שנקר - בי"ס גבוה להנדסה ולעיצוב
רחוב אנה פרנק 12, רמת גן 52526
טל.: 03-6110000 פקס: 03-7521141
דוא"ל: info@shenkar.ac.il

Software Engineering Department

Computer Organisation and Programming Course final assignment

Pocket Calculator application

Written by:

Yuval Berghaus 313247116

05 September 2021

Lecturer: Dr Yigal Hoffner

TABLE OF CONTENTS

1 Pocket Calculator application design

You write the document in English! It is for your own good that you practice using English as much as you can.

If you add or remove sections in the document - Generating the **Table of Contents** is achieved by going to the table of contents on page 1, clicking with the right mouse button, and selecting 'Update Field'.

If you have nothing to say in a speicif section – you can erase the section!

1.1 Major Design/Implementation Decisions

The following major design decisions were made during the design and implementation phases:

- Integers are represented internally by 2 words: one word for the absolute value and the other word specifies the sign of the value (i.e. positive or negative)
- All parameters to the subroutines are passed one OR it is used because they are global variables
- The division was made in "long division" technique
- Minus_flagLeft and minusFlagRight are very important components since they represent each of the operands sign
- I made each of the validations
- The multiplication algorithm was made according to Dr Hoffner's class

1.2 The High-level Algorithms

```
Int main() {
    Printf("Op available: + - * /. Press X to exit\nOp:");
    TOperator = GetOperator();
    Printf("Num:");
    Num = GetSignedInt();
    Printf("\nNum2:");
    Num2 = GetSignedInt();
    FI();                              //This will calculate the result accordingly
 }

short signed int GetSignedInt() {
    minus_Flag = OFF;
    Tnum = 0;
    cc = getc();
    IF (cc == "-")
    THEN minus_Flag = ON;
    cc = getc(); // Because of this line, there is...!
    FI
    WHILE (CC != CR) // ...no need to change the algorithm that converts the input!
        DO
            Tnum = Tnum *10 + (cc-30h);
            cc = getc();
        OD
        IF (minus_Flag == ON)
        THEN Num = -Num;
        FI
}
```

```
putSignedInt(Number) {
        IF (Number < 0)
                THEN Stripped = -Number;
                        Putc("-");                      //output the – sign to the screen
        ELSE  Stripped = Number;
        FI
        For(Count = It_count ; Count >= 0 ; Count--)
        DO
                Digit = 0;
                While(Stripped-10**Count > 0)
                DO
                        Stripped = Stripped – 10**Count;
                        digit++;
                OD
                Output(digit);
        OD;
}
FI() {

}
```

2 The User Guide

The **user guide** is intended to enable me to run the application without you being next to me.

It should contain 2 things (at least):

1. The first thing is to input an operator in the keyboard and wait for a response.
2. After you have input an operator you will be requested to input Num1 and press Enter (carriage return)
3. Later, “Num2:” will appear on the screen so you need to input the second number and press carriage return.
4. Once you clicked carriage return (Enter) than the program will calculate according to the operator you typed and will print your result corresponding to your request