



به نام خدا

تمرین اول ساختمان داده‌ها و الگوریتم

الگوریتم‌ها و حل مسئله - پیچیدگی زمانی - الگوریتم‌های بازگشتی

استاد:

دکتر روستایی

استادیار:

عرشیا عموزاد

۱. توابع زیر را از نظر رشد با هم مقایسه کنید

$$g(n) = n^{\log n}$$
$$f(n) = (\log n)^n(n)$$

۲. کدام یک از گزاره های زیر درباره ی توابع ذکر شده صحیح است؟

$$f(n) = 4^{\log n}$$
$$g(n) = (\log n)^{\log n}$$
$$h(n) = \log^2 n$$

$$f(n) = O(g(n)), g(n) = \Omega(h(n)) \quad \bigcirc$$
$$g(n) = \Omega(h(n)), h(n) = \Omega(f(n)) \quad \bigcirc$$
$$f(n) = O(h(n)), g(n) = \Omega(f(n)) \quad \bigcirc$$
$$h(n) = O(g(n)), f(n) = \Theta(g(n)) \quad \bigcirc$$

۳. کد زیر الگوریتم جستجوی خطی را نشان می دهد
فرض کنید با احتمال ۵۰ درصد عنصر target در آرایه ی arr است. بطور متوسط چند عنصر برای یافتن target بررسی می شوند؟

```
1 def linear_search(arr, target):
2     for index in range(len(arr)):
3         if arr[index] == target:
4             return index
5     return -1
```

۴. آیا می توان ادعا کرد که برای هر دو تابع $f(n)$ و $g(n)$ یا $f(n) = O(g(n))$ یا $g(n) = O(f(n))$ یا هر دو؟

۵. توابع زیر را از نظر نرخ رشد به ترتیب صعودی مرتب کنید.

$$f_5(n) = n^{\log \log n}, f_4(n) = n^{\log n}, f_3(n) = (\log n)^{\log n}, f_2(n) = 2^{\sqrt{n}}, f_1(n) = n^{1.01}$$

۶. در یک زمستان سرد، خرس قطبی n قطعه گوشت دقیقاً به اندازه های ۱، ۲ تا n را در غاری ذخیره کرده است. او هر روز یکی از این قطعه گوشت ها را به صورت تصادفی انتخاب می کند. اگر اندازه ی گوشت عدد فردی بود، آن را کاملاً می خورد. اگر زوج بود، آن را دقیقاً نصف می کند، یک نصف آن را می خورد و نصف دیگر را مجدداً در غار قرار می دهد. اگر گوشتی موجود نباشد خرس می میرد. با این الگوریتم، برای n های خیلی بزرگ روزهای باقیمانده از عمر خرس ما تابع کدام یک از گزینه های خواهد بود؟

$$O(n^2) \quad \bigcirc \quad O(n \log n) \quad \bigcirc \quad O(\log n) \quad \bigcirc \quad O(n) \quad \bigcirc$$

۷. پیچیدگی زمانی توابع زیر را پیدا کنید.

$$1. \quad T(n) = 4T\left(\frac{n}{2}\right) + \log^2 n!$$

$$2. \quad T(n) = 3\sqrt[3]{n^2}T(\sqrt[3]{n}) + n \cdot \log n$$

$$3. \quad T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{n}{6}\right) + n^{\sqrt{\log n}}$$

$$4. \quad H(0) = 0, H(n) = \frac{1}{2-H(n-1)}$$

$$5. \quad f(n) = \log(n!)$$

۸. الگوریتم زیر $\binom{n}{m}$ را برای اعداد صحیح $n \geq m$ محاسبه میکند:

```
1 function combination(n, m:integer): integer;
2 begin
3     if (m = n) or (m = 0) then
4         return 1
5     else
6         return combination(n-1, m) + combination(n-1, m-1)
7     end;
```

برای $n > m$ دلخواه عمل جمع چند بار تکرار می شود؟

$$n(n-m) \bigcirc \quad \binom{n}{m} - 1 \bigcirc \quad nm \bigcirc \quad \binom{n}{m} \bigcirc$$

۹. درباره‌ی اجرای الگوریتم مرتب سازی ادغامی (Merge Sort) بر روی آرایه‌ای به طول n به سوالات زیر پاسخ دهید.

۱. اگر در مرحله‌ی تقسیم، الگوریتم به جای نصف شدن دقیق به نسبت های دلخواه مانند $\frac{1}{3}$ و $\frac{2}{3}$ یا $\frac{1}{4}$ و $\frac{3}{4}$ تقسیم شود پیچیدگی زمانی الگوریتم چگونه تغییر می کند؟

۲. فرض کنید همه‌ی عناصر آرایه از قبل بصورت صعودی مرتب شده باشند. در این صورت تعداد مقایسه های الگوریتم نسبت به حالت تصادفی چه تغییری می کند؟

۱۰. با اجرای برنامه‌ی زیر چه عددی چاپ می شود؟

```
1 #include <stdio.h>
2
3 int func(int, int);
4
5 void main() {
6     int n = 2;
7     int m = 3;
8     printf("%d\n", func(n, m));
9 }
10
11 int func(int m, int n) {
12     if (m == 0) return n + 1;
13     if (n == 0) return func(m - 1, 1);
14     return func(m - 1, func(m, n - 1));
15 }
```