



به نام خدا
سیستم‌های توزیع شده
۲-۱۳۹۸

تمرین سوم
مدرس: صابر صالح

نکات مهم

لطفا ابتدا به نکات زیر توجه کنید:

- برای پیاده‌سازی این تمرین از زبان Python استفاده نمایید.

- پاسخ تمرین را فقط در Quera آپلود کنید.

- به شیوه‌ی ورودی و خروجی دقت کنید.

- مهلت ارسال تمرین تا پایان روز ۱۳۹۹/۰۳/۰۲ می‌باشد.

موفق باشید.

۱ مقدمه

در این تمرین قصد داریم توافق همه‌ی اعضای شبکه بر روی یک مقدار را پیاده کنیم. حالتی خاص از این مسئله را در تمرین «انتخاب رهبر» مشاهده کردید.

در اینجا شبکه‌ای از گره‌ها وجود دارد که در آن هر گره می‌تواند عددی را پیشنهاد دهد. در نهایت از بین اعداد پیشنهادی، همه باید بر روی یک مقدار توافق کنند. الگوریتم Paxos به همین منظور ارائه شده است.

۲ Paxos

برای توافق نهایی لازم است سه فاز زیر به ترتیب توسط یکی از گره‌ها طی شود:

◀ فاز ۱: اقدام برای رهبری

◀ فاز ۲: به دست آوردن آراء لازم برای رهبری و ارسال مقدار پیشنهادی

◀ فاز ۳: به دست آوردن آراء لازم برای مقدار پیشنهاد داده شده و اعلام توافق نهایی

گفته شد که هر گره مقداری را می‌تواند پیشنهاد دهد. برای این امر لازم است پیش از آن، گره به عنوان رهبر انتخاب شده باشد. اعضای شبکه در کنار رای مثبت به وی، مقداری را با بزرگترین شناسه که توسط رهبرهای پیشین پیشنهاد شده (در صورت وجود) به رهبر جدید ارسال می‌کنند.

حال رهبر از بین مقادیر پیشنهادی یا مقدار پیشنهادی خود، یکی را انتخاب کرده و به همه می‌فرستد و مجدداً منتظر پاسخ اعضا می‌ایستد. چنانچه مقدار پیشنهادی مورد قبول اکثریت باشد، آن مقدار به عنوان توافق نهایی برای همه ارسال می‌شود.

به طور کلی وظایف گره‌ها را در دو بخش **رهبر** و **فقط پاسخ‌دهنده** بیان می‌کنیم.

۱.۲ گره رهبر

۱.۱.۲ درخواست برای رهبری

هر گره همراه با اعلام درخواست خود برای رهبری، یک عدد را به سایر گره‌ها ارسال می‌نماید. زین پس از این عدد با عنوان «شناسه‌ی رهبر» یاد خواهیم کرد. گره مدعی رهبری باید بزرگترین شناسه‌ای که تا کنون دریافت کرده به علاوه‌ی یک را به همراه پیام خود ارسال کند.

این پیام از نوع POTENTIAL_LEADER بوده و value آن همان شناسه‌ی رهبر است و به همه‌ی گره‌ها فرستاده می‌شود.

```
{"nid": "<node_id>", "type": "POTENTIAL_LEADER", "value": "<leader_id>"}
```

۲.۱.۲ دریافت پاسخ به درخواست رهبری

پس از ارسال درخواست رهبری، گره باید به اندازه‌ی معینی منتظر پاسخ دیگر گره‌ها بماند. چنانچه تعداد پاسخ‌های دریافتی از نصف تعداد کل اعضای شبکه بیش‌تر باشد آن گره به عنوان رهبر انتخاب شده و وارد فاز بعدی یعنی «انتخاب مقدار پیشنهادی» می‌شود. در غیر این صورت، فرآیند رهبری را متوقف کرده و به گره عادی تبدیل می‌شود. رهبر باید مقادیر پیشنهادی رهبران پیشین را از پاسخ‌ها استخراج کرده و برای مرحله‌ی بعد ذخیره نماید.

۳.۱.۲ ارسال مقدار پیشنهادی

این پیام که به همه‌ی گره‌های همسایه فرستاده می‌شود، به فرمت زیر است. leader_id همان شناسه‌ی رهبری (که فرآیند رهبری با آن آغاز شده) و chosen_v مقدار پیشنهادی است.

```
{"nid": "<node_id>", "type": "V_PROPOSE", "value": "<leader_id>, <chosen_v>"}
```

مقدار پیشنهادی برابر با مقدار متناظر با بزرگ‌ترین شناسه‌ی رهبری است که گره حین فرآیند رهبری دریافت

کرده است. چنانچه هیچ گره ای مقداری را نفرستاده بود، رهبر مقدار پیشنهادی خود که برابر حاصل ضرب شناسه‌ی گره (node_id) در اندازه‌ی شبکه است را ارسال می‌کند.

۴.۱.۲ دریافت پاسخ به مقدار پیشنهادی

رهبر پس از ارائه‌ی مقدار پیشنهادی خود منتظر دریافت Acknowledgment گره‌ها می‌ماند. اگر تا پیش از زمانی معین، تعداد Acknowledgment‌ها به بیش از نصف تعداد کل اعضای شبکه برسد، این مقدار به عنوان مقدار نهایی اتخاذ می‌گردد.

چنانچه در زمان تعیین شده نتواند تعداد آرای لازم را جمع‌آوری کند، فرآیند متوقف شده و گره از حالت رهبری خارج می‌شود.

۵.۱.۲ ارسال مقدار نهایی

پس از نهایی شدن یک عدد برای توافق همگانی، رهبر پیامی از نوع V_DECIDE و با value مقدار نهایی، به گره‌ها ارسال می‌کند.

```
{"nid": "<node_id>", "type": "V_DECIDE", "value": "<final_v>"}
```

در این مرحله دیگر نیازی به منتظر بودن برای دریافت Acknowledgment‌ها وجود ندارد.

۶.۱.۲ دریافت پیام POTENTIAL_LEADER از سایر گره‌ها

در صورتی که شناسه‌ی رهبری دریافتی بزرگ‌تر از شناسه‌ی رهبری خود باشد، رهبری را متوقف کرده و مانند سایر گره‌ها به آن پاسخ می‌دهد. در غیر این صورت از این پیام چشم می‌پوشد.

۲.۲ گره‌های فقط پاسخ‌دهنده

منظور از فقط پاسخ‌دهنده گره‌هایی است که در این لحظه رهبر نیستند.

۱.۲.۲ دریافت پیام POTENTIAL_LEADER

اگر شناسه‌ی رهبری موجود در پیام بزرگ‌تر از شناسه‌ی رهبری‌هایی که تا کنون گره دیده باشد، در پاسخ پیامی از نوع POTENTIAL_LEADER_ACK می‌فرستد. value پیام به شکل زیر تعیین می‌شود:

– چنانچه تا کنون مقداری به این گره پیشنهاد نشده باشد، value زوج ۰ و -۱ را قرار می‌دهیم.

```
{"nid": "<node_id>", "type": "POTENTIAL_LEADER_ACK", "value": "0, -1"}
```

– در غیر این صورت، زوج مقدار بزرگ‌ترین شناسه رهبری که به او مقداری پیشنهاد نموده و مقدار پیشنهادی او را در پیام قرار می‌دهد.

```
{"nid": "<node_id>", "type": "POTENTIAL_LEADER_ACK", "value": "<max_leader_id_proposed>,"  
  "<v_from_max_leader_id_proposed>"}
```

لازم به ذکر است در این حالت پیام رهبرهای پیشین را نادیده گرفته و تنها به رهبر جدید پاسخ می‌دهد.

۲.۲.۲ دریافت پیام V_PROPOSE

گره‌ای که این پیام را دریافت کرده ضمن ذخیره‌ی مقدار پیشنهادی به همراه شناسه‌ی رهبر، پیامی از نوع V_PROPOSE_ACK با value برابر ۱- به رهبر ارسال می‌کند.

```
{"nid": "<node_id>", "type": "V_PROPOSE_ACK", "value": "-1"}
```

۳.۲.۲ دریافت پیام V_DECIDE

value این پیام را به عنوان مقدار نهایی که بر روی آن توافق شده ذخیره می‌کند.

۳.۲ نکات کلی

- ◀ گراف شبکه کامل و جهت‌دار است.
- ◀ برای هر گره یک زمان به عنوان دوره‌ی فاز ۱ در نظر گرفته شده است؛ این مقدار برابر مدت زمانی است که گره منتظر مانده و سپس برای رهبری اقدام می‌کند.
- ◀ در ورودی علاوه بر زمان بالا، دو وقفه‌ی دیگر وارد می‌شوند که مدت زمانی است که گره فرصت دارد به ترتیب رای‌های فاز ۱ و فاز ۲ را جمع‌آوری کند. پس از آن رای‌های دریافتی نادیده گرفته می‌شوند.
- برای اطلاع بیش‌تر می‌توانید به مقاله‌ی Paxos مراجعه نمایید.

۳ قالب پیام‌ها

برای یکسان‌سازی نحوه‌ی ورودی و خروجی از قالب زیر استفاده کنید.

```
{"nid": "<node_id>", "type": "<message_type>", "value": "<message_value>"}
```

nid همواره شناسه‌ی گره ارسال‌کننده است.

۴ ورودی‌ها و خروجی‌ها

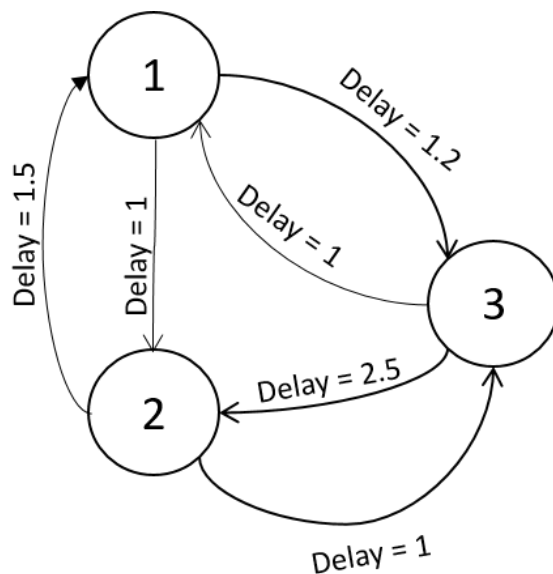
در خط اول ورودی تعداد گره‌های شبکه داده می‌شود. سپس در خطوط بعدی به ازای هر گره، در یک خط شناسه و زمان‌های انتظار و در خطوط بعدی مسیرهای ارتباطی به همراه تاخیر آن مسیر آورده می‌شود:

```
<size_of_network>
<nid> <timeout1> <timeout2> <timeout3>
<nid> <delay>
<nid> <delay>
```

...

گراف شبکه کامل است. نمونه‌ی ورودی:

```
3
1 10 6 5
2 1
3 1.2
2 14 5 5
1 1.5
3 1
3 18 6 6
1 1
2 2.5
```



در این مثال ۳ گره وجود دارد که مثلاً برای گره اول دوره‌ی اقدام برای انتخاب شدن به عنوان رهبری ۱۰ ثانیه، زمان انتظار برای دریافت پاسخ‌های POTENTIAL_LEADER_ACK، ۶ ثانیه و زمان انتظار برای دریافت پاسخ‌های V_PROPOSE_ACK، ۵ ثانیه است.

تمام پیام‌های دریافتی گره‌ها را در خروجی چاپ کنید. نمونه خروجی برای ورودی بالا:

```
node 2: 1 POTENTIAL_LEADER 1
node 3: 1 POTENTIAL_LEADER 1
node 1: 3 POTENTIAL_LEADER_ACK 0, -1
node 1: 2 POTENTIAL_LEADER_ACK 0, -1
node 2: 1 V_PROPOSE 1,3
node 3: 1 V_PROPOSE 1,3
node 1: 3 V_PROPOSE_ACK -1
node 3: 2 POTENTIAL_LEADER 2
node 1: 2 V_PROPOSE_ACK -1
node 1: 2 POTENTIAL_LEADER 2
```

```
node 2: 1 V_DECIDE 3
node 3: 1 V_DECIDE 3
* node 2: 3 POTENTIAL_LEADER_ACK 1, 3
```

* چاپ کردن این پیام الزامی نیست چون پس از decide به گره رسیده است.