >> amount;

double max=0; // told students could assume positive
for(int i=0; i < amount; i++)
{
    double next; cin >> next;
    if(max < next)
    {
        max = next;
    }
}
cout << max;
```
Problem (3) [10 points] Find 3 possible places for errors in the following code fragment. Explain specifically what causes the error and whether it is a syntax, runtime or logic error. You may assume all includes are done properly, the code is inside a valid function, it is using namespace std and there are no manually declared global variables.

```cpp
int 3number = 3;
cout << "Enter "<<3number*2<<" numbers: " << endl;
for(int i=0; i < 3number*2; i++) {
    double one, two, max;
    cin >> one >> two;
    if(one > two) {
        max = one;
    } else {
        max = two;
    }
}
cout << "The largest number was " << max << endl;
```

Grade breakdown:
3 points per part:
  2 points for description
  1 point for type of error
1 free point

Solution:
Syntax, "3number" incorrect identifier
Logic, loop reads 12 numbers, not 6
Logic, max is wrong....
Syntax, max is out of scope at end
Problem (4) [10 points] Rewrite this switch as logically equivalent set of if/else statements.

```cpp
char c;
cin >> c;

switch(c) {
    case 'a':
    case 'b':
        cout << "1";
        break;
    case 'c':
        cout << "2";
        // yes, no break
    case 'd':
        cout << "3";
        break;
    default:
        cout << "4";
        break;
}
```

Grade breakdown:
2 points - ensuring one statement will always run
2 points - correct output for 'a'
1 point - correct output for 'b'
2 points - correct output for 'c'
1 point - correct output for 'd'
2 points - correct output for anything else

Solution:
```cpp
if(c == 'a' || c == 'b') {
    cout << "1";
}
else if(c == 'c') {
    cout << "23";
}
else if(c == 'd')
```
```cpp
{ 
    cout << "3";
}
else 
{
    cout << "4";
}
```

**Problem (5) [10 points]** For each part, the conditions of the if-statements are not given. Without knowing the actual conditions, say give both the maximum and minimum amount of code blocks that can be executed.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>/<em>cond 1</em>/</td>
<td>/<em>cond 1</em>/</td>
<td>/<em>cond 1</em>/</td>
<td>/<em>cond 1</em>/</td>
<td>/<em>cond 1</em>/</td>
</tr>
<tr>
<td>//block 1</td>
<td>//block 1</td>
<td>//block 1</td>
<td>//block 1</td>
<td>//block 1</td>
</tr>
<tr>
<td>/<em>cond 2</em>/</td>
<td>else if(/<em>cond 2</em>/</td>
<td>else if(/<em>cond 2</em>/</td>
<td>else if(/<em>cond 2</em>/</td>
<td>else if(/<em>cond 2</em>/</td>
</tr>
<tr>
<td>//block 2</td>
<td>//block 2</td>
<td>//block 2</td>
<td>//block 2</td>
<td>//block 2</td>
</tr>
<tr>
<td>/<em>cond 3</em>/</td>
<td>else if(/<em>cond 3</em>/</td>
<td>if(/<em>cond 3</em>/</td>
<td>if(/<em>cond 3</em>/</td>
<td>else if(/<em>cond 3</em>/</td>
</tr>
<tr>
<td>//block 3</td>
<td>//block 3</td>
<td>//block 3</td>
<td>//block 3</td>
<td>//block 3</td>
</tr>
<tr>
<td>/<em>cond 4</em>/</td>
<td>else</td>
<td>else if(/<em>cond 4</em>/</td>
<td>else</td>
<td>if(/<em>cond 4</em>/</td>
</tr>
<tr>
<td>//block 4</td>
<td>//block 4</td>
<td>//block 4</td>
<td>//block 4</td>
<td>//block 4</td>
</tr>
</tbody>
</table>

**Grade breakdown:**
1 point per max
1 point per min

**Solution:**
Code A: max = 4, min = 0
Code B: max = 1, min = 1
Code C: max = 2, min = 0
Code D: max = 2, min = 2
Code E: max = 2, min = 0
Problem (6) [10 points] Fill in the blanks (i.e. ______) with the most appropriate pieces of code. The code should be fully functional after you fill in. You may only fill in the blanks, not change the existing code. All blanks must be less than 30 characters (no writing tons of code in them!).

```cpp
-------------
cout << "Odd numbers between 0 and " << m << " are: ";

-------------
{
  if(i%2 == 0)
  {

    -------------
    cout << i << " ";
  }
}
```

Grade breakdown:
2 points - declare and initialize m
3 points - loop
2 points - continue
3 points - correctly cout odd output

Solution:
```cpp
int m = 20;
for(int i=0; i < m; i++)
  continue;
```

Also:
```cpp
int m = 40;
for(int i=1; i < m; i+=2)
  // a comment
```

Problem (7) [10 points] Assume there exists a variable size with an integer value. Print a “T” rotated π² radians (90 degrees) to the left as shown in the example output. You must use size to determine how many X’s to print and not always 5 as shown in the output (you may assume size is odd).

Sample output for size=5:
X
X
X X X X X
X
X

Grade breakdown:
2 points - outer loop (over size)
3 points - inner loop (over size)
3 points - couting a vertical column
2 points - couting the base of the T correctly halfway

Solutions:
for(int i=0; i < size; i++)
{
    if(i == size/2)
    {
        for(int j=0; j < size; j++)
        {
            cout << "X ";
        }
    }
    else
    {
        cout << "X ";
    }
    cout << endl;
}