Algorithms – Assignment 2 (Solution)

(Basic Coding Questions)

Prof. Eunwoo Kim

Due: 9th April 2021

Submit a pdf file to the e-class that contains your answers for the following exercises including codes written in the C or Python programming language (screenshots okay).

[Solutions written in the C programming language]

1) [Programming] Write a program that takes a number n and displays the largest positive integer k satisfying the following equations: $2^k \leq n$. Display the results for three different n's: 10, 50, and 1025.

Solution:

```
#include <stdio.h>
int main() {
	int num, count = 0, temp = 2;
	printf("Input an integer : ");
	scanf("%d", &num);
	while (1) {
		if (temp > num) break;
		temp *= 2;
		count++;
	}
	printf("The maximum value of k is %d:", count);
	return 0;
}
```

2) [Programming] Palindrome refers to words that have the same results when we read from the beginning and read from the end, such as level, bob, and radar. Write a function that determines if the given word is palindrome or not. Display the results when you put two different words (one is palindrome and the other is not).

Solution:

```
#include <stdio.h>
#include <string.h>
void palinFunc(char arr[], int len) {
	int i, j = len - 1;
	for (i = 0; i < len / 2; i++) {
		if (arr[i] != arr[j]) {
			printf("%s is not palindrome.₩n", arr);
			return;
		}
		j--;
	}
	printf("%s is palindrome.₩n", arr);
	return;
}
int main(void) {
	char ary[] = "madam";
	int aryLen;
	aryLen = strlen(ary);
	palinFunc(ary, aryLen);
	return 0;
}
```

3) What is the output of the following code? The code shows the partial lines in a complete program. (Just give an answer, not a programming question)

```
int x = 3, y = 2, z = 5;
printf("%d₩n", ((x > y) ? x : y) > z ? ((y > x) ? x : y) : z);
```

Solution: 5

The syntax - *condition ? expression1 : expression2*
is equivalent to

```
if (condition)
        expression1;
else
        expression2;
```

The conditional operator evaluates one of two expressions, depending on whether the condition is true or false.

4) [Programming] Write a program that search for the integer 120 in the following list of integers using the binary search algorithm.

12 34 37 45 57 82 99 120 134

Solution:

```
#include <stdio.h>

int binarySearch(int arr[], int l, int r, int x)
{
    if (r >= l) {
        int mid = l + (r - l) / 2;

        if (arr[mid] == x)
            return mid;
        if (arr[mid] > x)
            return binarySearch(arr, l, mid - 1, x);
        return binarySearch(arr, mid + 1, r, x);
    }
    return -1;
}
int main(void)
{
    int arr[] = { 12, 34, 37, 45, 57, 82, 99, 120, 134 };
    int n = sizeof(arr) / sizeof(arr[0]);
    int x = 120;
    int result = binarySearch(arr, 0, n - 1, x);
```

```
    (result == -1) ? printf("Element is not present in array")
                   : printf("Element is present at index %d", result);
    return 0;
}
```

5) [Programming] Given an image represented by a 5×5 matrix, write a method to rotate the image by 90 degrees (clockwise). You can generate a matrix randomly.

Solution:

```
def rotate_matrix(matrix):
    '''rotates a matrix 90 degrees clockwise'''
    n = len(matrix)
    for layer in range(n // 2):
        first, last = layer, n - layer - 1
        for i in range(first, last):
            # save top
            top = matrix[layer][i]
            # left -> top
            matrix[layer][i] = matrix[-i - 1][layer]
            # bottom -> left
            matrix[-i - 1][layer] = matrix[-layer - 1][-i - 1]
            # right -> bottom
            matrix[-layer - 1][-i - 1] = matrix[i][- layer - 1]
            # top -> right
            matrix[i][- layer - 1] = top
    return matrix

print(your_matrix)
print(rotate_matrix(your_matrix))
```

6) [Programming] Write a function to find all pairs of an integer array whose sum is equal to a given number.

Function: pairSum([2, 4, 3, 5, 6, -2, 4, 7, 8, 9], 7)

Output: ['2+5', '4+3', '3+4', '-2+9']

Solution:

```
def findPairs(list, sum):
    for i in range(len(list)):
        for j in range(i+1,len(list)):
            if (list[i]+list[j]) == sum:
                print(list[i],list[j])
# findPairs(mylist, 7)
```