1) What does the following code print when called with i=1234, j=5?

```java
public int compute(int i, int j) {
    // base case
    if (i < j) return i;
    // recursion call
    return compute(i - j, j);
}
```

2) What does the recursive code below print when called with value = 10?

```java
public static int compute(int value) {
    // base case
    if (value == 0) return 0;
    // compute term
    int term = value;
    // recursive case
    return term + compute(value - 2);
}
```

3) What does the recursive code below print when called with value = 9?

```java
public static int compute(int value) {
    // base case
    if (value == 0) return 0;
    // compute term
    int term = value;
    // recursive case
    return term + compute(value - 2);
}
```
4) What does the recursive code below print when called with \(d = 100.0\) and \(n = 6\)?

```java
public static double compute(double d, int n) {
    // base case
    if (n == 0) return 0.0;
    // compute term
    double term = d;
    // recursion call
    return term + compute(d / 10.0, n - 1);
}
```

5) What does the recursive code below print when called with \(s = \text{"aabccddeeff"}\)?

```java
public static String munge(String s) {
    if (s == null || s.length() <= 1) // base case
        return s;
    else if (s.charAt(0) == s.charAt(1))
        return munge(s.substring(1, s.length()));
    else
        return s.charAt(0) +
                munge(s.substring(1, s.length()));
}
```

6) What does the recursive code below print when called with \(\text{number} = 13\) and \(\text{base} = 2\)?

```java
public static void convert(int number, int base) {
    int remainder = number % base;
    int quotient = number / base;
    if (quotient > 0) convert(quotient, base);
    System.out.print(remainder);
}
```