

هر سوال را در محل در نظر گرفته شده پاسخ دهید. پاسخ های خارج از محل تصحیح نمیشوند. نام و شماره دانشجویی را روی تمام برگه ها بنویسید. شماره دانشجویی باید با اعداد لاتین نوشته شود. مهلت این تمرین شنبه ۶ آبان ماه است.

زهرا حسینی - سهراب نمازی - سید صالح اعتمادی

۱. [۱۰] تابع های زیر را بر اساس نرخ رشد به صورت صعودی مرتب کنید.

$N, \sqrt{N}, N^2, N \log N, N \log N^2, 2/N, 2^N, 37, N^2 \log N, N^3$

در صورتی که برابر هستند نیز مشخص کنید.

۲. [۲۵] درستی یا نادرستی هریک از موارد زیر را اثبات کنید.

(آ)

$$f(n) + g(n) = \Theta(\max(f(n), g(n)))$$

(ب)

$$f(n) + O(f(n)) = \Theta(f(n))$$

(ج)

$$f(n) = O(f(n)^2)$$

(د)

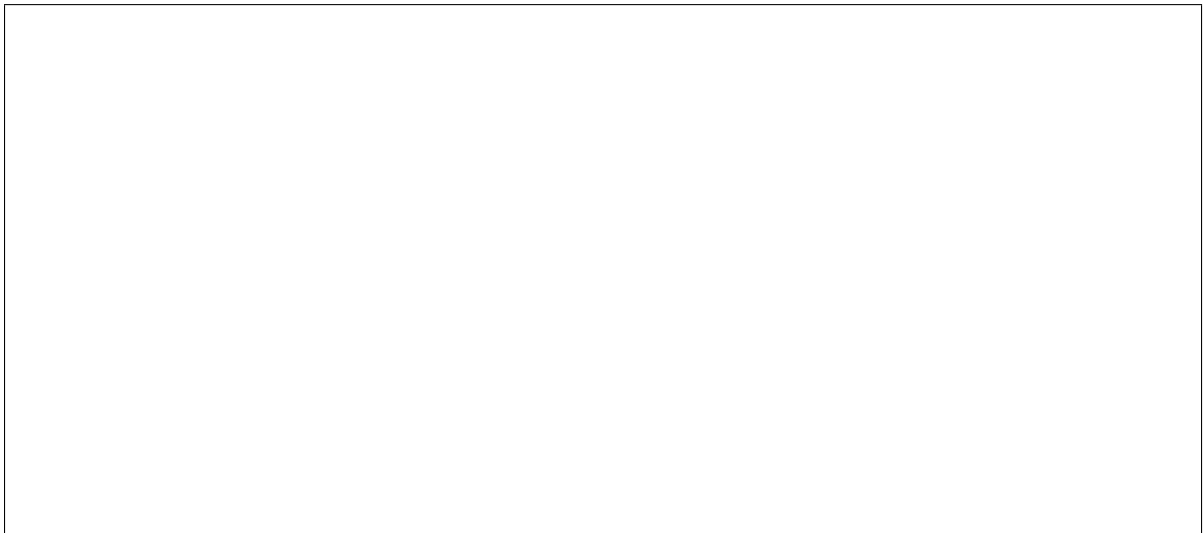
$$(n+1)^2 = O(n^2)$$

۳. [۱۵] در مسئله ی single machine scheduling به شما یک ماشین و مجموعه وظایف  $T = \{T_1, T_2, \dots, T_n\}$  با زمان شروع مربوطه،  $S = \{S_1, S_2, \dots, S_n\}$  و زمان پایان مربوطه  $F = \{f_1, f_2, \dots, f_n\}$  داده میشود. به عبارت دیگر وظیفه ی  $T_i$  در زمان  $s_i$  شروع میشود و در زمان  $f_i$  به پایان میرسد. این ماشین میتواند دقیقاً یک کار را در آن واحد انجام دهد. بر این اساس، وظایف برنامه ریزی شده روی این دستگاه نباید در تناقض باشند. هدف این است که تعداد وظایف برنامه ریزی شده برای این ماشین را به حداکثر برسانید. برای این مسئله الگوریتمی حریصانه (Greedy) طراحی کنید و درستی آن را بررسی کنید.



۴. [۱۰] الگوریتمی برای ورودی با سایز ۱۰۰، 0.5ms زمان صرف میکند. این الگوریتم برای ورودی با اندازه ی ۵۰۰ چقدر طول میکشد؟ در دو حالت زیر بررسی کنید.

Linear (آ)



(ب)  $O(n \log(n))$ 

۵. [۱۵] آرایه ای را فرض کنید که هر عضو آن علامت مثبت و یا علامت منفی بوده و طول مشخصی دارد. میخواهیم بیشترین تعداد منفی ممکن را با مثبت های موجود خنثی کنیم. اما نکته مسئله در این است که مثبت ها میتوانند منفی موجود در ماکسیمم فاصله معینی از خود را نهایتاً خنثی کنند که این فاصله مجاز را با  $k$  نمایش می دهیم. توجه داشته باشید که هر علامت مثبت میتواند یک منفی را خنثی کند. با توجه به هدف ما که خنثی کردن بیشترین تعداد منفی ممکن است، الگوریتم حریصانه ای برای حل این مسئله یافته و آن را توضیح دهید. سپس برای هر یک موارد بعد، پاسخ را با اجرای راه حلی که ارائه دادید به دست آورید.

(آ) الگوریتم شما:

(ب) Array = [+ , + , - , - , + , -], k = 2

(ج) Array = [+ , - , + , - , - , +], k = 3

۶. [۱۰] استراتژی کلی حل یک مسئله را به روش حریمانه، مرحله به مرحله برای راه حل خود در سوال قبل توضیح دهید.

۷. [۱۵] پیچیدگی زمانی الگوریتم های زیر را بدست آورید.

(Merge Sort ->  $O(n \log n)$ )

```
(آ) int[][] Numbers = new long[n][ ];  
for (int i = 0; i < n; i++):  
    int[] sorted = Mergesort(Numbers[ i ]);  
    foreach (int num in sorted) Console.WriteLine(num);
```

```
(ب) for (i = 1; i < n; i++):  
    for (j = 1; j < n; j += i):  
        do some  $O(1)$  task!
```