



هدف پروژه:

در این پروژه قصد داریم سنسورهای دمایی یک سیستم حساس را شبیه سازی کنیم. این سیستم دارای تعدادی سنسور دمایی در بخش های مختلف است. هدف ما این است که در دوره های زمانی مشخص دمای بخش های مختلف را از سنسورها بگیریم سپس آن ها را نرمالایز کنیم و در نهایت خروجی ها را در فایلی ذخیره کنیم. این کار را با استفاده از روش مانع (barrier) دو فازه انجام خواهیم داد. مانع یک نوع روش همگام سازی است که در آن یک نقطه به عنوان مانع در نظر گرفته می شود و تا زمانی که همه ی ریسمان ها به آن مانع نرسیده اند هیچکدام از ریسمان ها از آن مانع عبور نمی کنند.

پیاده سازی:

برای شبیه سازی این سیستم ابتدا ۱۰ عدد سنسور تعریف کنید. هر سنسور یک ریسمان جدا است. هر سنسور کارهای زیر را به ترتیب انجام می دهد:

۱. دمای بخش مربوط به خود را از ورودی می خواند.
۲. سپس منتظر می ماند که همه ی سنسورها دمای بخش خود را بخوانند.
۳. بعد از اینکه همه ی سنسورها ورودی های خود را خواندند، ورودی های خوانده شده را نرمالایز می کنند فرمول نرمالایز کردن به شکل زیر است:
$$\text{مقدار دمای خوانده شده توسط سنسور} = \frac{\text{مقدار نرمالایز شده}}{\text{ماکسیموم همه ی دماهای فعلی خوانده شده توسط همه ی سنسورها}}$$
۴. مقدار دمای نرمالایز شده را در فایل می نویسد.
۵. سپس منتظر می ماند که همه ی سنسور به این مرحله برسند.

۶. و در آخر 5 ثانیه منتظر می ماند تا بعد از آن دوباره مراحل را از اول شروع کند.

برای اینکه بتوانید نحوه عملکرد ریسمان هایی به سنسورها اختصاص داده شده است به درستی کنترل کنید باید این ریسمان ها را همگام سازی کنید. برای این کار از سمافورها استفاده کنید. و سعی کنید از کمترین تعداد سمافورها استفاده کنید.

توجه داشته باشید که در قسمت های مختلف کدتان ممکن است به `race condition` برخورد کنید برای هندل کردن آن نیز می توانید از سمافورها استفاده کنید. همچنین هر قدر که تعداد متغیرهای مشترک بین ریسمان ها بیشتر شود مشکلات بیشتری بوجود می آید مثلاً کنترل `race condition` ها سخت تر می شود. پس سعی کنید در راه حلتان متغیرهای مشترک بین ریسمان ها تا حد امکان کم باشد.

ورودی ها و خروجی ها:

- در فایل `input.txt` ورودی های مورد نیاز که دمای بخش های مختلف است آمده است به این شکل که ورودی ۱۰ سطر است و هر سطر برای یک سنسور است. در هر سطر ۲۰ عدد وجود دارد که به ترتیب دمای آن سنسور در زمان های متوالی است.
- برای نوشتن خروجی ها، هر سنسور فایل مربوط به خودش را دارد که می توانید اسم آن را با شماره سنسور یکسان بگذارید. (شماره سنسور کاربرد دیگری ندارد)

سایر نکات:

- در این پروژه به زبان `C++` (میتوانید از ورژن ۱۱ این زبان استفاده کنید) کد بزنید و کد هایتان با `g++` قابل کامپایل کردن باشد.

- نکته‌هایی که در جلسه توجیهی یا فروم مطرح می‌شوند مهم هستند. پس توصیه می‌شود در این جلسه شرکت کنید.
- نام فایلی که آپلود میکنید شماره دانشجویی‌تان باشد و به فرمت zip باشد.
- کد کسی را کپی نکنید. حتی یک تابع!