



به نام خدا

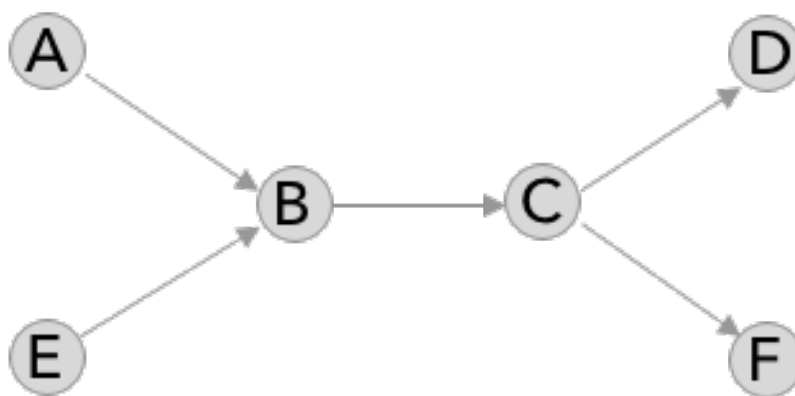
پروژه سوم درس سیستم‌های عامل

مهلت تحویل: ۲۴ آذر



هدف پروژه

در این پروژه قصد داریم یک سیستم حمل‌ونقل جاده‌ای را شبیه‌سازی کنیم. این سیستم حمل‌ونقل از مسیرهای مشخصی تشکیل شده است که هرکدام از این مسیرها به بخش‌های مختلفی تقسیم شده‌اند. این مسیرها می‌توانند در بخش‌هایی با یکدیگر اشتراک داشته باشند. برای مثال می‌توانید نقشه زیر را در نظر بگیرید. در این نقشه دو مسیر مشخص داریم: یکی شامل نقاط A، B، C و D. مسیر دیگر نیز شامل نقاط C، B، E و F است. همان‌طور که مشاهده می‌کنید این دو مسیر در بخش C-B با یکدیگر اشتراک دارند. از طرفی همه این مسیرها یک‌طرفه هستند.



در ابتدای اجرای شبیه‌سازی، در نقاط ابتدایی هر مسیر تعداد مشخصی خودرو تولید می‌شود. از طرفی هر بخش از مسیر گنجایش یک خودرو را در سرتاسر خود دارد و اگر یک خودرو در یک بخش وجود داشته باشد و خودروهای دیگر قصد ورود به آن بخش از مسیر را داشته باشند، باید صبر کنند تا آن بخش خالی شود. علت به‌وجود آمدن این قانون این بوده است که فرهنگ رانندگی در این سرزمین بسیار بد بوده است و تا شعاع چندصدمتری هر خودرو باید خالی از خودروهای دیگر باشد. در غیر این صورت تصادفات شدیدی رخ خواهد داد.

این‌که خودروها بتوانند این سطح از امنیت را تجربه کنند البته بی‌هزینه نبوده است. شرکت راه‌گستران امنیت‌محور به‌جای گرفتن عوارض، از خودروها درخواست می‌کند که در ازای عبور از هر بخش، محاسباتی را برای آن‌ها انجام دهند. این محاسبات میزان آلاینده‌گی هر خودرو در ازای عبور هر خودرو از آن جاده را مشخص می‌کند.

هدف این پروژه این است که تمامی خودروها به‌سلامت به مقصد خود برسند؛ به‌گونه‌ای که در هیچ لحظه‌ای دو خودرو در یک بخش از مسیر وجود نداشته باشد.

پیاده‌سازی

برای کنترل ترافیک هر بخش از مسیر از Monitor استفاده می‌کنیم. Monitor یک ساختار است که برای هم‌گام‌سازی در برنامه‌نویسی موازی به کار می‌رود. برای مطالعهٔ بیش‌تر در این‌باره می‌توانید به [این لینک](#) مراجعه کنید که شامل توضیحات کاملی است.

هم‌چنین خودروها متناظر با ریسمان‌ها هستند که برای ورود به هر بخش از مسیر باید از Monitor آن بخش استفاده کنند.

محاسباتی که خودروها باید در هر بخش از مسیر انجام دهند، طبق رابطهٔ زیر است. این رابطه میزان آلاینده‌های تولیدشده برحسب کیلوگرم را محاسبه می‌کند و در آن h عددی طبیعی است که متناسب با سختی هر بخش از مسیر است. (۱ نشان‌دهنده‌ی بیش‌ترین سختی است.) ضریب p نیز نشان‌دهنده‌ی میزان آلاینده‌ی خودرو است که در زمان ایجاد خودرو باید به‌صورت تصادفی بین اعداد ۱ تا ۱۰ تولید شود. (خودروی با ضریب کم‌تر آلاینده‌تر است.) دقت داشته باشید که علاوه بر آلودگی تولیدشده در هر بخش، مجموع آلودگی تولیدشده در کل سیستم را نیز باید ذخیره کنید.

$$\sum_{k=0}^{10^7} \left\lfloor \frac{k}{10^6 \cdot p \cdot h} \right\rfloor \quad (h \leq 4)$$

ورودی‌ها و خروجی‌ها

ورودی این شبیه‌ساز یک فایل است که از دو قسمت تشکیل شده است. در قسمت اول آن، در هر خط نام دو سر یک بخش از مسیر و عدد h که مشخص‌کنندهٔ سختی آن بخش از مسیر است مشخص شده است. در قسمت دوم فایل که با کاراکتر # جدا شده است، هر دو خط مربوط به یک مسیر از نقشه است. در یک خط نقاط تشکیل‌دهندهٔ این مسیر و در خط بعد تعداد خودروهایی که باید در ابتدای آن مسیر تولید شوند مشخص شده است. دقت کنید که این مسیرها ثابت هستند و مثلاً اگر دوراهی وجود داشته باشد، طبق مسیری که به هر خودرو نسبت داده شده است باید عمل شود. در ادامه می‌توانید فایل ورودی مربوط به نقشهٔ بالا را مشاهده کنید:

A - B - 4
B - C - 1
C - D - 2
E - B - 1
C - F - 3

A - B - C - D
3
E - B - C - F
4

همان‌طور که از نقشه استنباط می‌شود، ممکن است خودروها بتوانند مسیری مانند <A, B, C, F> را هم طی کنند ولی ما مسیرهای ثابت و مشخصی داریم که در فایل ورودی مشخص می‌شوند و خودروها فقط حق تردد در آن‌ها را دارند.

خروجی شما باید به ازای هر خودرو در هر مسیر، یک فایل با نام <car-number>-<path-number> باشد که در آن path-number نشان‌دهنده‌ی شماره‌ی مسیر به‌ترتیب فایل ورودی و car-number نشان‌دهنده‌ی شماره‌ی خودرو به ترتیب ایجاد است.

در هر خط از هر فایل باید طبق الگوی زیر، زمان ورود و خروج هر خودرو به هر بخش را ذخیره کنید:

<entrance-node-name>, <entrance-time>, <exit-node-name>, <exit-time>, <emission>, <total-emission>
--

entrance-node-name و exit-node-name دو سر یال‌هایی هستند که خودروها می‌پیمایند. entrance-time و exit-time هم زمان‌های ورود و خروج هستند. دقت کنید که زمان‌های ورود و خروج باید بر حسب میلی‌ثانیه و در فرمت [Epoch time](#) باشند. نهایتاً emission میزان آلاینده‌ی محاسبه‌شده در آن بخش از مسیر و total-emission نیز مجموع کل آلودگی تولیدشده در سیستم پس از خروج از آن بخش است.

سایر نکات

- در این پروژه به زبان C++ (می‌توانید از نسخه‌ی ۱۱ این زبان استفاده کنید) کد بزنید و کدهایتان با g++ قابل کامپایل کردن باشد.
- نکته‌هایی که در جلسه‌ی توجیهی یا فروم مطرح می‌شوند مهم هستند. پس توصیه می‌شود حتماً در این جلسه شرکت کنید.
- کد کسی را کپی نکنید. حتی یک تابع!
- برای تحویل پروژه ابتدا یک مخزن خصوصی در سایت [GitLab](#) ایجاد کنید و پروژه‌ی خود را در آن push کنید. (برای عبور از تحریم می‌توانید از سرویس رایگان [شکن](#) استفاده کنید.) سپس حساب UT_OS_TA را با دسترسی Maintainer به مخزن خود اضافه کنید. نهایتاً کافیست در محل بارگذاری پروژه در سایت CECM، آدرس مخزن و شناسه‌ی آخرین commit خود را ثبت کنید.
- می‌توانید هرگونه سؤال یا ابهام درباره‌ی این پروژه را به یکی از ایمیل‌های soltanloo@ut.ac.ir یا aligharizadeh1997@gmail.com ارسال کنید.