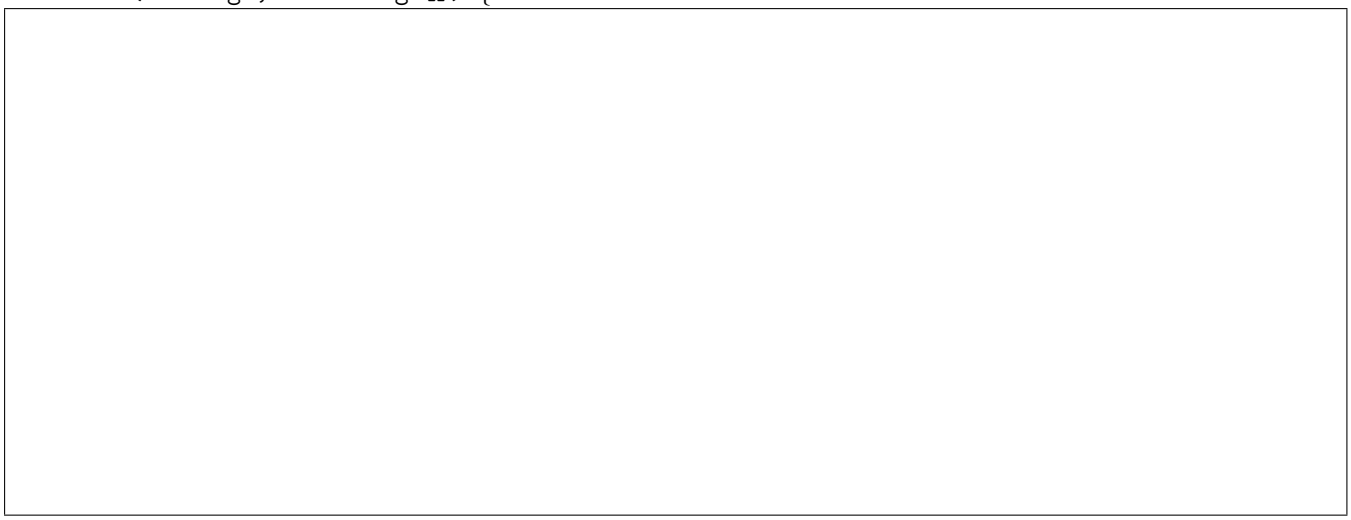
ISU Programming Assessment, Dec 06, 2019

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 one `A` then some `B`'s. The first pattern is one `A` followed by `n B`'s. Each new pattern has one `A` and one fewer `B` than the previous pattern. **Example:** if `n=3`, then the program will print `ABBBABBAB`

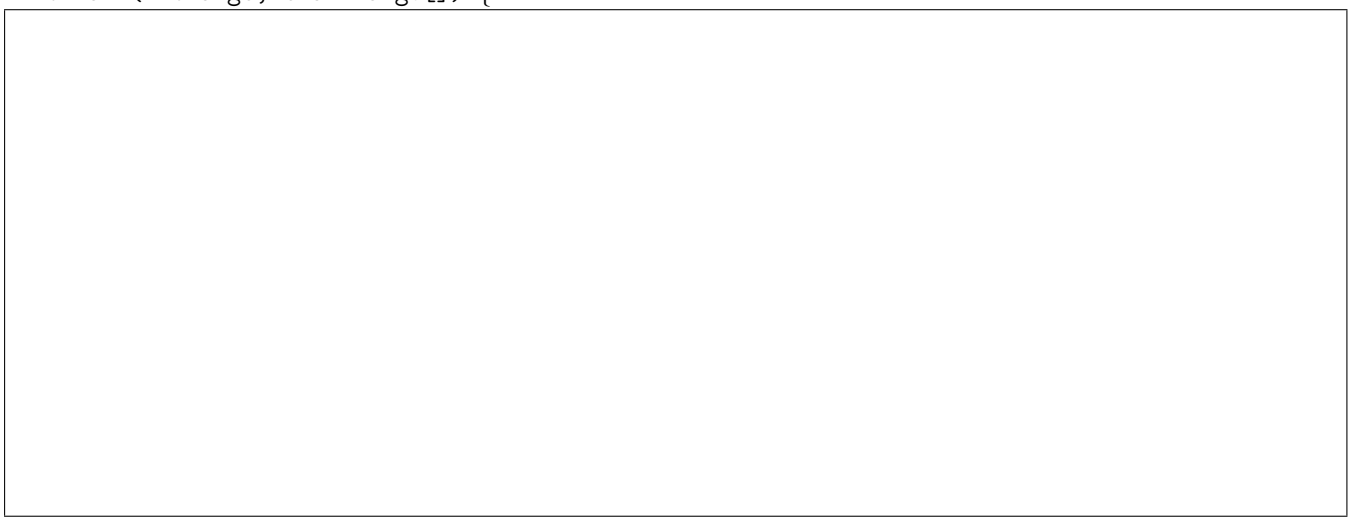
```
int main(int argc, char *argv[]) {
```



```
  return 0;
}
```

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

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

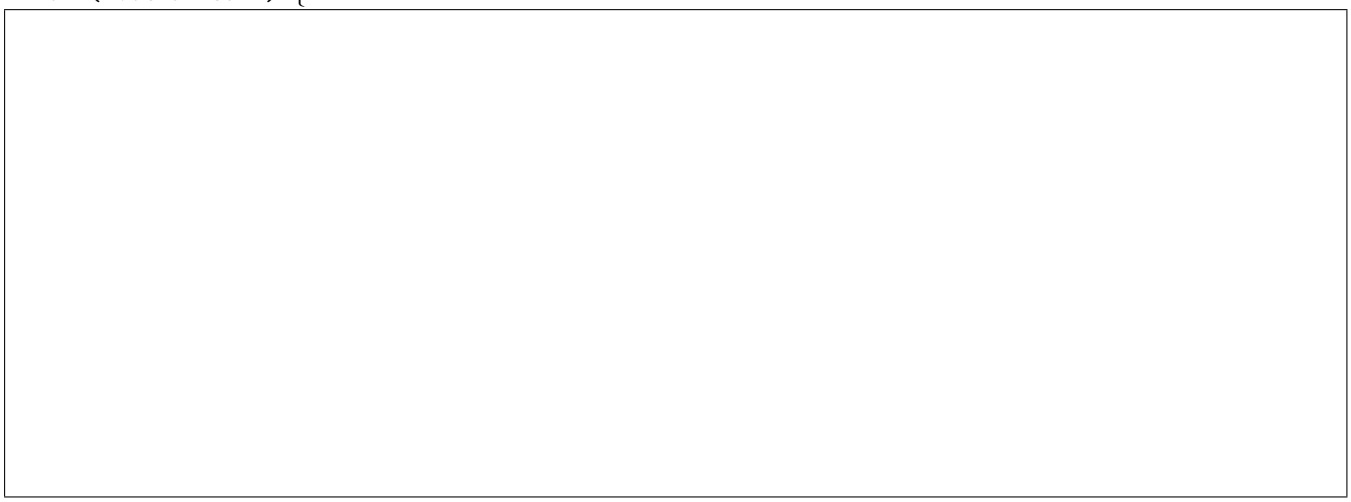


```
  return 0;
}
```

3. Write the function `sum3` that is passed the address of the head node of the list. The function adds all data in the list that is divisible by 3. It should return the final total. **Example:** If the data is: 30, 10, 6, 9, 4 then the function will return 45.

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

int  (node_t *curr) {
```



```
}
```

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

```
int numNodeW1C( bst_node_t *curr)
```

that is passed address 0 or the address of the root node of the BST. It returns the number of nodes that have exactly one child.

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

5. Write the function

```
int num101(int n)
```

where `n` is a 32–bit int. The function counts the number of `101` sequences in the 32 bits of n. It should return the final count. **Example:** there are 3 `101` sequences in the 8 bits `10101011`.