



دانشکده مهندسی کامپیوتر

درس تحلیل و طراحی الگوریتم ها

---

تمرین سری اول

---

مدرس ..... دکتر مرضیه ملکی مجد

طراحان ..... پارسا شفیعی یکتا - پرهام هدایتی

تاریخ انتشار ..... ۱۴۰۲/۱۲/۰۸

تاریخ تحویل ..... ۱۴۰۲/۱۲/۲۶



## فهرست مطالب

نکات تمرین.....	۳
تمرین اول.....	۴
سوال ۱ (تئوری).....	۵
طول بزرگترین زیر دنباله مشترک (LCS).....	۶
Rod Cutting.....	۸
قاشق زنی.....	۹
سیاست کثیف.....	۱۱
عذاب ارواح.....	۱۴
حساب کتاب.....	۱۷
زیر رشته خوب.....	۱۹
دنباله گم شده.....	۲۱

## نکات تمرین

➤ این تمرین شامل مباحث:

• Dynamic Programming

می باشد.

➤ برای پاسخگویی به سوالات این تمرین نیاز است که به مطالب اسلایدهای:

• DP1

• DP2

مسلط باشید.

➤ پاسخ تمامی سوالات عملی تنها به زبانهای C# و C++ قابل قبول می باشد

➤ علیرغم اعتماد کامل تیم تی ای به شما دانشجویان عزیز ، تمامی کدهای شما با سایر

دانشجویان بصورت خودکار مقایسه خواهند شد. شد ، لذا از کپی نمودن کد دوستانتان یا

استفاده مستقیم از منابع مختلف خودداری کنید و تمامی پاسخ ها ، کد خودتان باشد و در

صورت هم فکری با دوستانتان یا ایده گیری از منابع مختلف نام دانشجو و منبع استفاده

شده در ابتدای آن بخش به صورت کامنت شده بنویسید

➤ در تمامی تمرینات سعی شده است که سوالات ساده تر در ابتدا و سوالات دشوارتر در

انتهای فایل قرار گیرند

➤ در صورت وجود هرگونه سوال در مورد تمرینات ، سعی کنید تا جایی که امکان دارد

سوال خود را در گروه پرسید چرا که شاید سوال شما ، سوال دوستان نیز باشد و

دوستانتان نیز بتوانند از پاسخ سوال شما بهره ببرند .

- سوالات را در سامانه کوئرا و در قسمت تمرین سری اول آپلود نمایید.
- با توجه به مبحث تمرین استفاده از هر گونه روشی به جز روش خواسته شده مجاز نیست .
- از آنجایی که هر سوال توسط یک تی ای طرح شده است ، تنها تی ای طراح آن سوال می تواند شما را بصورت دقیق راهنمایی کند به همین منظور طراح هر سوال در زیر نوشته شده است تا در صورت ابهام و پرسش در مورد هر سوال ، در صورتی که نیاز به پرسش سوال بصورت انفرادی در پیوی هست ، به تی ای مربوطه مراجعه بفرمایید
- سوال ۱ (تئوری ۱) . پارسا شفیعی یکتا
- سوال ۱ (تئوری ۲) . پارسا شفیعی یکتا
- سوال ۱ (تئوری ۳) . پرهام هدایتی
- سوال ۲ . پارسا شفیعی یکتا
- سوال ۳ . پرهام هدایتی
- سوال ۴ . پارسا شفیعی یکتا
- سوال ۵ . پرهام هدایتی
- سوال ۶ . پرهام هدایتی
- سوال ۷ . پارسا شفیعی یکتا
- سوال ۸ . پرهام هدایتی
- سوال ۹ . پارسا شفیعی یکتا

## ۱- سوالات تئوری

۱) ویژگی هایی که یک مسئله dynamic programming دارد را نام ببرید و به طور خلاصه توضیح دهید.

۲) چطور در برخی مسائلی که recursive حل میشوند dynamic programming میتواند الگوریتم را بهبود ببخشد.

۳) نسخه ای از مسئله ضرب ماتریس ها در نظر بگیرید که هدف این است که طوری دنباله ماتریس ها را پرانتز گذاری کنیم که تعداد ضرب های مورد نیاز به جای کمینه شدن بیشینه شود آیا این مسئله دارای زیرساختار بهینه هست؟ (توضیح دهید)

## ۲- طول بزرگترین زیر دنباله مشترک (LCS)

در این مسئله به دنبال طول بزرگترین زیر دنباله مشترک هستیم. به عنوان ورودی به شما دو رشته ناتهی

$$A = a_1 a_2 a_3 \dots a_n$$

$$B = b_1 b_2 b_3 \dots b_m$$

تشکیل شده از رقم های ۰ تا ۹ داده میشود.

در این سوال شما باید به عنوان خروجی طول بزرگترین دنباله به اندازه  $p$  که در را که در شرایط زیر صدق میکند چاپ کنید.

$$1 \leq i_1 \leq i_2 \leq \dots \leq i_p \leq n, 1 \leq j_1 \leq j_2 \leq \dots \leq j_p \leq m$$

به طوری که

$$a_{i_1} = b_{j_1}, a_{i_2} = b_{j_2}, \dots, a_{i_p} = b_{j_p}$$

**ورودی:**

در خط اول به ترتیب طول رشته اول ( $n$ ) و دوم ( $m$ )، داده میشود. در خطوط دوم و سوم نیز به ترتیب رشته اول و دوم آمده است.

$$1 \leq n, m \leq 1000$$

**خروجی:**

طول بزرگترین زیر دنباله مشترک

**Example 1:**

**Input:**

7 6

1425682

132829

**Output:**

4

**توضیح:**

بزرگترین زیررشته مشترک دو دنباله 1282 است که طول آن برابر ۴ است.

### Rod Cutting – ۳

در این سوال به شما یک میله به طول  $n$  داده و لیستی از قیمت میله به ازای طول های مختلف داده میشود. شما میتوانید میله و هر تکه از آن را به تعداد دلخواه و به اندازه صحیح برش بزنید. از شما میخواهیم که بیشترین پولی که از فروش میله میتوان به دست آورد را به عنوان خروجی بدهید

#### ورودی:

در خط اول طول میله ( $n$ ) داده میشود

در خط بعد  $n$  تا قیمت به شما داده میشود که قیمت میله به اون طول است ( $i$ مین عدد قیمت میله به طول  $i$  است)

#### خروجی:

بیشترین پولی که از فروش میله موردنظر بدست می آید

#### Example 1:

##### Input:

8

1 5 8 9 10 17 17 20

##### Output:

22

#### توضیح:

بیشترین پولی که میتوان بدست آورد با تقسیم میله به دو قسمت به طول ۶ و ۲ میباشد که در مجموع به قیمت  $17+5$  فروخته میشود



## ۴- قاشق زنی

در مراسم چهارشنبه سوری امسال آیدا برای جمع کردن شیرینی به محله ای رفته که  $n$  خانه در آن وجود دارد. این  $n$  خانه به صورت دایره وار کنار یکدیگر قرار دارند. آیدا در صورتی میتواند زنگ در خانه ای را بزند و از آنها شیرینی دریافت کند که زنگ در خانه مجاور آن را نزده باشد در غیر این صورت صاحب خانه به او شیرینی ای نمیدهد. از آنجایی که آیدا بسیار عاشق شیرینی است به او در پیدا کردن بیشتر مقدار شیرینی ای که میتواند جمع آوری کند کمک کنید.

### ورودی:

در خط اول عدد  $n$  که بیانگر تعداد خانه هاست داده میشود.

در خط دوم نیز یک آرایه از اعداد داده میشود که بیانگر مقدار شیرینی است که هر خانه میدهد.

### خروجی:

در تنها خط خروجی باید حداکثر مقدار شیرینی هایی که امکان جمع آوری آنها است را چاپ کنید

#### Example 1:

Input:

3

2 3 2

Output:

3

## توضیح:

خانه های اول و سوم در حقیقت مجاور هستند (خانه ها بر روی دایره قرار دارند). پس بهینه ترین حالت زدن زنگ خانه دوم و گرفتن ۳ شیرینی است.

## Example 2:

Input:

4

1 2 3 1

Output:

4

## توضیح:

با انتخاب خانه های اول و سوم داریم  $4 = 3 + 1$

## ۵- سیاست کثیف

مرکز سری برای تحلیل های سری خودش تصمیم گرفته تمام سیاسیون رو حزب بندی کنه پس یک صف از سیاسیون تشکیل داده که هرکس یه قدرت نفوذی داره و به مرکز سری اعلام میکنه که میخواد تشکیل حزب بده و به تعداد قدرت و نفوذی که داره افراد پشت او در صف به حزب اون میپیوندند (به اجبار!!) و دیگه نمیتونن تشکیل حزب بدن و برای خودشون کسی بشن.

متأسفانه سازمان سری بی رحمه و به هیچکس اجازه نمیده با قدرتی کمتر از قدرت سازمان عضو حزبش بشن (اگه زیاد قوی باشی میکشنت که بعدا دردسر نشی!!) و هرکس که بدون حزب بمونه رو هم میکشه ولی ما باید با این سیاست کثیف مبارزه کنیم و با کمترین تعداد کشته شده همرو حزب بندی کنیم درواقع ما قبل از اینکه حزب بندی صورت بگیره به تعداد لازم سیاسیون رو میکشیم و چون ما ادم درستکاری هستیم قدرت سیاسی اونا برا ما مهم نیستن و همه از دید ما برابر هستنند. (برای درک بهتر سوالات نمونه ها را با دقت مطالعه کنید)

### ورودی:

در خط اول ورودی تعداد افراد تو صف به شما داده میشود

در خط دوم قدرت هر فرد به شما داده میشه

### خروجی:

در تنها خط خروجی شما حداقل تعداد کشته شدگان رو خروجی میدهید

## Example 1:

Input:

7

3 3 4 5 2 6 1

Output:

0

## توضیح:

قدرت نفر اول ۳ هست پس ۳ نفر بعدی از صف مجبورن عضو حزب نفر اول بشوند و از صف خارج میشوند  
اولین نفر باقی مانده قدرتش ۲ هست پس ۲ نفر بعدی هم عضو حزب نفر پنجم میشوند و صف خالی میشود  
پس خدایه شکر نیاز به دخالت ما نبود

## Example 2:

Input:

4

5 6 3 2

Output:

4

## توضیح:

قدرت نفر اول ۵ و نفر ۳ کلا ۳ نفر تو صف هستند پس نفر اول به اجبار کشته میشود قدرت نفر بعدی هم ۶ هست ولی دو نفر کلا تو صف هستند که نفر دوم هم کشته میشود همینطور ۳ و ۲ هم کشته میشوند و درواقع یک قتل عام رخ میدهد

## Example 3:

Input:

6

3 4 1 6 7 7

Output:

1

## توضیح:

اگر فقط نفر اول رو بکشیم نفر دوم با قدرت ۴ میتواند بقیه رو وارد حزب خودش بکنه

## ۶- عذاب ارواح

پارسا شکارچی روح ماهری هست که چند تا روح رو گیر انداخته و دارن عذاب میکشن و قصد نداره هیچکدوم رو آزاد کنه اما اون نمیدونه که تا زمانی که روح ها زندانی هستند طلسم هستش و عذاب میکشه و هر موقع که روحی ازاد میشه دنیای ارواح باخبر میشن و به پارسا حمله میکنند

یه نکته از دنیای ارواح اینه که روح ها سن های مختلفی دارن ( $0, 1, 2, \dots$ ) و همیشه کم سن ترین روح حمله میکند مگر اینکه روحی با اون سن زندانی باشد (یعنی باید تمام روح های اون سن ازاد باشن تا بتونند حمله کنند) حالا ما میخوایم همه روح ها رو بدون اینکه پارسا بفهمه ازاد کنیم و این کارو به ترتیبی انجام میدیم که پارسا کمترین آسیب رو ببینه (هر روح به اندازه سنش آسیب میزند)

لیست روح های زندانی رو میدیم و هر بار که یه روح رو ازاد میکنید به اندازه کوچک ترین عدد غیر منفی که در زندان نیست پارسا آسیب میبیند

### ورودی:

در خط اول تعداد روح های زندانی داده میشه

$$(1 \leq n \leq 2500)$$

در خط دوم سن هر روح زندانی داده میشه

### خروجی:

در تنها خط خروجی باید کمترین آسیبی که پارسا میبینه به عنوان خروجی داده بشه

## Example 1:

Input:

8

5 2 1 0 3 0 4 0

Output:

3

توضیح:

$[5, 2, 1, 0, 3, 0, 4, 0] \rightarrow [5, 2, 0, 3, 0, 4, 0] \rightarrow [5, 2, 0, 3, 4, 0] \rightarrow [5, 2, 3, 4, 0] \rightarrow [5, 2, 3, 4] \rightarrow [5, 2, 4] \rightarrow [2, 4] \rightarrow [4] \rightarrow []$ .

میزان آسیب :  $1+1+1+0+0+0+0+0=3$

ما تو مرحله اول وقتی روح ۱ رو آزاد میکنیم تمام روح های یک ساله آزاد شدن پس روح ها با قدرت ۱ به ما حمله میکنند در ۳ مرحله بعدی که • هارو آزاد میکنیم همچنان با ۱ حمله میکنند (چون کوچک ترین عدد غیر منفی که زندانی نیست ۱ عه) اما بعد از آزاد شدن سومین • دیگه همه روح های • ساله (نوزاد) آزاد شدن و میتونند با • حمله کنند که دیگه برا ما مهم نیست چون آسیب • به ما وارد میشود

## Example 2:

Input:

8

0 1 2 0 1 2 0 3

Output:

7

توضیح:

$[0, 1, 2, 0, 1, 2, 0, 3] \rightarrow [0, 2, 0, 1, 2, 0, 3] \rightarrow [0, 2, 0, 2, 0, 3] \rightarrow [2, 0, 2, 0, 3] \rightarrow [2, 2, 0, 3] \rightarrow [2, 2, 3] \rightarrow [2, 3] \rightarrow [3] \rightarrow []$ .

میزان آسیب :  $4+1+1+1+0+0+0+0 = 7$

اینجا ما اگر تو قدم اول ۱ رو آزاد کنیم باز هم روح با سن ۱ زندانی هست همچنین ۲ و ۳ هم زندانی هستند پس روح ها مجبور میشوند که با ۴ به ما حمله کنند اما بعد از آزاد شدن روح ۱ دوم همه ۱ ها آزادن پس میتونن از این به بعد با ۱ حمله کنن و ...



## ۷- حساب کتاب

پارسا یک آرایه از اعداد صحیح مثبت دارد. او میتواند به ابتدای هر یک از اعداد یکی از نماد های "-" یا "+" اضافه کند و یک عبارت ریاضی بسازد. پارسا میخواهد بداند چه تعداد عبارت ریاضی متمایز میتواند بسازد که حاصل آنها برابر مقدار `target` شود.

### ورودی:

در خط اول ورودی اندازه آرایه، `n` داده میشود.

در خط دوم مقدار `target` داده میشود.

در خط سوم ورودی نیز آرایه `A` با طول `n` از اعداد حسابی داده میشود.

$$1 \leq |A| \leq 30$$

$$0 \leq A_i \leq 1000$$

$$0 \leq \text{Sum}(A) \leq 1000$$

$$-1000 \leq \text{target} \leq 1000$$

### خروجی:

در تنها خط خروجی تعداد حالت های ممکن برای ساخت عبارتی که برابر با `target` باشد را چاپ کنید. تضمین میشود مقدار خروجی بیشتر از حداکثر مقدار `int32` نشود.

**Example 1:****Input:**

4

2

1 1 1 1

**Output:**

4

توضیح:

$$-1 +1 +1 +1 = 2$$

$$+1 -1 +1 +1 = 2$$

**Example 2:****Input:**

2

1

0 1

**Output:**

2

توضیح:

$$+0 +1 = 1$$

$$- 0 +1 = 1$$

## ۸- زیررشته خوب

ابولفضل یک رشته فرعی از یک رشته باینری را خوب می نامد اگر این رشته فرعی دقیقاً یک ۱ داشته باشد به ابولفضل کمک کنید تعداد رشته های باینری  $s$  را بشمارد به طوری که  $s$  دقیقاً حاوی  $n$  زیررشته خوب باشد و هیچ زیررشته خوبی با طول اکیداً بزرگتر از  $k$  نداشته باشد. توجه داشته باشید که رشته های فرعی بر اساس مکانشان در رشته متمایز می شوند، بنابراین اگر  $s=1010$  باشد باید هر دو وقوع را ۱۰ بشمارید.

### ورودی:

تنها خط شامل دو عدد صحیح

$$(1 \leq n \leq 2500) \quad n$$

و

$$(1 \leq k \leq n) \quad k$$

است - تعداد زیررشته های خوب مورد نیاز و حداکثر طول مجاز یک زیررشته خوب.

### خروجی:

تعداد رشته های باینری  $s$  به گونه ای که  $s$  دقیقاً حاوی  $n$  زیررشته خوب باشد و هیچ زیررشته خوبی با طول اکیداً بزرگتر از  $k$  ندارد. از آنجایی که این عدد صحیح می تواند خیلی بزرگ باشد، باقی مانده عدد به ۹۹۸۲۴۴۳۵۳ را خروجی بدهید

**Example 1:****Input:****1 1****Output:****1****توضیح:**

تنها رشته ای که بدون زیررشته خوب دارد و زیر رشته خوب بزرگ تر از ۱ نداشته باشد ۱ است.

**Example 2:****Input:****3 2****Output:****3****توضیح:**

رشته هایی که ۳ تا زیررشته خوب دارن و زیررشته خوب با طول بیشتر از ۲ ندارن : 110 , 011 , 111  
توجه کنید که ۱۰۰ هم سه تا زیر رشته خوب دارد ولی چون خود ۱۰۰ زیر رشته خوب با طول ۳ هست پس  
حساب نمیشود (و رشته ها محدودیت طول ندارند اینجا بر حسب اتفاق همه رشته ها طول ۳ داشتند)

## ۹- دنباله گم شده

احمد یک دنباله  $A = a_1, a_2, \dots, a_{n+m}$  داشت. او در اوقات فراغتش به دلایلی نامعلوم این دنباله رو به دو دنباله  $B = b_1, b_2, \dots, b_m$  و  $R = r_1, r_2, \dots, r_n$  تقسیم کرد به طوری که ترتیب نسبی اعداد در هر دو دنباله مانند ترتیب نسبی آنها در دنباله اصلی است یا به عبارتی ساده تر به عنوان مثال اگر  $r_j$  پس از  $r_i$  آمده باشد در دنباله اصلی نیز همینطور است. متأسفانه احمد که بی دقت است دنباله اصلی را گم کرده و میخواهد با استفاده از دنباله های  $R$  و  $B$  آن را باز یابد. در صورتی که چند روش برای ساخت دنباله  $A$  وجود داشته باشد او میخواهد موردی را پیدا کند که مقدار تابع زیر را ماکسیمم کند.

$$f(a) = \max(0, a_1, (a_1 + a_2), (a_1 + a_2 + a_3), \dots, (a_1 + a_2 + a_3 + \dots + a_{n+m}))$$

به احمد در پیدا کردن ماکسیمم مقدار تابع  $f(a)$  کمک کنید.

### ورودی:

در خط اول تعداد تست کیس ها  $t$  ( $1 \leq t \leq 1000$ ) داده میشود.

در خط اول هر تست طول آرایه اول  $n$  ( $1 \leq n \leq 100$ ) ،

در خط دوم هر تست یک آرایه از اعداد به طول  $n$  به طوری که  $-100 \leq r_i \leq 100$  ،

در خط سوم هر تست کیس طول آرایه دوم  $m$  ( $1 \leq m \leq 100$ ) ،

در خط چهارم هر تست یک آرایه از اعداد به طول  $m$  به طوری که  $-100 \leq b_i \leq 100$  داده میشود.

### خروجی:

خروجی شامل  $t$  خط است که در هر خط باید مقدار ماکسیمم تابع  $f(a)$  مربوط به آن تست را چاپ کنید.

### Example 1:

Input:

4

4

6 -5 7 -3

3

2 3 -4

2

1 1

4

10 -3 2 2

5

-1 -2 -3 -4 -5

5

-1 -2 -3 -4 -5

1

0

1

0



Output:

13

13

0

0

توضیح:

در دنباله های زیر اعدادی که به صورت به رنگ سبز نوشته شده آند مربوط به دنباله R میباشند.

برای تست کیس اول دنباله ای که تابع  $f(a)$  را ماکسیمم میکند به صورت  $[6, 2, -5, 3, 7, -3, -4]$  است.

برای تست های ۲ الی ۴ این دنباله ها به ترتیب به صورت زیر هستند.

$[10, 1, -3, 1, 2, 2]$

$[-1, -1, -2, -3, -2, -4, -5, -3, -4, -5]$

$[0, 0]$