

# **Practice for the Final Exam**

**Dohyung Kim**

# Problem 1

- The function named “getMaxMin” should compute max and average from an input array.
  - Please complete the code to run like the screenshot below
  - When you traverse the array, pointer arithmetic should be used instead of array index
  - Do not change the other part of the program

```

#include <stdio.h>
#include <stdlib.h>

void getMaxMin(float* list, int N, float* max, float*
min){
    /* complete the code */
    ...
}

void main() {
    int i, N;
    float *a, max, min;
    printf("how many elements will you input? ");
    scanf("%d", &N);
    a = malloc(sizeof(float)*N);
    for(i=0; i<N; i++){
        printf("Input [%d] th float value:", i+1 );
        scanf("%f", &a[i]);
    }
    getMaxAvg(a, N, &max, &min);
    printf("Max : %f, Min: : %f\n", max, min);
    free(a);
    return;
}

```

# Problem 2

- Write a program that finds the common prefix of the two input strings
  - Do not change the other part of the program

```
Input two strings: GoodMorning GoodBye
The longest common prefix is Good
```

```
#include<stdio.h>
#include<string.h>
#define MAX_LENGTH 100

void getLCP(char* x, char* y, char* res){
    /* complete the code */
    ...
}

void main(){
    char X[MAX_LENGTH];
    char Y[MAX_LENGTH];
    char LCP[MAX_LENGTH];

    printf("Input two strings: ");
    scanf("%s %s", X, Y);

    getLCP(X, Y, LCP);
    printf("The longest common prefix is %s\n", LCP);

    return;
}
```

# Problem 3

- This is a part of the program that adds two matrices. The main function A) takes elements of each matrix from the user and B) calls the function named “addMatrix”. Please complete the program.

```
What is the demension of input matrices : 2
Input the matrix A.
A[1][1]: 0
A[1][2]: 1
A[2][1]: 2
A[2][2]: 3
Input the matrix B.
B[1][1]: 10
B[1][2]: 10
B[2][1]: 20
B[2][2]: 20
The result matrix is:
10 11
22 23
```

```
#include<stdio.h>
#include<stdlib.h>

int* addMatrix(int* a, int* b, int N){
    /* complete the code */
    ...
    ...
}
```

```
void main(){
    int *matA, *matB, *matC, n;
    int i,j;

    printf("What is the demension of input matrices : ");
    scanf("%d", &n);

    matA = malloc(sizeof(int)*n*n);
    matB = malloc(sizeof(int)*n*n);

    printf("Input the matrix A.\n");
    /*complete the code*/
    ...

    printf("Input the matrix B.\n");
    /*complete the code*/
    ...

    matC = addMatrix(matA, matB, n*n);
    printf("The result matrix is: \n");
    /*complete the code*/
    ...

    return;
}
```

# Problem 4

- Please implement the function named “myStrCmp” having the same functionality with strcmp.
  - cf. The strcmp(string1, string2) returns
    - 0 if string1 is equal to string2
    - -1 if string1 appears before string2 in lexicographical order
    - 1 if string1 appears after string2 in lexicographical order
- The parameters of myStrCmp(), string1 and string2, have an arbitrary length.
- You must not use the built-in strcmp() in the standard C library
- Do not change the other part of the program

Input the first string: Good morning  
Input the second string: Good afternoon  
Good morning appears after Good afternoon

Input the first string: Good afternoon  
Input the second string: Good morning  
Good afternoon appears before Good morning

```
#include<stdio.h>
int myStrCmp(char *X, char *Y)
{
    /* complete the code */
    ...
}

int main()
{
    char X[100];
    char Y[100];

    printf("Input the first string: ");
    gets(X);
    printf("Input the second string: ");
    gets(Y);

    int ret = myStrCmp(X, Y);
    if (ret < 0)
        printf("%s appears before %s", X, Y);
    else if (ret > 0)
        printf("%s appears after %s", X, Y);
    else
        printf("%s", "X is equal to Y");
    printf("\n");
    return 0;
}
```