

## Assignment Set 1

*NOTE: The naming convention for each of the C code must follow the template below:*

**<serial number>\_<name>\_Asgn<Assignment number>\_Q<question number>.c**

*The serial number can be found in the excel sheet shared with you already. For example, Sahil Joshi with serial number 36 will name the C file for Q3 as: **36\_sahil\_Asgn1\_Q3.c***

1. **Wind chill.** Given the temperature  $t$  (in Fahrenheit) and the wind speed  $v$  (in miles per hour), the wind chill is defined as:

$$w = 35.74 + 0.6215 t + (0.4275 t - 35.75) v^{0.16}$$

Write a C program that reads two double command-line arguments  $t$  and  $v$  and prints the wind chill.

You may use the `pow( )` function to compute  $v^{0.16}$ . For this you must include `math.h`.

*File name: <serial\_number>\_<name>\_Asgn1\_Q1.c*

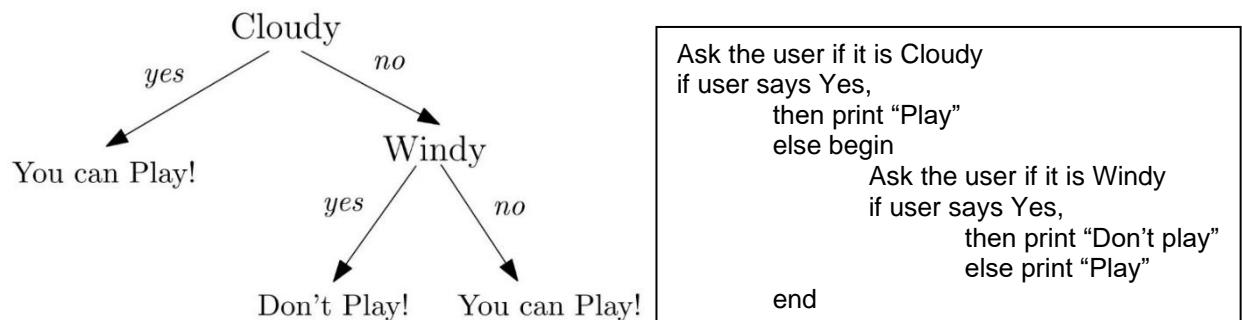
2. **Bounding Circle.** Write a C program, which reads the radius and coordinates of the centre for two circles. The program prints the radius and coordinates of the centre for the smallest circle that encloses the two circles.

*File name: <serial\_number>\_<name>\_Asgn1\_Q2.c*

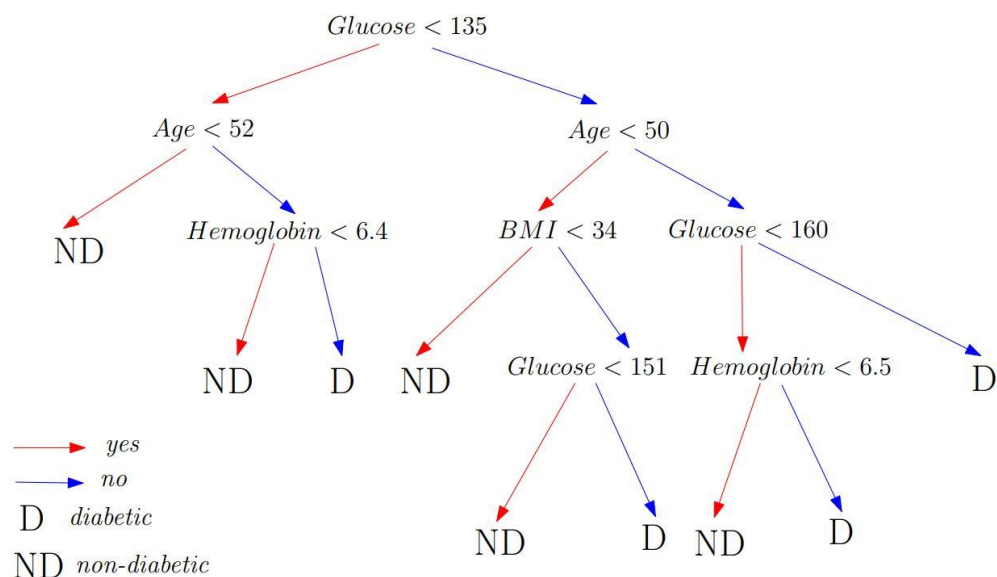
3. **Checksums.** The International Standard Book Number (ISBN) is a 10-digit code that uniquely specifies a book. The rightmost digit is a *checksum* digit which can be uniquely determined from the other 9 digits from the condition that  $d_1 + 2d_2 + 3d_3 + \dots + 10d_{10}$  must be a multiple of 11 (here  $d_i$  denotes the  $i$ th digit from the right). The checksum digit  $d_1$  can be any value from 0 to 10: the ISBN convention is to use the value X to denote 10. *Example:* the checksum digit corresponding to 020131452 is 5 since it is the only value of  $d_1$  between 0 and 10 for which  $d_1 + 2*2 + 3*5 + 4*4 + 5*1 + 6*3 + 7*1 + 8*0 + 9*2 + 10*0$  is a multiple of 11. Write a program, that reads a 9-digit integer, computes the checksum, and prints out the fully formatted 10-digit ISBN number, for example in the format: 0-201-31452-5.

*File name: <serial\_number>\_<name>\_Asgn1\_Q3.c*

4. **Implementing Decision Trees.** Decision trees are widely used to depict the process of decision making. For example, the following decision tree shows when you can play depending on weather artefacts. Also shown is a pseudocode which reads the artefacts and computes the decision.



Note that the user is asked whether it is Windy only when it is not Cloudy. You are required to write a program that implements the following decision tree:



This decision tree decides whether a person is diabetic (D) or non-diabetic (ND) based on several measurements. It starts by asking the user his / her glucose level and the age (observe that in the decision tree, these two measurements are always needed). Depending on the answers, it may reach a decision, or ask for the body mass index (BMI) or Haemoglobin level. These additional questions must therefore be nested in the appropriate branches of the if-then-else statement for the decision tree. In all cases, your program must convey the diagnosis, namely diabetic or non-diabetic to the user.

File name: <serial\_number>\_<name>\_Asgn1\_Q4.c