ISU Programming Assessment, Sept 27, 2019

Name: _____ CS class_____

Put all answers in boxes. Nothing you write outside the boxes will be counted. Did you bring an eraser?

1. **Write** a program that gets an integer, **n,** from the user and then prints **n** patterns. Each pattern consists of some XY's then some Z's. The first pattern is one XY followed by one Z. Each new pattern has one more XY and one more Z than the previous. **Example:** if **n=3,** then the program will print XYZXYXYZZXYXYXYZZZ.

```
int main(int argc, char *argv[]) {
```

```
    return 0;
}
```

2. **Get input a character at a time.** Write a program that counts the total number of sequences consisting of a lower case or upper case **e** followed by a decimal digit. It should print out the only the final count.

```
int main(int argc, char *argv[]) {
```

```
    return 0;
}
```

3. Write the function `numFalls` that is passed the address of the head node of the list. The function counts the number of times a number in the list is larger than the next number. It returns the final count.

```
typedef struct NODE {
    int data;
    struct  NODE *next;
}   node_t;
```

```
int  numFalls(node_t *curr) {
```

```
    }
```

4. A BST is constructed in the usual way using the node definition below. **Write** the function

```
        int numLeftChildren( bst_node_t *curr)
```

that is passed address 0 or the address of the root node of the BST. It returns the number of left children in the tree.

```
typedef struct  BST_NODE_T   {
 int   data;
 struct   BST_NODE_T   *left,  *right;
}   bst_node_t;
```

5. Write the function

```
int onesOnRight(int n)
```

where `n` is a 32–bit int. Returns 0 if the rightmost bit is a `0`. Otherwise it counts the number of `1`'s on the right and returns this value. Example: if the rightmost 8 bits is 11011111, the function will return a 5.