

---

CIS 210  
Fall 2011 Final Exam *Key*

Write your name at the bottom of **each page** before you begin. 1 point for each page.

1. [5 points] What does method `whatsit()` print?

```
public static void whatsit() {  
    int x = 18;  
    int y = 2;  
    x = x + y;  
    x = x + x;  
    x = x + y;  
    System.out.println(x);  
}
```

42

2. [8 points] What does method `parity()` print?

```
public static int reduce(int x) {  
    if (x < 2) {  
        return x;  
    }  
    return reduce(x-2);  
}  
  
public static void parity() {  
    int a = 35;  
    int res = reduce(a);  
    System.out.println(res);  
}
```

1

---

3. [10 points] If I run the main program in class MaxMag, what does it print?

```
class MaxMag {

    public static int mag(int i) {
        if (i < 0) {
            i = 0 - i;
        }
        return i;
    }

    public static int maxMag(int[ ] ar) {
        if (ar.length < 1) { return 0; }
        int max = ar[0];
        for (int i=1; i < ar.length; ++i) {
            if ( mag(ar[i]) > mag(max) ) {
                max = ar[i];
            }
        }
        return max;
    }

    public static void main(String[ ] args) {
        int[ ] vals = new int[ ] { -1, 2, -3, 4, -5, 4, -3, 2, -1 };
        int res = maxMag(vals);
        System.out.println(res);
    }
}
```

-5

---

4. [10 points] If I run the main program in class TestBus, what will it print?

```
public class Bus {
    int passengers;
    String color;

    /** Constructor */
    public Bus(String color) {
        passengers = 0;
        this.color = color;
    }

    /** Some passengers enter the bus.
     * @param n Number of passengers getting on the bus
     */
    public void enter(int n) {
        passengers = passengers + n;
    }

    /** Some passengers leave the bus.
     * @param n Number of passengers getting on the bus
     */
    public void leave(int n) {
        passengers = passengers - n;
    }

    /** How many passengers are on the bus?
     * @return The number of passengers on the bus.
     */
    public int aboard() { return passengers; }
}

class TestBus {
    public static void main(String[] args) {
        Bus redBus = new Bus("red");
        Bus blueBus = new Bus("blue");
        redBus.enter(3);
        blueBus.enter(4);
        redBus.leave(2);
        int reds = redBus.aboard();
        System.out.println(reds);
    }
}
```

1

---

5. [10 points] Write a Java static method or a Python function to determine the range of an array of integers. The range of a set of numbers is the difference between the largest and smallest values in that set. For example, if the values are [ -3, -5, 7, -8 ], then the range is 15. If there are less than two numbers, the range is defined to be zero.

I will provide header comments for a Java function, but you may write an equivalent Python function instead. Do not use any built-in or library functions or methods for finding the range, maximum, minimum, etc. Recall that if `ar` is an array in Java, `ar.length` is the number of elements in the array; `len(ar)` is the equivalent Python function. Be sure to make your method or function consistent with the Javadoc header comment.

```
/** Determine the range (max - min) of an array of integers.
 *  @param vals  An array of integers.
 *  @return The range (max - min) of the array.
 */
public static int range(int[ ] vals) {
    if (vals.length < 2) { return 0; }
    int max = vals[0];
    int min = vals[0];
    for (int i=0; i < vals.length; ++i) {
        if (vals[i] > max) { max = vals[i]; }
        if (vals[i] < min) { min = vals[i]; }
    }
    return max - min;
}
```

---

6. [10 points] In the imaginary word game *Words with Pets*, each vowel ('a', 'e', 'i', 'o' or 'u') is worth 2 points, and each consonant (any other letter) is worth 3 points. The score for a word is the sum of the values of all the letters in the word. For example, the word "ant" is worth 7 points, because it has one vowel ('a') and two consonants ('n' and 't'). The word "alien" is worth 12 points ( $3 \text{ vowels} \times 2 \text{ points} + 2 \text{ consonants} \times 3 \text{ points}$ ).

I have provided a Java method that determines whether a character is a vowel. You must provide another Java method that computes the score for a word. You should assume that every character in the word is either a vowel or a consonant. You must use the `isVowel` method to test each letter in the word.

It may be useful to remember that, if `str` is a variable of type `String`, `str.length()` returns the length of the string, and `str.charAt(i)` returns the character at position `i`. Type `String` does not implement the `Iterable` or `Iterator` interfaces.

```
/** Determine whether a character is a vowel.
 * @param ch The character to be tested
 * @return true if ch is a vowel, false if ch is a consonant
 */
static boolean isVowel(char ch) {
    if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' ) {
        return true;
    }
    return false;
}

/**
 * Determine the score of a word in Words with Pets.
 * 2 points for each vowel and 3 for each consonant.
 * @param word The word to be scored.
 * @return The integer score for the word
 */
static int scoreWord(String word) {
    int score = 0;
    for (int i=0; i < word.length(); ++i) {
        char ch = word.charAt(i);
        if ( isVowel(ch) ) {
            score += 2;
        } else {
            score += 3;
        }
    }
    return score;
}
```