BİLGE KÖROĞLU

Lecture Notes (May 2 – 4, 2007)

METHOD DECOMPOSITION and ARRAYS

Method Decomposition:

Every method should be understandable therefore sometimes wee need to decompose some methods into smaller methods as needed for clarity. As an example, let's look an example that requires translating English to Pig Latin.

Pig Latin is a language that moving the initial sounds of the word to the end and adding "ay" modifies each word.

Book ⟶ ookbay item ⟶ itemyay chair ⟶ airchay

In this program, our primary objective is writing method that can translate the words but as we can guess that the method will be too complicated therefore we need to decompose the method into smaller methods.

```
//*********************************************************************
//  PigLatin.java       Author: Bilge Koroglu
//
//  Demonstrates the concept of method decomposition.
//*********************************************************************

import java.util.Scanner;

public class PigLatin
{
  //-----------------------------------------------------------------
  //  Reads sentences and translates them into Pig Latin.
  //-----------------------------------------------------------------
```

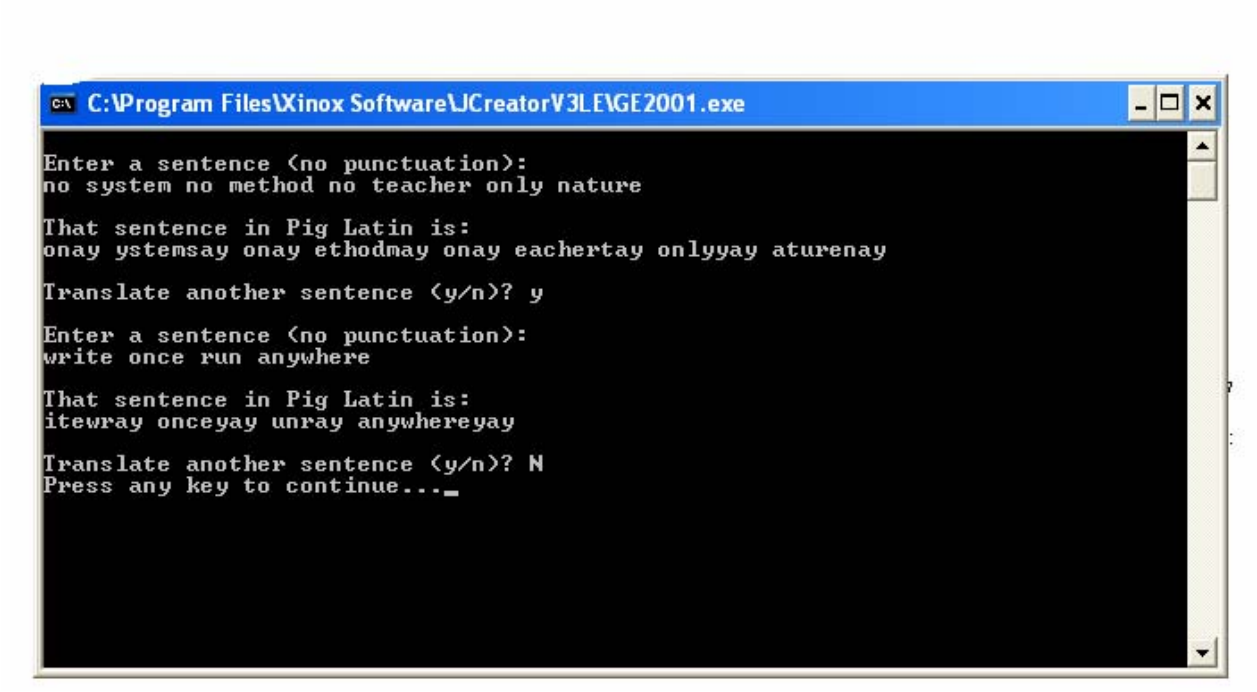
```
  public static void main (String[] args)
  {
    String sentence, result, another;

    Scanner scan = new Scanner (System.in);

    do
    {
      System.out.println ();
      System.out.println ("Enter a sentence (no punctuation):");
      sentence = scan.nextLine();

      System.out.println ();
      result = PigLatinTranslator.translate (sentence);
      System.out.println ("That sentence in Pig Latin is:");
      System.out.println (result);

      System.out.println ();
      System.out.print ("Translate another sentence (y/n)? ");
      another = scan.nextLine();
    }
    while (another.equalsIgnoreCase("y"));
  }
}
```

```
//********************************************************************
//  PigLatinTranslator.java       Author: Bilge Koroglu
//
//  Represents a translator from English to Pig Latin.
 // Demonstrates method decomposition.
  //********************************************************************

import java.util.Scanner;

public class PigLatinTranslator
{
  //-----------------------------------------------------------------
  //  Translates a sentence of words into Pig Latin.
  //-----------------------------------------------------------------
  public static String translate (String sentence)
  {
    String result = "";

    Scanner scan = new Scanner (sentence);

    sentence = sentence.toLowerCase();

    while (scan.hasNext())
    {
      result += translateWord (scan.next());
      result += " ";
    }

    return result;
  }
```

```
//-----------------------------------------------------------------
//  Translates one word into Pig Latin. If the word begins with a
//  vowel, the suffix "yay" is appended to the word.  Otherwise,
//  the first letter or two are moved to the end of the word,
//  and "ay" is appended.
//-----------------------------------------------------------------
private static String translateWord (String word)
{
  String result = "";

  if (beginsWithVowel(word))
    result = word + "yay";
  else
    if (beginsWithBlend(word))
      result = word.substring(2) + word.substring(0,2) + "ay";
    else
      result = word.substring(1) + word.charAt(0) + "ay";

  return result;
}

//-----------------------------------------------------------------
//  Determines if the specified word begins with a vowel.
//-----------------------------------------------------------------
private static boolean beginsWithVowel (String word)
{
  String vowels = "aeiou";

  char letter = word.charAt(0);

  return (vowels.indexOf(letter) != -1);
```

```
  }

  //-----------------------------------------------------------------
  //  Determines if the specified word begins with a particular
  //  two-character consonant blend.
  //-----------------------------------------------------------------
  private static boolean beginsWithBlend (String word)
  {
    return ( word.startsWith ("bl") || word.startsWith ("sc") ||
             word.startsWith ("br") || word.startsWith ("sh") ||
             word.startsWith ("ch") || word.startsWith ("sk") ||
             word.startsWith ("cl") || word.startsWith ("sl") ||
             word.startsWith ("cr") || word.startsWith ("sn") ||
             word.startsWith ("dr") || word.startsWith ("sm") ||
             word.startsWith ("dw") || word.startsWith ("sp") ||
             word.startsWith ("fl") || word.startsWith ("sq") ||
             word.startsWith ("fr") || word.startsWith ("st") ||
             word.startsWith ("gl") || word.startsWith ("sw") ||
             word.startsWith ("gr") || word.startsWith ("th") ||
             word.startsWith ("kl") || word.startsWith ("tr") ||
             word.startsWith ("ph") || word.startsWith ("tw") ||
             word.startsWith ("pl") || word.startsWith ("wh") ||
             word.startsWith ("pr") || word.startsWith ("wr") );
  }
}
```

Output of the Program:



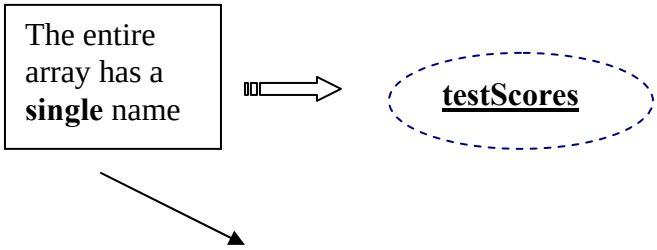
The *translate* method uses another method called *translateWord*.

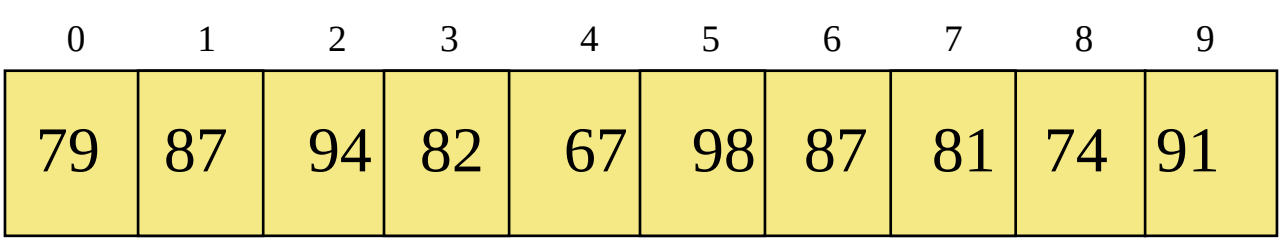
TranslateWord makes use of another private methods *beginsWithVowel* and *beginsWithBlend.* These methods are written as private because these methods are used only to support another method in the same class. This is an good example of method decomposition.

Arrays

Arrays are objects that help us to organize large amounts of information.

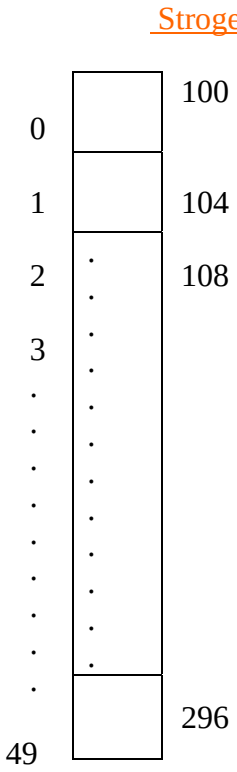
Declaring and Using Arrays:





int[] testScore = new int[50];

We store integers in the array. Consider each integer is 4 byte therefore its storage is like that:



Assume that we want to calculate exam results' average by using array's elements.

```
public static float avgScore(int[] testScore)
{
        float sum = 0.0;
        int i;
        for(i =0; i<testScore.length;i++)
                sum +=testScore[i];
        return(sum);
}
```

How to initialize the contents of the array?

```
int inputScore = 0;

for(int i =0;i<testScore;i++)
{
        System.out.println("Enter your score its number is "+(i+1));
        inputScore = scan.nextInt( );
        testScore[i] = inputScore;
}
```

An array can be set up to hold any primitive type or any class type that programmer create.For example we can store String objects like this:

```
        String myStrings = new String[5];
```

Questions

1. In the array declaration

```
String[] wordBag = new String[5];
```

What is

a. the array name?
b. the type of the array?
c. the length of the array?
d. the range of values an index accessing this array can have?
e. the first and the last index of array?

2. What is the output of the following code?

```
char[] letterBag = {  'a' , 'b' , 'c' }
for(int index = 0; index<letterBag.length; i++)
        System.out.print(letter[index] + ",");
```

3. What is the wrong with the following piece of code?

```
        int[] arr1 = new int[10];
                for(int i=1; i<=arr1.length; i++)
                        arr1[i] = 3*i;
```

4. Consider the following class definition:

```
public class Calculation
{
        public static void findSth(int n)
        {
                //Some code goes in here.
        }

        //The rest of the class is not important for the question.
}
```

Which of the following are acceptable method calls?

```
int[] a = {4, 5, 6};
int number = 2;

        Calculation.findSth(number);
        Calculation.findSth(a[2]);
        Calculation.findSth(a[3]);
        Calculation.findSth(a[number]);
        Calculation.findSth(a);
```

Note: Don't think the meanings of the method's and class's name.

5. Write a method definition for a *static void* method called *oneMore* , which has formal parameter for an array of integers and *increases* the value of each array elements *by one.*

6. What is the method decomposition?

7. How do you explain that in *PigLatinTranslator* class' some methods are written as private? Try to explain by considering *Method Decomposition* principle in Java?

8. When do we need to use an array?

9. What is the wrong the following method definition? It will compile but does

not work.

```
public static void tripleSize(int[] a)
{
        a= new int[a.length*3];
}
```

10. Imagine that you are supposed to get numbers of integers and then sort them in ascending order. Write a class which has main method and in the main method get numbers from the user, store them in an array then by using array, sort them in ascending order.

Answers

1. a. Array name is “wordBag”.
   b. Type of the array is String.
   c.The length of the array is 6.
   d. 0 through 4 inclusive.
   e.The first index is 0, the last index is 4.

2. The output of the code is “a, b, c”.

3. The for loop uses indices 1 through arr1.length, but the correct indices are 0 through arr1.length-1. The last index, arr1.length, is out of bounds. Correct one is the following :

```
int[] arr1 = new int[10];
        for(int i=0;i<arr1.length;i++)
                arr1[i] = 3*i;
```

4. Calculation.findSth(number);//legal
   Calculation.findSth(a[2]);//legal
   Calculation.findSth(a[3]);//illegal (index out of bounds)
   Calculation.findSth(a[number]);//legal
   Calculation.findSth(a);//illegal (the method's parameter isn't an array.

5. public static void oneMore(int[] a)

```
{
        for(int i=0;i<a.length;i++)
            a[i]= a[i]+1;
}
```

6. Method Decomposition is the process of dividing a complex method into several support methods. By this way, the design of the program and undestandablity of the program could be easy.

7. In *Method Decomposition*, some methods are used in another methods and these methods are written only support another's in the same class therefore they needn't to be used in another classes so they are written as *private.*

8. Sometimes we need to manage number a large amount of information. Instead of declaring and instantiating lots of variables, we can create an array, which is an object, to manage easily all of the information. Note that elemnts of the array should be the same type.

9. When this method is invoked, it has no effect on it's argument because the parameter is a local variable that contains a reference, it changed but since it is an reference, it goes away and actual parameter cannot be changed when the method ends.

10.

```
import java.util.Scanner;
public class deneme
{
    public static void main(String args[])
    {
        Scanner scan= new Scanner(System.in);

        int a;
```

```
		System.out.println("How many elements dou you want to create in
your array?");
		a=scan.nextInt();
		double[] numbers = new double[a];
		for(int i=0;i<a;i++)
		{
			System.out.println("Enter your real number");
			numbers[i]=scan.nextDouble();
		}

		for(int k=0;k<a;k++)
		{
			for(int i=k+1;i<a;i++)
			{
				if(numbers[i]>numbers[k])
				{
					double b = numbers[k];
					numbers[k]=numbers[i];
					numbers[i]=b;
				}
			}
		}
		for(int l=0;l<a;l++)
		{
			System.out.println(numbers[l]);
		}

	}
}
```