



ساختمان داده‌ها و الگوریتم‌ها

تمرین اول - پیچیدگی و الگوریتم‌های بازگشتی

حامد میرامیرخانی، ماردین نیچی

تاریخ تحویل: ۱۴۰۲/۸/۵

۲۰ نمره

۱.

پیچیدگی زمانی قطعه کدهای زیر را محاسبه کنید.

الف)

```
while (n > 0) {  
    for (int j = 0; j < n; j++)  
        System.out.println("*");  
    n = n / 2;  
}
```

ب)

```
sum=0;  
for(i=1; i<n;i++)  
    for( j = 1; j < i * i; j++ )  
        if( j % i == 0 )  
            for( k = 0; k < j; k++ )  
                sum++;
```

ج)

```
for(i = n; i > 2; i = i * 1/5)  
    for (int j = 1; j < n; j++)  
        for (int k = 1; k < n; k++)  
            j*=2;
```

۲.

۲۰ نمره

روابط زیر را رد یا اثبات کنید.

الف) $f(n) = O(g(n)) \Rightarrow 2^{f(n)} = O(2^{g(n)})$

ب) $f(n) \in O(g(n)) \Rightarrow g(n) \in \Omega(f(n))$

ج) $f(n) \neq O(g(n)) \Rightarrow f(n) = \Omega(g(n))$

۳.

۲۰ نمره

پیچیدگی روابط بازگشتی زیر را با استفاده از روش های ممکن به دست آورید.

الف) $T(n) = T(\sqrt{n}) + n$

ب) $T(n) = \sqrt{n}T(\sqrt{n}) + n\log(\log n)$

ج) $T(n) = T(\frac{n}{4}) + n(5 - \cos(n))$

۴.

۱۰ نمره

توابع زیر را براساس پیچیدگی زمانی مرتب کنید.

الف) $\log(n), n^n, \sum_{i=1}^n n - 2i, \log(n!), 2^n, \sqrt[n]{n}$

ب) $n\log(\log(n)), n^{\log(\log(n))}, n^\pi, (\log(n))!$

ج) $\sum_{i=1}^n \frac{n^i}{i!}, n2^n, n^{n\log\log n}, \log n!, n^2, 2^n$

۵.

۲۰ نمره

پیچیدگی زمانی توابع بازگشتی زیر را محاسبه کنید.

الف)

```
func(n) {
    if n <= 1
        return
    for i in 1 to n*n:
        //O(1)
    func(n-2);
}
```

ب)

```
func(a, b):
    if(b == 0):
        return a
    return func(b, a % b)
```

۶.

۱۰ نمره

فرض کنید یک دنباله ی n تایی شامل a_1, a_2, \dots, a_n از اعداد در اختیار دارید. الگوریتمی از مرتبه $O(n)$ طراحی کنید که بتواند حاصل عبارت مقابل را محاسبه کند.

$$\sum_{1 \leq l \leq r \leq n} f(l, r)$$

که $f(l, r)$ را اینگونه تعریف می کنیم:

$$f(l, r) = \sum_{i=l}^r a_i$$