



## تمرین شماره ۳

-۱

الف) تفاوت بین بارگیری پویا<sup>۱</sup> و پیوند پویا<sup>۲</sup> چیست؟ شرح دهید که هر کدام در چه مرحله‌ای از اجرای برنامه اعمال می‌شوند و چه اهدافی دارند.

ب) در هریک از قطعه کدهای a و b چه عملیاتی انجام شده و مربوط به کدام یک از موارد مطرح شده است؟ مزیت‌های هر کدام را بیان کنید.

(a)

```
Void * hndl dlopen("libname.so", RTLD_NOW);
```

```
Void * lib_func = dlsym(hndl, "func_name");
```

(b)

```
g++ sampleCode.c -ldynamicLibraryName
```

ج) در یک سیستم با چندین نسخه از یک کتابخانه پویا (مثل libmath.so.1 و libmath.so.2)، اگر برنامه‌ای با استفاده از پیوند پویا به libmath.so کامپایل شود، چه مشکلاتی ممکن است در زمان اجرا به وجود آید؟ چگونه می‌توان این مشکلات را مدیریت کرد؟

۲- اگر در یک مدل صفحه‌بندی<sup>۳</sup>، بیشترین اندازه صفحه برابر با چهار کیلوبایت باشد:

الف) بدترین حالت<sup>۴</sup> مربوط به تکه‌تکه شدن<sup>۵</sup> چه زمانی رخ می‌دهد؟ و هدررفت حدودی حافظه در این حالت چقدر است؟ (فرض کنید که کوچک‌ترین واحد برای ذخیره‌سازی یک بایت باشد).

ب) حالت میانگین<sup>۶</sup> را محاسبه نمایید.

ج) دستور ((getconf PAGESIZE)) را در رایانه خود اجرا کنید. عدد به دست آمده اندازه صفحه<sup>۷</sup> را نشان می‌دهد. آیا امکان تغییر آن فراهم است؟

د) چرا در گذر زمان، سیستم‌ها اندازه صفحه خود را افزایش داده‌اند؟ مزایا و معایب بزرگ یا کوچک بودن این مقدار چیست؟ آیا ارتباطی بین درجه چند برنامگی<sup>۸</sup> و اندازه صفحه وجود دارد؟

۳- پردازنده‌های زیر را در یک سیستم با مدیریت حافظه مبتنی بر ثابت<sup>۹</sup>‌های پایه<sup>۱۰</sup> و حد<sup>۱۱</sup> در نظر بگیرید.

پردازنده	پایه	حد
A	۲۰۰	۱۰
B	۳۰۰۰	۲۰
C	۷۰۰	۵۰

فرض کنید آدرس‌دهی در سطح بایت انجام شده و آدرس‌های منطقی پردازنده‌ها با استفاده از ثابت‌های پایه و حد به آدرس‌های فیزیکی نگاشت می‌شوند.

<sup>1</sup> Dynamic Loading

<sup>2</sup> Dynamic Linking

<sup>3</sup> Paging

<sup>4</sup> Worst-Case

<sup>5</sup> Fragmentation

<sup>6</sup> Average-Case

<sup>7</sup> Page Size

<sup>8</sup> Multi-Programming level

<sup>9</sup> Register

<sup>10</sup> Base

<sup>11</sup> Limit

الف) پس از یک تعویض بافتار<sup>۱۲</sup>، پردازش B شروع به اجرا خواهد کرد. برای آدرس‌های منطقی زیر که توسط پردازش B تولید شده‌اند، چه اتفاقی می‌افتد؟

۵ -

۱۵ -

۲۰ -

۲۵ -

ب) اگر ترتیب دسترسی به حافظه مشابه حالت‌های زیر باشد (از راست به چپ)، چه روش زمان‌بندی به‌ازای هر حالت در سامانه استفاده شده است؟  
\* آدرس‌ها، آدرس‌های فیزیکی هستند که باید به پردازش مربوطه نگاشت شوند.

- ۲۰۸، ۲۰۷، ۲۰۵، ۲۰۱، ۳۰۱۲، ۳۰۱۰، ۳۰۰۹، ۳۰۰۱، ۷۲۵، ۷۳۰، ۷۳۳، ۷۰۱

- ۲۰۱، ۲۰۳، ۲۰۶، ۲۰۹، ۳۰۱۰، ۳۰۱۲، ۳۰۰۱، ۳۰۱۸، ۷۰۰، ۷۴۵، ۷۴۰، ۷۰۸

ج) آیا تنظیم ثبات‌های پایه و حد توسط سخت‌افزار اختصاصی تنظیم می‌شوند یا توسط سیستم‌عامل؟

د) اگر در بخش (الف)، پردازش B در حالت کرنل اجرا شود و سیستم در حالت کرنل از رجیسترهای پایه و حد استفاده نکند (آدرس‌ها به‌عنوان آدرس‌های فیزیکی در نظر گرفته شوند)، پاسخ بخش (الف) تغییر می‌کند؟ پاسخ خود را توجیه کنید.

۴- یک سامانه صفحه‌بندی را در نظر بگیرید که زمان دسترسی به حافظه جانبی (دیسک) در آن برابر با پنج میلی‌ثانیه است. جدول صفحه‌ها در حافظه اصلی و با زمان دسترسی ۱۰۰ نانوثانیه قرار دارد. همچنین، برای بهبود زمان ترجمه آدرس، از حافظه انجمنی با زمان دسترسی ۲۰ نانوثانیه استفاده شده است. با فرض اینکه ۹۵٪ درخواست‌های ترجمه آدرس از حافظه انجمنی پاسخ داده می‌شوند و در صورت عدم وجود ترجمه در حافظه انجمنی، جدول صفحه‌ها در حافظه اصلی بررسی می‌شود. همچنین، ۱٪ از کل درخواست‌های دسترسی به حافظه، منجر به خطای صفحه می‌شوند که نیازمند دسترسی به حافظه جانبی خواهند بود. مقدار موثر زمان دسترسی برای اجرای دستور ((move 1024, 16)) چقدر است؟

۵- فضای آدرس منطقی شامل ۲۵۶ صفحه با اندازه صفحه چهار کیلوبایت را در نظر بگیرید که بر روی حافظه فیزیکی با ۶۴ قاب<sup>۱۳</sup> با همان اندازه صفحه نگاشت شده است. فرض کنید آدرس‌دهی در سطح بایت انجام می‌شود و سیستم از یک جدول صفحه<sup>۱۴</sup> تک‌سطحی استفاده می‌کند که در آن برخی صفحات ممکن است به دلیل محدودیت‌های سیستم به حافظه دیسک منتقل شوند. مشخص کنید در چنین سیستمی:

الف) چند بیت برای آدرس منطقی مورد نیاز است؟

ب) چند بیت برای آدرس فیزیکی مورد نیاز است؟

ج) اگر شماره صفحه در آدرس منطقی به‌صورت جداگانه کدگذاری شود، چند بیت برای آن نیاز است؟

د) اگر جدول صفحه در حافظه فیزیکی ذخیره شود و هر ورودی جدول صفحه چهار بایت باشد، حداقل اندازه جدول صفحه برای این سیستم چند کیلوبایت است؟

۶-

الف) تفاوت اصلی بین یک خطای جزئی صفحه<sup>۱۵</sup> و یک خطای کلی صفحه<sup>۱۶</sup> در لینوکس چیست؟ این خطا چه زمانی رخ می‌دهد؟  
ب) با اجرای دستور زیر در سیستم‌عامل لینوکس، کاربرد آن را شرح دهید.

<sup>12</sup> Context Switch

<sup>13</sup> Frame

<sup>14</sup> Page Table

<sup>15</sup> Minor Page Fault

<sup>16</sup> Major Page Fault

→ ps -eo pid,min\_flt,maj\_flt

ج) چه پردازنده‌هایی عموماً دارای شماره پردازنده<sup>۱۷</sup> کوچک‌تر هستند؟ چرا این پردازنده‌ها معمولاً خطای صفحه کمتری خواهند داشت؟

۷- یک سیستم فرضی با پنج قاب حافظه فیزیکی و ۱۰ صفحه حافظه مجازی داریم. دسترسی به صفحات مجازی به ترتیب رشته زیر انجام می‌شود (از چپ به راست).

“ADFBACGHDA BEHFBIDJ”

باتوجه به اطلاعات ارائه شده درباره این سیستم، الگوریتم‌های مختلف تخصیص حافظه را برای پیاده‌سازی در این سیستم شبیه‌سازی کنید؛ سپس به موارد زیر پاسخ دهید. همچنین، توجه داشته باشید که هر حرف، نماینده (شناسه) یک صفحه است.

الف) برای هر یک از الگوریتم‌های FIFO، LRU و Optimal، پس از اجرای الگوریتم، کدام صفحات در حافظه فیزیکی باقی خواهند ماند؟

ب) تعداد خطاهای صفحه<sup>۱۸</sup> را برای هر الگوریتم محاسبه نمایید.

- در صورتی که در فرایند تخصیص حافظه، صفحات مختلف با اولویت یکسان برای حذف یا اضافه شدن وجود داشته باشند، صفحه‌ای که کمترین شماره را داشته باشد به حافظه تخصیص داده می‌شود.
- در صورتی که در روند تخصیص حافظه فیزیکی برابری رخ دهد، برابری را به ترتیب الفبایی حل نمایید.

۸- فرض کنید یک رشته ارجاع صفحه برای یک پردازنده با m قاب دارید (که در ابتدا تمامی این m قاب خالی هستند). طول این رشته ارجاع صفحه برابر با p است و n شماره صفحه مجزا در آن رخ می‌دهند. به سوالاتی که در ادامه مطرح خواهند شد، برای هر الگوریتم جایگزینی صفحه پاسخ دهید.

الف) حد پایین برای تعداد خطاهای صفحه چقدر است؟

ب) حد بالا برای تعداد خطاهای صفحه چقدر است؟

۹- رشته ارجاع صفحه زیر را در نظر بگیرید (از چپ به راست).

- ۱، ۰، ۳، ۲، ۴، ۵، ۰، ۱، ۷، ۷، ۴، ۳، ۵، ۲، ۱، ۳، ۲، ۷

با فرض صفحه‌بندی برحسب تقاضا با سه قاب، چه تعداد خطای صفحه برای هر یک از الگوریتم‌های جایگزینی زیر رخ خواهد داد؟

- جایگزینی LRU

- جایگزینی FIFO

- جایگزینی Optimal

<sup>۱۷</sup> pid

<sup>۱۸</sup> Page Fault