Lab 16 Functions and Testing

Introduction to Computer Science I

# Objectives:

After performing this lab, the students should be able to

* write functions that perform base conversion
* devise and perform testing, including selecting appropriate test cases

# Names of Lab Group Members:

**Base conversion** involves converting a number in one base to its equivalence in another base. The base is we are most familiar with is base 10 (decimal). In digital systems, base 2 (binary) and base 16 (hexadecimal) are commonly used.

To convert a non-decimal number (base other than 10) to decimal (base 10), we multiple each digit by the base raised to the digit’s position and then add all the products. For example, given the binary number 11012(base 2), to convert it to decimal, we do the following:

1 x 23 + 1 x 22 + 0 x 21 + 1 x 20 = 8 + 4 + 0 + 1 = 13

The position of the rightmost digit is 0, and therefore the digit 1 is multiplied by 20. The next position on the left is 1, and therefore the digital 0 is multiplied by 21. Each digit is multiplied by its corresponding power of the base, and the sum of all the products is the result of the conversion.

To convert a decimal number to a non-decimal base, successive integer divisions by the non-decimal base are performed. For example, to convert the decimal number 13 to binary (base 2), we keep dividing the number by 2 until the quotient is 0 and using the reminders from the divisions to form the converted number:

13 =

11012

13 / 2 = 6 🡪 remainder = 1

6 / 2 = 3 🡪 remainder = 0

3 / 2 = 1 🡪 remainder = 1

1 / 2 = 0 🡪 remainder = 1

The first remainder becomes the rightmost digit of the converted number, and the rest of the remainders are appended in order to its left.

## Activity

##### Part A

Write the *pseudocode* for the function ConvertToDecimal that given a number in a given base converts the number to decimal. Assume base is no more than 10.

int ConvertToDecimal(int number, int base)

Take the Original number, divide by the basenumber, the remainder becomes the rightmost number. The number of times you can divide by the base becomes the next numerator.

Repeat these steps until the result of dividing is 0

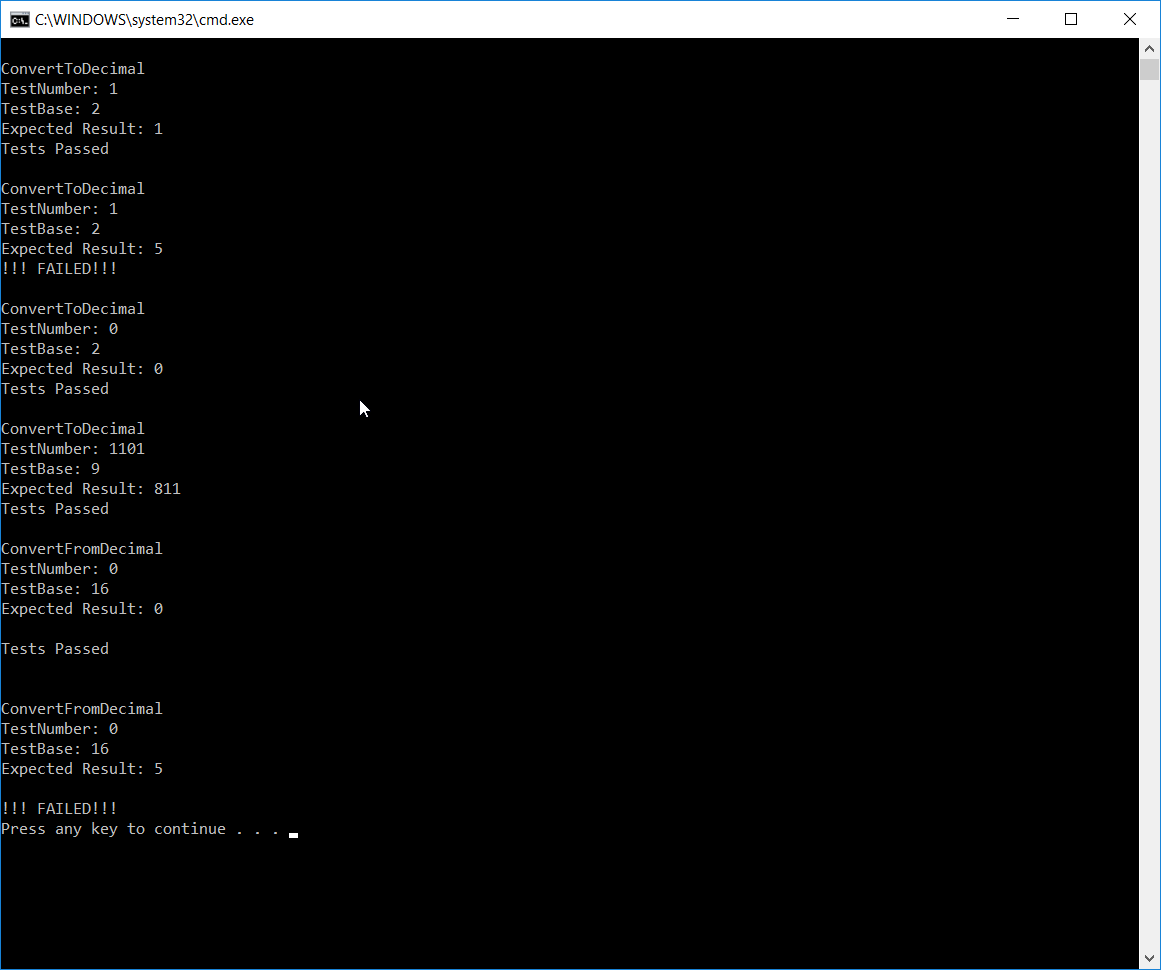
Write the *pseudocode* for the function ConvertFromDecimal that converts a decimal number to a desired base. Assume base is no more than 10.

int ConvertFromDecimal(int decimalNumber, int base)

Take DecimalNumber divide by base. Store this number.

Take DecimalNumber get the mod of base to get remainer.

Add both these numbers, this becomes the new remainder. This becomes the converted Number.



##### Part B

Provide your C++ source code and screenshots of your program outputs.

Write a program that implements the two functions described above in part A. Perform unit testing on each function. Describe the procedures that you took to test the two functions. In the table below, document the test cases that you have chosen for testing this program.

**Describe Testing Procedures:**

**Test Cases:**

ConvertToDecimal

|  |  |  |
| --- | --- | --- |
| Test Input | Expected Output | Reason for choosing this test input |
| 1 2 | 1 Pass | Simple test |
| 1 2 | 5 Fail | Control test |
| 0 2 | 0 Pass |  |
|  |  |  |
|  |  |  |

ConvertFromDecimal

|  |  |  |
| --- | --- | --- |
| Test Input | Expected Output | Reason for choosing this test input |
| 0 16 | 0 Pass | Simple test |
| 0 16 | 5 Fail | Control test |
|  |  |  |
|  |  |  |
|  |  |  |