

تحلیل پروژه سوم سیستم های عامل

برای جلوگیری از مسئله مطرح شده برای تحلیل می توانیم از ترند افزایش اولویت فرآیند ها (Aging) استفاده کنیم. ابتدا فرآیند ها با توجه به نوع الگوریتم انتخاب شده، در زیر صف های مربوط به اولویتشان مرتب میشوند و فرآیند هایی که آماده اجرا باشند اما منابعشان موجود نباشد به صف waiting منقل می شوند. صف waiting بعد از تعداد معینی واحد زمانی که نه خیلی زیاد (ممکن است منابع مورد نیاز فرآیند مدت طولانی ای باشد که موجوداند اما فرآیند از صف دیر خارج شود) و نه خیلی کم (به دلیل چک کردن زیاد صف waiting وقت زیادی از پردازنده گرفته میشود) است (مثلا ۴ سایکل)، چک می شود و فرآیند هایی که در آن لحظه منابع مورد نیازشان موجود است، اولویتشان یک واحد افزایش می یابد و وارد صف ready می شوند. همچنین باید متغیری جدا برای هر فرآیند تعریف کنیم که در ابتدا مقدار ۰ را دارد اما اگر فرآیندی برای مدت معینی یا بیشتر (مدتی که از زمان چک کردن صف waiting بیشتر است، برای مثال ۱۶ سایکل) در صف waiting بود، این متغیر یک شده و به ابتدای صف منقل می شود. در پردازنده ها چک می شود اگر فرآیندی دارای این متغیر با مقدار ۱ بود به تمام پردازنده های دیگر گفته می شود که بعد از انجام آخرین فرآیندشان فرآیند دیگری را اجرا نکنند تا منابع کافی موجود باشد و فرآیند بتواند اجرا شود. به این ترتیب بعد از مدت مشخصی همه فرآیند ها انجام می شوند و starvation رخ نمیدهد.

مزایا: این روش برخلاف روش های دیگر مانند قرار دادن فرآیند در ابتدای صف (از همان بار اول خروج از صف waiting) یا تعویض فرآیند با فرآیندی که در حال اجراست (running) کم کم اولویت فرآیند را افزایش می دهد و باعث starvation فرآیند هایی که اولویت بالاتری دارند یا دیرتر اجرا شدن آنها نمی شود.

معایب: ممکن است فرآیند مدت زیادی منتظر باشد تا به ابتدای صف برسد و منابع کافی برای آن موجود باشد. همچنین ممکن است بن بست (deadlock) ایجاد شود یعنی دو یا چند پردازنده همزمان اعلام کنند که بقیه پردازنده ها دیگر فرآیندی را اجرا نکنند. اما این مشکل به ندرت اتفاق میفتد و سیستم عامل آن را حل میکند.

حالت های مختلف:

Round-Robin: در این حالت از آنجایی که فرآیند ها کوانتوم کوانتوم انجام می شوند و در هر کوانتوم ممکن است فرآیند منابع مورد نیاز برای اجرا شدن را نداشته باشد و به صف waiting منتقل شود، امکان طولانی شدن اتمام اجرای فرآیند های بلند تر وجود دارد اما به starvation منتهی نمی شود.

FCFS: از آنجایی که فرآیندی که در صف waiting بوده قبل از فرآیند های دیگر صف ready ایجاد شده، الگوریتم ایجاد شده ترتیب first-come-first-served را به هم نمی زند و الگوریتم به خوبی عمل خواهد کرد.

SJF: در این الگوریتم اگرچه ممکن است فرآیندی که در ابتدای صف (با ابتدای زیرصف مربوط به اولویتش) قرار میگیرد کوتاه ترین زمان اجرا را نداشته باشد اما از starvation جلوگیری می شود.

HRRN: از آنجایی که این الگوریتم بر اساس مدت زمان اجرا و مدت زمان انتظار فرآیند در صف ready آنها را اولویت بندی می کند، الگوریتم پیشنهاد شده ممکن است جلوی اجرای فرآیند هایی را که شاید مدت زیادی در ready منتظر مانده اند بگیرد اما هیچ فرآیندی starve نمی شود.