



به نام خدا



ساختمان داده - تمرین شماره ۱

تاریخ تحویل: ۲۹ مهر ۹۷، ساعت ۱۲ ظهر

۱- ثابت کنید $T(n) = O(n \log^2 n)$

$$T(n) = 2T(n/2) + 2n \log_2 n, T(2) = 4$$

۲- این ادعا را ثابت یا با یک مثال نقض رد کنید.

$$g(n) = O(r(n)), f(n) = O(s(n)) \rightarrow f(n) - g(n) = O(s(n) - r(n))$$

۳- مرتبه زمانی کدهای زیر را مشخص کنید.

- A)

```
for (int i = 1; i <= n; i++)  
    for (int j = n; j > i; j--);
```
- B)

```
for (int i = 1; i <= n; i++)  
    for (int j = 1; j < n; j++)  
        i *= 2;
```
- C)

```
// A is an array of boolean with size n  
stack <bool> s;  
for (int i = 0; i < n; i++)  
    if (A[i] == false)  
        while (!s.empty())  
            s.pop();  
    else  
        s.push(A[i]);
```
- D)

```
for (int i = 1; i <= n; i++)  
    for (int j = 1; j*j < n; j++)  
        for (int z = 1; z < 100000; z++);
```

۴- پیچیدگی توابع زیر را به روش دلخواه بدست آورید.

A) $T(n) = 2T(n/2) + n$

B) $T(n) = T(n/3) + T(2n/3) + n$

C) $T(n) = \sqrt{n}T(\sqrt{n}) + n$

۵- با کمک قضیه زیر ثابت کنید $(\log n)^a = O(n^b)$

قضیه: برای هر دو ثابت $a > 1$, $c > 0$ و برای هر تابع صعودی $f(n) : f(n)^c = O(a^{f(n)})$. به عبارت دیگر یک تابع نمایی سریع تر از یک تابع چندجمله ای رشد میکند.

شاد باشید.