



دانشگاه صنعتی اصفهان  
دانشکده مهندسی برق و کامپیوتر

## تکلیف دوم درس سیستم عامل

نیمسال تحصیلی پاییز-۱۴۰۰  
مدرس: دکتر محمدرضا حیدرپور

دستیاران آموزشی:  
مجید فرهادی - دانیال مهرآیین - محمد نعیمی

نحوه تحویل:

پاسخ های خود به همراه برنامه های نوشته شده را در قالب یک فایل PDF در سامانه بارگذاری کنید. استفاده از  $\text{\LaTeX}$  اختیاری بوده و ۴۰ نمره اضافی دربرخواهد داشت. می توانید برای آشنایی با دستورات  $\text{\LaTeX}$  از این قالب آماده شده استفاده کنید.

۱. به سوالات زیر به صورت کوتاه پاسخ دهید. (۱۱۰ نمره)

(آ) تفاوت External Fragmentation و Internal Fragmentation در چیست؟ توضیح دهید.

(ب) به چه دلیل از چند سطح مختلف حافظه استفاده می شود؟ مزیت این کار چیست؟

(ج) چرا در حافظه اختصاص داده شده به هر پروسس، Stack و Heap را به صورت پشت سر هم قرار نمی دهند؟ با قراردادن آنها به صورت پشت سر هم چه مشکلاتی ممکن است رخ دهد؟

(د) در روش Segmentation برای مجازی سازی حافظه، قابلیت اشتراک کد بین دو پروسس چگونه فراهم می شود؟ لازم است سخت افزار چه قابلیت هایی داشته باشد؟

(ه) بزرگی یا کوچکی اندازه Paging هر کدام چه مزایا و معایبی دارند؟

(و) چرا رسیدگی به Page Fault توسط سخت افزار انجام نمی شود؟

(ز) در روش Paging برای مجازی سازی حافظه، کاربرد ثابت CR3 چیست؟

(ح) در فرایند دسترسی به حافظه، هر کدام از موارد زیر در چه شرایطی رخ می دهند؟ (در صورت ممکن نبودن هر مورد، دلیل آن را بیان کنید).

- TLB Miss & No Page Fault
- TLB Miss & Page Fault
- TLB Hit & No Page Fault
- TLB Hit & Page Fault

۲. با توجه به سیاست های جایگزینی صفحات، جداول زیر را کامل کنید. اندازه Cache برابر 4 است. (۶۰ نمره)

LRU			LFU			Optimal		
Access	Hit	State (after)	Access	Hit	State (after)	Access	Hit	State (after)
2			2			2		
3			3			3		
1			1			1		
5			5			5		
6			6			6		
2			2			2		
1			1			1		
5			5			5		
3			3			3		
2			2			2		
6			6			6		
5			5			5		
4			4			4		
3			3			3		
2			2			2		

۳. سیستمی دارای یک سطح TLB، یک سطح Cache و دو سطح Page Table است و از روش Paging برای مجازی سازی حافظه استفاده می کند. پارامترهای مختلف این سیستم در جدول زیر آمده است:

Parameter	Description	Value
$P_{\text{CacheMiss}}$	Cache Miss Probability	0.01
$P_{\text{TLBMiss}}$	TLB Miss Probability	0.01
$P_{\text{PageFault}}$	Page Fault Probability	0.00002
$T_{\text{Cache}}$	Cache Access Time	1 ns
$T_{\text{TLB}}$	TLB Access Time	1 ns
$T_{\text{DRAM}}$	Main Memory Access Time	100 ns
$T_{\text{Disk}}$	Hard Disk Access Time	10 ms

در هنگام وقوع TLB Miss، عمل به روزرسانی آن به طور خودکار توسط سخت افزار انجام می شود. Page Table ها در حافظه اصلی نگهداری می شوند و از حافظه Cache برای نگهداری آن ها استفاده نمی شود؛ بنابراین در صورت نبود آدرس یک صفحه در TLB، نیاز به دو بار دسترسی به حافظه اصلی داریم. همچنین فرض کنید Dirty Bit تمامی صفحات صفر است و نیازی به بازنویسی آن ها در حافظه جانبی نیست.

با توجه به فرضیات فوق و با کمک از اسلاید شماره ۱۱۲ فصل دوم، میانگین زمان دسترسی به صفحات حافظه را محاسبه کنید. (برای راحتی محاسبات، از  $T_{Cache}$  و  $T_{TLB}$  در مقابل  $T_{DRAM}$  و از  $T_{Cache}$ ،  $T_{TLB}$  و  $T_{DRAM}$  در مقابل  $T_{Disk}$  صرف نظر کنید.) (۸۰ نمره)

۴. فرض کنید که فضای آدرس دهی مجازی ۱۲ بیتی و اندازه هر صفحه ۳۲ بایت است. همچنین فرض کنید که از روش Paged Segmentation (Hybrid) استفاده می کنیم. ۲ بیت پرارزش آدرس مجازی، Segment Number را مشخص می کنند. هر PTE ۱ بایت است. هر PTE را که در نظر بگیریم، پرارزش ترین بیت آن به عنوان Valid Bit در نظر گرفته می شود و سایر بیت ها PFN را مشخص می کنند. اطلاعات Segment ها (در ستون Base، آدرس صفحه ابتدای Segment آمده است.) و محتوای برخی صفحات فیزیکی به شرح زیر است (اعداد در مبنای ۱۶ هستند و آدرس هر صفحه از حاصل ضرب PFN آن در اندازه آن (۳۲ بایت) به دست می آید):

Segment	SN	Base	Size
Code	01	64	6
Heap	10	1024	10
Stack	11	640	10

Page with PFN = 0:

8b 66 10 45 af e7 3c cb b1 a7 31 d2 2a e8 b2 65

f8 a0 cb 18 58 66 24 0e ae 2e c9 74 c3 1c ab a6

Page with PFN = 2:

6d 5d c3 8a f7 0d a5 f3 a9 62 06 2b 5d b9 2f 4e

ce ba b5 a6 a0 ec bd 69 98 43 73 cf 08 1c f1 35

Page with PFN = 20:

f0 ba 8e ea 20 20 19 f5 7f fa fc 5a 34 a8 89 c9

bf 5f 5e 76 32 4b cb 0d ec 06 bb 34 a7 93 81 89

Page with PFN = 32:

2d ce 12 17 3e 10 8c 42 53 8e a1 60 86 fc 7e 26

e9 b9 5e 1b ad 6b d5 8f 85 95 c6 7b f5 db f0 5a

Page with PFN = 67:

b9 9a 05 73 bd 4b 2d 29 5c 87 e5 7f 0f 3a 9a 43

c0 1f a3 61 44 52 51 e1 e6 05 bc 8a e0 40 3b 59

Page with PFN = 64:

be 91 ec 48 fb d6 d8 4b 8a 42 75 cc a2 85 4f f4

ce 9e f8 b9 07 a8 5e dd 59 de a3 a1 6a ec 7c 6f

(آ) فرض کنید قرار است به آدرس مجازی 0x45d دسترسی پیدا کنیم: (۴۰ نمره)

- برای ترجمه آدرس مجازی به آدرس فیزیکی، به چه صفحاتی رجوع می‌شود؟ توضیح دهید.
- این آدرس مجازی به این پروسس تخصیص داده شده است؟ اگر «خیر»، چرا؟ اگر «بله»، بایت مورد نظر در چه صفحه‌ای قرار دارد و مقدار آن چیست؟ توضیح دهید.

(ب) قسمت قبل را برای آدرس مجازی 0xc85 تکرار کنید. (۴۰ نمره)

۵. در روش Paged Segmentation (Hybrid)، فرض کنید اندازه هر Segment دارای توضیح یکنواخت بین صفر تا 4 گیگابایت باشد. همچنین اندازه PTE هر Page برابر 4 بایت باشد. اندازه Paging را به گونه‌ای تعیین کنید که سربار حافظه کمترین مقدار خود را داشته باشد. سربار حافظه را برابر مجموع Internal Fragmentation ها و فضای اشغالی توسط Page Table ها در نظر بگیرید. (۷۰ نمره)

۶. (اختیاری) در این سوال قصد داریم سیاست‌های مختلف Paging را بررسی نماییم. ابتدا فایل اجرایی مربوط به شبیه سازی را از اینجا [دانلود کرده](#) و سپس با توجه به توضیحات داده شده در این فایل، به سوال ۲ موجود در انتهای آن پاسخ دهید. همچنین می‌توانید برای آشنایی با طرز استفاده از این شبیه ساز به راهنمای شبیه ساز مراجعه کنید. تصویر مربوط به نتیجه شبیه سازی را به همراه تحلیل خود از نتیجه آن را در پاسخ نامه خود قرار دهید. (۳۰ نمره)

۷. (غیرتحویلی) کد زیر را در نظر بگیرید:

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
int main()
{
    char buffer1[2] = "A";
    char buffer2[2] = "B";
    \\ Do not change the following section
    strcpy(buffer1, "AAA");
    printf("%s\n%s\n", buffer1, buffer2);
    return 0;
}
```

بدون تغییر کدهای بعد از کامنت، متغیرهای برنامه را به فضای Heap منتقل کنید. آیا مشکل Buffer Overflow برطرف شده است؟ علت را شرح دهید.

---

۸. (غیرتحویلی) قطعه کد زیر برای ضرب ماتریسی را در نظر بگیرید:

```
float a[1024][1024], b[1024][1024], c[1024][1024];
multiply()
{
    unsigned i, j, k;
    for (i = 0; i < 1024; i++)
        for (j = 0; j < 1024; j++)
            for (k = 0; k < 1024; k++)
                c[i][j] += a[i,k] * b[k,j];
}
```

فرض کنید کد باینری تابع و Stack برنامه، هر کدام در یک Page قرار می‌گیرند و هر عدد اعشاری 4 بایت از حافظه را اشغال می‌کند. اگر اندازه صفحات 4 کیلوبایت باشد، TLB دارای 8 سطر و سیاست جایگزینی سطرهای آن LRU باشد، تعداد TLB Miss ها را محاسبه نمایید. (TLB در ابتدا خالی است).

۹. (غیر تحویلی) سیستمی که از روش Paging برای مجازی سازی حافظه خود استفاده می کند، میزان بهره وری سخت افزارهایش به شرح زیر است:

- CPU: 20%
- Disk: 97.7%
- Other I/O Devices: 5%

در رابطه با تاثیر هر یک از موارد زیر برای افزایش بهره وری CPU بحث کنید.

- نصب یک CPU سریع تر
- نصب یک Disk بزرگ تر
- افزایش درجه Multiprogramming
- کاهش درجه Multiprogramming
- نصب یک Main Memory بزرگ تر
- نصب یک Disk سریع تر یا استفاده همزمان از چند Disk
- استفاده از روش های Prepaging برای انتقال پیش از موعد صفحات
- افزایش اندازه Paging

موفق باشید.