MIDTERM2 A

Section A – 3 Points Each (do 7 out of 8)

Q1) What is the output of the loop below?

Q1) Click or tap here to enter text.

```
XXXXXXXXXX
 XXXXXXXXX
  XXXXXXXX
   XXXXXXX
    XXXXXX
     XXXXX
      XXXX
       XXX
        XX
         X
```

```
for (int row = 1; row <= 10; row++, cout << endl)
{
        for (int col = 1; col <= 10; col++)
                if (col < row)
                        cout << " ";
                else
                        cout << "X";
}
```

Q2) What is the output of the loop below?

Q2) 1 2 3 4 5 6 7 8 9 10

```
int i = 1;
while (true)
{
        if (i / 10 == 1)
                break;
        cout << i << " ";
        i++;
}
cout << i << endl;
```

Q3) What is the output of the loop below?

Q3) 1 3 5 7 9 10

```
int i = 0;
while (i / 10 == 0)
{
        i++;
        if (i % 2 == 0)
                continue;
        cout << i << " ";
}
cout << i << endl;
```

Q4) What is the output of the loop below?

Q4) 20

```
int i = 0;
do
{
        i++;
        if (i)
                continue;
        cout << i << " ";
} while (i < 20);
cout << i << endl;
```

Q5) What is the output of the loop below?

Q5) 0 2 4 6 8

```
for (int i = 0; i < 10; i += 2)
        cout << i << " ";
cout << endl;
```

Q6) What is the output of the loop below?

Q6) 0

```
for (int i = 0; i % 2 == 0; i++)
        cout << i << " ";
cout << endl;
```

Q7) What is the output of the loop below?

Q7) 0 1 2 3 4 5

```
int i = 0;
while (1 <= 1)
{
        cout << i << " ";
        if (i == 5)
                break;
        i++;
}
cout << endl;
```

Q8) What is the output of the loop below?

Q8) 10 9 8 7 6 5

```
int i = 10;
do
{
        cout << i << " ";
        i--;
} while (i > 4);
cout << endl;
```

Section B – 10 Points Each (do 3 out 4)

P1)

Print the numbers between start and finish using a while loop
Numbers should be separated by comma
PrintRange(4,6) ==> 4, 5, 6
PrintRange(6,4) ==> 6, 5, 4

```
void PrintRange(int start, int finish)
{
	bool firstTime = true;
	if (start < finish)
		while (start <= finish)
		{
			if (firstTime)
				firstTime = false;
			else
				cout << ", ";
			cout << start;
			start++;
		}
	else
		while (start >= finish)
		{
			if (firstTime)
				firstTime = false;
			else
				cout << ", ";
			cout << start;
			start--;
		}
	cout << endl;
}
```

P2)

Print odd numbers between start and finish using a while loop
Numbers should be separated by comma
Assume start is always less than finish
PrintOddRange(4,6) ==> 5
PrintOddRange(1,10) ==> 1, 3, 5, 7, 9

```
void PrintOddRange(int start, int finish)
{
	bool firstTime = true;
	while (start <= finish)
	{
		if (start % 2 == 1)
		{
			if (firstTime)
			{

			}
			else
				cout << ", ";
			cout << start;
			firstTime = false;
		}
		start++;
	}
}
```

P3**)**

Using a while loop, add digits of a number
AddDigits(7777) ==> 28
AddDigits(27) ==> 9
AddDigits(0) ==> 0

```
int AddDigits(int n)
{
	int sum = 0;
	while (n)
	{
		sum += n % 10;
		n /= 10;
	}
	return(sum);
}
```

P4)

Use a double for loop to output the following
For example PrintPattern(10) will output the following…

```
XXXXXXXXX
 XXXXXXXX
  XXXXXXX
   XXXXXX
    XXXXX
     XXXX
      XXX
       XX
        X
```

```
void PrintPattern(int n)
{
        for (int row = 1; row <= n; row++, cout << endl)
        {
                for (int col = 1; col <= n; col++)
                        if (col < row)
                                cout << " ";
                        else
                                cout << "X";
        }
}
```