Islamic University of Technology

Lab 11: Recap After Mid

CSE 4108 : Structured Programming Lab

Section 1B

MIA, NFS, IAC, ASH

February 20, 2025

1 Lab Tasks

1.1 MirrorFlip: The Recursive String Reversal Challenge

Write a recursive function to check if a string is palindrome or not. The function should take a string as input and return 1 if the string is palindrome or return 0. Avoid using loops—let recursion do all the work!. You are also restricted to use string.h headers and string related functions

Design the input and output accordingly to match the task.

```
int isPalindrome(char str[], int start, int end);
```

[Sample Program]
Input 1

```
hello
```

Output 1

```
No
```

Input 2

```
madam
```

Output 2

```
Yes
```

1.2 Bubble Tea and Sort

Sorting is a very common sub-task in almost any programming applications. One of the most naive or simple way to sort a series of data in an array is to use the Bubble Sort algorithm. As the name suggest, the algorithm kind of works like water bubbles floating to the surface. It works by repeatedly swapping adjacent elements that are out of order. A pseudo-code for bubble sort for sorting in ascending order (low to high) is given next.

```
// A is the array containing the values
// A.length means the length of the array
BubbleSort(A):
    for i = 0 to A.length - 1
        for j = A.length downto i + 1
            if A[j] < A[j-1]
                swap A[j] with A[j-1]
```

Now, Write a bubble sorting function that would take an array of integers and its length and sort the array in ascending order using Bubble Sort. From the main function print the sorted array. You may use VLAs or consider that the length of the array will not exceed 200.

```
void bubbleSort(int arr[], int length);
```

[Sample Program]
Input 1

```
Enter array length: 5
Enter values for the array: 1 3 2 -1 2
```

Output 1

```
Sorted array: {-1, 1, 2, 2, 3}
```

Input 2

```
Enter array length: 8
Enter values for the array: 9 8 7 6 5 4 3 2
```

Output 2

```
Sorted array: {2, 3, 4, 5, 6, 7, 8, 9}
```

[Hint] *Bubble sort sorts the array in place, meaning it modifies the original array.*

1.3 Later Letter

Write a C program that would read a sentence from the user and then print out the count of each of the letters. Lower and upper case letter should be treated as same. Punctuation marks and other white space characters should be handled carefully. You can try to make the output pretty.

You can think that the sentence length will never exceed 200 and then read character by character until a newline is met and store it in the array.

[Sample Program]
Input

```
Enter a sentence: Hello World
```

Output

```
Letter counts:
A : 0   B : 0   C : 0   D : 1   E : 1
F : 0   G : 0   H : 1   I : 0   J : 0
K : 0   L : 3   M : 0   N : 0   O : 2
P : 0   Q : 0   R : 1   S : 0   T : 0
U : 0   V : 0   W : 1   X : 0   Y : 0
Z : 0
```

[Hint] *You can utilize useful functions from* `<ctype.h>` *like* `isalpha(char)` *and/or* `tolower(char)` *or* `toupper(char)` *to aid in implementing this program. This problem is very similar to digit counting or repeated digit detection problem.*

1.4 Omni-Directional

Write a program that reads `N` an integer and then a `N` × `N` array of integers from the user and then prints the row sums and the column sums.

You may use VLAs or consider that `N` will not be more than 20.

[Sample Program]
Input

```
Enter N: 5
Enter row 1: 8 3 9 0 10
Enter row 2: 3 5 17 1 1
Enter row 3: 2 8 6 23 1
Enter row 4: 15 7 3 2 9
Enter row 5: 6 14 2 6 0
```

Output

```
Row totals: 30 27 40 36 28
Col totals: 34 37 37 32 21
```