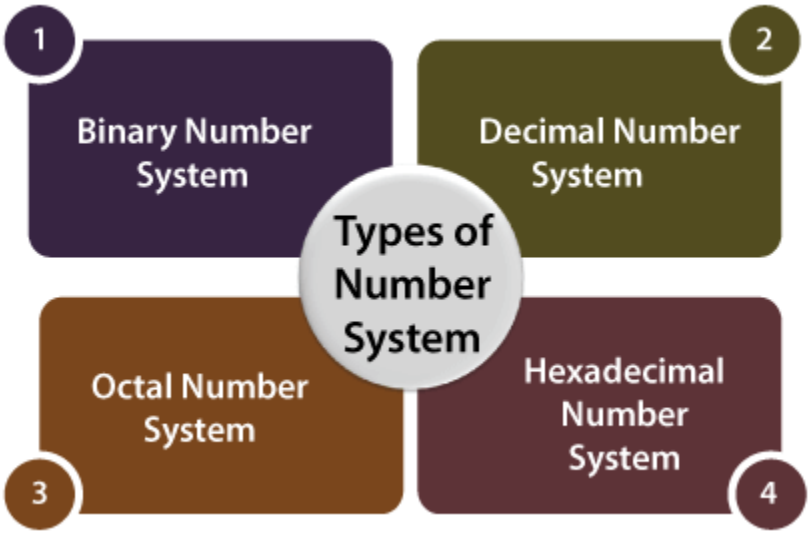
PROJECT REPORT

GROUP MEMBERS :

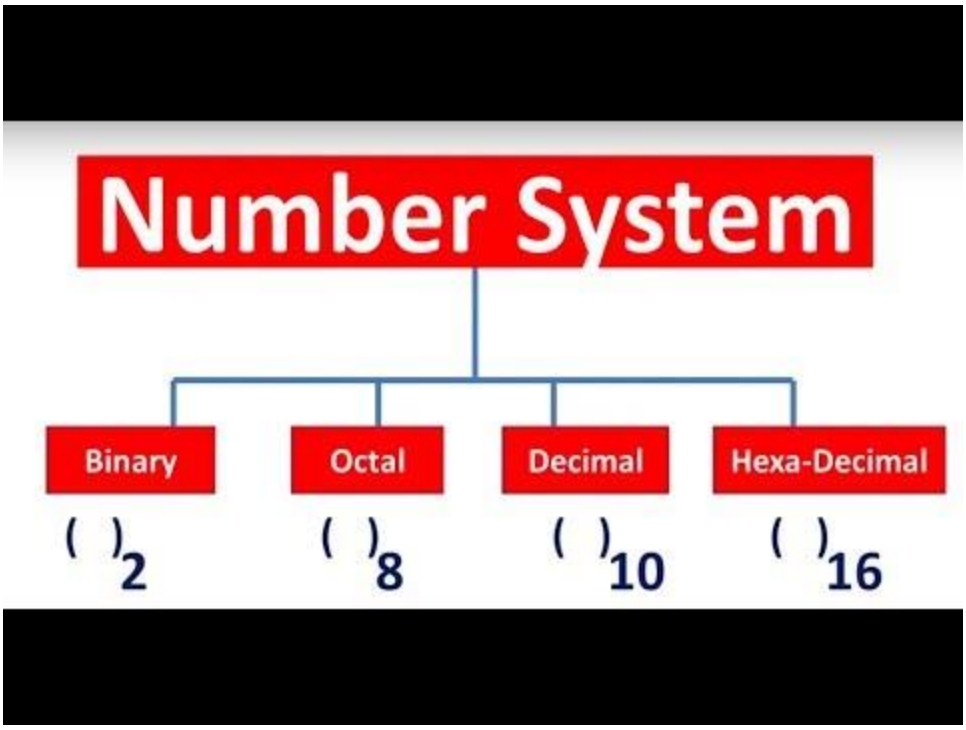
- 11189 AMMAD-UL-HASAN
- 11101 JAVERIA FAREED
- 11278 SIMRA

PROJECT NAME: NUMBER SYSTEM CONVERSION



Computer architecture support following number system

1. Binary number system
2. Octal number system
3. Decimal number system
4. Hexadecimal number system



Binary number system:

A **binary number system** is one of the four types of **number system**. In **computer** applications, where **binary numbers** are represented by only two symbols or digits, i.e. 0 (zero) and 1(one). The binary numbers here are expressed in the base-2 numeral system. For example, $(101)_2$ is a binary number. Each digit in this system is said to be a bit

Octal number system:

Octal Number System has a base of eight and uses the number from 0 to 7. The octal numbers, in the number system, are usually represented by binary numbers when they are grouped in pairs of three. For example, 12_8 is expressed as 001 010_2, where 1 is equivalent to 001 and 2 is equivalent to 010.

Decimal number system:

In the **decimal number system**, the numbers are represented with base 10. The way of denoting the decimal numbers with base 10 is also termed as decimal notation. This number system is widely used in computer

applications. It is also called the base-10 number system which consists of 10 digits, such as, 0,1,2,3,4,5,6,7,8,9.

.

Hexadecimal number system:

Hexadecimal Number System is one the type of Number Representation techniques, in which there value of base is 16. That means there are only 16 symbols or possible digit values, there are 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Where A, B, C, D, E and F are single bit representations of decimal value 10, 11, 12, 13, 14 and 15 respectively. It requires only 4 bits to represent value of any digit.

CONVERSIONS:

- Decimal into Binary/Hex number system
- Binary into Decimal/Hex number system
- Hexadecimal into Binary/Decimal number system
- Octal into binary/decimal

- **Decimal to binary**

Steps to convert decimal to binary:

a) Take decimal number as dividend.
b) Divide this number by 2 (2 is base of binary so divisor here).
c) Store the remainder in an array (it will be either 0 or 1 because of divisor 2).
d) Repeat the above two steps until the number is greater than zero.
e) Print the array in reverse order (which will be equivalent binary number of given decimal number).

For example:

- Convert decimal number 112 into binary number.

Division	Remainder (R)
112 / 2 = 56	0
56 / 2 = 28	0
28 / 2 = 14	0
14 / 2 = 7	0
7 / 2 = 3	1
3 / 2 = 1	1
1 / 2 = 0	1

This will be 1110000 which is equivalent binary number of decimal integer 112.

- **Decimal to hexadecimal**

Steps to convert decimal to hexadecimal:

a) Take decimal number as dividend.

b) Divide this number by 16 (16 is base of hexadecimal so divisor here).

c) Store the remainder in an array (it will be: 0 to 15 because of divisor 16, replace 10, 11, 12, 13, 14, 15 by A, B, C, D, E, F respectively).

d) Repeat the above two steps until the number is greater than zero.

e) Print the array in reverse order (which will be equivalent hexadecimal number of given decimal number).

For example:

- Convert decimal number 540 into hexadecimal number

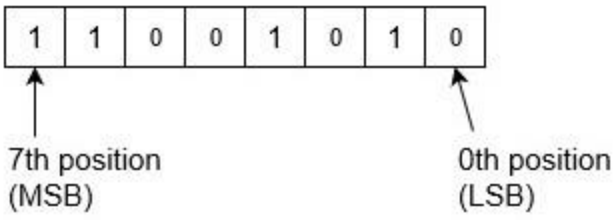
Division	Remainder (R)
540 / 16 = 33	12 = C
33 / 16 = 2	1
2 / 16 = 0	2
0 / 16 = 0	0

This will be 021C (or only 21C) which is equivalent hexadecimal number of decimal integer 540.

- **Binary to decimal:**

For example:

Convert binary number 11001010 into decimal number.



$= (11001010)_2$
$= 1x2^7+1x2^6+0x2^5+0x2^4+1x2^3+0x2^2+1x2^1+0x2^0$
$= 128+64+0+0+8+0+2+0$
$= (202)_{10}$

- **Binary to Hexadecimal:**

Example − Convert binary number 1101010 into hexadecimal number.

First convert this into decimal number:
$= (1101010)_2$
$= 1x2^6+1x2^5+0x2^4+1x2^3+0x2^2+1x2^1+0x2^0$
$= 64+32+0+8+0+2+0$
$= (106)_{10}$
Then, convert it into hexadecimal number
$= (106)_{10}$
$= 6x16^1+10x16^0$
$= (6A)_{16}$ which is answer.

- **Hexadecimal to binary:**

Steps to convert hexadecimal to binary:

Step 1: Take given hexadecimal number.

Step 2: Find the number of digits in the decimal.

Step 3: If it has n digits, multiply each digit with 16^{n-1} where the digit is in the nth position.

Step 4: Add the terms after multiplication.

Step 5: The result is the decimal number equivalent to the given hexadecimal number. Now we have to convert this decimal to binary number.

Step 6: Divide the decimal number with 2

Step 7: Note the remainder

Step 8: Do the above 2 steps for the quotient till the quotient is zero

Step 9: Write the remainders in the reverse order.

Step 10: The result is the required binary number.

For example:

Convert $A2B_{16}$ to an equivalent binary number.

Solution: Given hexadecimal number = $A2B_{16}$

First, convert the given hexadecimal to the equivalent decimal number.

$A2B_{16} = (A \times 16^2) + (2 \times 16^1) + (B \times 16^0)$

$= (A \times 256) + (2 \times 16) + (B \times 1)$

$= (10 \times 256) + 32 + 11$

$= 2560 + 43$

$= 2603$(Decimal number)

Now we have to convert 2603_{10} to binary

2 | 2603
2 | 1301 -- 1
2 | 650 -- 1
2 | 325 -- 0
2 | 162 -- 1
2 | 81 -- 0
2 | 40 -- 1
2 | 20 -- 0
2 | 10 -- 0
2 | 5 -- 0
2 | 2 -- 1
2 | 1 -- 0
2 | 0 -- 1

- **Hexadecimal to decimal:**

To convert a hexadecimal to a decimal manually, you must **start by multiplying the hex number by 16**. Then, you raise it to a power of 0 and increase that power by 1 each time according to the hexadecimal number equivalent.

For example:

Convert hexadecimal number F1 into decimal number.

$(F1)_{16}$
$= (1111\ 0001)_2$ or $(011\ 110\ 001)_2$
Because in binary, value of F and 1 are 1111 and 0001 respectively.
Then convert it into decimal number multiplying power of its position of base.
$= (1x2^7+1x2^6+1x2^5+1x2^4+0x2^3+0x2^2+0x2^1+1x2^0)_{10}$
$=(241)_{10}$

- **Octal to Binary:**

Octal to Decimal Conversion

- Count the number of digits present in the given number. Let the number of digits be ‘n’.
- Now multiply each digit of the number with 8^{n-1}, when the digit is in the nth position from the right end of the number. If the number has a decimal part, multiply each digit in the decimal part by `$8^{-m}$` when the digit is in the m^{th} position from the decimal point.
- Add all the terms after multiplication.
- The obtained value is the equivalent decimal number.

Decimal to Binary Conversion

- Take the above-produced decimal number and divide it by 2.
- Note down the remainder.
- Continue the above two steps for the quotient till the quotient is zero.
- Write the remainder in the reverse order.

- The received number is the equivalent binary number for the given octal number.

For example:

Q.1: Convert 41_8 to a binary number.
Solution: Given number is 41_8
$41_8 = (4 * 8^1) + (1 * 8^0)$
$= 4 * 8 + 1 * 1$
$= 32+1$
$= 33$(Decimal number)
Now convert this decimal number into its equivalent binary number. Let us draw a table to show the conversion of decimal to binary as given below.

Decimal Number divided by 2	Quotient	Remainder
33 divided by 2	16	1
16 divided by 2	8	0
8 divided by 2	4	0
4 divided by 2	2	0
2 divided by 2	1	0
1 divided by 2	0	1

Therefore, the equivalent binary number is 100001_2.

Hence, $41_8 = 100001_2$

- **Octal to Decimal:**

Octal to Decimal Conversion

- Count the number of digits present in the given number. Let the number of digits be ‘n’.
- Now multiply each digit of the number with 8^{n-1}, when the digit is in the nth position from the right end of the number. If the number has a decimal part, multiply each digit in the decimal part by `$8^{-m}$` when the digit is in the m^{th} position from the decimal point.
- Add all the terms after multiplication.
- The obtained value is the equivalent decimal number.

  For example:

Q.1: Convert 41_8 to a binary number.

Solution: Given number is 41_8

$41_8 = (4 * 8^1) + (1 * 8^0)$

$= 4 * 8 + 1 * 1$

$= 32+1$

= 33(Decimal number)

>>>>>>>>>>>>>>>>>>>>>>>>>>>>>**********<<<<<<<<<<<<<<<<<<<<<<<<<<<

- **We use 6 procedures in our code which are highlighted in code below.**
- **We use different predefined functions as well like: writehex, writedec ,readdec, readhex etc.**

OUR PROJECT CODE:

```
INCLUDE Irvine32.inc

.DATA

	Heading byte "<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<-----NUMBER SYSTEM CONVERTER----->>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>",0dh,0ah,0

	Group_Members byte "<<<<<<<<<<------ GROUP MEMBERS ------>>>>>>>>>>",0dh,0ah

				byte "------ 1.Muhammad Ammad Ul hasan (11189) ------ ",0dh,0ah

				byte "---------- 2.Javeria Fareed (11101) ----------",0dh,0ah

				byte "---------- 3.Simra (11278) ----------",0dh,0ah,0

	OPTIONS byte "Select the type of conversion you want to perform from the following 4 ",0dh,0ah

				byte "1. Convert Binary to Hexadecimal and Decimal ",0dh,0ah

				byte "2. Convert Decimal to Hexadecimal and Binary ",0dh,0ah

				byte "3. Convert Hexadecimal to Binary and Decimal ",0dh,0ah

				byte "4. Convert Octal to Hexadecimal and Decimal ",0dh,0ah

				byte "5. Exit ",0dh,0ah,0

	SELECT_OPTION byte "Please enter the conversion you want to perform: ",0

	OPTION_TEMP dword ?

	THANK_YOU_MESSAGE byte "<<<<<<<<<<----------  THANK YOU FOR USING OUR CALCULATOR ---------->>>>>>>>>>",0dh,0ah,0

	ERROR_OPTIONS byte "<<<<<<<<<<----------  Please select from the options given. ----------->>>>>>>>>>",0dh,0ah,0

	DECIMAL_NUMBER_CONVERTED BYTE "Decimal Number IS: ",0

	HEXADECIMAL_NUMBER_CONVERTED BYTE "Hexadecimal Number IS: ",0

	BINARY_NUMBER_CONVERTED BYTE "Binary Number IS: ",0

	;------------------------------***----------------------------

	;--------------------------BINARY DATA--------------------------

	BINARY_INPUT byte "Enter the BINARY number you want to convert: ",0

	ERROR_NOT_BINARY_NUMBER BYTE "Invalid Binary can only contain 0 and 1",0

	BINARY_NUMBER_ARRAY BYTE 33 DUP(?)

	BINARY_LENGTH DWORD 0
```

```
        BASE DWORD 2
        DECIMAL_NUMBER DWORD ?
        COUNT DWORD 0
        ;-------------------------------***-----------------------------
        ;--------------------------DECIMAL DATA---------------------------
        DECIMAL_INPUT byte "Enter the DECIMAL number you want to convert: ",0
        DECIMAL_TEMP dword ?
        ;-------------------------------***-----------------------------
        ;--------------------------HEXADECIMAL DATA---------------------------
        HEXADECIMAL_INPUT byte "Enter the HEXADECIMAL number you want to convert: ",0
        HEXADECIMAL_TEMP dword ?
        ;-------------------------------***-----------------------------
        ;--------------------------OCTAL DATA---------------------------
        OCTAL_INPUT byte "Enter the OCTAL number you want to convert: ",0
        ERROR_NOT_OCTAL_NUMBER BYTE "Invalid OCTAL Number",0
        OCTAL_NUMBER BYTE 33 DUP(?)
        OCTAL_LENGTH DWORD 0
        BASE_OCTAL DWORD 8
        DECIMAL_NUMBER_OCTAL DWORD ?
        COUNT_OCTAL DWORD 0
.CODE
main PROC
        mov eax,0
        mov edx,offset Heading
        call writestring
        mov edx,offset Group_Members
        call writestring

        MAIN_LABEL_FOR_CONVERTER:
```

```
call crlf
mov edx,offset OPTIONS
call writestring
call crlf
mov edx,offset SELECT_OPTION
call writestring
call readdec
call crlf
mov OPTION_TEMP,eax

COMPARE_1_LABEL:
        mov eax,OPTION_TEMP
        cmp eax,1
        JE OPTION_1_LABEL
COMPARE_2_LABEL:
        mov eax,OPTION_TEMP
        cmp eax,2
        JE OPTION_2_LABEL
COMPARE_3_LABEL:
        mov eax,OPTION_TEMP
        cmp eax,3
        JE OPTION_3_LABEL
COMPARE_4_LABEL:
        mov eax,OPTION_TEMP
        cmp eax,4
        JE OPTION_4_LABEL
COMPARE_5_LABEL:
        mov eax,OPTION_TEMP
        cmp eax,5
```

```
        JE QUIT_LABEL_OPTION_5
        JGE GREATER_THAN_5_ERROR_LABEL


OPTION_1_LABEL:
        mov edx, OFFSET BINARY_INPUT
        call WriteString
        mov edx,OFFSET BINARY_NUMBER_ARRAY
        mov ecx,SIZEOF BINARY_NUMBER_ARRAY
        call ReadString
        mov BINARY_LENGTH,eax
        mov ecx, BINARY_LENGTH
        mov eax,0
        mov esi,0
        call BINARY_TO_DECIMAL_CONVERT_PROCEDURE
        jmp MAIN_LABEL_FOR_CONVERTER
OPTION_2_LABEL:
        mov edx,offset DECIMAL_INPUT
        call writestring
        call readdec
        mov DECIMAL_TEMP,eax
        call DISPLAY_DECIMAL_TO_BINARY_AND_HEXADECIMAL
        jmp MAIN_LABEL_FOR_CONVERTER
OPTION_3_LABEL:
        mov edx,offset HEXADECIMAL_INPUT
        call writestring
        call readhex
        mov HEXADECIMAL_TEMP,eax
        call DISPLAY_HEXADECIMAL_TO_BINARY_AND_DECIMAL
        jmp MAIN_LABEL_FOR_CONVERTER
```

```
                OPTION_4_LABEL:
                        mov edx, OFFSET OCTAL_INPUT
                        call WriteString
                        mov edx,OFFSET OCTAL_NUMBER
                        mov ecx,SIZEOF OCTAL_NUMBER
                        call ReadString
                        mov OCTAL_LENGTH,eax
                        mov eax,0
                        mov esi,0
                        mov ecx, OCTAL_LENGTH
                        call OCTAL_TO_DECIMAL_CONVERT_PROCEDURE
                        jmp MAIN_LABEL_FOR_CONVERTER
                GREATER_THAN_5_ERROR_LABEL:
                        mov edx,offset ERROR_OPTIONS
                        call writestring
                        call crlf
                        jmp MAIN_LABEL_FOR_CONVERTER
QUIT_LABEL_OPTION_5:
mov edx,offset THANK_YOU_MESSAGE
call writestring
call crlf
EXIT
main ENDP

BINARY_TO_DECIMAL_CONVERT_PROCEDURE PROC
        OUTER_CONVERSION_LABEL:
                cmp ecx,0
                je DISPLAY_BINARY_TO_DECIMAL_AND_HEXADECIMAL
                mov COUNT,ecx
```

```
CONDITION_1:
        cmp BINARY_NUMBER_ARRAY[esi],'0'
        je INCRMENT_LABEL

CONDITION_2:
        cmp BINARY_NUMBER_ARRAY[esi],'1'
        jne NOT_BINARY_ERROR

        mov ecx, BINARY_LENGTH
        sub ecx,esi
        dec ecx

        mov eax,1
        .while(ecx >= 0)
                cmp ecx,0
                je stop

                mov ebx,BASE
                mul ebx
                dec ecx
        .endw

        stop:
                add DECIMAL_NUMBER,eax
                jmp INCRMENT_LABEL

NOT_BINARY_ERROR:
        mov edx, OFFSET ERROR_NOT_BINARY_NUMBER
```

```
                call WriteString
                call Crlf
                exit

        INCRMENT_LABEL:
                inc esi
                mov ecx,COUNT
                dec ecx
                jmp OUTER_CONVERSION_LABEL

        call DISPLAY_BINARY_TO_DECIMAL_AND_HEXADECIMAL
        ret
BINARY_TO_DECIMAL_CONVERT_PROCEDURE ENDP

DISPLAY_BINARY_TO_DECIMAL_AND_HEXADECIMAL PROC
                mov edx, OFFSET DECIMAL_NUMBER_CONVERTED
                call WriteString
                mov eax, DECIMAL_NUMBER
                call WriteDec
                call Crlf
                mov edx,offset HEXADECIMAL_NUMBER_CONVERTED
                call WriteString
                mov eax, DECIMAL_NUMBER
                call writehex
                call crlf
                ret
DISPLAY_BINARY_TO_DECIMAL_AND_HEXADECIMAL ENDP

DISPLAY_DECIMAL_TO_BINARY_AND_HEXADECIMAL PROC
```

```
        mov edx,offset BINARY_NUMBER_CONVERTED
        call writestring
        call writebin
        call crlf
        mov eax,DECIMAL_TEMP
        mov edx,offset HEXADECIMAL_NUMBER_CONVERTED
        call writestring
        call writehex
        call crlf
        ret
DISPLAY_DECIMAL_TO_BINARY_AND_HEXADECIMAL ENDP

DISPLAY_HEXADECIMAL_TO_BINARY_AND_DECIMAL PROC
        mov edx,offset BINARY_NUMBER_CONVERTED
        call writestring
        call writebin
        call crlf
        mov eax,HEXADECIMAL_TEMP
        mov edx,offset DECIMAL_NUMBER_CONVERTED
        call writestring
        call writedec
        call crlf
        ret
DISPLAY_HEXADECIMAL_TO_BINARY_AND_DECIMAL ENDP

OCTAL_TO_DECIMAL_CONVERT_PROCEDURE PROC
        OUTER_CONVERSION_LABEL:
                cmp ecx,0
                je DISPLAY_OCTALL_TO_HEXADECIMAL_AND_DECIMAL
```

```
mov COUNT_OCTAL,ecx

CONDITION_1:
        cmp OCTAL_NUMBER[esi],'0'
        je INCRMENT_LABEL

CONDITION_2:
        cmp OCTAL_NUMBER[esi],'1'
        jge CONDITION_3
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,0
        jmp INNER_CONVERSION_LABEL
CONDITION_3:
        cmp OCTAL_NUMBER[esi],'2'
        jge CONDITION_4
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,1
        jmp INNER_CONVERSION_LABEL
CONDITION_4:
        cmp OCTAL_NUMBER[esi],'3'
        jge CONDITION_5
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,2
```

```
        jmp INNER_CONVERSION_LABEL
CONDITION_5:
        cmp OCTAL_NUMBER[esi],'4'
        jge CONDITION_6
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,3
        jmp INNER_CONVERSION_LABEL
CONDITION_6:
        cmp OCTAL_NUMBER[esi],'5'
        jge CONDITION_7
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,4
        jmp INNER_CONVERSION_LABEL
CONDITION_7:
        cmp OCTAL_NUMBER[esi],'6'
        jge CONDITION_8
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,5
        jmp INNER_CONVERSION_LABEL
CONDITION_8:
        cmp OCTAL_NUMBER[esi],'7'
        jge CONDITION_9
        mov ecx, OCTAL_LENGTH
```

```
        sub ecx,esi
        dec ecx
        mov eax,6
        jmp INNER_CONVERSION_LABEL
CONDITION_9:
        cmp OCTAL_NUMBER[esi],'8'
        jge NOT_OCTAL_ERROR
        mov ecx, OCTAL_LENGTH
        sub ecx,esi
        dec ecx
        mov eax,7
        jmp INNER_CONVERSION_LABEL


INNER_CONVERSION_LABEL:
        cmp ecx,0
        je stop
        mov ebx,BASE_OCTAL
        mul ebx
        dec ecx
        jmp INNER_CONVERSION_LABEL

stop:
        add DECIMAL_NUMBER_OCTAL,eax
        jmp INCRMENT_LABEL



NOT_OCTAL_ERROR:
```

```
                mov edx, OFFSET ERROR_NOT_OCTAL_NUMBER
                call WriteString
                call Crlf
                call WaitMsg
                exit

        INCRMENT_LABEL:
                inc esi
                mov ecx,COUNT_OCTAL
                dec ecx
                jmp OUTER_CONVERSION_LABEL

        call DISPLAY_OCTALL_TO_HEXADECIMAL_AND_DECIMAL
        ret
OCTAL_TO_DECIMAL_CONVERT_PROCEDURE ENDP

DISPLAY_OCTALL_TO_HEXADECIMAL_AND_DECIMAL PROC
        mov edx, OFFSET DECIMAL_NUMBER_CONVERTED
        call WriteString
        mov eax, DECIMAL_NUMBER_OCTAL
        call WriteDec
        call Crlf
        mov edx, OFFSET HEXADECIMAL_NUMBER_CONVERTED
        call WriteString
        mov eax, DECIMAL_NUMBER_OCTAL
        call WriteHex
        call crlf
        mov edx, OFFSET BINARY_NUMBER_CONVERTED
        call WriteString
```

```
        mov eax, DECIMAL_NUMBER_OCTAL
        call WriteBin
        call crlf
        ret
DISPLAY_OCTALL_TO_HEXADECIMAL_AND_DECIMAL ENDP

END mainf
```

>>>>>>>>>>>>>>>>>>>>**********<<<<<<<<<<<<<<<<<<<<<<

OUTPUT:

```
Microsoft Visual Studio Debug Console
<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<<-----NUMBER SYSTEM CONVERTER----->>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
<<<<<<<<<<------ GROUP MEMBERS ------>>>>>>>>>>
------ 1.Muhammad Ammad Ul hasan (11189) ------
----------- 2.Javeria Fareed (11101) ----------
--------------- 3.Simra (11278) ---------------

Select the type of conversion you want to perform from the following 4
1. Convert Binary to Hexadecimal and Decimal
2. Convert Decimal to Hexadecimal and Binary
3. Convert Hexadecimal to Binary and Decimal
4. Convert Octal to Hexadecimal and Decimal
5. Exit

Please enter the conversion you want to perform: 1

Enter the BINARY number you want to convert: 0010101
Decimal Number IS: 21
Hexadecimal Number IS: 00000015

Select the type of conversion you want to perform from the following 4
1. Convert Binary to Hexadecimal and Decimal
2. Convert Decimal to Hexadecimal and Binary
3. Convert Hexadecimal to Binary and Decimal
4. Convert Octal to Hexadecimal and Decimal
5. Exit

Please enter the conversion you want to perform: 5

<<<<<<<<<<<---------- THANK YOU FOR USING OUR CALCULATOR ---------->>>>>>>>>>


C:\Users\PCS\source\repos\Project6\Debug\Project6.exe (process 9072) exited with code 0.
Press any key to close this window . . .
```

>>>>>>>>>>>>>>>>>THE END<<<<<<<<<<<<<<<<<<<<