



دانشکده علوم ریاضی

مدرس: دکتر شهرام خزایی

ساختمان داده‌ها

تمرین سری یک

مهلت ارسال: ۲۳ آبان

گردآورنده: آرش کورش نژاد، مهدی طاهری، محمدحسین مهدی‌خانی

• از بین ۷ پرسش زیر، به ۵ پرسش به دلخواه پاسخ دهید.

پرسش ۱

دنباله $a_1 \dots a_n$ از اعداد طبیعی متمایز داده شده است. هدف ما پیدا کردن بزرگترین k است به طوری که $1 \leq i \leq n-k$ وجود داشته باشد که $a_i < a_{i+k}$. برای مثال دنباله $a_1 \dots a_7$ که به این شکل تعریف شده است را در نظر بگیرید:

$$a_1 = 14, \quad a_2 = 6, \quad a_3 = 8, \quad a_4 = 1, \quad a_5 = 12, \quad a_6 = 7, \quad a_7 = 5$$

در این دنباله جواب ۴ است زیرا $a_2 = 6 < 7 = a_6$.

(الف)

الگوریتمی از مرتبه زمانی $O(n)$ ارائه کنید که برای هر t ، کمینه اعداد در بین $a_1, a_2, \dots, a_t$ را محاسبه کند و آن را در m_t ذخیره کند. (در حقیقت $m_t = \min_{i=1}^t a_i$).

(ب)

الگوریتمی از مرتبه زمانی $O(n \log n)$ ارائه دهید تا ماکزیمم k خواسته شده را پیدا کند.

پرسش ۲

روابط بازگشتی زیر را ابتدا با روش درخت محاسبه حدس بزنید و سپس با استقرا ثابت کنید.

(الف)

$$T(n) = 3T\left(\frac{n}{4}\right) + O(n \log n)$$

(ب)

$$T(n) = T\left(\frac{n}{3}\right) + \Omega(n^2)$$

(ج)

$$T(n) = \sqrt{n}T(\sqrt{n}) + 1 \circ n$$

پرسش ۳

تابع f به شما داده شده است. فرض کنید عدد c یک عدد حقیقی دلخواه کوچک است. تابع f_c^* به این گونه تعریف می‌شود.

$$f_c^*(n) = \min\{i \geq 0 : f^{(i)}(n) \leq c\}$$

به زبان ساده، این تابع کمترین تعداد دفعاتی را نشان می‌دهد که باید تابع f را روی n اعمال کنیم تا خروجی از c کمتر شود. اثبات کنید:

$$\lg^*(n) \in \mathcal{O}(\lg n)$$

پرسش ۴

توابع زیر را از لحاظ نرخ رشد مرتب کنید. اگر چند تابع نرخ رشد یکسانی دارند، دنباله‌ی مرتب شده‌ی خود را به طوری بنویسید که این تساوی نرخ مشخص باشد.

$\lg(\lg^* n)$	$2^{\lg^* n}$	$(\sqrt{2})^{\lg n}$	n^2	$n!$	$(\lg n)!$
$(3/2)^n$	n^3	$\lg^2 n$	$\lg(n!)$	2^{2n}	$n^{1/\lg n}$
$\ln \ln n$	$\lg^* n$	$n \cdot 2^n$	$n^{\lg \lg n}$	$\ln n$	1
$2^{\lg n}$	$(\lg n)^{\lg n}$	e^n	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$
$\lg^*(\lg n)$	$2^{\sqrt{2} \lg n}$	n	2^n	$n \lg n$	2^{2n+1}

پرسش ۵

فرض کنید $a_1, \dots, a_n$ دنباله‌ای از n عدد متمایز باشد. اگر $i < j$ و $a_i > a_j$ آنگاه جفت (i, j) را یک نابجایی می‌نامیم.

(الف)

کدام جایگشت از مجموعه‌ی $\{1, 2, \dots, n\}$ بیشترین تعداد نابجایی را دارد؟ تعداد نابجایی‌های آن چند تا است؟

(ب)

نشان دهید هر الگوریتم مرتب‌سازی که تنها مجاز به جابجایی عناصر مجاور است، در بدترین حالت باید $\Omega(n^2)$ جابجایی انجام دهد (در نتیجه زمان اجرای آن در بدترین حالت از مرتبه‌ی $\Omega(n^2)$ است).

پرسش ۶

دنباله‌ی $a_1, \dots, a_n$ از اعداد طبیعی داده شده است. الگوریتمی از مرتبه زمانی $\Theta(n \log n)$ ارائه کنید که تعداد نابجایی‌های (توضیح داده شده در سوال ۵) دنباله‌ی داده شده را محاسبه کند.

پرسش ۷

(الف)

با ذکر یک ناوردایی مناسب، ثابت کنید الگوریتم ادغام زیر درست کار می‌کند.

(ب)

اگر از این الگوریتم ادغام در مرتب‌سازی ادغامی استفاده شود، زمان اجرای الگوریتم را بدست آورید.

(ج)

یک الگوریتم درجا با زمان اجرای خطی برای الگوریتم ادغام بیابید. (جست‌وجو در منابع آنلاین و آفلاین بلامانع است).

Procedure: InPlaceMerge(A, p, q, r)

$A[p..q]$ and $A[q + 1..r]$ are sorted

$i \leftarrow p$

$j \leftarrow q + 1$

while $i < j$ and $j \leq r$ **do**

if $A[i] \leq A[j]$ **then**

$i \leftarrow i + 1$ # element in left part is already in place

else

 # $A[j]$ should come before $A[i]$

$temp \leftarrow A[j]$

$k \leftarrow j$

while $k > i$ **do**

$A[k] \leftarrow A[k - 1]$ # shift right

$k \leftarrow k - 1$

$A[i] \leftarrow temp$

 # update indices after insertion

$i \leftarrow i + 1$

$j \leftarrow j + 1$

$q \leftarrow q + 1$ # boundary moves right
