

MTH 4300: Algorithms, Computers and Programming II

Fall 2025

Section: STRA

Midterm 1 (October 9th, 2025)

Grading Section (For Instructor Use)

| Question   | Points | Score |
|------------|--------|-------|
| Question 1 | 10     | / 10  |
| Question 2 | 10     | / 10  |
| Question 3 | 10     | / 10  |
| Question 4 | 10     | / 10  |
| Total      | 40     | / 40  |

Comments:

## Question 1 (10 points)

Implement a function that processes text content based on different operation modes using enums. The function should handle various text transformations and return processed results.

You can assume that the strings will only be separated by spaces.

### Function Signature:

```
enum ProcessMode { FIND_SHORTEST_WORD, FIND_LONGEST_WORD };  
std::string processContent(const std::string& content, ProcessMode mode);
```

### Operation Descriptions:

- **FIND\_SHORTEST\_WORD**: Return the shortest word in the content
- **FIND\_LONGEST\_WORD**: Return the longest word in the content

### Sample Output:

Input: content = "Hello World Test", mode = FIND\_SHORTEST\_WORD

Output: "Test"

Input: content = "Hello World Programming", mode = FIND\_LONGEST\_WORD

Output: "Programming"

## Question 2 (10 points)

Implement a recursive function to solve the Collatz Conjecture ( $3n+1$  problem). The function should return the number of steps it takes to reach 1. You may not augment the initial function signature, but you're free to add more functions to solve the problem.

### Function Signature:

```
int collatzSteps(int n);
```

### Rules:

- If  $n$  is even: divide by 2
- If  $n$  is odd: multiply by 3 and add 1
- Continue until  $n$  reaches 1

### Sample Output:

```
Input: n = 4  
Steps: 4 → 2 → 1  
Output: 2
```

```
Input: n = 5  
Steps: 5 → 16 → 8 → 4 → 2 → 1  
Output: 5
```

```
Input: n = 1  
Output: 0 (already at 1)
```

### Question 3 (10 points)

Given a string, compress it using run-length encoding. Replace consecutive identical characters with the character followed by its count. You can assume there will only be lower case alphabetic characters in the string.

#### Function Signature:

```
std::string runLengthEncode(const std::string& input);
```

#### Sample Output:

```
Input: input = "aaabbc"  
Output: "a3b2c1"
```

```
Input: input = "hello"  
Output: "h1e1l1o1"
```

```
Input: input = "aabbcc"  
Output: "a2b2c2"
```

```
Input: input = "xyz"  
Output: "x1y1z1"
```

#### Question 4 (10 points)

Write a function that rotates a 1D array representing a square matrix 90 degrees clockwise. The matrix is stored in row-major order.

**Function Signature:**

```
void rotateMatrix90(int matrix[], int size);
```

**Note:** For a square matrix of dimension  $\text{size} \times \text{size}$ , element at position  $(i, j)$  is stored at index  $i * \text{size} + j$  in the 1D array.

**Sample Output:**

Input: matrix = [1, 2, 3, 4] (represents 2×2), size = 2

Original: [[1,2], [3,4]] → Rotated: [[3,1], [4,2]]

Output: (if you print the matrix it would contain) [3, 1, 4, 2]

Input: matrix = [1, 2, 3, 4, 5, 6, 7, 8, 9] (represents 3×3), size = 3

Original: [[1,2,3], [4,5,6], [7,8,9]] → Rotated: [[7,4,1], [8,5,2], [9,6,3]]

Output: (if you print the matrix it would contain) [7, 4, 1, 8, 5, 2, 9, 6, 3]