

مستند پروژه نهایی درس تحلیل و طراحی الگوریتمها

دانشجو

یاسین بلورچی

استاد

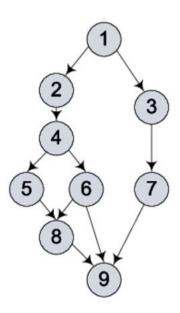
دکتر سعدون عزیزی

استاد حل تمرین

نوید رزمان

شرح کلی و هدف پروژه:

برنامه موازی در نظر بگیرید که شامل n کار (task) است جایی که هر کار دارای اندازه مشخصی میباشد و واحد آن – MI در Million Instruction است. از طرف دیگر، m ماشین مجازی ناهمگن (ماشین های مجازی با قدرت محاسباتی مختلف) را در نظر بگیرید که واحد هر یک از آنها Million Instruction Per Second – MIPS است. با در نظر گرفتن وابستگی بین کار ، میتوان آنها را به صورت یک گراف جهتدار بدون دور (Directed Acyclic Graph – DAG) مدل کرد که در آن گرهها بیانگر کارها و لینکها بیانگر وابستگی بین آنها است. نمایی از این مدل را میتوانید در شکل زیر مشاهده نمائید.



مسئله این است که چگونه این n کار را روی m ماشین مجازی زمانبندی کنیم به گونه ای که ماکسیمم زمان کاری m تا ماشین (makespan) بهینه شود؟ (به عبارت دیگر، کمینه کردن زمان اجرای پرکار ترین ماشین).

ورودی های این مسئله به ترتیب عبارت است از:

- یک ماتریس n x n که بیانگر گراف وابستگی است.
- یک آرایه با اندازه n که عنصر i ام آن بیانگر اندازه i امین کار است.
- یک آرایه با اندازه m که عنصر i ام آن بیانگر قدرت حاسباتی i امین ماشین مجازی است.

روش حل مسئله (الگوريتم):

برای حل این مسئله لازم است که مسئله را به چند قدم تقسیم کنیم و سپس مرحله به مرحله پیش برویم.

در مرحلهی نخست با گرفتن ماتریس وابستگی ها، کارها را به صورت توپولوژیکالی مرتب میکنیم اما برای اینکه جواب الگوریتم ما به جواب بهینه نزدیک تر باشد تمامی مرتب سازی های توپولوژیکالی گراف ورودی را بدست آورده و هر کدام را ذخیره میکنیم.

در مرحلهی بعدی الگوریتم به ازای هر کدام از مرتب سازی های توپولوژیکالی بدست آمده به صورت زیر در داخل ماشین های مجازی موجود قرار میدهیم:

- ابتدا زمان پایان کار تمامی ماشین ها را صفر کرده و زمان شروع و پایان تمامی کار ها را برابر با بینهایت میکنیم.
 - به ازای هر کار تمامی وابستگی هایش را پیدا میکنیم.
- به ازای تمامی وابستگی هایی که کار انتخاب شده دارد بیشینهی زمان پایان آنها را پیدا میکنیم؛ بنابراین حداقل زمان شروع هر کار بدست میآید.
- به ازای هر ماشین، اندازه ی کار را بر قدرت آن ماشین تقسیم میکنیم و با زمان اتمام کار آن ماشین جمع میکنیم تا زمان
 اتمام کار برای ماشین انتخاب شده بدست بیاید.
 - به ازای هر ماشین بیشینهی بین زمان اتمام کار قبلی ماشین و حداقل زمان شروع کار را بدست می آوریم.
- حال با داشتن مقادیر بالا برای هر کار ماشینی را انتخاب میکنیم که علاوه بر کمترین زمان شروع هر کار، کمترین زمان
 اتمام را به ما بدهد.
- پس از انتخاب بهترین ماشین، اطلاعات مربوط به زمان اتمام کار ماشین و زمان اتمام کار را در آرایه های مخصوص به
 آنها ذخیره میکنیم.
- در انتها پس از اتمام تمامی کارها، آرایهی مربوط به زمان اتمام کار هر ماشین را نگاه کرده و بیشترین مقدار را به عنوان خروجی (maximum makespan) نمایش میدهیم.

پیچیدگی زمانی الگوریتم:

پیچیدگی زمانی الگوریتم حریصانهی گفته شده در قسمت بالا یک چند جملهای از مرتبهی ۲ است زیرا برای هر کار تمامی ماشین های موجود را چک کرده و بهترین ماشین را با توجه به معیارهای گفته شده انتخاب میکند.

اما پیچیدگی زمانی کل الگوریتم پیاده سازی شده به صورت نمایی بوده و در بدترین حالت از مرتبهی n! است زیرای اگر هیچ وابستگی ای بین کار ها موجود نباشد تمام مرتب سازی های توپولوژیکالی ممکن برابر با جایگشت تمامی کار ها میشود.

در نهایت برای بدست آوردن جوابی مناسب و نسبتا بهینه میتوان الگوریتم حریصانهی گفته شده را تنها برای یک مرتب سازی توپولوژیکالی بدست آورد و پیچیدگی زمانی کل الگوریتم را به n^۲ کاهش داد.