Encoding Schemes and Number Systems

Encoding Schemes

- **How does a keyboard work:**
    1. **Feeds in data -> converted into code ->converted into binary**
- **Encoding: The mechanism of converting data into an equivalent cipher using specific code is called encoding**
- **In order to ensure that all keyboards use the same system of encoding, i.e. each key has one value, there are certain standard coding schemes used throughout a region to ensure uniformity:**
    1. **American Standard Code for Information Interchange (ASCII):**
        - **Most commonly used coding scheme**
        - **7 bit code i.e. 128 total possible value**
        - **only applicable to English character set**
        - **A - 65,a-97**
    2. **Indian Script Code for Information Interchange (ISCII):**
        - **To facilitate use of Indian languages**
        - **8 bit code- 256 characters**
        - **Uses 128 ASCII values and uses the rest for Indian language characters.**
        - **Additional codes have been assigned in the upper region (160-255) for *aksharas* of the language**
    3. **UNICODE:**
        - **Due to the presence of several encoding schemes, users with different encoding systems were unable to communicate**
        - **Therefore, a standard called UNICODE was developed to incorporate all existing characters in the world**
        - **Can be used on any device, OS, or software**
        - **Superset of ASCII (0-128 have same values)**
        - **UTF-8,UTF-16,UTF-32**
        - **Hex values**

Number Systems

Key Terms:

1. **Literals:** **A unique set of characters belonging to a particular number system**
2. **Radix:** **The count of literals. Eg: Binary has two literals(0,1) so its radix is 2, i.e. it is a Base-2 system**

Positional Number System:

- **Value of each symbol depends on position**
- **0 position is the number to the immediate left of the decimal.**
- **The positional value is the radix of the number system to the power of the position number**

SYSTEM	RADIX	LITERALS
Binary	2	0,1
Octal	8	0-7
Decimal	10	0-9
Hexadecimal	16	0-9 and A-F

Kinds of Number Systems:

1. **Binary Number System**
    - **Literals: 0,1**
    - **Radix: 2**
2. **Octal Number System:**
    - **Literals: 0,1,2,3,4,5,6,7**
    - **Radix: 8**
3. **Decimal Number System:**
    - **Literals: 0,1,2,3,4,5,6,7,8,9**
    - **Radix: 10**
4. **Hexadecimal Number System:**
    - **Literals: 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F**
    - **Radix: 16**

CONVERSION CHEAT SHEET

Can be made in exams within a minute

Decimal	Binary	Octal	Hexadecimal
0	0000	0	0
1	0001	1	1
2	0010	2	2
3	0011	3	3
4	0100	4	4
5	0101	5	5
6	0110	6	6
7	0111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
14	1101	15	D
14	1110	16	E
15	1111	17	F

Applications of Hexadecimal:

1. **Makes it easier to memorise a memory address (eg 1100000011110001 == C0F1)**
2. **Used to describe colors in RGB format**

Conversions Between Systems

1. **Decimal to other**
    1. **Divide the number by the radix of the system**
    2. **Note the remainder**
    3. **Keep on dividing until the quotient is zero**
    4. **write remainders in reverse order**
2. **Other to decimal:**
    1. **Write the position number for each literal**
    2. **Get positional value of each literal by raising its radix to the power of its positional number**
    3. **Multiply each literal with the respective positional value to get a decimal value**
    4. **Add all to get the number**
3. **Other to Other:**
    - **Binary to Octal:**
        - **make groups of 3s and find respective octal code for each group**
    - **Octal to binary:**
        - **Find the 3 bit binary value for each octal number**
    - **Binary to Hexadecimal:**
        - **Split binary digits into groups of four**
        - **Write respective codes for each**
        - **0001 1010 1100 == 1 A C**
    - **Hexadecimal to Binary:**
        - **Find the 4 bit binary value for each hexadecimal character**
4. **Decimal with fraction to other number systems:**
    1. **Repeatedly multiply the fractional part by the radix until the fractional part becomes 0.**
    2. **if it does not become zero, then stop after 10 multiplications**
5. **Other to decimal in fraction:**
    1. **use position value method with negative powers**

6. **Fractional Binary to Octal or Hexadecimal:**
    1. **For octal make groups of 3s and find respective binary code**
    2. **For hex make groups of 4 and find respective binary code**