



## Lab 10 Blood Pressure Monitoring Part 2<sup>1</sup>

Due: Friday, November 11, 2011 at 23:00

*"A Balanced diet is a cookie in each hand."*

### 1. Outcomes

1. Perform a simple maintenance activity on an existing Java program.
2. Construct software which utilizes array List to store a set of data
3. Iterate over a List in a forward and backward direction.

### 2. Introduction

This lab involves a simple change to an existing Java program (or hopefully simple). You will be changing the parameters slightly for the program, and ending up with hopefully a better program in the end.

In essence, you will be modifying your program to use an array list to store blood pressure readings, modifying your code so a variable number of readings can be made, and fixing an outstanding defect with your previous program. Extra credit will also be available.

### 3. Assignment

#### 3.1. Step 1: Fix something that was broken

The first step to this assignment is to fix the operation of the hypertensive crisis. A Hypertensive crisis should be declared whenever a single blood pressure reading occurs that meets the clinical definition for a hypertensive crisis.

#### 3.2. Step 2: Change to an array list

The second step to this lab is to modify your code so that you use an array list instead of an array. This should be limited to making approximately 4 - 5 changes in your source code, though it may be more difficult.

#### 3.3. Step 3: Modifying the number of entries

The third step is to modify your code so that the number of blood pressure readings can be changed. The previous lab was fixed at 5. Now we want to be able to change that. This will

---

<sup>1</sup> Or Healthy Living 102



involve prompting the user for the number of readings and then obtaining that number from the user.

### **3.4. Forward or backward printing**

This step involves modifying your code so that the blood pressure entries are either printed in a forward or backward direction. The user will be prompted which way (forward or reverse) they would like to see readings displayed, and the program will print them in that order when requested.

### **3.5. Extra Credit: Reading from a non-keyboard stream**

If you complete the first three steps and would like to complete another step, the course website has a class which allows you to obtain an instance of a scanner. In particular, this scanner allows, in a very poor way, one to read from a file.

For extra credit, you can modify your code such that you prompt the user for a person's name, namely sally, bob, or tim. This prompting is then passed into the provided code, which returns an instance of a scanner. This scanner should replace your calls to the keyboard scanner, allowing data to be read from an external location.

## **4. Deliverables**

1. Name, date, title, and course information.
2. Samples of the program executing showing the output written to the console.
3. A short description of what went wrong and what went right during the lab. (One sentence each for what went right and what went wrong, unless more than this applies.)
4. One sentence describing something that went wrong with your first implementation, making this lab difficult.
5. A description of what you learned from this lab.

Your source code should also be submitted in a zip archive.