Assignment 1 - Solutions to Part A

1. Operator Precedence – 10 Marks

Determine the operator precedence for the following C++ expressions and evaluate each expression based on this precedence. Assume that the variables are defined as follows:

int a = 5; int b = 4; int c = 3; int d = 2; int e = 1; double f = 10;

```
(a)   a – b – c – d
(b)   a – (b – c)  - d
(c)   a + b  *  c /d  % e
(d)  (a + b) * (c /d) % e
(e)  (a + b + c + d + e)/ f
(f)  'a'–'b'+ c + d + e + f
(g)  (a < b) || !(c == d)
(h)  (a == b) ? c : d
(i)   a += b
(j)   a++ + ++a + a
```

Answers:

```
(a)   a – b – c – d = -4
        1   2   3
(b)   a – (b – c)  - d = 2
        2    1     3
(c)   a + b  *  c /d  % e = 5
        4    1    2   3
(d)  (a + b) * (c /d) % e = 0
        2    3    1   4
     or 1    3    2   4
(e)  (a + b + c + d + e)/ f = 1.5
        1   2   3   4   5
(f)  'a'–'b'+ c + d + e + f = 15
        1   2   3   4   5
(g)  (a < b) || !(c == d) = 1 (or true)
        1    4  3   2
(h)  (a == b) ? c : d = 2
        1     2   3
     or 1     2   2
     or 1     2
(i)   a += b = 9
        1
(j)   a++ + ++a + a = 19 (or 31 if assume (i) done first)
       2  3  1  4
```

2. Generating Random Numbers – 5 Marks

What #include file is required to create random numbers?

Answer: #include <cstdlib>

or #include <stdlib.h>

What function must be called to seed rand's random number generator?

Answer: srand()

What is the range of the result of each of the following expressions?

(a) `rand() % 20`	Range: [0,19]
(b) `rand() % 6 + 1`	Range: [1, 6]
(c) `rand() % 100 – 50`	Range: [-50,49]

3. Definite Loops – 5 Marks

Write a for loop to print the odd numbers from 1 to 99 (inclusive) as a space separated list. You do not need to write a running program, just the for-loop and for-loop body.

The solution presented below is the preferred solution but is just one possible solution. Other solutions that produce the same output are acceptable. The final line is optional as it was not part of the specification.

Solution:

```
for(int i=1; i<100; i+=2)
        cout << i << " ";
cout << endl;
```

Alternative solutions acceptable but may be less clear:

```
for(int i=1; i<=99; i++) {
   cout << i << " ";
   // Modifying the loop variable within a
   // FOR loop is considered poor style.
   i++;
}
```

Using IF to skip every second iteration of a FOR loop is also poor style:

```
for(int i=1; i<=99; i++)
   if (i%2 == 1)
      cout << i << " ";
```

4. Indefinite Loops – 5 Marks

Transform the following while loop into an equivalent do-while loop such that it produces the same output.

```
int num = 1;
while (num++ < 20)
{
   cout << num << " ";
}
cout << endl;
```

Solution:

Note that the above while loop outputs the following:

```
   2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

This is because num starts at 1, but gets incremented by the while statement, so the first value of num to be displayed to cout is 2. So we need the do-while to output the same sequence.

Students must remember to place a single space between each number, and output the final new line character (endl) for full marks.

Here is one such solution:

```
   int num = 1;
   do {
       cout << ++num << " ";
   } while (num < 20);
   cout << endl;
```