CSC 160 FINAL EXAM PRACTICE PROBLEMS AND STUDY GUIDE

GODFREY MUGANDA

1. Practice Problems

You should treat this as closely as possible to a real test and try to do the problems without consulting your notes. Only consult the notes or book, or ask for help when you are stuck. In that case, make sure you understand what you are missing.

1. Write a method

```
void increment(int [] arr)
```

that takes an array of integers in the range $0 \ldots, 9$

that represents the digits in the odometer of a car. For example, an array of size 5 whose values are

$a[0]$	$a[1]$	$a[2]$	$a[3]$	$a[4]$
0	8	9	9	9

represents the mileage on a car with 8999 miles. The `increment` method adds 1 to the "odometer". Thus two successive calls to the `increment` method will set the arrays to

$a[0]$	$a[1]$	$a[2]$	$a[3]$	$a[4]$
0	9	0	0	0

and

$a[0]$	$a[1]$	$a[2]$	$a[3]$	$a[4]$
0	9	0	0	1

Note: You cannot assume that the array has 5 entries. This is just an example.

2. In grade school, (or is it middle school) you learn to add numbers that are expressed in decimal. Here is an example of how to add 5-digit numbers

[0]	[1]	[2]	[3]	[4]
0	8	9	9	9
4	5	7	0	3
5	4	7	0	2

The very first row shows the position of the digits of a number in an array; the second and third rows show the digits that make up the two numbers to add; and the third row shows the sum.

Write a method

```
int [] sum(int [] number1, int [] number2)
```

that takes two arrays that hold the digits of two numbers as parameters. The two arrays are the same size. The method returns a third array of the same size that contains the digits of the sum of the two numbers in the first two arrays.

A carry out of the leftmost digit is just ignored.

3. Write a method

```
void reverse(int [] arr)
```

that rearranges the values in the array so that there are in reverse order. For example, an array with values

```
[12, 45, 67, 32]
```

is rearranged so that it becomes

```
[32, 67, 45, 12]
```

4. Write a method

```
void rotate(int [] arr)
```

that rotates the values in an array one place in a clockwise fashion. This means that every value in the array moves one place to the right, and the rightmost value in the array moves to the leftmost position in the array.

For example, rotating

```
[12, 45, 67, 32]
```

transforms the array into

```
[32, 12, 45, 67]
```

5. Write a method

```
void rotate(int [] arr, int n)
```

that rotates an array n places in a clockwise version.

6. Suppose that a file called `"input.txt"` contains an integer n, followed by n strings, with each string on its own line. Write a file that opens the file, reads its contents, and then writes an output file `"output.txt"` that contains the exact same information, except the strings are in reverse order. For example, if the input file contains

```
5
Reagan
Bush
Clinton
Bush
Obama
```

then the output file will contain

```
5
Obama
Bush
Clinton
```

```
Bush
Reagan
```

7. Solve the previous problem if it is possible to have more than one string on the same line, but all strings are single words separated by at least one space or by a line break. Thus the file might contain

```
5 Reagan Bush Clinton Bush
Obama
```

8. A prime number is a positive integer greater than 1 that has no divisors other than 1 and itself.

Write a method

```
boolean isPrime(int number)
```

that returns true if the given number passed as parameter is prime, and returns false otherwise.

9. Write a method

```
boolean contains (int [ ] arr, int x)
```

which returns true if the parameter `x` is in the given array, and false otherwise.

10. Write a method

```
boolean containsAtLeastOne(int [] arr, int [] b)
```

that returns true if the array `arr` contains at least one element of the array `b`, and false otherwise. You may use your solution to problem 9.

11. Write a method Write a method

```
boolean containsAll(int [] arr, int [] b)
```

that returns true if the array `arr` contains every element of the array `b`, and false otherwise. You may use your solution to problem 9.

2. Concepts

Additional concepts include sorting, linear search, binary search, recursion, base case for recursion.

The final exam will be comprehensive, so consult all other study guides.