

GIT Department of Computer Engineering
CSE 222/505 - Spring 2020

Homework 2

Upload Due date: March 13 2020 – 12:00

(You will both bring your handwritten solutions to Nur Banu Albayrak (118) and upload the scanned versions to Moodle. You can bring the handwritten solutions due to the end of March 13.)

PART 1 : Analyze the following algorithms. Write worst-case, average-case, best-case analysis if significant. Express your results using most proper asymptotic notation. Explain your solutions. For the 1st to the 4th algorithms use table method and show table.

1.

```
somefunction(rows, cols)
{
    for(i = 1; i <= rows; i++)
    {
        for( j = 1; j <= cols; j++)
            print(*)

        print(newline)
    }
}
```

2.

```
somefunction(a, b)
{
    if (b == 0)
        return 1
    answer = a
    increment = a
    for(i = 1; i < b; i++)
    {
        for(j = 1; j < a; j++)
        {
            answer += increment
        }
        increment = answer
    }
    return answer
}
```

3.

```

somefunction(arr[], arr_len)
{
    val = 0

    for (i = 0; i < arr_len / 2; i++)
        val = val + arr[i]

    for (i = n / 2; i < arr_len; i++)
        val = val - arr[i]

    if (val >= 0)
        return 1
    else
        return -1
}

```

4.

```

somefunction(n)
{
    c = 0
    for (i = 1 to n*n)
        for (j = 1 to n)
            for (k = 1 to 2*j)
                c = c+1

    return c
}

```

5.

```

otherfunction(xp, yp)
{
    temp = xp
    xp = yp
    yp = temp
}

somefunction(arr[], arr_len)
{
    for (i = 0; i < arr_len - 1; i++)
    {
        min_idx = i

        for (j = i+1; j < arr_len; j++)
            if (arr[j] < arr[min_idx])
                min_idx = j

        otherfunction(arr[min_idx], arr[i])
    }
}

```

6.

```
otherfunction(a, b)
{
    if b == 0:
        return 1

    answer = a
    increment = a

    for i = 1 to b:
    {
        for j = 1 to a:
            answer += increment

        increment = answer
    }
    return answer
}

somefunction(arr, arr_len)
{
    for i = 0 to arr_len):
        for j = i to arr_len):
            if otherfunction(arr[i], 2) == arr[j]:
                print(arr[i], arr[j])
            elif otherfunction(arr[j], 2) == arr[i]:
                print(arr[j], arr[i])
}
```

7.

```
otherfunction(X, i)
{
    s = 0
    for(j = 1; j <= i; j=j*2)
        s = s + X[j]
    return s
}

somefunction(arr[], arr_len)
{
    for(i = 0; i <= arr_len-1; i++)
        A[i] = otherfunction(arr, i) / (i + 1)
    return A
}
```

8.

```
somefunction(n)
{
    res = 0
    j = 1

    if(n < 10)
        return n + 10

    for(i = 9; i > 1; i--)
        while (n % i == 0)
            n = n / i
            res = res + j * i
            j *= 10

    if(n > 10)
        return -1
    return res
}
```

PART 2 : Design an algorithm for each of the problems. Write your algorithms in pseudo code. Obtain the complexities of the algorithms. Write worst-case, average-case, base-case analysis if significant. Express your results using most proper asymptotic notation. Explain your solutions.

1. Assume you have an array of points in 2d space. Find the closest point in the array to a given point.
2. The i^{th} element of an array A is a local minimum if, $A[i] \leq A[i+1]$ and $A[i] \leq A[i-1]$.
 - a. Find a local minimum in a given array A.
 - b. Find all local minimums in a given array A.
3. Find if a given array of integers contains two numbers whose sum is a given number b.
4. A sequence of positive integers in increasing order, a_1, a_2, \dots, a_n is called a "Sum Chain of Length n" if for all k ($1 < k \leq n$), there exist i, j ($1 \leq i \leq j \leq k$) such that $a_k = a_i + a_j$

Example: {1, 2, 3, 5, 10, 13, 15} : (2=1+1, 3=2+1, 5=3+2, 10=5+5, 13=10+3, 15=10+5)

Find if a given sequence of n numbers is a "Sum Chain of Length n". Use the algorithm you design for the third question in this part.