

# CECS 524 Unit 1 Assignment 1

Name: Spuritha Mudireddy

CSULB ID: 030743269

1. Write, in the programming language of your choice, an interpreter for Brainfu\*k. It is very simple, on the order of a late first semester of programming assignment. Did mine in Java, about 60 lines of code. Run it with this input below - what is the output?

```
+++++++[>++++[>+>++++>++++>+<<<<-]>+>+>->+ [<]<-]>>.>---.+++++++..+++.>>.<-.<.  
+++.-----.-----.>>+.>+.
```

## Code:

```
package Unit1;  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
  
public class BrainF {  
    public static void main(String[] args) throws IOException  
    {  
        byte b[]=new byte[30000];  
        int dp=0;  
        BufferedReader rd = new BufferedReader(  
            new InputStreamReader(System.in));  
        String input = rd.readLine();  
        int p=0;  
  
        for(int i=0;i<input.length();i++)  
        {  
            switch(input.charAt(i))  
            {  
                case '>':  
                    dp++;  
                    break;  
  
                case '<':  
                    dp--;
```

```

        break;

    case '+':
        b[dp]++;
        break;

    case '-':
        b[dp]--;
        break;

    case '.':
        System.out.print((char)b[dp]);
        break;

    case ',':
        b[dp]=(byte)(rd.readLine().charAt(0));
        break;

    case '[':
        if (b[dp] == 0)
        {
            i++;
            while (p > 0 || input.charAt(i) != ']')
            {
                if (input.charAt(i) == '[')
                    p++;
                else if (input.charAt(i) == ']')
                    p--;
                i++;
            }
        }

        break;

    case ']':
        if (b[dp] != 0)
        {
            i--;
            while (p > 0 || input.charAt(i) != '[')
            {
                if (input.charAt(i) == ']')
                    p++;
                else if (input.charAt(i) == '[')
                    p--;
                i--;
            }
        }

```

}

$$\}$$

### Output:

