

# Logic Building

## 31-Jan 2022 to 05 Feb 2022

### Day-5

#### Trainers:

1. Dr. Abhay Kothari
2. Prof. Mubeen Ahmed Khan

SIRT SAGE University,  
Department of Computer Science and  
Engineering

# Contents

- Number System
- Binary to Decimal
- Octal to Decimal
- Hexadecimal to Decimal
- Decimal to binary
- Octal to binary
- Hexadecimal to Binary
- Octal to decimal
- Hexadecimal to decimal
- Binary to Decimal
- Decimal to Binary
- Octal to binary
- Decimal to Hexadecimal
- Hexadecimal to binary
- Octal to Decimal
- Octal to Hexadecimal

# Common Number Systems

System	Base	Symbols	Used by humans?	Used in computers?
Decimal	10	0, 1, ... 9	Yes	No
Binary	2	0, 1	No	Yes
Octal	8	0, 1, ... 7	No	No
Hexa-decimal	16	0, 1, ... 9, A, B, ... F	No	No

# Quantities/Counting (1 of 3)

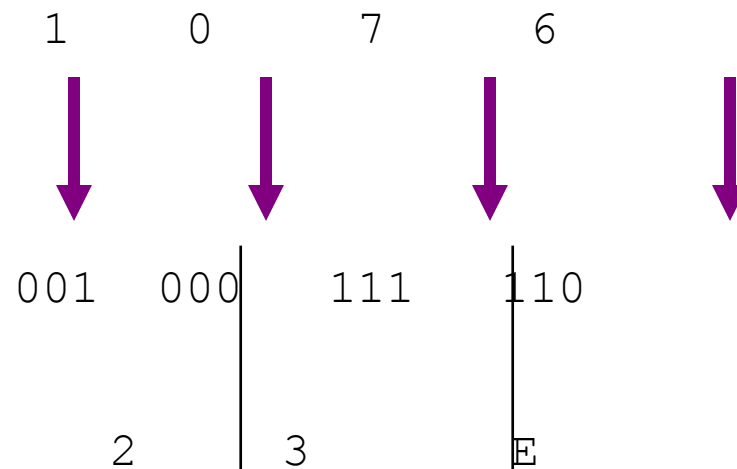
Decimal	Binary	Octal	Hexa- decimal
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7

# Quantities/Counting (2 of 3)

Decimal	Binary	Octal	Hexa- decimal
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

# Example

$$1076_8 = ?_{16}$$



$$1076_8 = 23E_{16}$$

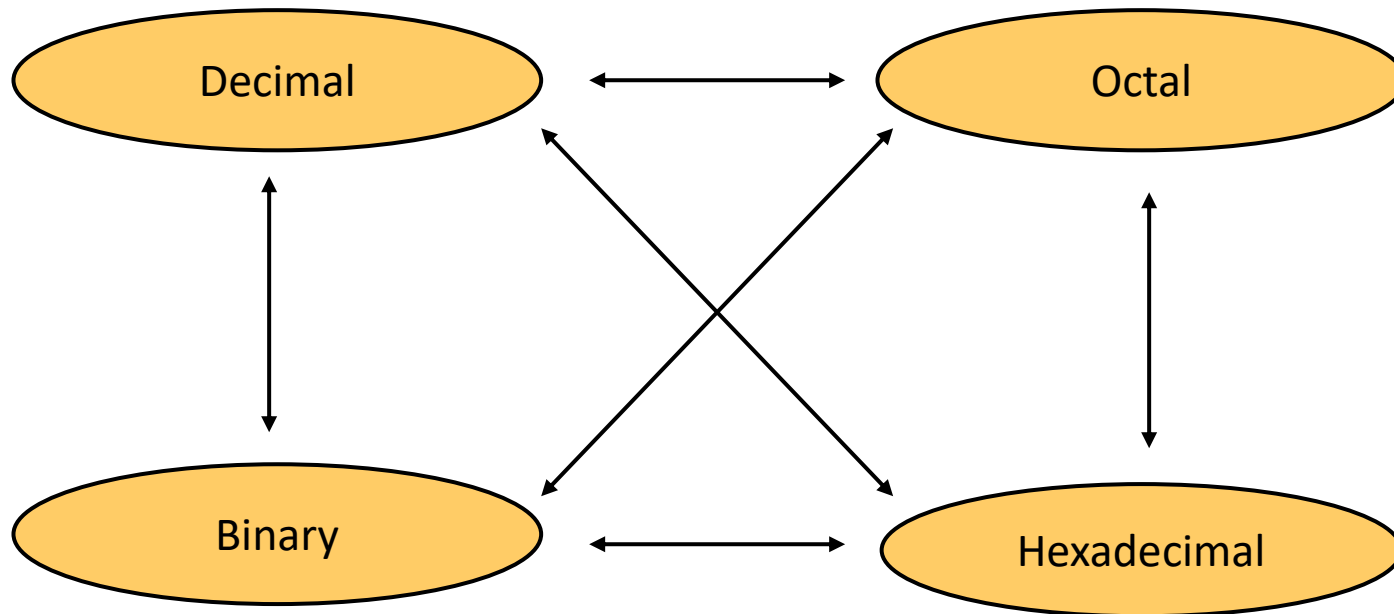
# Quantities/Counting (3 of 3)

Decimal	Binary	Octal	Hexa- decimal
16	10000	20	10
17	10001	21	11
18	10010	22	12
19	10011	23	13
20	10100	24	14
21	10101	25	15
22	10110	26	16
23	10111	27	17

Etc.

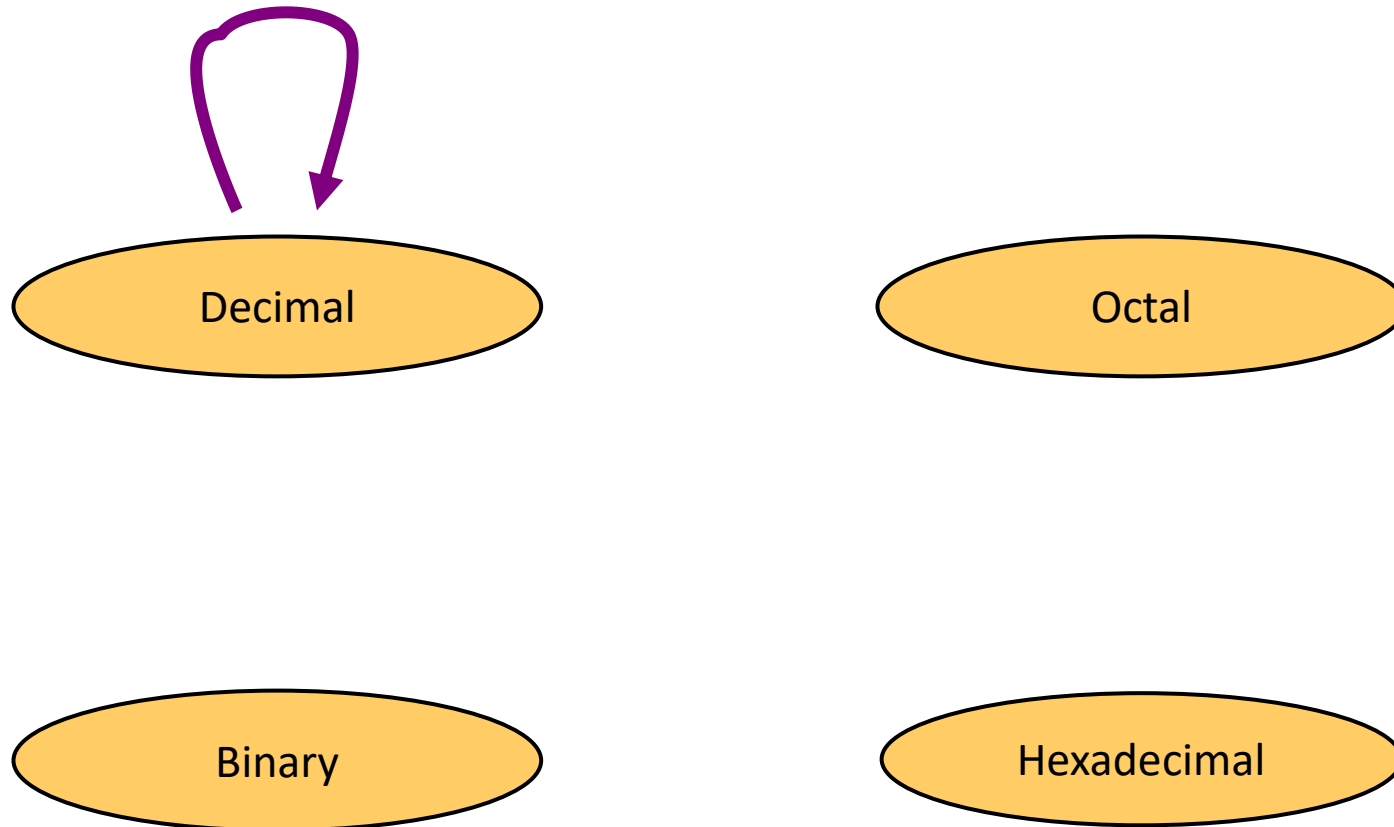
# Conversion Among Bases

- The possibilities:





# Decimal to Decimal (just for fun)



$125_{10} \Rightarrow$

5	$\times 10^0 =$	5
2	$\times 10^1 =$	20
1	$\times 10^2 =$	100

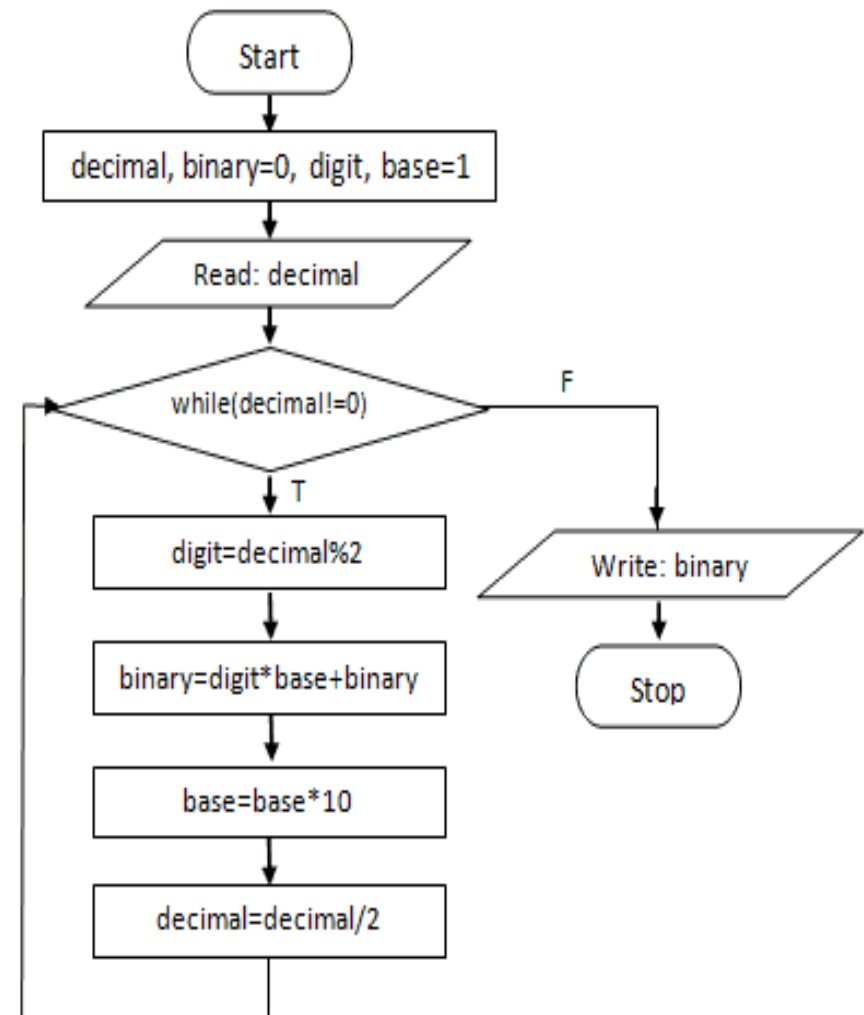
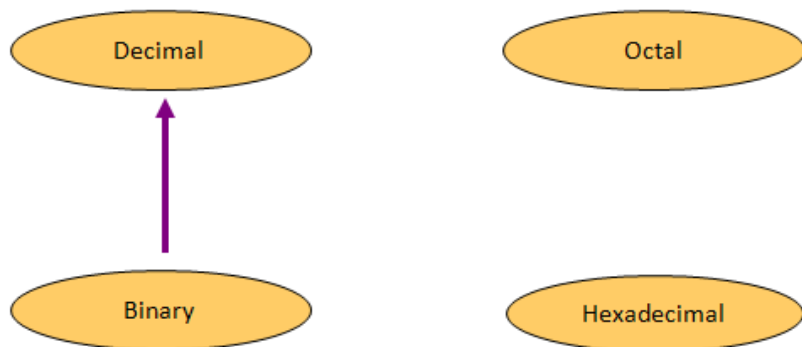
Weight

Base

125

# Decimal to Binary

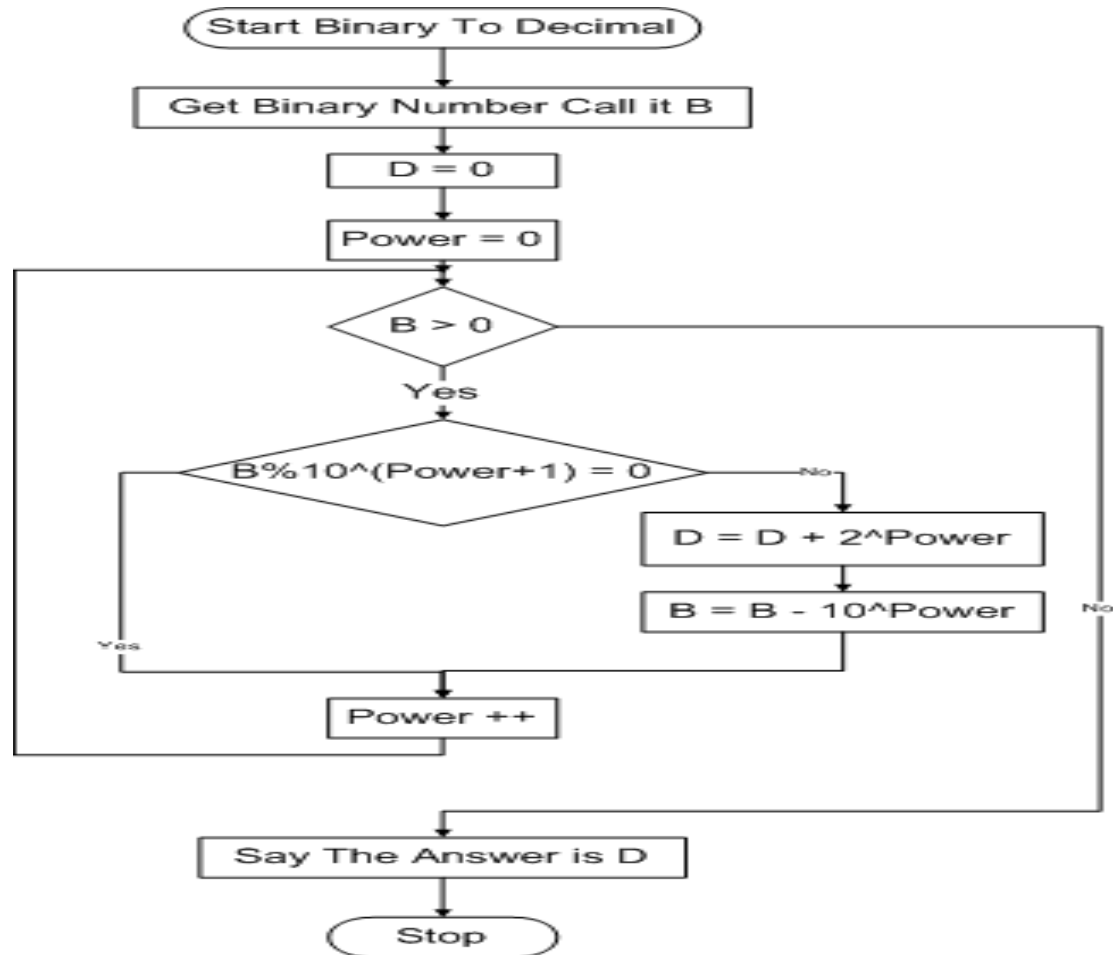
## Binary to Decimal



# Binary to Decimal

- Technique
  - Multiply each bit by  $2^n$ , where  $n$  is the “weight” of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results

# Binary to Decimal



# Example

Bit "0"

$101011_2 \Rightarrow$

$$1 \times 2^0 = 1$$

$$1 \times 2^1 = 2$$

$$0 \times 2^2 = 0$$

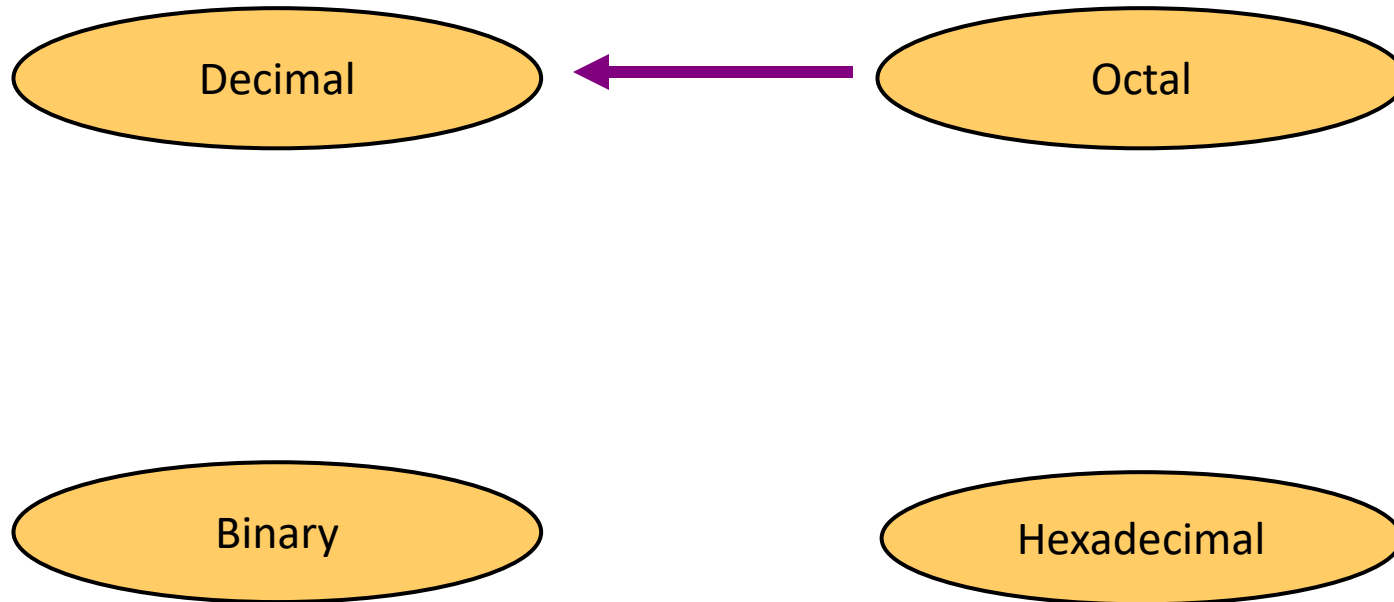
$$1 \times 2^3 = 8$$

$$0 \times 2^4 = 0$$

$$1 \times 2^5 = 32$$

$43_{10}$

# Octal to Decimal



# Octal to Decimal

- Technique
  - Multiply each bit by  $8^n$ , where  $n$  is the “weight” of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results

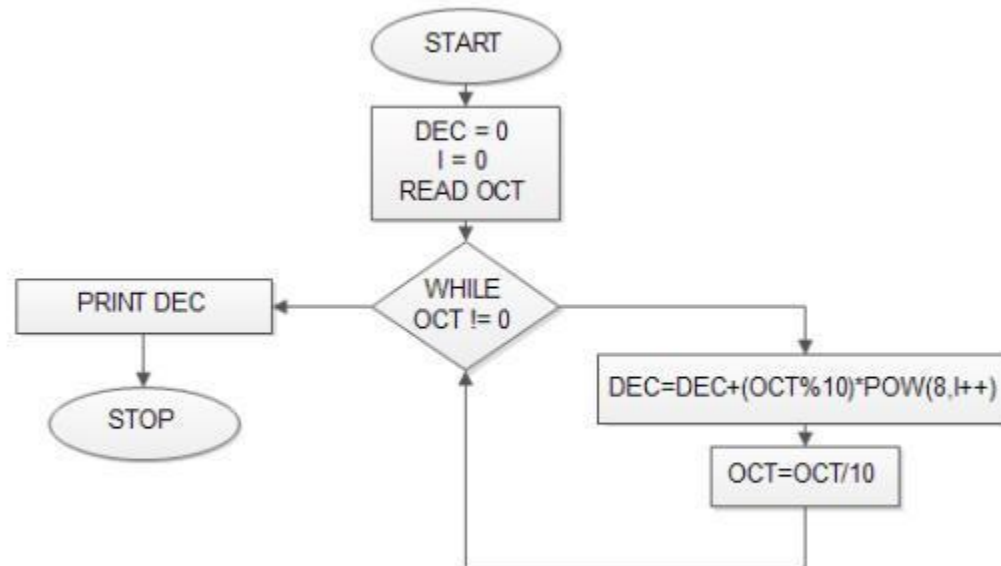


# Example

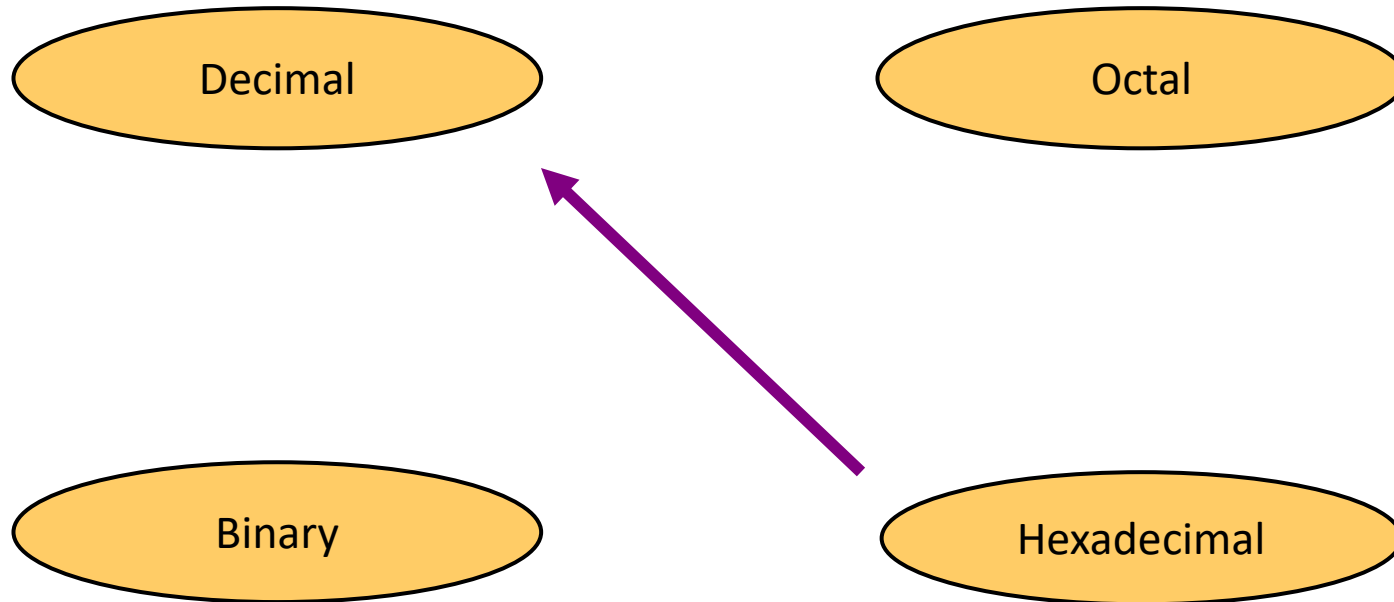
$$724_8 \Rightarrow$$

$$\begin{array}{rcl} 4 \times 8^0 & = & 4 \\ 2 \times 8^1 & = & 16 \\ 7 \times 8^2 & = & 448 \\ \hline & & 468_{10} \end{array}$$

# Octal to Decimal



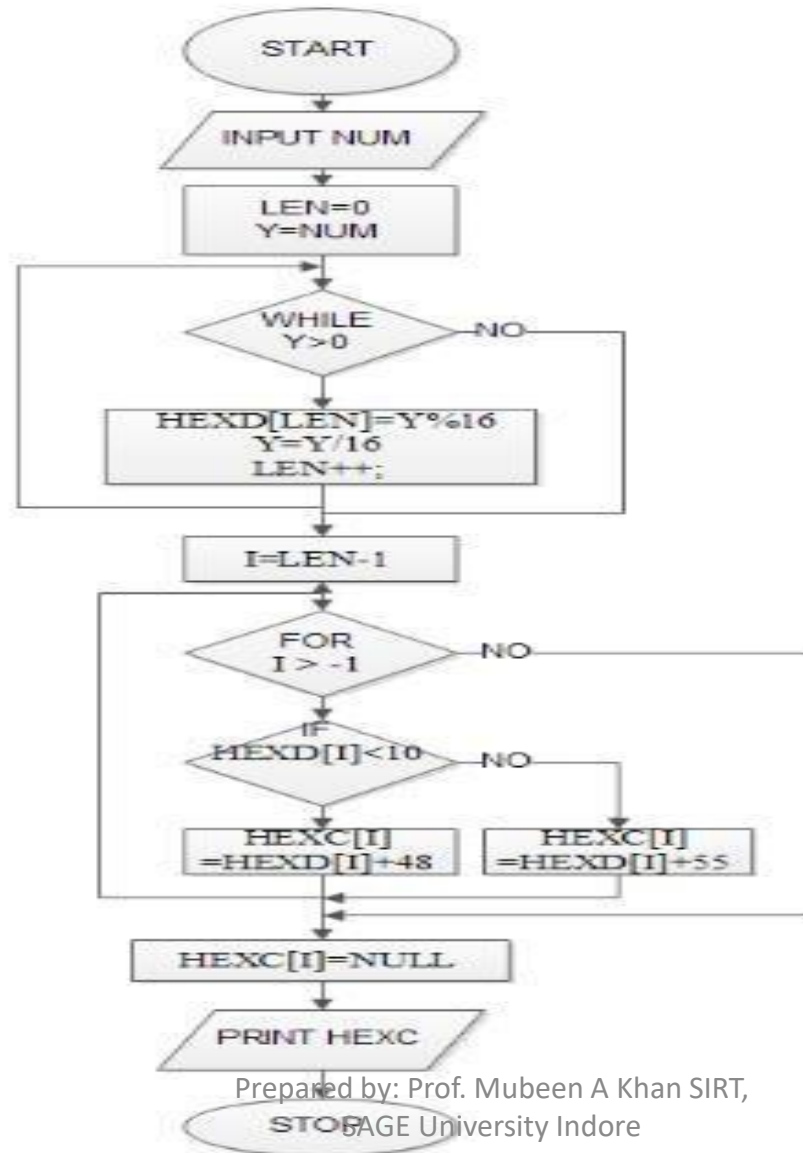
# Hexadecimal to Decimal



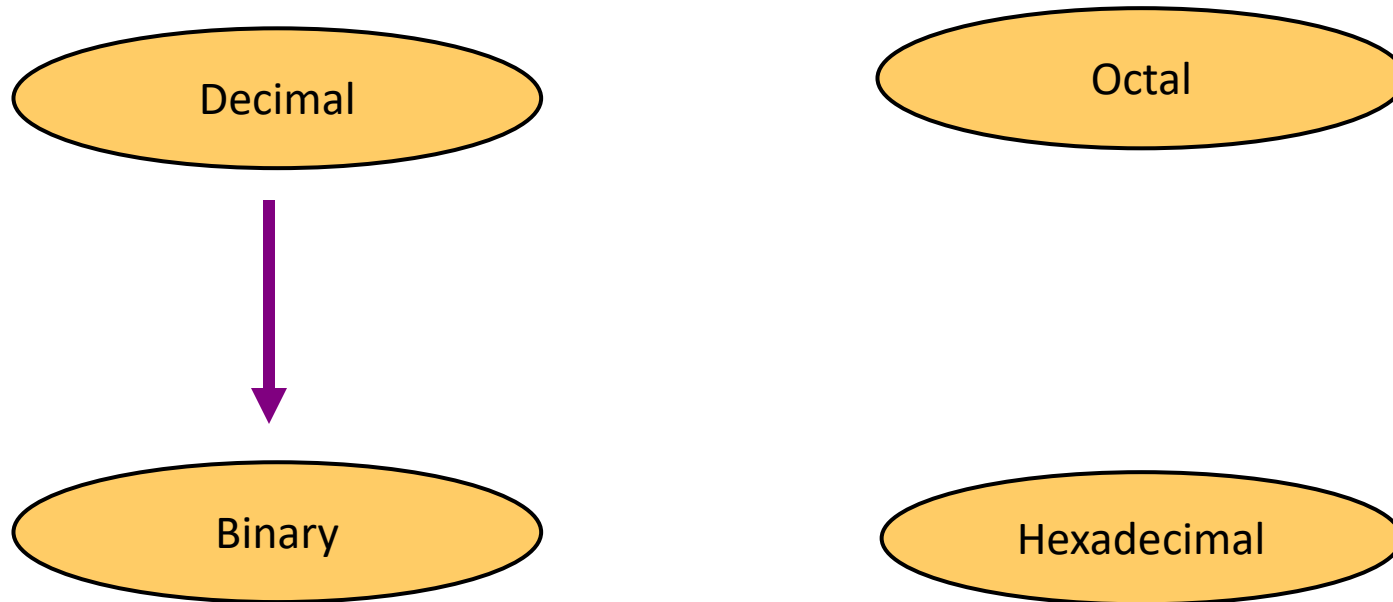
# Hexadecimal to Decimal

- Technique
  - Multiply each bit by  $16^n$ , where  $n$  is the “weight” of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results

# Hexadecimal into Decimal



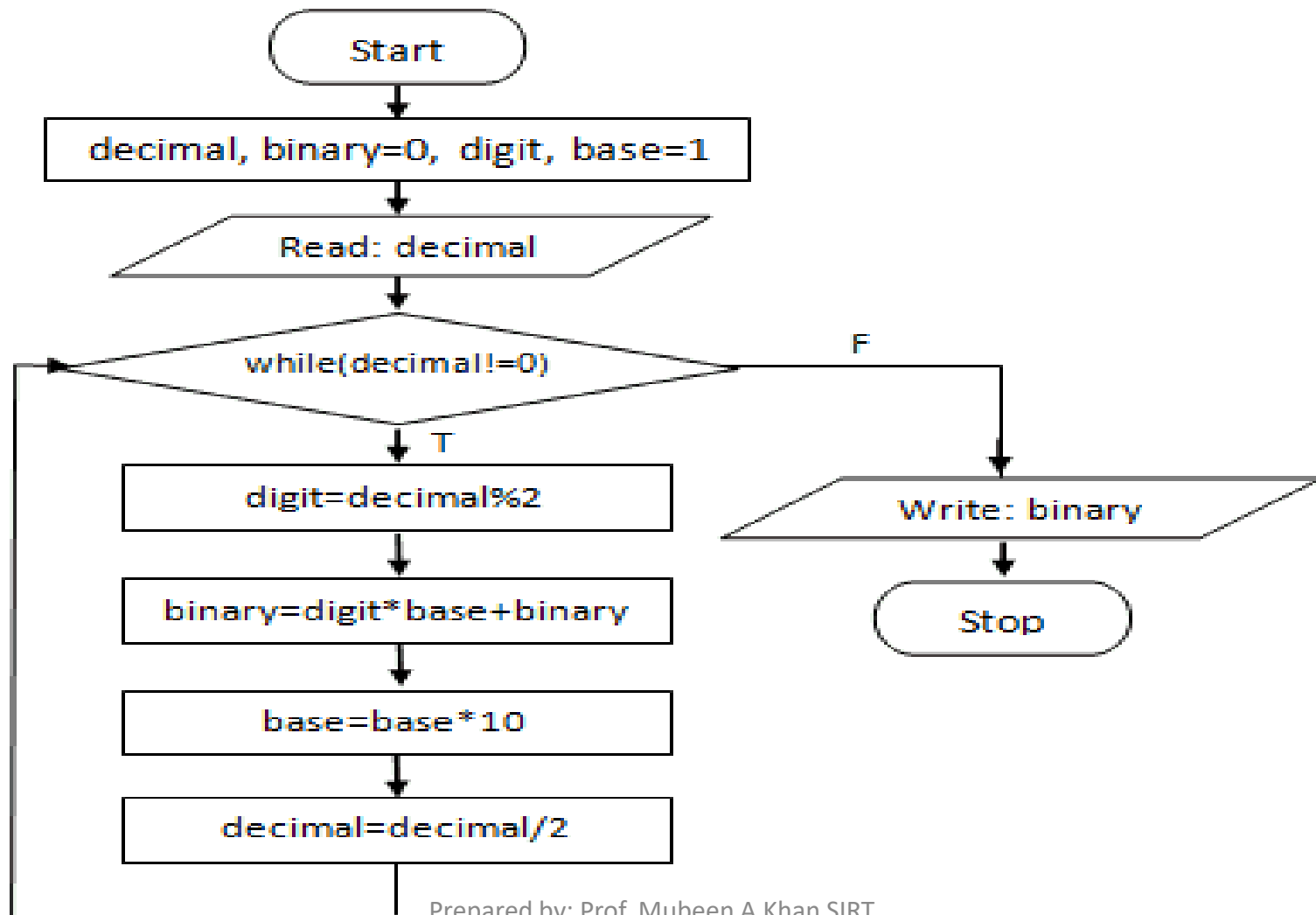
# Decimal to Binary



# Decimal to Binary

- Technique
  - Divide by two, keep track of the remainder
  - First remainder is bit 0 (LSB, least-significant bit)
  - Second remainder is bit 1
  - Etc.

# Decimal to Binary

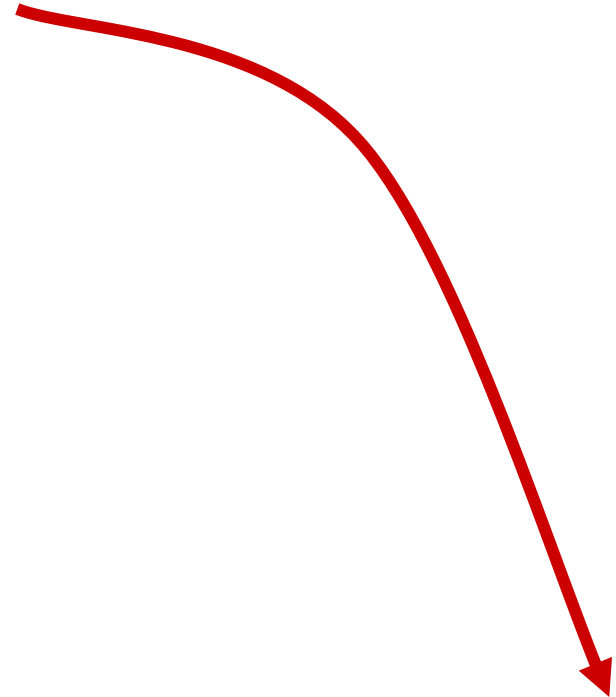




# Example

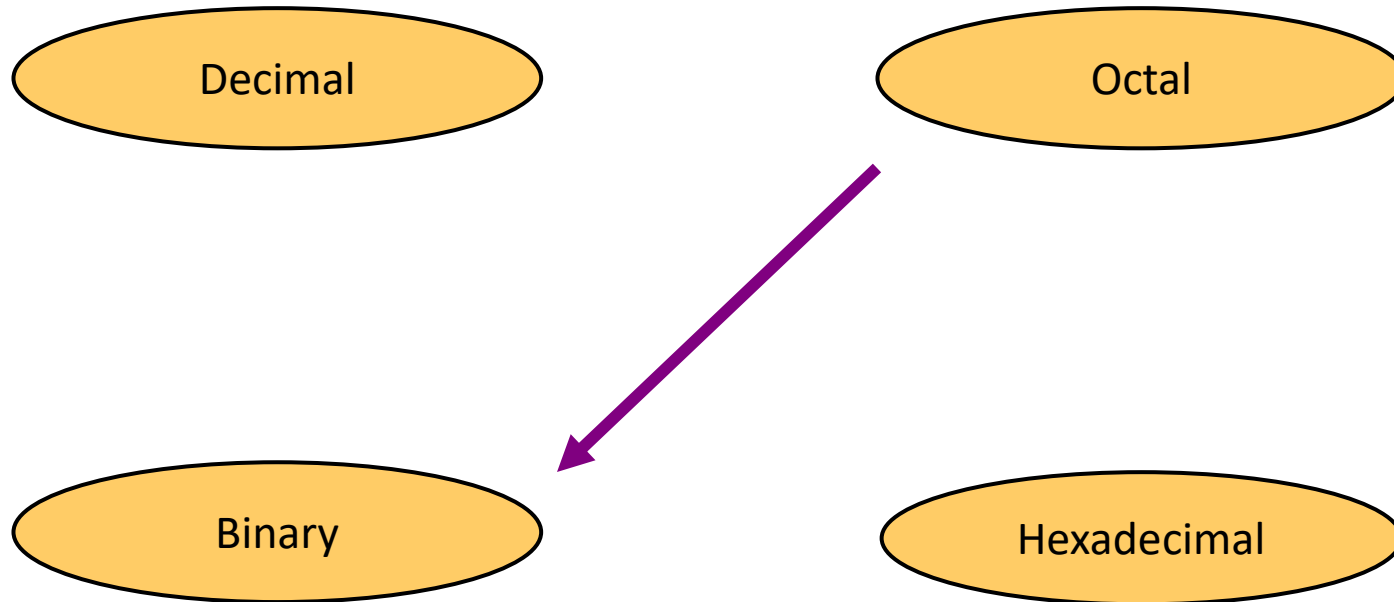
$$125_{10} = ?_2$$

2	125	
2	62	1
2	31	0
2	15	1
2	7	1
2	3	1
2	1	1
	0	1



$$125_{10} = 1111101_2$$

# Octal to Binary

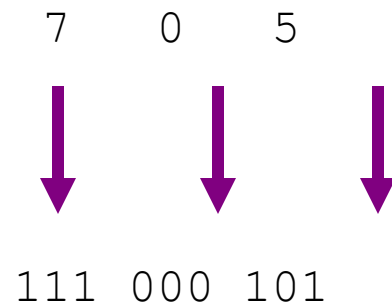


# Octal to Binary

- Technique
  - Convert each octal digit to a 3-bit equivalent binary representation

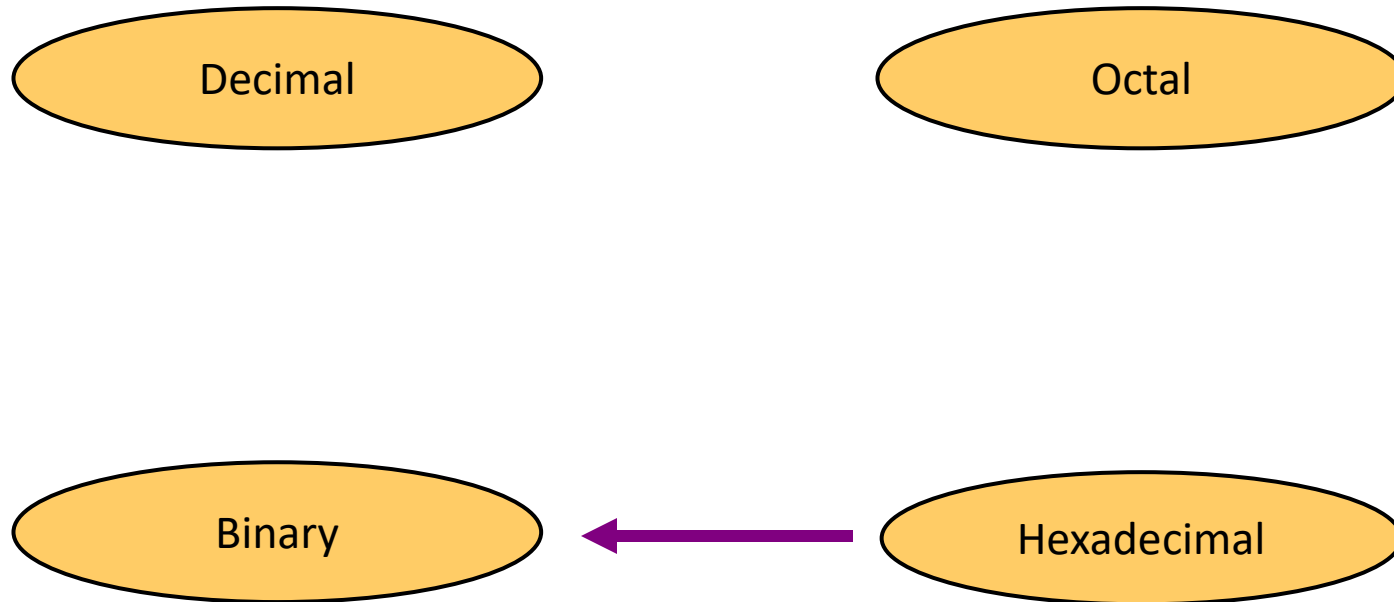
# Example

$$705_8 = ?_2$$



$$705_8 = 111000101_2$$

# Hexadecimal to Binary

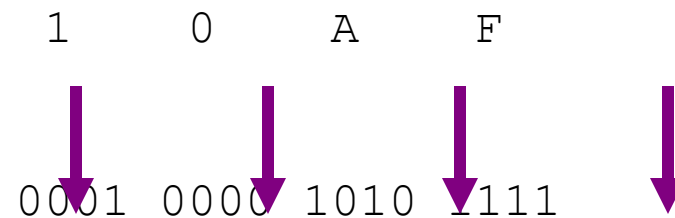


# Hexadecimal to Binary

- Technique
  - Convert each hexadecimal digit to a 4-bit equivalent binary representation

# Example

$$10AF_{16} = ?_2$$



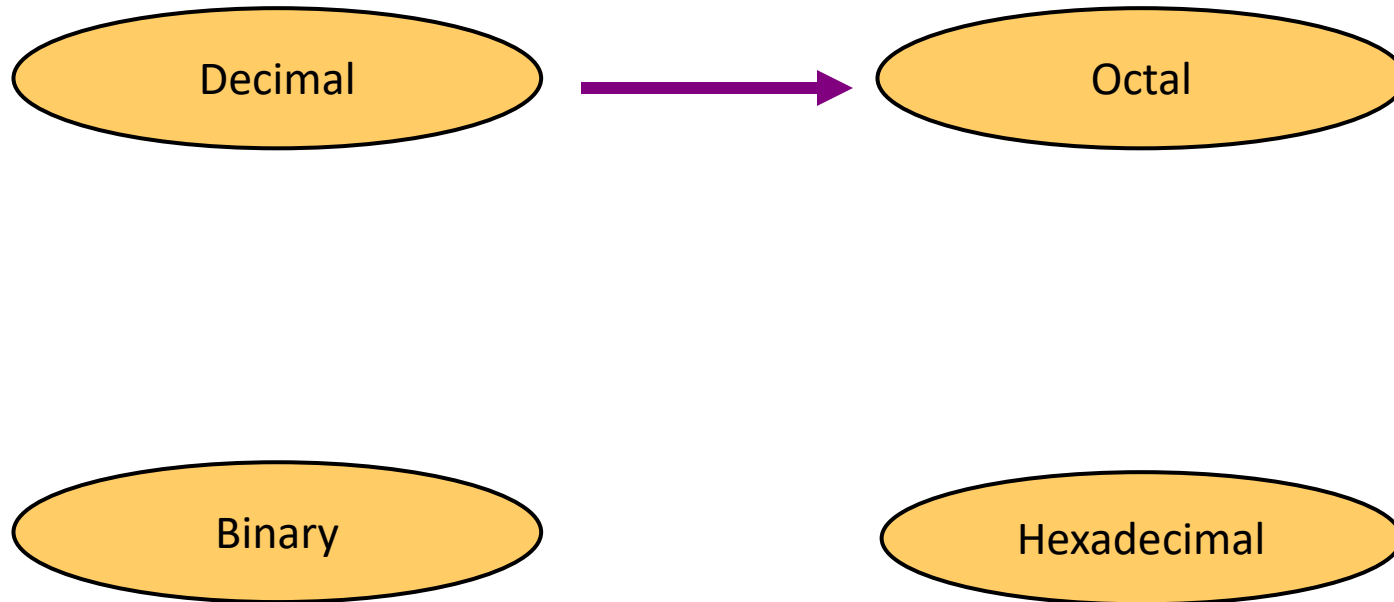
$$10AF_{16} = 0001000010101111_2$$

# Octal to Decimal

- Technique
  - Multiply each bit by  $8^n$ , where  $n$  is the “weight” of the bit
  - The weight is the position of the bit, starting from 0 on the right
  - Add the results



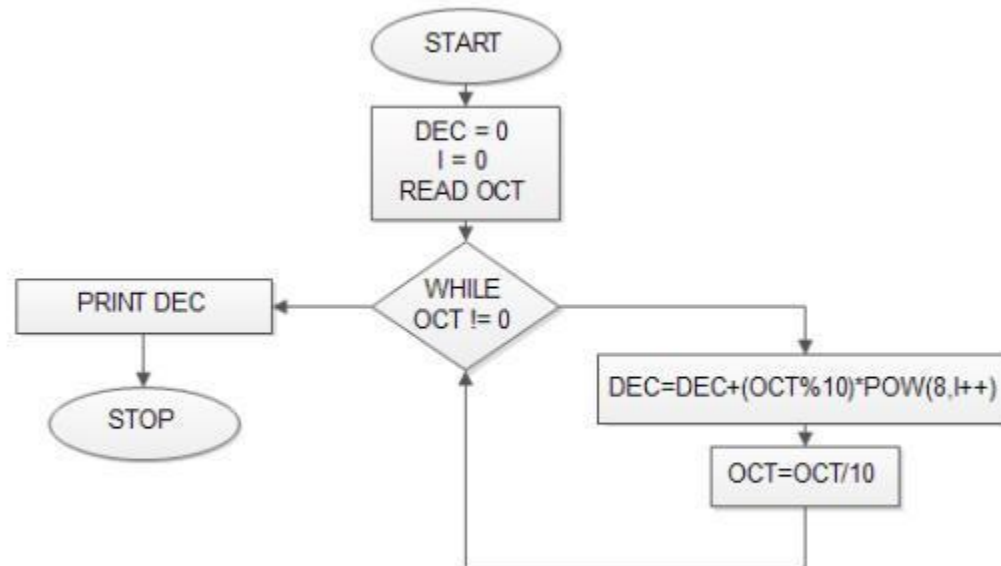
# Decimal to Octal



# Decimal to Octal

- Technique
  - Divide by 8
  - Keep track of the remainder

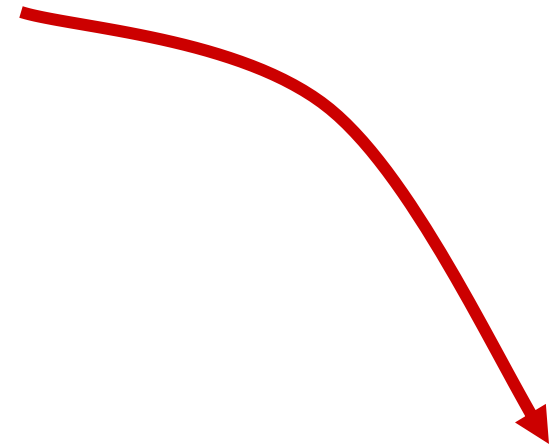
# Decimal to Octal



# Example

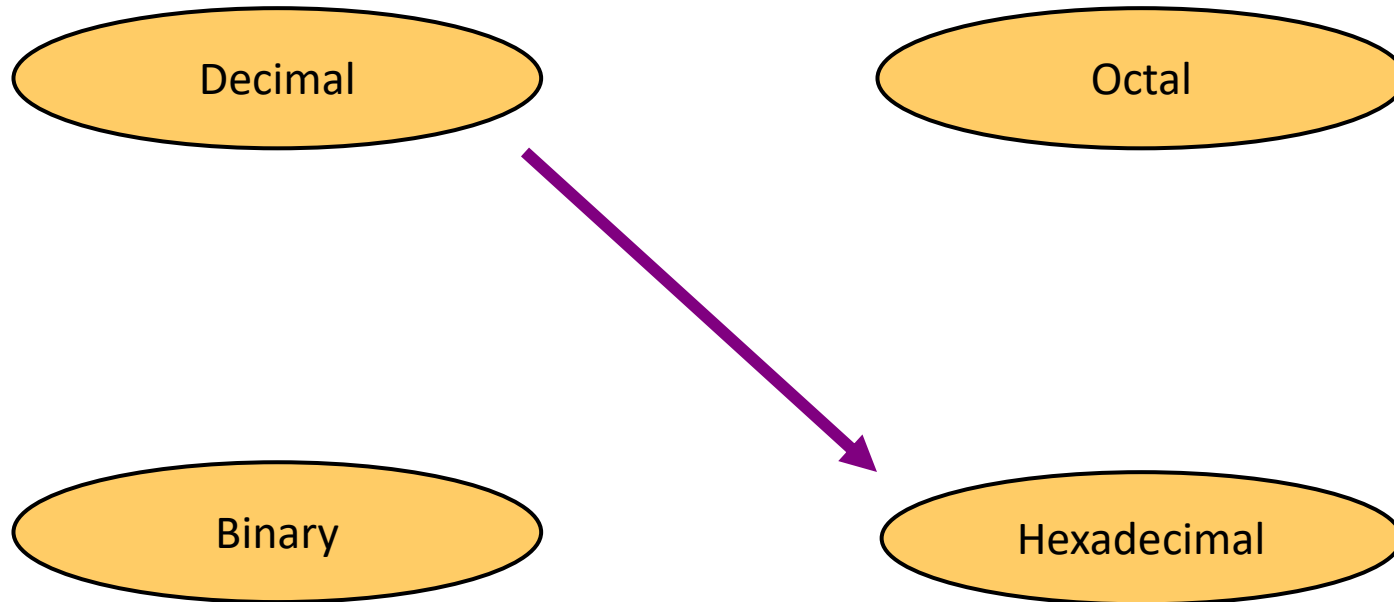
$$1234_{10} = ?_8$$

$$\begin{array}{r|l} 8 & 1234 \\ \hline & 154 \quad 2 \\ 8 & \quad 154 \quad 2 \\ \hline & 19 \quad 2 \\ 8 & \quad 19 \quad 2 \\ \hline & 2 \quad 3 \\ 8 & \quad 2 \quad 3 \\ \hline & 0 \quad 2 \end{array}$$



$$1234_{10} = 2322_8$$

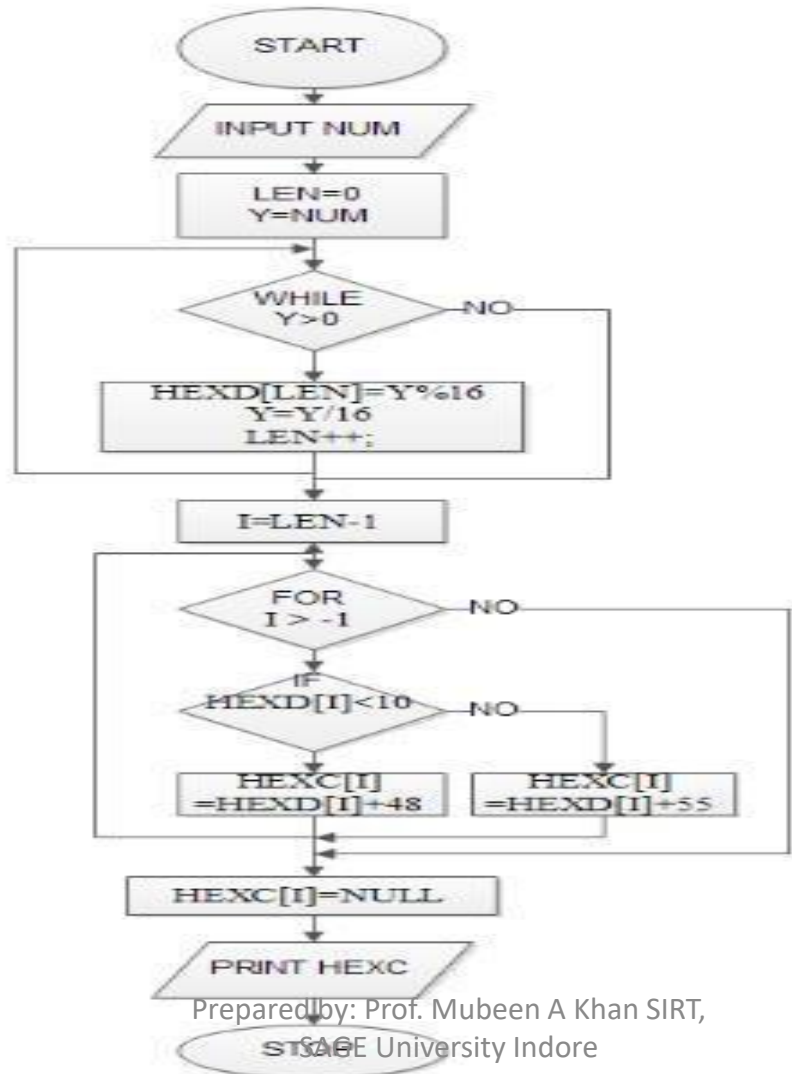
# Decimal to Hexadecimal



# Decimal to Hexadecimal

- Technique
  - Divide by 16
  - Keep track of the remainder

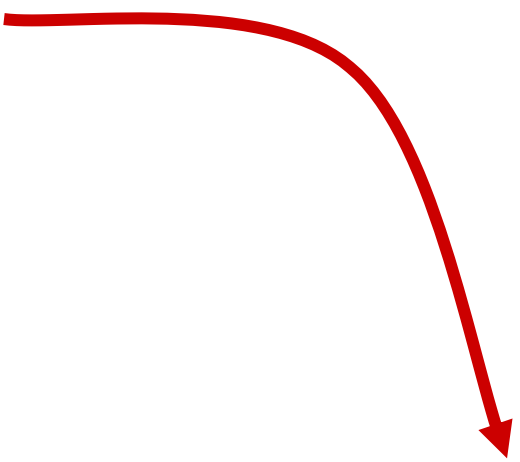
# Decimal to Hexadecimal



# Example

$$1234_{10} = ?_{16}$$

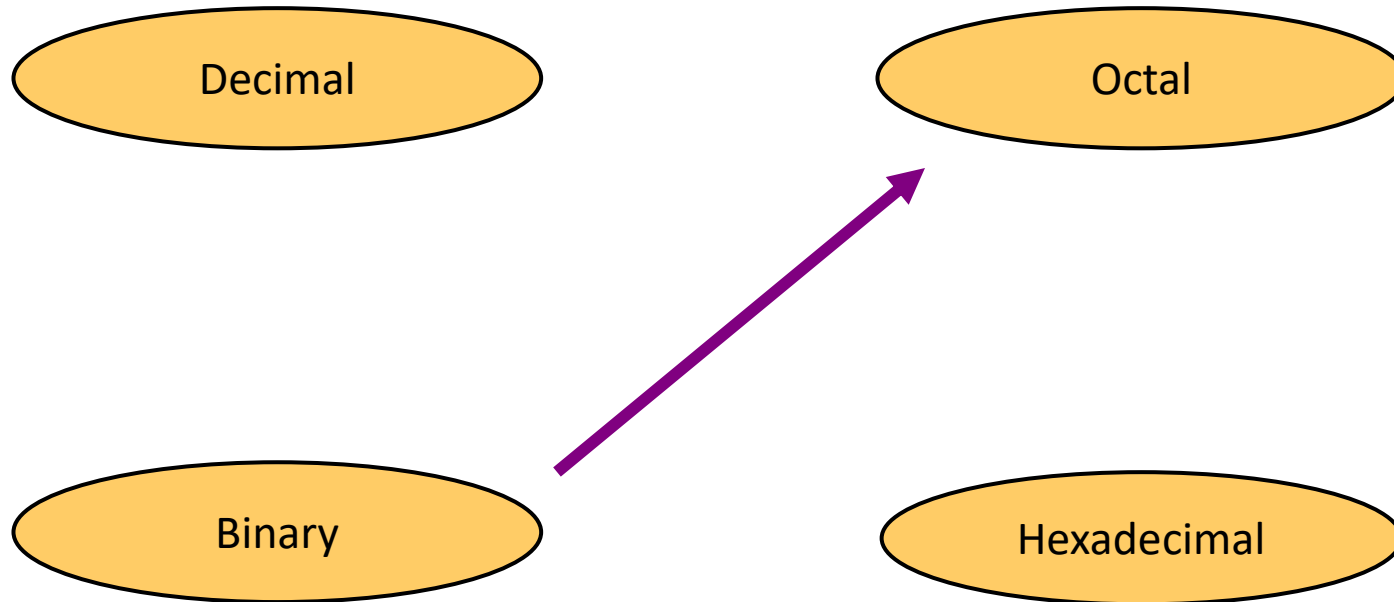
$$\begin{array}{r}
 16 \overline{) 1234} \\
 \underline{77} \phantom{2} \\
 16 \overline{) 413} \\
 \underline{4} \phantom{13} = D \\
 16 \overline{) 04} \\
 \underline{0} \phantom{4}
 \end{array}$$



$$1234_{10} = 4D2_{16}$$



# Binary to Octal

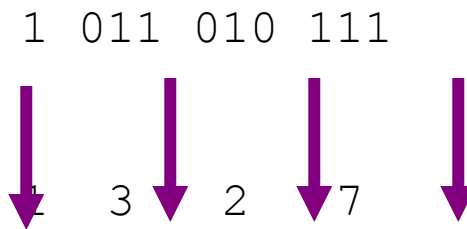


# Binary to Octal

- Technique
  - Group bits in threes, starting on right
  - Convert to octal digits

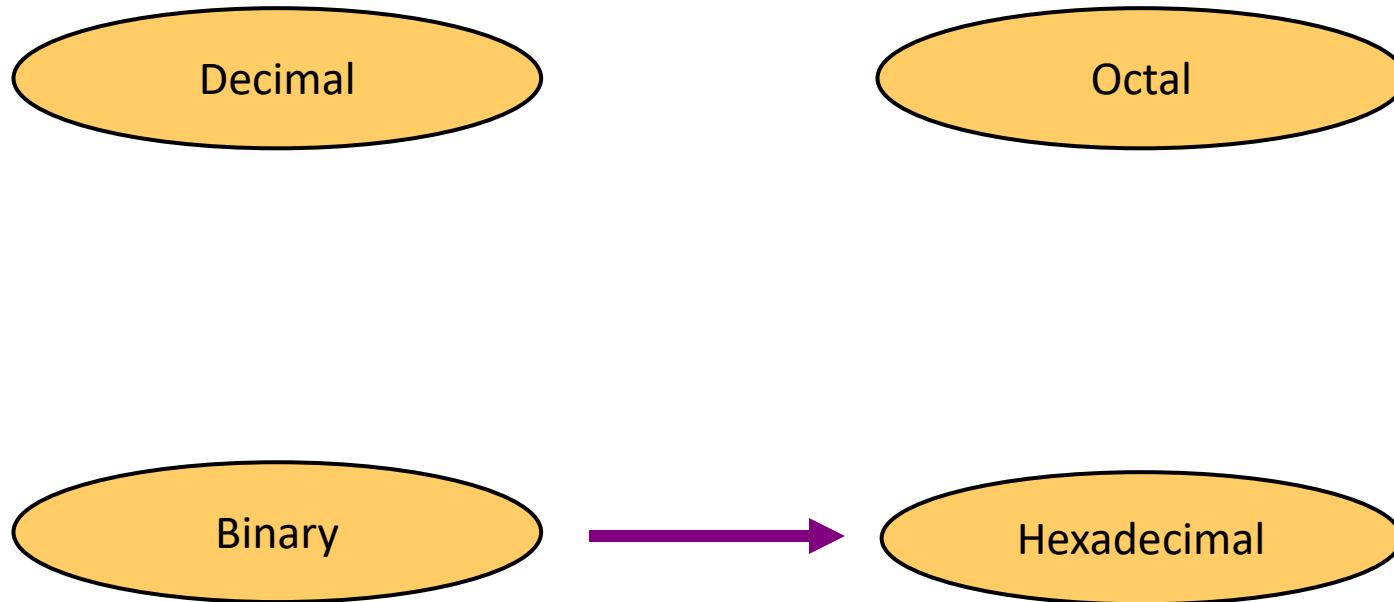
# Example

$$1011010111_2 = ?_8$$



$$1011010111_2 = 1327_8$$

# Binary to Hexadecimal



# Binary to Hexadecimal

- Technique
  - Group bits in fours, starting on right
  - Convert to hexadecimal digits

# Example

$$1010111011_2 = ?_{16}$$

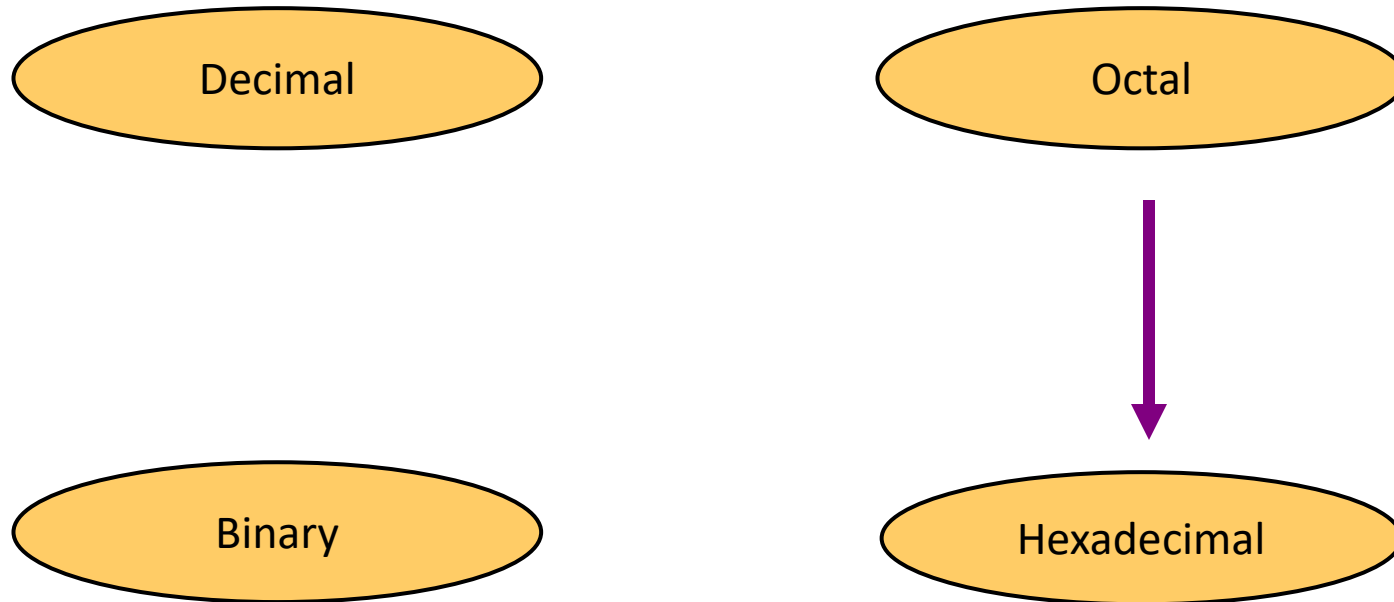
10 1011 1011

↓                      ↓                      ↓

2                      B                      B

$$1010111011_2 = 2BB_{16}$$

# Octal to Hexadecimal



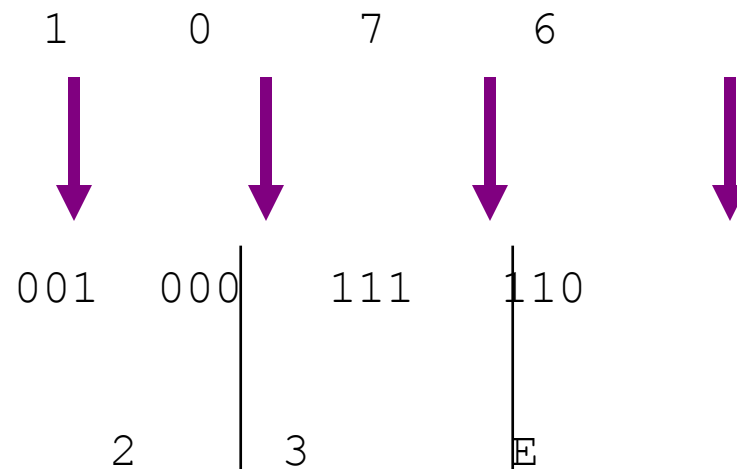
# Octal to Hexadecimal

- Technique
  - Use binary as an intermediary



# Example

$$1076_8 = ?_{16}$$



$$1076_8 = 23E_{16}$$

# THANK YOU

# Day-5 Lab Assignments

- Program for Binary to Decimal
- Program for Decimal to Binary conversions
- Program for Decimal to Octal and
- Program for Octal to Decimal
- Program for Hexadecimal to Octal and
- Program for Octal to Hexadecimal

# Program for Decimal to Binary/Binary to Decimal

```
#include <stdio.h>
int main()
{
    int n, c, k;
    printf("Enter an integer in decimal number
        system\n");
    scanf("%d", &n);
    printf("%d in binary number system is:\n", n);
    for (c = 31; c >= 0; c--)
    {
        k = n >> c;
        if (k & 1)
            printf("1");
        else
            printf("0");
    }
    printf("\n");
    return 0;
}
```

```
#include <stdio.h>
#include <math.h>
int convertBinaryToDecimal(long long n);
int main()
{
    long long n;
    printf("Enter a binary number: ");
    scanf("%lld", &n);
    printf("%lld in binary = %d in decimal", n,
        convertBinaryToDecimal(n));
    return 0;
}
int convertBinaryToDecimal(long long n)
{
    int decimalNumber = 0, i = 0, remainder;
    while (n != 0)
    {
        remainder = n % 10;
        n /= 10;
        decimalNumber += remainder * pow(2, i);
        ++i;
    }
    return decimalNumber;
}
```

# Decimal to Octal/Octal to Decimal

```
#include <stdio.h>
#include <math.h>

int convertDecimalToOctal(int decimalNumber);
int main()
{
    int decimalNumber;

    printf("Enter a decimal number: ");
    scanf("%d", &decimalNumber);

    printf("%d in decimal = %d in octal", decimalNumber,
        convertDecimalToOctal(decimalNumber));

    return 0;
}

int convertDecimalToOctal(int decimalNumber)
{
    int octalNumber = 0, i = 1;

    while (decimalNumber != 0)
    {
        octalNumber += (decimalNumber % 8) * i;
        decimalNumber /= 8;
        i *= 10;
    }

    return octalNumber;
}
```

```
#include <stdio.h>
#include <math.h>

long long convertOctalToDecimal(int octalNumber);
int main()
{
    int octalNumber;

    printf("Enter an octal number: ");
    scanf("%d", &octalNumber);

    printf("%d in octal = %lld in decimal", octalNumber,
        convertOctalToDecimal(octalNumber));

    return 0;
}

long long convertOctalToDecimal(int octalNumber)
{
    long long decimalNumber = 0, i = 0;
    while(octalNumber != 0)
    {
        decimalNumber += (octalNumber%10) * pow(8,i);
        ++i;
        octalNumber/=10;
    }
    i = 1;
    return decimalNumber;
}
```

# Decimal to Hexadecimal/Hexadecimal to Decimal

```
#include<stdio.h>
#include<math.h>
int main()
{
int decimal_number, remainder,
    hexadecimal_number = 0;
int count = 0;
printf("Enter a Decimal Number:\t");
scanf("%d", &decimal_number);
while(decimal_number != 0)
{
Remainder=decimal_number % 16;
hexadecimal_number = hexadecimal_number +
    remainder * pow(10, count);
decimal_number = decimal_number/16;
count++;
}
printf("\nHexadecimal Equivalent:\t%d\n",
    hexadecimal_number);
return 0;
}
```

```
#include <stdio.h>
#include <math.h>
#include <string.h>
int main()
{ char hex[17];
long long decimal, place;
int i = 0, val, len;
decimal = 0;
place = 1;
printf("Enter any hexadecimal number: "); gets(hex);
len = strlen(hex);
len--;
for(i=0; hex[i]!='\0'; i++)
if(hex[i]>='0' && hex[i]<='9')
{ val = hex[i] - 48;
}
else if(hex[i]>='a' && hex[i]<='f')
{ val = hex[i] - 97 + 10; }
else if(hex[i]>='A' && hex[i]<='F')
{ val = hex[i] - 65 + 10; }
decimal += val * pow(16, len);
len--;
}
printf("Hexadecimal number = %s\n", hex);
printf("Decimal number = %lld", decimal);
return 0;
}
```

THANK YOU