دانشکده‌ی علوم ریاضی

آنالیز الگوریتم‌ها　　　　مدرس: دکتر شهرام خزایی

تمرین سری یک

گردآورنده : پارسا زارع‌زاده　　　　مهلت ارسال : ۱۲ اسفند

پرسش ۱

آ) (۱۰ نمره) توابع زیر را بر حسب نرخ رشد مجانبی از کوچک به بزرگ مرتب کنید.(به طور دقیق اثبات کنید)

$f(n) = n + 1000$

$g(n) = \sqrt{n}$

$h(n) = \log^4 n$

$p(n) = (n!)!$

$q(n) = \sum_{i=1}^{n} i^3$

$s(n) = (n^n)!$

$r(n) = 28r(n/3) + n^3 \qquad r(0) = 5$

$t(n) = 6t(n/7) + n \qquad t(0) = 1$

ب) (۱۰ نمره) رابطه‌های بازگشتی زیر را تحلیل مجانبی کنید.

$T(n) = T(\alpha n) + T(\beta n) + O(n) \quad \alpha, \beta > 0, \alpha + \beta = 1$

$T(n) = T(\alpha n) + T(\beta n) + O(n) \quad \alpha, \beta > 0, \alpha + \beta < 1$

$T(n) = aT(\frac{n}{b}) + O(n^d \log n)$

پرسش ۲

زمان اجرایی الگوریتم های زیر را تحلیل کنید: (نیازی به اثبات درستی یا توصیف کد نیست)

آ) (۵ نمره) W یک ماتریس مجاورت برای گراف است.

```
FLOYD-WARSHALL(W)
n = W.rows
D^(0) = W
for k = 1 to n
    let D^(k) = (d_ij^(k)) be a new n × n matrix
    for i = 1 to n
        for j = 1 to n
            d_ij^(k) = min(d_ij^(k-1), d_ik^(k-1) + d_kj^(k-1))
return D^(n)
```

ب) (۷ نمره)

```
TABLE-INSERT(T, x)
if T.size == 0
    allocate T.table with 1 slot
    T.size = 1
if T.num == T.size
    allocate new-table with 2 · T.size slots
    insert all items in T.table into new-table
    free T.table
    T.table = new-table
    T.size = 2 · T.size
insert x into T.table
T.num = T.num + 1
```

پ) (۸ نمره)

```
COMPUTE-PREFIX-FUNCTION(P)
m = P.length
let π[1 .. m] be a new array
π[1] = 0
k = 0
for q = 2 to m
    while k > 0 and P[k + 1] ≠ P[q]
        k = π[k]
    if P[k + 1] == P[q]
        k = k + 1
    π[q] = k
return π
```

پرسش ۳

به یک دنباله $\langle x_1, x_2, ..., x_n \rangle$ را گوگولی می گوییم اگر و تنها اگر مقدار i وجود داشته باشد به طوری که به ازای هر $1 \leq j \leq i-1$ داشته باشیم $x_j \leq x_{j+1}$ و به ازای هر $i \leq j \leq n-1$ داشته باشیم $x_j \geq x_{j+1}$ یا به عبارتی دیگر دنباله تا یک جایی صعودی و از آنجا به بعد نزولی باشد.
حال با توجه به تعریف به سوالات زیر پاسخ دهید:

آ) (۸ نمره) برای الگوریتم زیر که برای مرتب سازی دنباله های گوگولی است سودوکد مناسب نوشته، سپس درستی الگوریتم را اثبات کنید و در انتها زمان اجرا الگوریتم را تحلیل کنید.

الگوریتم مرتب سازی دنباله های گوگولی: الگوریتم بر اساس روش تقسیم و حل است. به این صورت است که در ابتدا دنباله گوگولی را در نظر می‌گیریم. اگر تنها یک عضو داشت همان را خروجی می‌دهیم و گرنه آن را از وسط به دو بخش تقسیم می‌کنیم. سپس عضو های متناظر دو بخش را با هم مقایسه می‌کنیم.(یعنی عضو اول بخش اول با

عضو اول بخش دوم و عضو دوم بخش اول با عضو دوم بخش دوم و...) پس از مقایسه اگر عضو بخش اول بزرگتر از عضو بخش دوم بود آن دو عضو را با یکدیگر جابه‌جا می‌کنیم. سپس همین کار را برای بخش اول و بخش دوم به صورت بازگشتی اعمال می کنیم. در انتها ادعا می‌کنیم که دنبال مرتب شده است.

ب) (۷ نمره) حال با استفاده از الگوریتم بخش (آ) الگوریتمی طراحی کنید یک دنباله دلخواه را مرتب کند. سودوکد و اثبات درستی و تحلیل زمان اجرا الگوریتم را بنویسید.
(راهنمایی: مانند الگوریتم مرتب سازی ادغامی عمل کنید ولی بخش ادغام کردن را باید تغییر دهید)

پ) (۵ نمره) حال توضیح دهید الگوریتم بخش (ب) چه برتری نسبت به الگوریتم مرتب سازی ادغامی دارد.

پرسش ۴

(۲۰ نمره) بازی پیچ و مهره به این صورت است که n مهره و n پیچ با سایزهای مختلف داریم و هر پیچ دقیقا یک مهره متناظر دارد. هدف بازی این است که همه مهره ها را در پیچ متناظر آن قرار دهیم. در هر مرحله می‌توانید یک مهره و یک پیچ بردارید و تشخیص دهید که مهره بزرگتر پیچ است یا کوچکتر از پیچ است و یا دقیقا همان اندازه است. دقت کنید که نمی‌توانید دو پیچ یا دو مهره را با هم مقایسه کنید و تشخیص دهید کدام یک بزرگتر است. الگوریتمی طراحی کنید که به طور میانگین در $\Theta(n \log n)$ مرحله تمام مهره ها را داخل پیچ‌ها قرار دهد.

پرسش ۵

(۲۰ نمره) به n نقطه در صفحه وضعیت پایدار می‌گوییم اگر به ازای هر جفت نقطه یکی از شروط زیر برقرار باشد: (نقطه اول به صورت (x_1, y_1) و نقطه دوم به صورت (x_2, y_2) است)

- $x_1 = x_2$
- $y_1 = y_2$
- نقطه دیگری مثل (x_3, y_3) وجود دارد که:
  $\min(x_1, x_2) \leq x_3 \leq \max(x_1, x_2)$ و $\min(y_1, y_2) \leq y_3 \leq \max(y_1, y_2)$

حال به شما n نقطه داده شده است. الگوریتمی طراحی کنید که در زمان اجرایی $O(n \log n)$ و با اضافه کردن $O(n \log n)$ نقطه به مجموعه اولیه یک مجموعه پایدار تشکیل دهد.(دقت کنید که همگی نقاط باید متمایز باشند)