**44-542 Object Oriented Programming**

**Arrays**

1. Consider the array a, with the elements shown below. Each element is a string that represents the name of a horse or dog.



* 1. Write the statements necessary to create an array named a that will contain 14 references to String objects.
  2. Assume that we have initialized the array entries, so that the entries point to the String objects shown above. Write a statement that will change the array entries, so that the element indexed by 13 refers to the string “Lydia”.
  3. Write a for loop that will print out each String referenced by a, each on a separate line, and in the same order as they are stored in the array. Use the length field to determine the number of iterations for your loop (rather than just explicitly writing the number 14). Use the traditional form of the for loop, including the loop counter initialization, the controlling condition, and the incrementing/decrementing expression.
  4. Rewrite the code in the previous step, using an enhanced for loop.

1. In this problem, you will create an array that stores int values and then process the array in various ways.
   1. Write the statements necessary to create an array named numbers that contains 20 ints.
   2. Write a for loop that assigns values to each of the entries in numbers. Each entry is calculated by multiplying the index by 3 and then adding 1. Your code should work even if we alter the length of the array. Use a traditional for loop.
   3. Write an enhanced for loop to print the values stored in the array. All values should be printed on the same line with one space between them.
   4. Draw a picture of the array numbers, similar to the one shown on the first page of this worksheet. Remember that ints are primitive data types, not objects. They are stored directly in the array itself.
2. Write a complete Java program that does the following. Note: When using a traditional for loop and traversing the entire array, use the length field to determine your controlling condition, rather than explicitly writing the numerical value of the length of the array.
   1. Use the Scanner class to read 11 integer values from the file arrayData.txt and store them in an array of ints of length 11 named numArray. Contents of arrayData.txt are shown at the end of this document. Note that the file contains 11 integer values followed by 16 mammal names. The mammal names will be used later. Here is how your code should begin:

import java.io.\*;

import java.util.\*;

public class Worksheet16

{

public static void main(String[] args) throws IOException

{

Scanner myFileReader = new Scanner(

new File("arrayData.txt"));

* 1. Print the contents of numArray in reverse order, with one space between each pair of values printed.
  2. Print each of the negative values in numArray, with one space between each pair of values printed.
  3. Print each of the values in numArray that have an even index, with one space between each pair of values printed.
  4. Use the Scanner instance myFileReader that you created previously to read the 16 mammals from the file arrayData.txt and store the names in an array of Strings of length 16 named mammalArray.
  5. Print out the names of the mammals, with each name on a separate line.
  6. Print out the names of the mammals for all mammals that start with the letter H – Z, with each name on a separate line.

Desired output for this program is shown at the end of this document.

Contents of arrayData.txt

8

-3

4

-6

12

4

-8

12

2

13

-1

Horse

Cat

Ferret

Dog

Whale

Rabbit

Monkey

Panther

Antelope

Giraffe

Zebra

Rhinoceros

Camel

Dolphin

Goat

Sheep

Desired output

-1 13 2 12 -8 4 12 -6 4 -3 8

-3 -6 -8 -1

8 4 12 -8 2 -1

Horse

Cat

Ferret

Dog

Whale

Rabbit

Monkey

Panther

Antelope

Giraffe

Zebra

Rhinoceros

Camel

Dolphin

Goat

Sheep

Horse

Whale

Rabbit

Monkey

Panther

Zebra

Rhinoceros

Sheep