

Appendix 1

Computer Science

Section 1: Downloading and Installing JDK and NetBeans.

Installing JDK8

1. Go to <http://www.oracle.com/technetwork/java/javase/downloads/index.html> , download and install Java development kit 8 (JDK 8).

Installing NetBeans 8

1. Go to <https://netbeans.org/> and download and install **NetBeans IDE 8.x** (where x is any number).
2. You will use the IDE to write Java code therefore download and install **Java SE**.

Support

Section one: Getting Started with Java can help you if you get stuck.

<http://www.homeandlearn.co.uk/java/java.html>

Section 2: Test and develop simple programs

Task 1: Computing Body Mass Index

Body Mass Index (BMI) is a measure of health based on height and weight. It can be calculated by taking your weight in kilograms and dividing it by the square of your height in metres.

$$BMI = \frac{\text{weight (kg)}}{(\text{height})^2 (m)^2}$$

The interpretation of BMI for people 20 years or older is as follows.

BMI	Interpretation
Below 18.5	Underweight
18.5 – 24.9	Normal
25.0 – 29.9	Overweight
Above 30.0	Obese

The following program will prompt the user to enter their weight in pounds and height in inches and displays the BMI of the user.

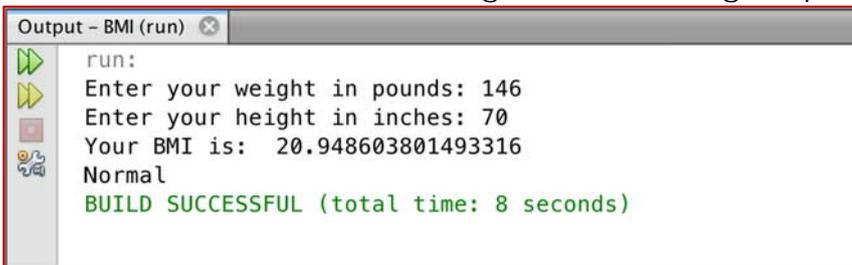
1 pound = 0.45359237 kg

1 inch = 0.0254 meter

```
1 package bmi;
2
3 import java.util.Scanner;
4
5 public class BMI {
6
7     public static void main(String[] args) {
8         Scanner input = new Scanner(System.in);
9
10        // Prompt the user to enter their weight in pounds
11        System.out.print("Enter your weight in pounds: ");
12        double weight = input.nextDouble();
13
14        //Prompt the user to enter their height in inches
15        System.out.print("Enter your height in inches: ");
16        double height = input.nextDouble();
17
18        final double KG_PER_POUNDS = 0.45359237; // Constant
19        final double M_PER_INCH = 0.0254; // Constant
20
21        // Compute BMI
22        double weightInKg = weight * KG_PER_POUNDS;
23        double heightInMetres = height * M_PER_INCH;
24        double bmi = weightInKg / (heightInMetres * heightInMetres);
25
26        // Display result
27        System.out.println("Your BMI is: " + bmi);
28        if (bmi < 18.5)
29            System.out.println("Underweight");
30        else if (bmi < 25)
31            System.out.println("Normal");
32        else if (bmi < 30)
33            System.out.println("Overweight");
34        else
35            System.out.println("Obese");
36    }
37 }
```

Output

Enter the information below to get the following output.



```
Output - BMI (run) x
run:
Enter your weight in pounds: 146
Enter your height in inches: 70
Your BMI is: 20.948603801493316
Normal
BUILD SUCCESSFUL (total time: 8 seconds)
```

Write a narration to explain how the program work

Your narration should be detailed enough to give a novice a good understanding of what the program is doing.

Task 2: Lottery Number Generator

Design and create a program to randomly generate lottery numbers.

It must generate six random numbers between 1 and 49 inclusive + one for the bonus ball, note that the numbers should not repeat in any draw.

Write a narration to explain how the program work

Your narration should be detailed enough to give a novice a good understanding of what the program is doing.

Section 3: Design, code and document a program

Task 1: Opposites

Your teacher asks you to develop a program that will help her KS2 students to practice opposite words for their examination. The program should randomly select two different pairs of words from the lists below and display a question based on the selection.

Word lists

[hot, summer, hard, dry, simple, light, weak, male, sad, win, small, ignore, buy, succeed, reject, prevent, exclude]

[cold, winter, soft, wet, complex, darkness, strong, female, happy, lose, big, pay attention, sell, fail, accept, allow, include]

For example if **hot** and **weak** are selected then the question displayed is
“**hot** is to cold as **weak** is to _____?”

When the user types their answer the program should display whether the user is correct or not.

The program should start by asking the user for their name. The program should then display ten random questions. After displaying 10 questions the program should display the user's name and their final score out of 10.

This is the algorithm for task 1, you are required to use this algorithm and write the Java program.

```
score ← 0
word_list1 ← [hot, summer, hard, dry, heavy, light, weak, male, sad, win, small, ignore, buy,
              succeed, reject, prevent, exclude]

word_list2 ← [cold, winter, soft, wet, light, darkness, strong, female, happy, lose, big, pay
              attention, sell,
              fail, accept, allow, include]

PROCEDURE make_question(number1, number2)
    OUTPUT word_list1[number1], "is to", word_list2[number1], "as",
           word_list1[number2], "is to _____?"
END PROCEDURE

INPUT name
FOR index FROM 1 TO 10
```

```
pick1 ← random(0, LEN(word_list1) - 1) #LEN returns the length of the list
pick2 ← random(0, LEN(word_list2) - 1) #LEN returns the length of the list
{random(a,b) generates a random number between a and b inclusive}
```

```
WHILE pick2 = pick1
    pick2 ← random(0, LEN(word_list2) - 1)
END WHILE
```

```
{Now make a question}
make_question(pick1, pick2)
correct_answer ← word_list2[pick2]
INPUT user_answer
IF user_answer = correct_answer THEN
    score ← score + 1
    OUTPUT "Correct answer"
ELSE
    OUTPUT "Wrong answer"
END IF
NEXT index
```

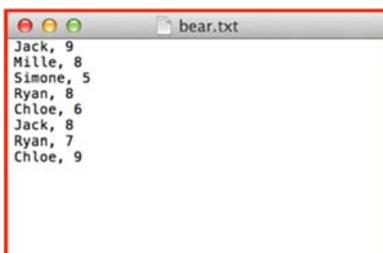
```
END FOR
OUTPUT name, "you got", score, "out of 10"
```

Task 2

The teacher teaches three different classes (bee, bear, duck) and wants to use this test as an assessment. Before this test can be used as an assessment she has to ensure that no question is repeated, in the same test. For example if the following should not be allowed in the same test.

hot is to cold as light is to _____ ?
winter is to summer as hot is to
_____ ?

The teacher would also like to store the result into a text file so that she can analyse the data at a later stage. The figure below shows an example of a part of the file for bear.



You are required to:

1. Write the algorithm that will solve this problem
2. Write the Java code that will solve the problem
3. Write an evaluation of your solution