



## 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.

## پرسش‌ها

### ۱. پردازنده

#### شرح مسئله

شرکت اینتل (intel) پردازنده‌های خود را در قالب 3 سری core i3 و core i5 و core i7 می‌سازد. فرض می‌کنیم هر نسل جدید پردازنده‌های این شرکت طی 3 سال عرضه می‌شود، پردازنده core i3 سال اول، پردازنده core i5 سال دوم و پردازنده core i7 سال سوم یک نسل عرضه می‌شوند. در نهایت پس از عرضه کامل یک نسل، نسل بعدی عرضه می‌شود. ترتیب عرضه پردازنده‌ها در لیست زیر مشخص شده است:

1- سال اول: پردازنده core i3 نسل اول

2- سال دوم: پردازنده core i5 نسل اول

3- سال سوم: پردازنده core i7 نسل اول

4- سال چهارم: پردازنده core i3 نسل دوم

این روند به همین شکل ادامه پیدا می‌کند.

توان پردازشی پردازنده core i3 هر نسل، تابعی از نسل قبلی خود است:

$$f_{i3}(n) = 1.5 \times f_{i3}(n - 1)$$

توان پردازشی پردازنده core i5 هر نسل، تابعی از نسل قبل خود و همچنین توان پردازشی پردازنده core i3 نسل فعلی است:

$$f_{i5}(n) = 1.2 \times f_{i5}(n - 1) + 0.2 \times f_{i3}(n)$$

به همین ترتیب، توان پردازشی پردازنده core i7 هر نسل نیز از رابطه زیر بدست می‌آید:

$$f_{i7}(n) = 1.3 \times f_{i7}(n - 1) + 0.3 \times f_{i5}(n) + 0.2 \times f_{i3}(n)$$

توان پردازشی پردازنده‌های نسل اول نیز به صورت زیر است:

$$f_{i3}(1) = 1, f_{i5}(1) = 1.4, f_{i7}(1) = 1.7$$

وظیفه شما پیدا کردن توان پردازشی پردازنده‌ای است که در ورودی به شما داده شده است.

## قالب ورودی

ورودی شامل یک خط است که عدد n به عنوان سال مورد نظر به شما داده شده است.

## قالب خروجی

خروجی باید شامل دو خط باشد که خط اول نوع پردازنده آن سال و خط دوم که شامل یک عدد اعشاری است، توان پردازشی آن پردازنده را نشان می‌دهد.

## ورودی و خروجی نمونه

ورودی	خروجی
5	core i5 1.98

توضیحات:

- پردازنده خواسته شده در واقع پردازنده core i5 نسل دوم است:

$$f_{i3}(2) = 1.5 \times 1 = 1.5, f_{i5}(2) = 1.2 \times 1.4 + 0.2 \times 1.5 = 1.98$$

ورودی	خروجی
7	core i3 2.25

توضیحات:

- پردازنده خواسته شده در واقع پردازنده core i3 نسل سوم است:

$$f_{i3}(3) = 1.5 \times 1.5 = 2.25$$

## ۲. کدگذاری

### شرح مسئله

برای ارسال داده به صورت محرمانه از روش های مختلفی استفاده می شود. یک روش ساده برای کدگذاری یک رشته به این شکل است که به ازای یک رشته، حروف سمت چپ به صورت **یکی در میان** با حرفی از سمت راست رشته swap می شوند. این کار تا جایی ادامه دارد که حرکت دو سمت از هم عبور کنند.

در صورتی که رشته شامل فقط یک حرف باشد، حاصل کدگذاری با خود رشته ورودی یکسان است.

در صورتی که رشته شامل دو حرف باشد، حاصل کدگذاری معادل جابه جا شدن آن دو حرف است.

برای فهم بهتر این روش به ورودی و خروجی های نمونه دقت کنید.

### قالب ورودی

ورودی شامل یک خط است که رشته مورد نظر برای کدگذاری است.

### قالب خروجی

خروجی شامل یک خط است که رشته کدگذاری شده با روش مطرح شده است.

### ورودی و خروجی نمونه

ورودی	خروجی
xyz	zyx

### توضیحات:

حرکت از سمت چپ ابتدا روی حرف x و حرکت از سمت راست روی حرف z است.

این دو حرف با هم swap می شوند و در ادامه، حرکت سمت چپ دو واحد به راست میاید (روی حرف z) و حرکت سمت

راست یک واحد به سمت چپ میاید (روی حرف y).

از آنجا که حرکت دو سمت از هم عبور کردند پس کار اتمام یافته و رشته کنونی که zyx است نتیجه کدگذاری می شود.

ورودی	خروجی
abcdefg	gbfdeca

توضیحات:

حرف a با g، حرف c با f، و حرف e با خودش swap می شود.

ورودی	خروجی
123456789	928476531

توضیحات:

حرف 1 با 9، حرف 3 با 8، و حرف 5 با 7 swap می شود.

## ۳. سودوکو

### شرح مسئله

سودوکو جدول اعدادی است که یکی از بهترین تمرین‌های مغز و تقویت آی کیو است. امروزه این بازی یکی از سرگرمی‌های رایج در کشورهای مختلف جهان به شمار می‌آید. نوع متداول سودوکو یک جدول ۹×۹ است که کل جدول هم به ۹ جدول کوچک‌تر ۳×۳ تقسیم شده است. در این جدول چند عدد به طور پیش‌فرض قرار داده شده که باید باقی اعداد را با رعایت سه قانون زیر یافت:

قانون اول: در هر سطر جدول اعداد ۱ الی ۹ بدون تکرار قرار گیرد.

قانون دوم: در هر ستون جدول اعداد ۱ الی ۹ بدون تکرار قرار گیرد.

قانون سوم: در هر ناحیه ۳×۳ جدول اعداد ۱ الی ۹ بدون تکرار قرار گیرد.

### قالب ورودی

یک جدول ۹×۹ با فرمت مشابه مثال به شما داده خواهد شد.

### قالب خروجی

خروجی باید شامل جدول پر شده با رعایت سه قانون فوق باشد.

### ورودی و خروجی نمونه

ورودی	خروجی
3 9 0 0 5 0 0 0 0 0 0 0 2 0 0 0 0 5 0 0 0 7 1 9 0 8 0 0 5 0 0 6 8 0 0 0 2 0 6 0 0 3 0 0 0 0 0 0 0 0 0 0 0 4 5 0 0 0 0 0 0 0 0 6 7 0 1 0 5 0 4 0 1 0 9 0 0 0 2 0 0	3 9 1 8 5 6 4 2 7 8 6 7 2 3 4 9 1 5 4 2 5 7 1 9 6 8 3 7 5 4 9 6 8 1 3 2 2 1 6 4 7 3 5 9 8 9 3 8 5 2 1 7 6 4 5 4 3 6 9 2 8 7 1 6 7 2 1 8 5 3 4 9 1 8 9 3 4 7 2 5 6

ورودی	خروجی
5 3 0 0 7 0 0 0 0	5 3 4 6 7 8 9 1 2
6 0 0 1 9 5 0 0 0	6 7 2 1 9 5 3 4 8
0 9 8 0 0 0 0 6 0	1 9 8 3 4 2 5 6 7
8 0 0 0 6 0 0 0 3	8 5 9 7 6 1 4 2 3
4 0 0 8 0 3 0 0 1	4 2 6 8 5 3 7 9 1
7 0 0 0 2 0 0 0 6	7 1 3 9 2 4 8 5 6
0 6 0 0 0 0 2 8 0	9 6 1 5 3 7 2 8 4
0 0 0 4 1 9 0 0 5	2 8 7 4 1 9 6 3 5
0 0 0 0 8 0 0 7 9	3 4 5 2 8 6 1 7 9

## نکات و نحوه تحویل

- توجه داشته باشید که الگوریتم‌های بازگشتی یا Backtracking در حالت کلی لزوماً کارآمد<sup>1</sup> نیستند. هدف شما در این تکلیف رسیدن به یک الگوریتم صحیح با استفاده از روش‌های بازگشتی و Backtracking است، نه یک الگوریتم کارآمد با سریع‌ترین زمان حل ممکن.
- کد هر سوال را در یک فایل مجزا با فرمت Q#.cpp قرار دهید. برای مثال نام فایل حاوی کد پاسخ سوال 1 می‌شود Q1.cpp. سپس کدهای خود را در قالب یک فایل فشرده با نام A2-SID.zip در صفحه‌ی ایلرن درس بارگذاری کنید که SID شماره دانشجویی شماست؛ برای مثال اگر شماره‌ی دانشجویی شما ۸۱۰۱۰۰۰۰ باشد، نام پرونده کد شما باید A2-810100000.zip باشد که شامل کد شما است.
- برنامه شما باید در سیستم عامل لینوکس و با مترجم g++ با استاندارد c++11 ترجمه و در زمان معقول برای ورودی‌های آزمون اجرا شود.
- در این تمرین نیز مانند دیگر تمرین‌ها تمیزی کد، شکستن مرحله‌به‌مرحله مسئله و طراحی مناسب، در کنار تولید خروجی دقیق و درست، بخش مهمی از نمره شما را تعیین خواهد کرد.
- درستی برنامه شما از طریق آزمون‌های خودکار سنجیده می‌شود؛ بنابراین از درستی کامل قالب خروجی برنامه خود اطمینان حاصل کنید و از دادن خروجی‌هایی که در صورت پروژه ذکر نشده‌است اجتناب کنید.

<sup>1</sup> Efficient

- هدف این تمرین یادگیری شماسست. لطفاً تمرین را خودتان انجام دهید. در صورت کشف تقلب مطابق سیاست درس با آن برخورد خواهد شد.