



دانشگاه فردوسی مشهد

گروه مهندسی کامپیوتر

پروژه سوم درس سیستم عامل

عنوان

زمانبندی

نگارش

الهامه متقین

فاطمه صوفیان

استاد درس

دکتر محمد الله بخش

بهار 1400

1- مقدمه

در این پروژه برای زمانبندی سفارشات غذا در یک رستوران از الگوریتم های مختلف زمانبندی استفاده می کنیم. در طی این پروژه الگوریتم های اول کمترین مهلت، اول کمترین لختی و نرخ یکنواخت پیاده سازی شده اند.

2- زمانبندی

در یک سیستم کامپیوتری هزاران و شاید میلیون ها پردازش در یک لحظه در حال اجرا باشند. اگر پردازنده بخواهد همه کارها را به ترتیب انجام دهد، دیگر نخواهیم توانست به طور همزمان با برنامه های مختلف کار کنیم! زمان بندی پردازنده یا به طور دقیق تر، الگوریتم های زمانبندی در سیستم عامل برای مدیریت درست فرآیندها در هنگام پردازش به وجود آمده اند.

یک فرآیند (Process) اساساً یک برنامه ی در حال اجراست. منظور از برنامه در حال اجرا، کاری است که توسط زمان بند کار، انتخاب و وارد گردونه ی اجرا شده است ولی هنوز پایان نیافته و از سیستم خارج نشده است. این فرآیند الزاماً در حال حاضر CPU را در اختیار ندارد. الگوریتم های زمان بندی پردازنده یکی از مواردی است که باید در مورد مدیریت فرآیندها در سیستم عامل مورد بررسی قرار بگیرد.

هدف از زمانبندی پردازنده، تخصیص فرآیندها به پردازنده در طول زمان است که گونه ای که هدف های سیستم از قبیل زمان پاسخ، توان عملیاتی و کارایی پردازنده را برآورده سازد. زمان بندی پردازنده، اساس سیستم های عامل چند برنامه ای است.

2-1- الگوریتم های غیرانحصاری

در الگوریتم های غیرانحصاری زمانبندی پردازنده، ممکن است یک فرآیند که در حال اجراست توسط سیستم عامل متوقف شده و به حالت آماده منتقل شود. این عمل برای تخصیص پردازنده به یک فرآیند دیگر و یا انجام عملیات های I/O و وقفه صورت می گیرد. در این گونه الگوریتم ها، اجرای فرآیندها ممکن است به چند بخش تقسیم شده و در چند بار عملیات تخصیص پردازنده صورت بگیرد.

2-2- الگوریتم اول کمترین مهلت

الگوریتم ابتدا زودترین مهلت Earliest deadline first در این الگوریتم پردازش در سیستم عاملی ابتدا اجراء می شود که فرصتش از همه کمتر است یعنی نزدیکترین مهلت را دارد. این مهلت برای وقایع متناوب برابر زمان رخداد واقعه بعدی می باشد.

2-3- الگوریتم نرخ یکنواخت

الگوریتم نرخ یکنواخت (Rate monotonic) در این الگوریتم به هر پردازش در سیستم عامل اولویتی متناسب با فرکانس رخداد آن واقع نسبت داده می‌شود. مثلاً به پردازشی که هر 20 میلی ثانیه تکرار می‌شود، اولویت 50 و به پردازشی که هر 100 میلی ثانیه تکرار می‌شود، اولویت 10 داده می‌شود. این الگوریتم از نوع غیرانحصاری است.

2-4- الگوریتم اول کمترین لختی

تعریف مقدار لختی یک فرایند برابر است با حداکثر مقدار زمانی که فرایند می‌تواند در آن مدت منتظر گرفتن CPU بماند و اجرا نشود. مثلاً اگر یک پردازش در سیستم عامل به 200 میلی ثانیه وقت CPU احتیاج داشته باشد، و 250 میلی ثانیه نیز مهلت داشته باشد که کارش را تمام کند، زمان سستی او برابر $250 - 200 = 50$ میلی ثانیه می‌باشد. در این الگوریتم پردازشی ابتدا اجراء می‌گردد که کوچکترین زمان سستی را دارد.

2-5- پیاده سازی الگوریتم ها در پروژه

تابع اجرای زمانبندی و اختصاص CPU برای هر سه الگوریتم مشابه یکدیگر پیاده سازی شده است. آنچه میان نتیجه اجرای الگوریتم های مختلف، تفاوت ایجاد می‌کند، تابع `compareTo` در کلاس `Food` است. برای الگوریتم اول کمترین مهلت این تابع بر حسب مقدار مهلت فرایندها، برای الگوریتم نرخ یکنواخت این تابع بر حسب مقدار دره تناوب فرایندها و برای الگوریتم اول کمترین لختی بر حسب مقدار لختی فرایندها، پیاده سازی شده است. همچنین سایر شروط گفته شده در صورت پروژ نیز در پیاده سازی این تابع در نظر گرفته شده است، از جمله اینکه فرایندها با اولویت یکسان، اگر فرایندی در میان اجرا باشد، اولویت بالا تری می‌گیرد.

نمونه ای از پیاده سازی این تابع به صورت زیر است.

```
//least laxity
@Override
public int compareTo(Food o) {
    if(o.isJustArrive() && !this.isJustArrive())
        return -1;
    else if(this.isJustArrive() && !o.isJustArrive())
        return 1;
    else if(o.isJustArrive() && this.isJustArrive || o.getInterval() ==
this.getInterval()) {
        if (o.isCooking() && !this.isCooking())
            return -1;
        else if (!o.isCooking() && this.isCooking())
            return 1;
        else
            return this.getTime() - o.getTime();
    }
    return (this.getDeadline() - this.getTime()) - (o.getDeadline() -
o.getTime());
}
```

در تابع اصلی نیز فرایندها بر اساس اولویت تعییت شده در صف قرار می گیرند و به ترتیب اجرا می شوند. در صورت رسیدن یک فرایند جدید و یا بروز هرگونه خطا، وقفه ای رخ می دهد و بر اساس اولویت های جدید ادامه داده می شود.

*** پروژه روی گیت هاب در آدرس <https://github.com/elahemotaghin/Scheduling.git> قرار گرفته است.