

## برنامه‌سازی پیشرفته

تمرین کامپیوتری شماره ۳



مدرس: رامتین خسروی

طراحان: سروش صحرائی، هادی بابالو، کوروش علی‌نقی،  
عرفان عسگری، مبینا مهرآذر، فرشته باقری

مهلت تحویل: چهارشنبه ۱۶ آبان ۱۴۰۳، ساعت ۲۳:۵۹

### مقدمه

این تمرین برای آشنایی با برنامه‌نویسی بازگشتی طراحی و در قالب سه سوال مجزا تهیه شده است که پیشنهاد می‌شود برای درک بهتر مفاهیم برنامه‌نویسی بازگشتی، زمان کافی را برای پاسخ دادن به آن‌ها اختصاص دهید. توجه کنید که پرسش‌ها حتماً باید به روش بازگشتی حل شوند، هر چند ممکن است روش‌های دیگری نیز برای حل آن‌ها وجود داشته باشد.

### تمرین اول: شطرنج با مانع



شیرفرهاد که از شطرنج عادی خسته شده بود، تصمیم گرفت مانند اسب‌سواری و دوندگی با مانع، شطرنج با مانع برگزار کند. در شطرنج با مانع، یک سری خانه‌ها بسته شده‌اند و نمی‌توان روی آن‌ها مهره‌ای قرار داد، اما مانع‌ها جلوی تهدید مهره‌ها را نمی‌گیرند. برای مثال اگر دو رخ در یک ردیف قرار داشته باشند و بینشان مانع وجود داشته باشد همچنان یکدیگر را تهدید می‌کنند.

شیرفرهاد بعد از خفن شدن در شطرنج با مانع و گذاشتن مگنوس کارلسن در جیبش، تصمیم گرفت مسئله هشت وزیر را با مانع حل کند. برای این کار او به شما یک صفحه شطرنج با مانع می‌دهد و از شما تعداد حالات قرار دادن هشت وزیر به طوری که هیچ دو وزیری یکدیگر را تهدید نکنند را می‌خواهد.

## ورودی

ورودی شامل هشت خط است که هر خط یک رشته هشت حرفی است. این ۶۴ حرف نشان‌دهنده خانه‌های جدول هستند. به ازای هر خانه عادی در خانه متناظرش یک کاراکتر '!' به شما داده شده و به ازای هر خانه با مانع، در خانه متناظرش یک کاراکتر '\*' به شما داده می‌شود.

مثال ورودی
.....
.....
..*.....
.....
.....
.....**.
..*.....
.....

## خروجی

تعداد راه حل‌های مسئله هشت وزیر در جدول داده شده را خروجی بدهید.

مثال خروجی
65

## تمرین دوم: کیوون دو برره



کیوون که به واسطه گفتن "پول وده" و گرفتن پول زور از مردم شهرت داشت، این بار دچار مشکل بزرگی شده است. روزی کیوون در حالی که از مردم پول طلب می‌کرد، متوجه شد که امروزه همه پرداخت‌ها با کارت بانکی و به صورت الکترونیکی انجام می‌شود و دیگر کسی پول نقد به همراه ندارد. او از شدت شرمندگی و بی‌پولی تصمیم گرفت به جای مطالبه پول از مردم، تحصیل کرده و ریاضیات بیاموزد تا راهی تازه برای گذران زندگی پیدا کند.

یکی از تمرینات ریاضی که استاد به کیوون برره داده، محاسبه تعداد نتایج مختلف یک عبارت ریاضی با استفاده از پرانتزگذاری‌های متفاوت است. این عبارت ریاضی شامل اعداد طبیعی یک رقمی و عملگرهای +، - و \* است. هیچ دو عملگری پشت سر هم نمی‌آیند و بین هر دو عملگر متوالی، یک عدد وجود خواهد داشت. حالا کیوون شما را گیر انداخته و با گفتن "جواب وده" از شما می‌خواهد که به زور این مسئله را برایش حل کنید. دقت کنید که ترتیب عملگرها باید ثابت بماند و فقط پرانتزگذاری می‌تواند به روش‌های مختلفی انجام شود.

### ورودی

در خط اول عدد  $n$  می‌آید که نشان‌دهنده تعداد اعداد موجود در عبارت است (عبارت با رقم شروع شده و با - یا + یا بقیه عملگرها شروع نمی‌شود). در خط بعدی عبارت ریاضی به صورت یک رشته و در یک خط به شما داده می‌شود. ( $n \leq 10$ )

#### مثال ورودی

4

$2*3-4*5$

## خروجی

فقط یک عدد صحیح که تعداد نتایج مختلف قابل به دست آمدن از عبارت با پرانتزگذاری‌های متفاوت را نشان می‌دهد.

### مثال خروجی

4

در این مثال، نتایج مختلف زیر با پرانتزگذاری‌های متفاوت به دست می‌آید:

$$2 * (3 - (4 * 5)) = - 34$$

$$(2 * 3) - (4 * 5) = - 14$$

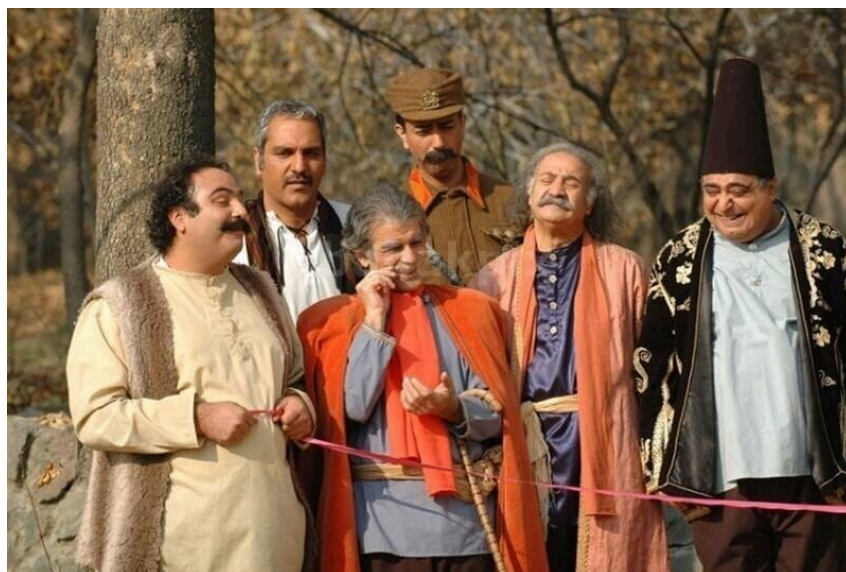
$$(2 * (3 - 4)) * 5 = - 10$$

$$2 * ((3 - 4) * 5) = - 10$$

$$((2 * 3) - 4) * 5 = + 10$$

در نتیجه، تعداد نتایج مختلف برابر با ۴ است.

## تمرین سوم: شیرینی شایان



آقا شایان که از عهد قجر در شرکت دوتانت کار می‌کرد بالاخره توانست با حقوقش ماشین بخرد. همانطور که می‌دانید رسم شیرینی گرفتن در دانشگاه تهران شعبه برره بسیار مرسوم است و شایان باید به همه شیرینی بدهد. شعبه برره دانشگاه تهران را می‌توان به شکل یک خیابان دید که در آن  $n$  دانشکده پشت سر هم قرار دارند. در دانشکده  $i$ ام  $a_i$  دانشجو وجود دارد. ماشین آقا شایان در حال حاضر جلوی دانشکده اول پارک است و آقا شایان می‌خواهد به دانشکده  $n$  برود. ماشین آقا شایان که نه بوق دارد نه صدلی با هر استارت حداکثر می‌تواند دو دانشکده جلو برود و بعد از آن خاموش می‌کند. برای مثال وقتی آقا شایان در دانشکده اول است فقط می‌تواند به دانشکده‌های دوم و سوم برود. در صورتی که آقا شایان در یک مرحله از دانشکده  $i$ ام به دانشکده  $j$ ام برود باید  $|a_i - a_j|$  تا شیرینی بدهد. از آنجا که آقا شایان به حقوق بخور و نمیر دوتانت قانع است به او کمک کنید با دادن کمترین تعداد شیرینی ممکن به هدفش برسد تا ورشکست نشود.

### ورودی

در خط اول عدد  $n$  می‌آید که تعداد دانشکده‌های دانشگاه تهران شعبه برره است.  $2 \leq n \leq 30$   
پس از آن در خط دوم  $n$  عدد به شما داده می‌شوند که  $i$ امین آن‌ها برابر  $a_i$  است.  $1 \leq a_i \leq 10000$

### مثال ورودی

6

30 10 60 10 60 50

## خروجی

یک عدد خروجی بدهید که برابر حداقل تعداد شیرینی‌ای است که آقا شایان باید بدهد.

مثال خروجی

40

## نکات و نحوه تحویل

- تمامی تحویل‌های فردی و گروهی از طریق گیت‌هاب انجام شده و امکان بارگذاری کد در سامانه ای‌لرن وجود نخواهد داشت.
- دقت کنید که دسترسی مخزن<sup>1</sup> شما باید به صورت خصوصی<sup>2</sup> تعریف شود و نام مخزن باید به صورت `<SID>-A3-F03-AP` باشد. همچنین، کاربر `AP-UT` را به مخزن اضافه کنید.
- کد هر سوال را در یک فایل مجزا با فرمت `Q#-<SID>.cpp` قرار دهید. برای مثال نام فایل حاوی کد پاسخ سوال ۱ می‌شود `Q1-810102000.cpp`. سپس فایل‌های کدهای خود را در مخزن ساخته شده قرار دهید. توجه کنید که فایل‌های کدهای شما باید در صفحه اول مخزن باشد و نباید در هیچ پوشه‌ای قرار داده شده باشد.
- سعی کنید مرحله به مرحله کد را `commit` کنید و از `commit` کردن کل کد در انتهای پروژه خودداری کنید. همچنین، پیام `commit`-های شما باید خوانا باشد و تغییر ایجاد شده را توضیح دهد. دقت کنید که بخشی از نمره شما را استفاده صحیح از گیت تشکیل می‌دهد. برای مطالعه در خصوص نحوه صحیح `commit` کردن، می‌توانید [این لینک](#) را مطالعه کنید.
- برای تحویل پروژه کافی است در بخش "متن بر خط" در سامانه ای‌لرن، در خط اول لینک مخزن در گیت‌هاب و در خط دوم هش مربوط به آخرین `commit` پروژه را قرار دهد. نمونه‌ای از فرمت خواسته شده در ادامه آمده است. لازم به ذکر است که هم تاریخ قرار دادن متن خواسته شده در ای‌لرن و هم تاریخ آخرین `commit` شما باید پیش از ددلاین تمرین باشد. نمونه متن خواسته شده در سامانه ای‌لرن (سه بخش `<username>` و `<repository_name>` و `<last_commit_hash>` را جایگزین کنید):

```
https://github.com/<username>/<repository_name>  
<last_commit_hash>
```

- با توجه به اینکه کدها به صورت خودکار از گیت‌هاب دریافت می‌شوند، عدم رعایت فرمت ذکر شده سبب کسر 5 درصد از نمره شما خواهد شد.
- برنامه شما باید در سیستم عامل لینوکس و با مترجم `g++` با استاندارد `C++20` ترجمه و در زمان معقول برای ورودی‌های آزمون اجرا شود.
- درستی برنامه شما از طریق آزمون‌های خودکار سنجیده می‌شود؛ بنابراین از درستی کامل قالب خروجی برنامه خود اطمینان حاصل کنید و اکیدا از دادن خروجی‌هایی که در صورت پروژه ذکر نشده است اجتناب کنید.

---

<sup>1</sup> Repository

<sup>2</sup> Private

- ممکن است نکات جدیدی در خصوص پروژه در تالار گفتگو مطرح شود که در نمره‌دهی و نحوه آزمون پروژه شما موثر خواهد بود.
- هدف این تمرین یادگیری شماست. لطفاً تمرین را خودتان انجام دهید. در صورت کشف تقلب مطابق سیاست درس با آن برخورد خواهد شد.

## نمرات

- تمیزی کد
  - رعایت کردن نام‌گذاری صحیح و انسجام<sup>3</sup>
  - عدم وجود کد تکراری
  - رعایت دندانه‌گذاری<sup>4</sup>
  - استفاده صحیح از متغیرهای ثابت به جای Magic Value-ها
  - ساختاردهی کد در قالب توابع کوتاه که فقط یک کار را انجام می‌دهند
- درستی کد
  - آزمون‌های خودکار
- طراحی بازگشتی
  - پیاده‌سازی الگوریتم بازگشتی و عدم استفاده از حلقه (به جز ورودی/خروجی)
- گیت و گیت‌هاب
  - استفاده از commit message‌های مناسب
  - هر کامیت یک کار مشخص انجام بدهد و کامیتی چندین کار انجام ندهد.

دقت کنید که موارد ذکر شده لزوماً کل نمره شما را تشکیل نمی‌دهند و ممکن است با تغییراتی همراه باشند.

---

<sup>3</sup> Consistency

<sup>4</sup> Indentation