Name:_____ UNM Username:_____

Answer all questions in the space provided. Write clearly and legibly, you will not get credit for illegible or incomprehensible answers. Print your name at the top of every page.
This is a closed book exam. However, each student is allowed to bring one page of notes to the exam. Also, you are permitted the use of a "dumb" calculator to perform basic arithmetic.

Question:	1	2	3	4	5	6	7	Total
Points:	20	15	6	9	10	15	10	85
Score:								

1. Select the *single best* answer for each of the following questions.

   (a) What is the keyword used to indicate that a variable cannot be reassigned? (2)
   A. boolean B. constant C. final D. int E. public F. static G. void

   (b) What is the keyword used to indicate that a method does not return a value? (2)
   A. class B. final C. main D. null E. public F. static G. void

   (c) If you want to use an existing class from the Java standard libraries (for example, Scanner), what keyword will you use near the top of your file? (2)
   A. `export` B. `import` C. `include` D. `require` E. `use` F. `using`

   (d) What is the name of the first method run when a Java program starts? (2)
   A. class B. final C. main D. public E. static F. start G. void

   (e) What is the value of the following expression? (2)

```
1 + "2"+ 3 + 4 * 5 + 6
```

   A. `32` D. `"1229"` G. Some other value.
   B. `"123206"` E. `"1241"` H. The value of this expression is undefined.
   C. `"123456"` F. `"1266"` I. This expression would result in a compilation error.

   (f) If you wanted to store the number of students enrolled in CS152 in a variable, which of these types would be best? (2)
   A. `boolean` B. `char` C. `double` D. `float` E. `long` F. `String`

   (g) If you wanted to store the average GPA of students enrolled in CS152 in a variable, which of these types would be best? (2)
   A. `boolean` B. `char` C. `double` D. `int` E. `long` F. `short`

   (h) If you wanted to store the number of people living in a household in a variable while minimizing the amount of memory used, which of these types would be best? (2)
   A. `boolean` B. `double` C. `int` D. `long` E. `short` F. `String`

   (i) Which of the following is a type of loop in Java? (2)
   A. break B. continue C. if D. repeat E. switch F. until G. while

   (j) Which of the following expressions would evaluate to `1.5`? (2)
   A. `3 / 1` B. `(double)(3 / 1)` C. `(double)3 / 1` D. `(int)3.0 / 1` E. `1.0 + 1 / 2`

2. Do the following code snippets compile and run without error? Answer yes or no. If you answered no, explain what is wrong.

(a) (3)

```
public static void foo(int n) {
    for(int i = n; i > 0; i--) {
        if(i == 4) break;
    }
    System.out.println(i);
}
```

(b) (3)

```
public static int square(double n) {
    return n * n;
}
```

(c) (3)

```
public static void foo(int n) {
    int i = n;
    do {
        System.out.println(n + ", " + i);
    } while(i > 1);
    System.out.println(i);
}
```

(d) (3)

```
int[] numbers = int[25];
```

(e) (3)

```
String s = "char";
char c = s.charAt(3);
```

3. Write a single java statement (so only one line of code) that will declare and initialize an array variable named `nums` that can hold 25 floating point values. (6)

4. Consider the following code. Reminder: `System.out.print` will print its argument without going to a new line afterwards.

```
for (int n = 3; n < 10; n += 2) {
    System.out.print(n + " ");
}
```

   (a) What would this loop print? (4)

   (b) Rewrite the code using a while loop instead of a for loop. (5)

5. The following Java program compiles and runs. What is its output? (10)

```
public class BranchTest {

    public static void main(String[] args) {
        int kirk = 7;
        int spock = 10;
        int mccoy = 16;

        switch ( mccoy - kirk ) {
        case 3:
            System.out.println("space");
        case 2:
        case 4:
            System.out.println("the");
            break;
        case 9:
        case 5:
            System.out.println("final");
        default:
            System.out.println("frontier");
        }

        System.out.println("boldly");

        if(mccoy > spock || kirk % 2 == 0) {
            System.out.println("go");
            if(spock > kirk) {
                System.out.println("where");
            } else {
                System.out.println("no");
            }
        } else if( kirk + spock > mccoy ) {
            System.out.println("one");
            if(kirk <= mccoy) {
                System.out.println("has");
            }
            System.out.println("gone");
        } else {
            System.out.println("before");
        }
    }
}
```

6. The following Java program compiles and runs. What is its output? (15)

```
public class ArrayTest {

    public static int foo(int a, int[] vals) {

        int c = a / 6;
        int b = vals[c] % vals.length;

        System.out.println("foo: " + vals[b] + ", a = " + a + ", b = " + b);

        vals[b] += a;

        return c;
    }

    public static void main(String[] args) {

        int a = 12;
        int b = 20;
        int c = -123;
        int[] arr = {8, 4, 9, 11, 10};

        c =  foo(a, arr);
        System.out.println("A: " + a + ", " + c);

        c =  foo(b, arr);
        System.out.println("B: " + b + ", " + c);

        for(int i = 0; i < arr.length; i++) {
            System.out.println("arr[" + i + "] = " + arr[i]);
        }
    }
}
```

7. Write a method that takes a String and returns a new string where every character in the original string is doubled. (10)

   The following examples are to give you a feel for how it works.

   `doubleChar("Hello!")` → `"HHeellllloo!!"`
   `doubleChar("abc")` → `"aabbcc"`
   `doubleChar("")` → `""`

   To do this, copy exactly one line of code from each of the pairs below, place them in the correct order, and add indentation and closing curly braces as needed to make a correct Java method with the desired behavior.

   A. `for(int i = 0; i < orig.length; i++) {`
   B. `for(int i = 0; i < orig.length(); i++) {`

   C. `public static String doubleChar(String orig) {`
   D. `public static char doubleChar(char orig) {`

   E. `char c = orig.charAt(i);`
   F. `char c = s.charAt(i);`

   G. `s = s + c + c;`
   H. `s += c;`

   I. `return c;`
   J. `return s;`

   K. `String s = "";`
   L. `String s = orig;`