

دستگرمی

- محدودیت زمان: 1 ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت
- طراح: میثاق محقق

برای پاسخ به این سوال حتما از کد قالب داده شده استفاده کنید.

در این قسمت باید به PDF تمرین و قالب داده شده مراجعه کرده و توابع مربوط به کلاس‌ها را که خالی هستند با کد مناسب کامل کنید.

توجه کنید که شما نیازی به هندل کردن ورودی و خروجی برنامه ندارید و صرفاً متدهای کلاس‌های قالب داده شده را پیاده‌سازی می‌کنید. مقدار ریترن هر متد به طور خودکار چاپ شده و شما در متدهای کلاس از `print` استفاده نمی‌کنید.

ورودی

با توجه به قالب داده شده، ابتدا یک یا چند آبجکت از نوع پشته و صف و لینکد لیست ایجاد می‌شود. سپس توابع مشخص شده برای هر کدام از آبجکت‌ها صدا زده می‌شوند و در صورت داشتن خروجی چاپ می‌شوند. همه توابع استفاده شده در قالب آمده است و توضیح مربوط به هر کدام در PDF تمرین آورده شده است.

مثال

ورودی نمونه ۱

```
make queue q1
call q1.enqueue(1)
call q1.enqueue(2)
call q1.enqueue(3)
call q1.empty()
call q1.size()
```

```
call q1.one_line_str()
call q1.dequeue()
call q1.dequeue()
call q1.dequeue()
call q1.empty()
call q1.size()
```

خروجی نمونه ۱

```
False
3
1 2 3
1
2
3
True
0
```

ورودی نمونه ۲

```
make stack s1
call s1.push(1)
call s1.push(2)
call s1.push(3)
call s1.empty()
call s1.size()
call s1.one_line_str()
call s1.pop()
call s1.pop()
call s1.pop()
call s1.empty()
call s1.size()
```

خروجی نمونه ۲

False

3

1 2 3

3

2

1

True

0

```
@media print { .ms-editor-squiggler { display:none !important; } } .ms-editor-squiggler { all: initial;
display: block !important; height: 0px !important; width: 0px !important;
```

```
@media print { .ms-editor-squiggler { display:none !important; } } .ms-editor-squiggler { all: initial;
display: block !important; height: 0px !important; width: 0px !important;
```

```
@media print { .ms-editor-squiggler { display:none !important; } } .ms-editor-squiggler { all: initial;
display: block !important; height: 0px !important; width: 0px !important;
```

مرد ری فکتور

- محدودیت زمان: ۱ ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت
- طراح: میثاق محقق

برای پاسخ به این سوال حتما از کد قالب داده شده استفاده کنید.

رئیس سازمان مسعود از کد پروژه ایراد گرفته و گفته که متن کد در صفحه گوشی همراه او جا نمی‌شود! این مشکل به خاطر indentation زیاد و تو رفتگی‌های کد هست که اگر از فلسفه دیدن کد در گوشی بگذریم، حرف بدی هم نیست! برای همین مسعود تصمیم گرفت که کدش یک لایه indent بیشتر نداشته باشد! برای این کار مسعود می‌خواهد هر scope جدیدی که داریم (که منجر به تو رفتگی جدیدی می‌شود) را با یک فراخوانی تابع جایگزین کند.

ورودی

در ورودی یک نمونه از کد مسعود که در زبان غیررسمی Painthon زده شده قرار می‌گیرد. سینتکس این زبان به صورت زیر است. همیشه در انتهای هر بلاک `# end_scope` قرار می‌گیرد و همه if-ها برای سادگی شرط True دارند. برای درک بهتر این زبان به ورودی و خروجی‌های نمونه رجوع کنید.

- Function Definition
- If Statement
- Simple Statements

خروجی

در خروجی، نسخه ری فکتور شده کد مسعود می‌آید که به ازای هر if statement یک تابع داریم و خود if-ها با فراخوانی تابع جایگزین شده اند.

توضیح کد قالب

کد قالب در کنار صورت پروژه به شما داده شده است. این کد قالب ورودی را خوانده و آن را پردازش می‌کند. شما فقط کافیست که تابع `print_result` را پیاده‌سازی کنید و نیازی نیست به هیچ جای دیگر کد دست بزنید. کارکرد کد به این صورت است که کلاس `Runner` ورودی را خوانده و سپس بنا بر نوع `statement` (که می‌تواند تعریف تابع، گزاره شرطی، `statement` ساده و یا `end scope` باشد) متد متناظر کلاس `Handler` را صدا می‌زند تا در آنجا پردازش بشود.

در نهایت همه `statement`-ها به عنوان یک `Node` ذخیره می‌شوند که هر `Node` می‌تواند چندین فرزند `Node` داشته باشد. مثلاً فرزندان یک `IfStatement` شامل محتوای داخل آن گزاره شرطی است. `SimpleStatement`-ها فرزندی ندارند.

شما در تابع `print_result` یک ورودی `program_node` می‌گیرید که فرزندان آن `FunctionStatement` هستند. حال شما باید با استفاده از یک صف، کد ری‌فکتور شده را در این تابع پرینت کنید.

مثال

ورودی نمونه ۱

```
1 def main():
2     first_statement
3     if True:
4         inner_statement
5         # end_scope
6     # end_scope
```

خروجی نمونه ۱

```
1 def main():
2     first_statement
3     condition_1()
4
5 def condition_1():
6     inner_statement
```

توضیح

همانطور که می‌بینیم، گزاره شرطی به یک تابع تبدیل شده و از indentation زیاد کد جلوگیری شده است. فیلد `node.name` همه گزاره‌های شرطی در کد قالب، مقدار `'condition'` را دارند که ما اینجا باید با استفاده از یک شمارنده مقدار `_1` را به انتهای نام تابع اضافه کنیم تا نام آنها یکسان نباشد.

ورودی نمونه ۲

```
1 def myfunc():
2     statement1
3     statement2
4     statement3
5
6     if True:
7         i_am_first_if
8
9         if True:
10            this_is_inner_if
11            this_is_inner_if2
12            if True:
13                very_nested
14                # end_scope
15            # end_scope
16
17            i_am_first_if_end
18            # end_scope
19
20     if True:
21         another_if
22
```

```
# end_scope

statement100
# end_scope
```

خروجی نمونه ۲

```
1  def myfunc():
2      statement1
3      statement2
4      statement3
5      condition_1()
6      condition_2()
7      statement100
8
9  def condition_1():
10     i_am_first_if
11     condition_3()
12     i_am_first_if_end
13
14  def condition_2():
15     another_if
16
17  def condition_3():
18     this_is_inner_if
19     this_is_inner_if2
20     condition_4()
21
22  def condition_4():
23     very_nested
```

توضیح

در این مثال نحوه هندل شدن شرط‌های nested را می‌بینیم که مانند قبل است. از آنجا که برای پیاده‌سازی از یک صف استفاده می‌کنید نحوه شماره‌گذاری توابع نیز به شکلی که در خروجی مشخص است در خواهد آمد.

نکته: در صورت وجود بیش از یک تابع، کد مربوط به ری‌فکتور کردن یک تابع را در چرخه به ازای هر تابع به ترتیب

اجرا کنید.

نبرد فاضلابی

• محدودیت زمان: 2 ثانیه

• محدودیت حافظه: ۲۵۶ مگابایت

تعدادی موش در لوله فاضلاب به صورت صف قرار گرفته و هرکدام به جهت راست یا چپ حرکت می‌کنند. در صورت برخورد دو موش با یکدیگر، آنها با هم می‌جنگند و موش با قدرت کمتر می‌میرد (اگر قدرت دو موش برابر بود، هردو می‌میرند). پس از اتمام این نبردها، موش‌های باقی‌مانده به سمت مخفی‌گاه استاد اسپلینتر حرکت می‌کنند و در نهایت با او می‌جنگند.

استاد اسپلینتر از شما می‌خواهد تا به او کمک کنید که بفهمد با چه موش‌هایی خواهد جنگید تا بتواند خود و لاک‌پشت‌های نینجا را برای این نبرد آماده کند. صف موش‌های داخل لوله، به صورت یک لیست به شما داده می‌شود و هر موش با یک عدد نشان داده می‌شود (عدد منفی نشان‌دهنده حرکت به سمت چپ و عدد مثبت نشان‌دهنده حرکت به راست می‌باشد و همچنین مقدار عدد برابر مقدار قدرت موش است).

ورودی

ورودی به صورت یک خط که نشان‌دهنده لیست موش‌ها (Rats) با طول L داده می‌شود. لیست با '[' شروع شده و با ']' به اتمام می‌رسد. همچنین اعضای لیست با ',' از هم جدا می‌شوند.

$$2 \leq L \leq 10^4$$

$$-10^3 \leq Rats[i] \leq 10^3$$

$$Rats[i] \neq 0$$

خروجی

همانند ورودی، یک لیست از موش‌ها در یک خط باید چاپ شود (لیست با '[' شروع شده و با ']' به اتمام می‌رسد و

همچنین اعضای لیست با '' از هم جدا می‌شوند). هر موش نیز باید مشابه ورودی، با یک عدد نشان داده شود.

مثال

ورودی نمونه ۱

[7,8,-7]

خروجی نمونه ۱

[7,8]

توضیح: موش 8 با موش 7 برخورد کرده و موش 7- می‌میرد.

ورودی نمونه ۲

[5,-5]

خروجی نمونه ۲

[]

توضیح: موش 5 با موش 5- برخورد کرده و چون قدرت هر دو برابر است، هر دو می‌میرند.

ورودی نمونه 3

[12,5,-10]

خروجی نمونه 3

[12]

توضیح: موش 5 با موش 10- برخورد کرده و موش 5 می‌میرد و سپس موش 12 با موش 10- برخورد می‌کند و 10- می‌میرد.

کتاب شبانه

- محدودیت زمان: 3 ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

دیر وقت است اما علی خوابش نمی‌برد، او تصمیم می‌گیرد که از کتابخانه کتابی بردارد و بخواند. در یک طبقه از کتابخانه کتاب‌های علی قرار گرفته است و او یک برچسب عددی به معنای شدت جذابیت کتاب روی هر کدام چسبانده است. از آنجایی که علی نمی‌خواهد بقیه را بیدار کند چراغ را روشن نمی‌کند و با چراغ قوه از سمت چپ ردیف کتاب‌هایش شروع به جستجو می‌کند و با توجه به نور چراغ قوه، در هر لحظه می‌تواند k کتاب را ببیند و از آنجایی که می‌خواهد در اتاقش با وجود روشنایی و آرامش بیشتر کتاب خود را انتخاب کند، در هر بار دیدن کتاب‌ها، از بین k کتابی که می‌بیند بهترین کتاب را برمی‌دارد تا به اتاق ببرد (فرض کنید به ازای هر کتاب، بی‌نهایت کتاب دیگر با همان امتیاز پشت آن قرار دارد و با برداشتن یک کتاب، کتاب دیگری با همان امتیاز جایگزین آن می‌شود) و سپس یک قدم به راست حرکت می‌کند تا بتواند کتاب بعدی را نیز ببیند (سمت چپی ترین کتاب از دید او حذف می‌شود). لیست کتاب‌هایی که علی به اتاق خود می‌برد را با رعایت ترتیب پیدا کنید.

ورودی

در خط اول ورودی، کتاب‌های علی در کتابخانه به صورت یک لیست (Books) با طول L داده می‌شود (لیست با '[' شروع شده و با ']' به اتمام می‌رسد. همچنین اعضای لیست با ',' از هم جدا می‌شوند).

در خط دوم ورودی، k داده می‌شود.

$$1 \leq L \leq 10^5$$

$$-10^4 \leq Books[i] \leq 10^4$$

$$1 \leq k \leq L$$

خروجی

برابر است با لیست کتاب‌هایی که علی به اتاقش می‌برد. این لیست در یک خط چاپ می‌شود (لیست با '[' شروع شده و با ']' به اتمام می‌رسد. همچنین اعضای لیست با ',' از هم جدا می‌شوند).

مثال

ورودی نمونه ۱

[2, -1, 3, 12, -4, 10, 2, 34]

3

خروجی نمونه ۱

[3, 12, 12, 12, 10, 34]

توضیح:

$$[2, -1, 3, 12, -4, 10, 2, 34] \rightarrow [2, -1, 3] \rightarrow \max([2, -1, 3]) \rightarrow 3$$

$$[2, -1, 3, 12, -4, 10, 2, 34] \rightarrow [-1, 3, 12] \rightarrow \max([-1, 3, 12]) \rightarrow 12$$

$$[2, -1, 3, 12, -4, 10, 2, 34] \rightarrow [3, 12, -4] \rightarrow \max([3, 12, -4]) \rightarrow 12$$

$$[2, -1, 3, 12, -4, 10, 2, 34] \rightarrow [12, -4, 10] \rightarrow \max([12, -4, 10]) \rightarrow 12$$

$$[2, -1, 3, 12, -4, 10, 2, 34] \rightarrow [-4, 10, 2] \rightarrow \max([-4, 10, 2]) \rightarrow 10$$

$$[2, -1, 3, 12, -4, 10, 2, 34] \rightarrow [10, 2, 34] \rightarrow \max([10, 2, 34]) \rightarrow 34$$

شهر وارونه

- محدودیت زمان: 2 ثانیه
- محدودیت حافظه: ۲۵۶ مگابایت

روزی روزگاری در یک شهر کوچک ، خانه‌هایی وجود داشتند که هرکدام یک شماره به خود اختصاص داده بودند و با راه‌هایی به خانه‌های دیگر می‌رسیدند. مردم این شهر همیشه با نظمی خاص در کنار هم زندگی می‌کردند؛ به ترتیب از خانه‌ی اول تا خانه‌ی آخر، یکی یکی به هم متصل بودند.

روزی، پادشاه شهر، «آقای معکوس»، تصمیم گرفت که نظم این شهر را کمی تغییر دهد. او می‌خواست خانه‌های مردم را گروه‌بندی کند و ترتیب خانه های هر گروه را معکوس کند. به این ترتیب مردم می‌توانستند همسایه‌های جدیدی پیدا کنند و تجربه‌ای تازه از شهرشان داشته باشند.

آقای معکوس برای اجرای این نقشه، هر دفعه که تصمیم به عمل گرفت، مقدار عدد x را انتخاب کرده و خانه‌ها را به گروه‌هایی با اندازه x می‌شکند و خواستار او معکوس شدن ترتیب هر گروه است. همچنین اگر پس از گروه بندی، چند خانه باقی ماندند که تعداد آنها از x کمتر بود، او خواستار به هم زدن ترتیب آنها نیست.

آقای معکوس شهر را (خانه ها و راه های بین آنها) و مقدار عدد x به شما می‌دهد و شما باید شهر معکوس شده را حساب کرده و به او بدهید (تغییر مقدار داخل گره‌ها مجاز نیست و فقط می‌توانید خود گره‌ها را تغییر دهید).

ورودی

ورودی تنها در یک خط داده می‌شود که اول، لیست خانه ها (n تا خانه) داده می‌شود (لیست با '[' شروع شده و با ']' به اتمام می‌رسد و با ',' از هم جدا می‌شوند) و سپس مقدار x به صورت " $x = \text{some number}$ " داده می‌شود (خانه‌ها و عدد x با " , " از هم جدا می‌شوند).

$$1 \leq x \leq n \leq 5 * 10^3$$

$$0 \leq \text{Node.value} \leq 10^3$$

خروجی

در یک خط باید چاپ شود که برابر با لیست خانه‌ها پس از معکوس کردن است (لیست با '[' شروع شده و با ']' به اتمام می‌رسد و اعضای لیست با ',' از هم جدا می‌شوند).

مثال

ورودی نمونه ۱

$[1, 3, 2, 4, 5, 6, 7, 8], x = 3$

خروجی نمونه ۱

$[2, 3, 1, 6, 5, 4, 7, 8]$

توضیح: ۸ خانه داریم و آنها به دو گروه $\{1, 2, 3\}$ و $\{4, 5, 6\}$ دسته بندی می‌شوند. و سپس آنها را معکوس می‌کنیم (خانه‌های ۷, ۸ داخل هیچ گروهی جای نمی‌گیرند و ترتیبشان به هم نمی‌خورد).

ورودی نمونه ۲

$[1, 2, 3, 4, 5, 6, 7, 8], x = 2$

خروجی نمونه ۲

$[2, 1, 4, 3, 6, 5, 8, 7]$

توضیح: ۸ خانه داریم و آنها به چهار گروه $\{1, 2\}$ و $\{3, 4\}$ و $\{5, 6\}$ و $\{7, 8\}$ دسته بندی می‌شوند و آنها را معکوس می‌کنیم.