Control Statement: Complex Version

Problem 1: Matrix Boundary Words Concatenation

Problem Statement: Given a 2D matrix of strings, write a program to concatenate the boundary elements of the matrix.

Input:

- An integer `n` representing the number of rows and columns (matrix is n x n).
- A 2D array `matrix` of strings.

Output:

- A string representing the concatenation of the boundary elements.

Sample Input:

```
4
a b c d
e f g h
i j k l
m n o p
```

Sample Output:

```
abcdhlponmiej
```

Problem 2: Longest Palindromic Substring

Problem Statement: Given a string, write a program to find the longest palindromic substring.

Input:

- A string `s`.

Output:

- A string representing the longest palindromic substring.

Sample Input:

```
babad
```

Sample Output:

```
bab
```

Problem 3: Anagram Groups

Problem Statement: Given an array of strings, group the anagrams together.

Input:

- An integer `n` representing the number of elements in the array.
- An array `arr` of strings.

Output:

- A 2D array where each subarray contains strings that are anagrams of each other.

Sample Input:

```
6
eat tea tan ate nat bat
```

Sample Output:

```
eat tea ate
tan nat
bat
```

Problem 4: Zigzag Conversion

Problem Statement: Given a string and a number of rows, write a program to print the string in a zigzag pattern.

Input:

- A string `s`.
- An integer `numRows`.

Output:

- A string representing the zigzag conversion.

Sample Input:

```
PAYPALISHIRING
3
```

Sample Output:

```
PAHNAPLSIIGYIR
```

Problem 5: Find Missing Number

Problem Statement: Given an array of `n-1` integers in the range from 1 to `n`, write a program to find the missing number.

Input:

- An integer `n` representing the total number of elements.
- An array `arr` of `n-1` integers.

Output:

- An integer representing the missing number.

Sample Input:

```
5
1 2 4 5
```

Sample Output:

```
3
```

Problem 6: Reverse Words in a Sentence

Problem Statement: Given a string sentence, reverse the order of words.

Input:

- A string `sentence`.

Output:

- A string with the words in reverse order.

Sample Input:

```
The quick brown fox
```

Sample Output:

```
fox brown quick The
```

Problem 7: Largest Number Formed From Array

Problem Statement: Given an array of non-negative integers, arrange them such that they form the largest number.

Input:

- An integer `n` representing the number of elements in the array.
- An array `arr` of integers.

Output:

- A string representing the largest number formed.

Sample Input:

```
5
3 30 34 5 9
```

Sample Output:

```
9534330
```

Problem 8: Product of Array Except Self

Problem Statement: Given an array of integers, write a program to return an array such that each element at index `i` is the product of all the numbers in the original array except the one at `i`.

Input:

- An integer `n` representing the number of elements in the array.
- An array `arr` of integers.

Output:

- An array of integers representing the product of array except self.

Sample Input:

```
4
1 2 3 4
```

Sample Output:

```
24 12 8 6
```

Problem 9: Longest Common Prefix

Problem Statement: Given an array of strings, write a program to find the longest common prefix string amongst an array of strings.

Input:

- An integer `n` representing the number of elements in the array.
- An array `arr` of strings.

Output:

- A string representing the longest common prefix.

Sample Input:

```
3
flower flow flight
```

Sample Output:

```
fl
```

Problem 10: Evaluate Reverse Polish Notation

Problem Statement: Given an array of strings where each string is either an operand or an operator, write a program to evaluate the expression in Reverse Polish Notation (RPN).

Input:

- An integer `n` representing the number of elements in the array.
- An array `arr` of strings representing the RPN expression.

Output:

- An integer representing the result of the RPN expression.

Sample Input:

```
5
2 1 + 3 *
```

Sample Output:

```
9
```