Anton Benfey 3539741
Paul Armstrong 3537294

Power.java:

```
// ********************************************************************
// Power.java
//
// Reads in two integers and uses a recursive power method
// to compute the first raised to the second power.
// ********************************************************************
import java.util.Scanner;

/**
* @author Paul Armstrong
* @author Anton Benfey
*/

public class Power
{
  public static void main(String[] args)
  {
    int base, exp;
    int answer;

    Scanner scan = new Scanner(System.in);

    System.out.print("Welcome to the power program! ");
    System.out.println("Please use integers only.");

    //get base
    System.out.print("Enter the base you would like raised to a power: ");
    base = scan.nextInt();

    //get exponent
    System.out.print("Enter the power you would like it raised to: ");
    exp = scan.nextInt();

    answer = power (base,exp);
    System.out.println(base + " raised to the " + exp + " is " + answer);
  }

  // -----------------------------------------
  // Computes and returns base^exp
  // -----------------------------------------
  public static int power(int base, int exp)
  {
    int pow;
    if (exp == 0){
```

```
            pow = 1;
     } else {
            pow = base * power(base,exp-1);
     }
     return pow;
   }
}
```

Fib.java:

```
// ********************************************************************
// Fib.java
//
// A utility class that provide methods to compute elements of the
// Fibonacci sequence.
// ********************************************************************

/**
* @author Paul Armstrong
* @author Anton Benfey
*/
public class Fib
{
   //-----------------------------------------------------------------
   // Recursively computes fib(n)
   //-----------------------------------------------------------------
   public static int fib1(int n)
   {
      int answer;
      if (n == 0){
            answer = 0;
      }
      else if (n == 1){
            answer = 1;
      } else {
            answer = (fib1(n-1) + fib1(n-2));
      }

     return answer;
   }

   public static int fib2(int n)
   {
     int[] args = new int[n];
     if (n == 0){
```

```
        return 0;
    }

    else if (n == 1){
        return 1;

    } else {
        args[0] = 0;
        args[1] = 1;
        for (int i = 2; i < n; i++){
       args[i] = (args[i-1]+args[i-2]);
        }
    }

    return (args[n-1]+args[n-2]);
  }
}
```

Palindrome.java + output:

```
import java.util.Scanner;
/**
* @author Paul Armstrong
* @author Anton Benfey
*/
public class Palindrome
{
   public static boolean isPal(String s)
   {
      if(s.length() == 0 || s.length() == 1){
         return true;
      }

      if(s.charAt(0) == s.charAt(s.length()-1)){
         return isPal(s.substring(1, s.length()-1));
      }

      return false;
   }

   public static void main(String[]args)
   {
      Scanner sc = new Scanner(System.in);
      System.out.println("Enter a word: ");
      String input = sc.nextLine();
      if(isPal(input))
         System.out.println(input + " is a palindrome");
      else
```

```
            System.out.println(input + " is not a palindrome");
        }
    }
```

```
[abenfey@id415m18 Lab4]$ javac *.java
[abenfey@id415m18 Lab4]$ java Palindrome
Enter a word:
kayak
kayak is a palindrome
[abenfey@id415m18 Lab4]$ java Palindrome
Enter a word:
anton
anton is not a palindrome
[abenfey@id415m18 Lab4]$ java Palindrome
Enter a word:
noon
noon is a palindrome
[abenfey@id415m18 Lab4]$ java Palindrome
Enter a word:
racecar
racecar is a palindrome
[abenfey@id415m18 Lab4]$ java Palindrome
Enter a word:
abcdefghijklmnopqrstuvwxyzzyxwvutsrqponmlkjihgfedcba
abcdefghijklmnopqrstuvwxyzzyxwvutsrqponmlkjihgfedcba is a palindrome
[abenfey@id415m18 Lab4]$
```