

## تمرینات سری اول درس ساختمان های داده

1. پیچیدگی زمانی کد های زیر را بدست آورید:

(الف)

```
int function(int number)
{
    int sum = 0;
    for(int i = 0; i < 1000000; i++)
    {
        sum += number;
    }
    return sum;
}
```

(ب)

```
for(int i = 0; i < n; i++)
{
    for(int j = 0; j < n; j++)
    {
        std::cout << "Hello there\n";
    }
}
```

(پ)

```
int sum = 5;
for(int i = 0; i < n; i++)
{
    for(int j = 0; j < i; j++)
    {
        for(int k = 0; k < 5; k++)
        {
            sum++;
        }
    }
}
```

(ت)

```
for(int i = 1; i < n; i *= 2)
{
    for(int j = 1; j < i; j *= 3)
    {
        print("Hello, world!");
    }
}
```

(ث)

```
void function(int number)
{
    if(number > 0)
    {
        for(int i = 0; i < number; i++)
        {
            print(number);
        }

        function(number - 1);
    }
}
```

2. پیچیدگی زمانی روابط بازگشتی زیر را بدست آورید. ( حداقل یک مورد را با درخت بازگشت حل کنید. )

a)  $T(n) = T\left(\frac{n}{2}\right) + \lg(n!)$

b)  $T(n) = 2T\left(\frac{n}{2}\right) + 2n \lg n$ ,  $T(2) = 4$

c)  $T(n) = 2T\left(\frac{n}{2}\right) + n \lg \lg n$

d)  $T\left(\frac{n}{2}\right) + T\left(\frac{n}{3}\right) + n$

e)  $T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\log n}$

f)  $T(n) = 4T\left(\frac{n}{2}\right) + (\lg n!)^2$

g)  $T(n) = T(\lg n) + n$

3. درستی یا نادرستی عبارت زیر را با ذکر دلیل مشخص نمایید:  
 «الگوریتم A و B هر دو در تمامی حالات ( Best, Worst, Average ) دارای پیچیدگی زمانی  $O(n)$  هستند. بنابراین سرعت اجرا (Execution Time) این دو الگوریتم در تمامی حالات با هم برابر است.»

4. درستی یا نادرستی عبارات زیر را مشخص کنید.

1.  $3n^3 + n^2 = \theta(n^3)$

2.  $20n^2 + 100 = \Omega(n^2)$

3.  $n^2 \log n = O(n^2)$

4.  $n^{3^n} + 10 \times 3^n = O(n^{3^n})$

5.  $n = O(n \log n)$

6.  $n^2 = O(n \log n)$

5. تعداد مراحل برنامه ی زیر را محاسبه کنید. ( جمع تعداد اجرای دستورات )

```
int[][] function(int n, int m)
{
    int array[n][m];
    for(int i = 0; i < n; i++)
    {
        array[i][0] = 0;
        for(int j = 1; j < m - 2; j++)
        {
            array[i][j] = random(0, 5);
        }
    }

    return array;
}
```

6. توابع زیر را از نظر مرتبه رشد مرتب کنید.

$$lglgn, lglg^2n, lglgn^2, \sqrt{lgn}, lg^2lgn$$

7. مرتبه اجرایی کد زیر را بر حسب  $\theta$  بدست آورید.

```
# K-way merge sort, K is a constant
# N = len(A)
def merge_sort_k(A, k):
    if len(A) == 1:
        return A
    j = ceil(len(A)/k) # j = N/k

    B = [[] for i in range(k)] # [], [], ...
    for i in range(0, len(A), step=j):
        B[i] = merge_sort_k(A[i:i+j], k)

    result = []
    for array in B:
        merge(result, array) # merge result with array
    return result
```

8. توابع  $h(n) = \lg^2 n$  و  $g(n) = (\lg n)^{\lg n}$ ,  $f(n) = 4^{\lg n}$  را در نظر بگیرید. کدام یک از گزاره های زیر صحیح است؟

$$\begin{aligned} g(n) &\in \Omega(h(n)), h(n) \in \Omega(f(n)) \quad (1) \\ f(n) &\in (g(n)), f(n) \in \Omega(f(n)) \quad (2) \\ f(n) &\in h(n), g(n) \in \Omega(f(n)) \quad (3) \\ h(n) &\in O(g(n)), f(n) \in \theta(g(n)) \quad (4) \end{aligned}$$

9. ثابت کنید:  $n! = \omega(2^n)$