



دانشگاه شاهرود

به نام خداوند بخشاینده مهربان

درس: سیستم عامل

استاد: دکتر وحید رنجبر

ترم: بهار ۱۴۰۰

دستیاران استاد: علیرضا مهدی‌نسب، محمدرضا اسماعیلی

« پروژه سوم »

مدیریت حافظه

مقدمه

در یک سیستم تک‌برنامه‌ای، حافظه به دو بخش تقسیم می‌شود:

۱. یک بخش برای سیستم عامل (ناظر، مقیم، هسته)

۲. یک بخش برای برنامه در حال اجرای کاربر

در یک سیستم چندبرنامه‌ای، بخش کاربر باید تقسیم‌بندی شود تا چندین برنامه را هم‌زمان در خود جای دهد.

وظیفه تقسیم‌بندی حافظه به زیربخش‌ها به صورت پویا، توسط سیستم عامل صورت می‌گیرد که به این عمل مدیریت حافظه می‌گویند.

برای مدیریت حافظه، لازم است حافظه را بخش‌بندی کنیم. یکی از این روش‌ها، بخش‌بندی پویا است.

در این بخش‌بندی:

- بخش‌ها دارای طول و تعداد متغیر هستند.
- حافظه تخصیص‌یافته به هر فرایند، دقیقاً برابر با میزان نیاز آن فرایند است.

- پس از تخصیص و آزادسازی‌های مکرر فرایندها، حفره‌هایی در حافظه پدید می‌آیند که پراکندگی خارجی نامیده می‌شوند.
- فشرده سازی: معمولا برای مقابله با پراکندگی خارجی، سیستم عامل فرایندها را انتقال می‌دهد تا در کنار یک‌دیگر قرار گیرند و تمام حافظه آزاد در یک بلوک قرار گیرد.

شرح پروژه

برای جای‌گذاری فرایندها در بخش‌بندی پویا، الگوریتم‌های متعددی وجود دارد که دو مورد آن‌ها عبارت‌اند از:

۱. اولین برازش
۲. بهترین برازش

«هدف ما در این پروژه، پیاده‌سازی این دو الگوریتم جای‌گذاری می‌باشد»

الگوریتم اول: اولین برازش

- حافظه را از ابتدا مرور می‌کند و اولین بلوک آزاد و کافی را به فرایند اختصاص می‌دهد.
- ساده‌ترین، سریع‌ترین و بهترین الگوریتم است.

الگوریتم دوم: بهترین برازش

- کوچک‌ترین بلوکی را که فرایند در آن جای می‌شود، انتخاب می‌کند.
- بیشترین هزینه اجرا و بدترین کارایی را دارد.

ورودی

این پروژه برای اجرا به ۲ ورودی نیاز دارد.

۱. سایز RAM جهت اجرای فرایندها (بر حسب مگابایت)

این ورودی، از کاربر گرفته شده و به عنوان سایز RAM (بخش مربوط به اجرای فرایندها) شناخته می‌شود.

۲. فایل txt که حاوی مجموعه‌ای از تخصیص‌ها و آزادسازی‌های فرایندهاست.

این ورودی، یک فایل متنی با پسوند txt می‌باشد. در این فایل اطلاعات مربوط به تخصیص‌ها و آزادسازی‌های فرایندها نوشته شده است.

نمونه فایل ورودی

```
REQUEST A 23  
REQUEST B 30  
RELEASE A  
REQUEST C 17
```

توضیحات

در این نمونه ورودی، ابتدا فرایند A برای اجرا ۲۳ مگابایت از فضای RAM را درخواست می‌کند. پس از آن، فرایند B برای اجرا ۳۰ مگابایت از فضای RAM را درخواست می‌کند که می‌بایست این فضاهای درخواستی به فرایندها اختصاص داده شود. سپس؛ اجرای فرایند A به پایان رسیده و فضای تخصیص داده شده به فرایند A، بازپس گرفته خواهد شد. و در آخر؛ فرایند C برای اجرا ۱۷ مگابایت از فضای RAM را درخواست می‌کند.

خروجی

در نهایت، پس از پایان تخصیص‌ها و آزادسازی‌ها، دو قسمت می‌بایست در خروجی چاپ شود.

۱ قسمت مربوط به خروجی الگوریتم اولین برازش (First Fit)

۲ قسمت مربوط به خروجی الگوریتم بهترین برازش (Best Fit)

و اما اطلاعاتی که در هر قسمت می‌بایست چاپ شود:

الف) تعداد دفعاتی که عمل فشرده‌سازی انجام شده است.

ب) وضعیت RAM در پایان کار

به عنوان مثال؛ برای نمونه ورودی ذکر شده در بخش قبل، خروجی زیر می‌بایست چاپ شود: (فرض کنید سائز RAM هفتاد مگابایت می‌باشد).

```
*** First Fit ***  
Compression No: 0  
RAM Status: C, 6, B, 17  
  
*** Best Fit ***  
Compression No: 0  
ARAM Status: 23, B, C
```

توضیحات

نحوه استفاده از RAM و جای‌گذاری فرایندها در آن، دقیقا مشابه با توضیحات اسلاید ۱۷ و ۱۸ از فصل هفتم می‌باشد.

مثلا برای زمانیکه خروجی پروژه Status RAM: C, 6, B, 17 باشد، وضعیت RAM بدین صورت بوده است:



نکاتی چند

- ✓ فرض را بر آن بگیرید که امکان پاسخ‌گویی به تمام Requestها وجود دارد. (تمام فرایندهایی که برای اجرا Request می‌دهند، فضای کافی برای اجرای آن‌ها وجود دارد).
- ✓ در صورتیکه فضای کافی برای اجرای یک فرایند جدید وجود نداشت، **می‌بایست عمل فشردن سازی انجام شود** و سپس فضا به آن فرایند اختصاص داده شود.
- ✓ نحوه شبیه‌سازی RAM کاملا اختیاری بوده و بستگی به سلیقه و نوع طراحی دانشجو دارد.
- ✓ هیچ محدودیتی شامل نام فرایندها نمی‌شود؛ جز آنکه در نام فرایند space وجود ندارد.
- ✓ مقدار فضایی که هر فرایند برای اجرا Request می‌دهد، عددی صحیح است و نمی‌تواند به صورت اعشاری باشد.

✓ سائز RAM (که به عنوان ورودی از کاربر گرفته می‌شود)، عددی صحیح بین ۱ و ۱۰۲۴ می‌باشد.

توجه

- برای انجام پروژه می‌توانید از زبان Java یا Python استفاده کنید.
- گزارشی از فعالیت خود تهیه کرده و توابع مورد استفاده را مختصراً در قالب PDF توضیح دهید.
- بخشی از نمره پروژه، مربوط به ارائه و دفاع از آن می‌باشد. زمان دقیق ارائه متعاقباً اعلام می‌گردد.
- مهلت آپلود پروژه در سامیاد: ۱۸ تیرماه ۱۴۰۰

نمره اضافی

الف) پیاده‌سازی الگوریتم در پی برازش

الگوریتم سوم: در پی برازش

- حافظه را از محل آخرین جاییابی به بعد مرور کرده و اولین بلوک با اندازه کافی را انتخاب می‌کند.
- بلوک‌های بزرگ حافظه، سریعاً به بلوک‌های کوچک‌تر تقسیم می‌شوند.

ب) مقایسه الگوریتم‌ها با یکدیگر

برای هر سه الگوریتم نام‌برده (اولین برازش، بهترین برازش، در پی برازش)، علاوه بر اطلاعات خواسته‌شده، پارامترهای زیر نیز می‌بایست محاسبه شوند:

- ❖ تعداد شکاف‌های خارجی
- ❖ میانگین اندازه شکاف‌های خارجی

ج) نوشتن خروجی در فایل txt

تمام قسمت‌های مربوط به خروجی می‌بایست در یک فایل متنی نوشته شوند.

توضیحات

➤ یک فایل متنی با نام output.txt ایجاد و تمام قسمت‌های مربوط به خروجی الگوریتم‌ها در این فایل نوشته شود.

➤ نحوه نوشتن خروجی در این فایل، کاملاً اختیاری است.

شاد و سلامت باشید.