Assignment No. 5

Numbers / Loops

1. Write a program to check whether a given number is a **happy number**.
(A happy number eventually reaches 1 when replaced by the sum of the squares of its digits repeatedly.)

2. Write a function to find the **nth Fibonacci number** using recursion.

3. Write a program to count how many numbers between 1 and 100 are **palindromes**.

4. Write a program to find the **smallest** and **largest** numbers in a given list without using built-in functions like `min()` or `max()`.

5. Write a function to check whether a given number is a **Kaprekar number**.
(A number whose square can be split into two parts that add up to the original number.)

Strings

6. Write a program to find all **unique substrings** of a given string.

7. Write a function to reverse each word in a sentence **without changing the word order**.
*(Example: `"Python is fun"` → `"nohtyP si nuf"`)

8. Write a function to check if a string is a **pangram**.
(A pangram contains every letter of the English alphabet at least once.)

9. Write a program to count the number of **words**, **digits**, and **special characters** in a string.

10. Write a function to **compress a string** using counts of repeated characters.
*(Example: `"aaabbcddd"` → `"a3b2c1d3"`)

Patterns / Logic

11. Write a program to print an **inverted pyramid** of stars.
(For n=5)

```
*****
****
***
```

```
**
*
```

12. Write a program to generate a **diamond-shaped star pattern**.
(For n=3)

```
  *
 ***
*****
 ***
  *
```

13. Write a function to print a **spiral matrix** of size n × n.
(For n=3)

```
1  2  3
8  9  4
7  6  5
```

14. Write a function to check if a **Sudoku** row, column, or grid is valid.
(Input: a list of 9 numbers, output: True/False)

15. Write a program to create a **number pyramid** where each row contains consecutive integers:
(For n=4)

```
1
2 3
4 5 6
7 8 9 10
```