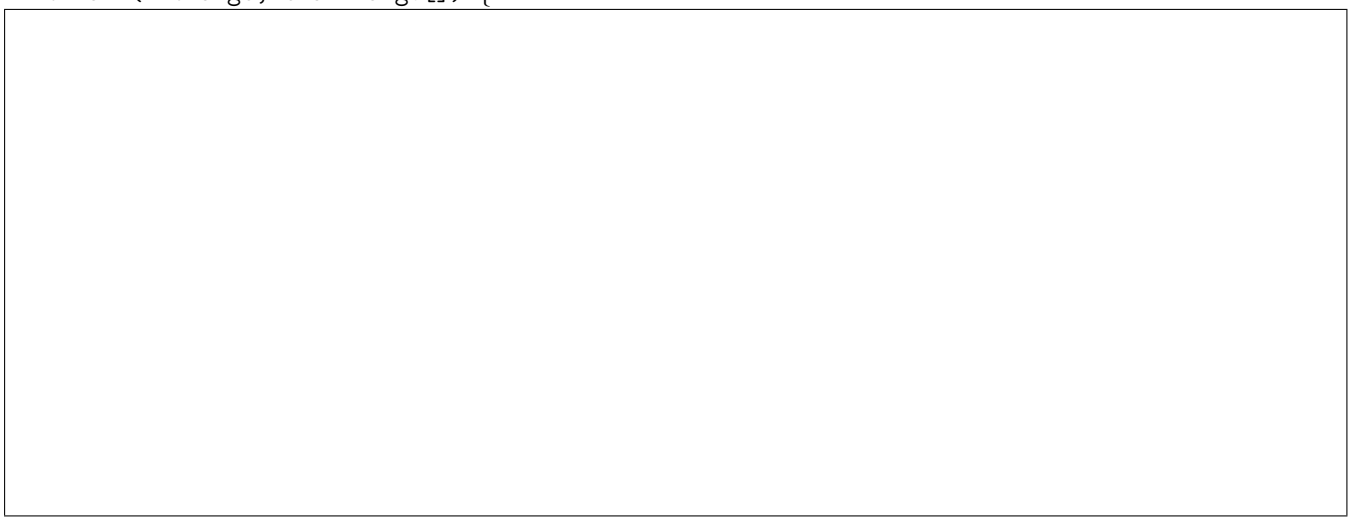
ISU Programming Assessment, Oct 18, 2019

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 contains upper case letters. The first pattern is `A`. The second pattern is `AB`. Each new pattern starts with `A` and contains one more letter than the previous pattern. **Example:** if `n=5`, then the program will print `AABABCABCDABCDE`. *NOTE:* the statement `printf("%c", 'A'+j);` prints the character whose character code is `'A'+j`.

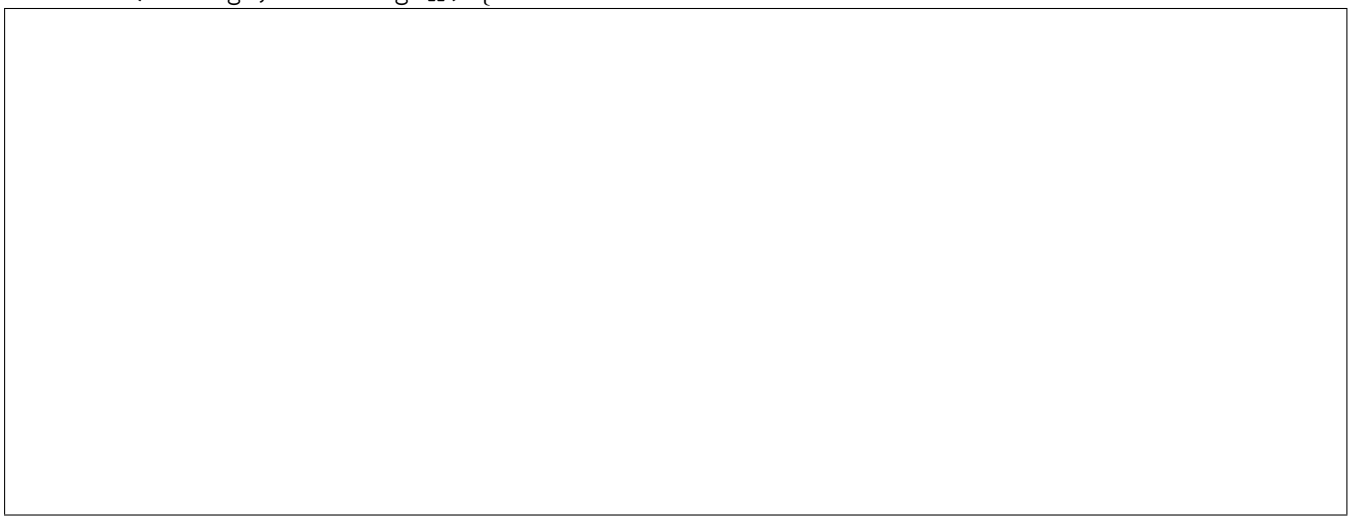
```
int main(int argc, char *argv[]) {
```



```
  return 0;
}
```

2. **Get input a character at a time.** Write a program that counts the total number of times that the string `dog` appears in its input. It should print out only the final count.

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

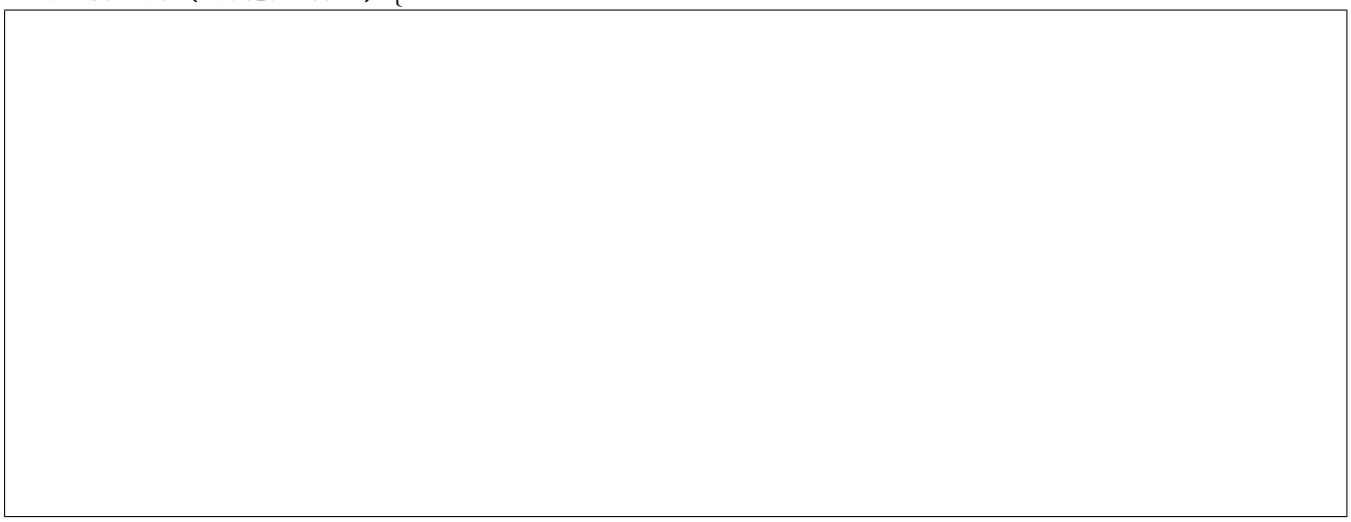


```
  return 0;
}
```

3. Write the function `sumEven` that is passed the address of the head node of the list. The function finds the sum of all the even numbers in the list. It returns the final sum.

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

int  sumEven(node_t *curr) {
```



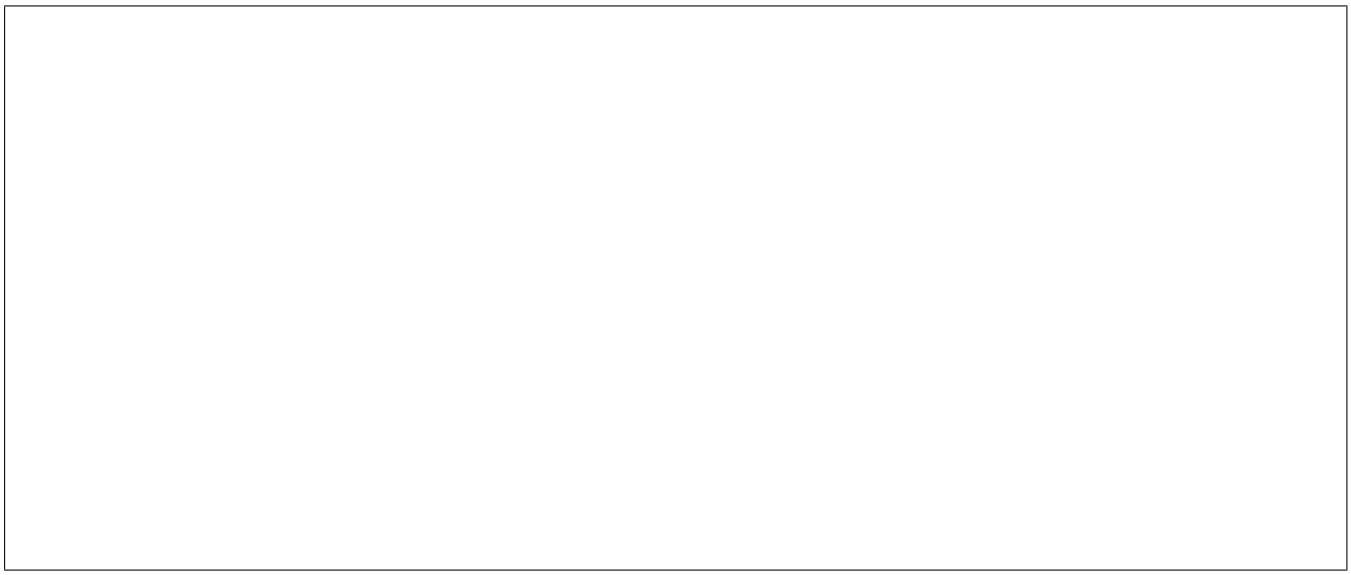
```
}
```

4. A BST is constructed in the usual way using the node definition below. The **level** of a node is how far below the root node it is. For instance, the children of the root are at level 1. **Write** the function

```
int numNodes5( bst_node_t *curr, int level)
```

that is passed the address of the root node, and 0 (level of root). It returns the number of nodes at level 5 in the tree.

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



5. Write the function

```
int dist(int n1, int n2)
```

where `n1, n2` are 32–bit ints. It returns the number of bits where n1 and n2 differ.