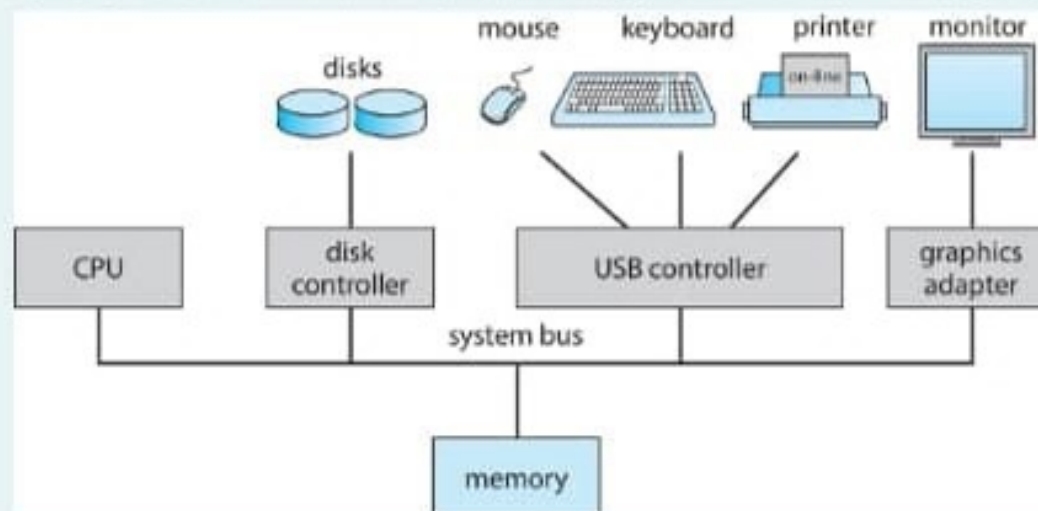


در شکل زیر، CPU و یک I/O Device می‌توانند همزمان از حافظه (memory) بخوانند یا در آن بنویسند. (۱.۵ دقیقه)



یک گزینه را انتخاب کنید:

☐ صحیح

☒ غلط

پاسخ درست گزینه «غلط» است.

سوال 1

درست

نمره 1.00 از 1.00

علامت زدن

سوال

سوال 2

درست

نمره 2.00 از 2.00

۳ علامت زدن
سوال

وقتی که فرایندی در اثر فراخوانی سیستمی fork ایجاد می‌شود، کدام یک از موارد زیر توسط پردازنده فرزنده به ارث برده نمی‌شود؟ (۳ دقیقه)

a . Process address space ☐

b . Open file descriptors ☐

c . Process ID ☒

d . User ID ☐

e . Signal handlers ☐



پاسخ شما صحیح می‌باشد

پاسخ درست »

Process ID « است.

سوال 3

درست

نمره 1.00 از 1.00

۳ علامت زدن

سوال

یکی از معایب روش نگاشت many-to-many از نخ‌های سطح کاربر (user space thread) به نخ‌های سطح هسته (kernel space thread) این است که توسعه دهندگان نرم‌افزار عملاً نمی‌توانند به تعداد مورد نیاز نخ (thread) تعریف کنند چرا که نخ‌های سطح هسته نمی‌توانند به شکل موازی بر روی سیستم‌های چند هسته‌ای (۲ یا بیشتر هسته CPU) اجرا شوند. (۱.۵ دقیقه)

یک گزینه را انتخاب کنید:

☐ صحیح

☒ غلط

پاسخ درست گزینه «غلط» است.

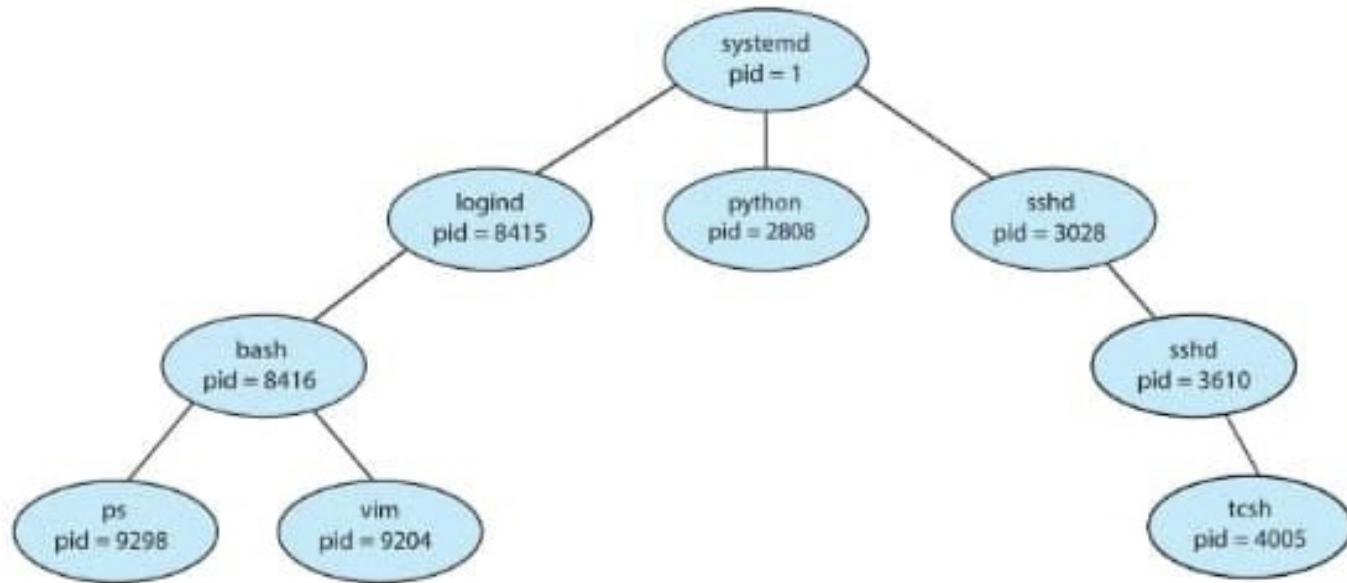
سوال 4

درست

نمره 2.00 از 2.00

۳ علامت زدن
سوال

فرض کنید که سیستم عامل لینوکس اجازه ادامه اجرای پردازش فرزند را بعد از اتمام اجرای پردازش پدر را نمی‌دهد. با توجه به درخت زیر از پردازش‌ها، اگر پردازش ۸۴۱۵ خود را terminate کند (اجرایش را پایان دهد)، سیستم عامل کدام پردازش‌ها را terminate خواهد کرد (به اجرایشان پایان خواهد داد)؟



a. تنها پردازش ۲۸۰۸

b. تنها پردازش‌های ۸۴۱۶ و ۹۲۰۴

c. تنها پردازش‌های ۸۴۱۶ و ۹۲۹۸

d. تنها پردازش 8416

e. پردازش‌های ۸۴۱۶، ۹۲۹۸ و ۹۲۰۴



پاسخ شما صحیح می باشد.

پاسخ درست »

پردازش‌های ۸۴۱۶، ۹۲۹۸ و ۹۲۰۴ است.

کد زیر از Pthreads API استفاده می‌کند. کدام گزینه در مورد مقدار چاپ شده برای متغیر value در خطوط A و B و C صحیح است؟ (۴.۵ دقیقه)

```
#include <sys/types.h>
#include <sys/wait.h>
#include <pthread.h>
#include <stdio.h>
#include <unistd.h>

int value = 5;
void *runner1(void *param);
void *runner2(void *param);

int main()
{
    pthread_t tid1, tid2;
    pthread_attr_t attr1, attr2;

    pthread_attr_init(&attr1);
    pthread_create(&tid1, &attr1, runner1, NULL);

    pthread_attr_init(&attr2);
    pthread_create(&tid2, &attr2, runner2, NULL);

    printf("A: value = %d\n", value); /* LINE A */

    pthread_join(tid1, NULL);
    pthread_join(tid2, NULL);

    return 0;
}

void *runner1(void *param) {
    value += 10;
    printf("B: value = %d\n", value); /* LINE B */

    pthread_exit(0);
}

void *runner2(void *param) {
    value += 10;
    printf("C: value = %d\n", value); /* LINE C */

    pthread_exit(0);
}
```

- ☐ a. خروجی در هر سه خط A و B و C برای اجراهای مختلف، بواسطه وجود race condition می‌تواند متفاوت باشد.
- ☐ b. خروجی در خط C همیشه برابر ۱۵ است و برای خطوط A و B، خروجی برای اجراهای مختلف می‌تواند متفاوت باشد.
- ☒ c. خروجی در خط B همیشه برابر ۱۵ است و برای خطوط A و C، خروجی برای اجراهای مختلف می‌تواند متفاوت باشد.
- ☐ d. خروجی در خط A همیشه برابر ۲۵ است و برای خطوط B و C، خروجی برای اجراهای مختلف می‌تواند متفاوت باشد.

✖

پاسخ شما صحیح نیست

پاسخ درست «خروجی در هر سه خط A و B و C برای اجراهای مختلف، بواسطه وجود race condition می‌تواند متفاوت باشد.» است.

سؤال 6

درست

نمره 2.00 از 2.00

۳ علامت زدن

سؤال

شما از یک مهندس کامپیوتر خواسته‌اید که Unix shell را پیاده کند. این شخص برای ایجاد یک وظیفه (job) جدید (مانند اجرا کردن یک برنامه یا دستور (cmd) در ترمینال یا shell)، به جای اینکه ابتدا fork و سپس exec را صدا بزند، مطابق کد زیر ابتدا exec و سپس fork را صدا زده است. (۳ دقیقه)

```
shell (..) {  
    .. ..  
    exec (cmd, args);  
    fork();  
    .. ..  
}
```

نظر شما در مورد ادعای درست کار کردن پیاده‌سازی بالا چیست؟ (۳ دقیقه)

یک گزینه را انتخاب کنید:

☐ صحیح

☒ غلط

پاسخ درست گزینه «غلط» است.

برنامه زیر از Pthreads API استفاده می‌کند. هنگامی که زمان اجرای خط B برسد، چه تعداد نخ (thread) در سیستم بواسطه این برنامه فعال خواهد بود؟ (۴.۵ دقیقه)

```
#include <sys/types.h>
#include <sys/wait.h>
#include <pthread.h>
#include <stdio.h>
#include <unistd.h>

int value = 5;
void *runner1(void *param);
void *runner2(void *param);

int main()
{
    pthread_t tid1, tid2;
    pthread_attr_t attr1, attr2;

    pthread_attr_init(&attr1);
    pthread_create(&tid1, &attr1, runner1, NULL);

    pthread_attr_init(&attr2);
    pthread_create(&tid2, &attr2, runner2, NULL);

    printf("A: value = %d\n", value); /* LINE A */

    pthread_join(tid1, NULL);
    pthread_join(tid2, NULL);

    return 0;
}

void *runner1(void *param) {
    value += 10;
    printf("B: value = %d\n", value); /* LINE B */

    pthread_exit(0);
}

void *runner2(void *param) {
    value += 10;
    printf("C: value = %d\n", value); /* LINE C */

    pthread_exit(0);
}
```



- ☒ a. ۲ یا ۳ نخ (بسته به نحوه زمانبندی نخ‌های ایجاد شده)
- ☐ b. همیشه ۲ نخ
- ☐ c. ۱ یا ۲ یا ۳ نخ (بسته به نحوه زمانبندی نخ‌های ایجاد شده)
- ☐ d. همیشه ۱ نخ (thread)
- ☐ e. همیشه ۳ نخ

سوال 8

درست

نمره 2.00 از 2.00

۳ علامت زدن

سوال

بخش text از فضای ادرس یک پردازنده (process address space) شامل چه چیزی می‌شود؟ (۳ دقیقه)



a. کد اجرایی برنامه

b. پیام‌های متنی (text) که به این پردازنده ارسال شده است.

c. پیام‌هایی که از طریق IPC به این پردازنده ارسال می‌شود

d. داده‌های تعریف شده به صورت پویا، مانند `int* arr = new(10*sizeof(int))`

e. داده‌های تعریف شده به صورت ایستا (مانند `constant static x=0`)

پاسخ شما صحیح می باشد

پاسخ درست »

کد اجرایی برنامه « است.

سوال 9

درست

نمره 2.00 از 2.00

۳ علامت زدن
سوال

سیستم عامل لینوکس از کدام مدل نگاشت (mapping) نخ سطح کاربر (user space thread) به نخ سطح هسته (kernel space thread) پشتیبانی می‌کند؟ (۳ دقیقه)



☒ a. One-to-one

☐ b. One-to-many

☐ c. Many-to-one

☐ d. همه گزینه‌ها

☐ e. Many-to-many

پاسخ شما صحیح می باشد

پاسخ درست »

«One-to-one» است.

سؤال 10

درست

نمره 2.00 از 2.00

۳ علامت زدن
سؤال

کدام گزینه یک سیگنال غیرهمزمان (Asynchronous signal) است؟ (۳ دقیقه)

- ☐ a. دسترسی به محتوای یک اشاره‌گر NULL در یک برنامه
- ☐ b. رخداد تقسیم بر صفر (divide by 0) در یک برنامه
- ☐ c. دسترسی غیر مجاز به حافظه (illegal memory access) در یک برنامه
- ☒ d. فشردن شدن <C><control> در ترمینال برای بستن برنامه‌ای که از طریق ترمینال باز شده است.



پاسخ شما صحیح می باشد

پاسخ درست »

فشردن شدن <C><control> در ترمینال برای بستن برنامه‌ای که از طریق ترمینال باز شده است.» است.

سؤال 11

درست

نمره 1.00 از 1.00

۳ علامت ردن

سؤال

هنگامی که یک I/O Device از DMA برای انتقال داده به/از حافظه استفاده می‌کند، CPU می‌تواند با اجرای برنامه‌های دیگر، کار مفیدی انجام دهد.

یک گزینه را انتخاب کنید:

☒ صحیح

☐ غلط

پاسخ درست گزینه «صحیح» است.

برنامه زیر از Pthreads API و fork استفاده می‌کند. هنگامی که زمان اجرای خط B برسد، چند پردازش (process) بواسطه این برنامه در سیستم فعال خواهد بود؟
توضیحات: فرض کنید که هیچکدام از فراخوانی‌های سیستمی با شکست مواجه نمی‌شوند.

```
#include <sys/types.h>
#include <sys/wait.h>
#include <pthread.h>
#include <stdio.h>
#include <unistd.h>

int value = 5;
void *runner(void *param);

int main()
{
    pid_t pid;
    pthread_t tid;
    pthread_attr_t attr;

    printf("A: value = %d\n", value); /* LINE A */

    pid = fork();

    if (pid == 0) {
        pthread_attr_init(&attr);
        pthread_create(&tid, &attr, runner, NULL);
        pthread_join(tid, NULL);
        printf("B: value = %d\n", value); /* LINE B */
        return 0;
    }
    else if (pid > 0) {
        wait(NULL);
        printf("C: value = %d\n", value); /* LINE C */
        return 0;
    }
    return 0;
}

void *runner(void *param) {
    value += 10;
    pthread_exit(0);
}
```

2.a ☒3.b ☐1.c ☐4.d ☐

پاسخ شما صحیح می باشد

پاسخ درست »

2 است.

سوال 13

درست

نمره 2.00 از 2.00

۳ علامت زدن
سوال

کدام گزینه می‌تواند راهکاری برای کاهش سربار context switch باشد؟ (۳ دقیقه)

☐ a. پیاده‌سازی سیستم عامل با روش‌های پیچیده‌تر

☐ b. هیچکدام

☐ c. اضافه کردن فیلدهای بیشتر و جدیدتر به PCB پردازنده‌ها

☒ d. طراحی سخت‌افزار به گونه‌ای که بتوان چندین context را همزمان در CPU داشت (multiple contexts loaded at once).



پاسخ شما صحیح می‌باشد

پاسخ درست »

طراحی سخت‌افزار به گونه‌ای که بتوان چندین context را همزمان در CPU داشت (multiple contexts loaded at once). « است.

برنامه زیر از Pthreads API و fork استفاده می‌کند. کدام گزینه در مورد مقدار چاپ شده متغیر value در خطوط A و B و C صحیح نیست؟ (۴.۵ دقیقه)

توضیحات: فرض کنید که هیچکدام از فراخوانی‌های سیستم با شکست مواجه نمی‌شوند.

```
#include <sys/types.h>
#include <sys/wait.h>
#include <pthread.h>
#include <stdio.h>
#include <unistd.h>

int value = 5;
void *runner(void *param);

int main()
{
    pid_t pid;
    pthread_t tid;
    pthread_attr_t attr;

    printf("A: value = %d\n", value); /* LINE A */

    pid = fork();

    if (pid == 0) {
        pthread_attr_init(&attr);
        pthread_create(&tid, &attr, runner, NULL);
        pthread_join(tid, NULL);
        printf("B: value = %d\n", value); /* LINE B */
        return 0;
    }
    else if (pid > 0) {
        wait(NULL);
        printf("C: value = %d\n", value); /* LINE C */
        return 0;
    }
    return 0;
}

void *runner(void *param) {
    value += 10;
    pthread_exit(0);
}
```

- ☐ a. مقدار متغیر value در خط A همیشه (برای همه اجراها) عدد ۵ خواهد بود.
- ☐ b. در این برنامه برای متغیر value حالت رقابت (race condition) وجود ندارد بنابراین آنچه که چاپ می‌شود، برای همه اجراها یکسان است.
- ☐ c. مقدار متغیر value در خط B همیشه (برای همه اجراها) عدد 15 خواهد بود.
- ☒ d. مقدار متغیر value در خط C همیشه (برای همه اجراها) عدد 15 خواهد بود.



پاسخ شما صحیح می باشد

پاسخ درست »

مقدار متغیر value در خط C همیشه (برای همه اجراها) عدد 15 خواهد بود. « است.

سؤال 15

درست

نمره 3.00 از 3.00

علامت زدن
سؤال

برنامه ساده پایتون زیر را در نظر بگیرید. تابع printf چند مرتبه صدا زده می‌شود؟ (۴.۵ دقیقه)

```
import os

i=0
while i < 2:
    print(i)
    os.fork()
    i += 1
```

4 .a ☐

2 .b ☐

3 .c ☒

5 .d ☐



پاسخ شما صحیح می باشد

پاسخ درست »

3 « است.

سوال 16

درست

نمره 2.00 از 2.00

۳ علامت زدن
سوال

هنگامی که پردازنده‌ها (process) از طریق حافظه مشترک (shared memory) با هم ارتباط برقرار می‌کنند، کدام گزینه در مورد کنترل این ارتباط و بحث همگام‌سازی صحیح است؟

- ☐ a. تمامی کنترل‌ها و همگام‌سازی (synchronization) توسط سیستم عامل انجام می‌شود و جای نگرانی برای برنامه‌ها (پردازنده‌ها) نیست.
- ☐ b. دسترسی read-only تنها راهکار در دسترس برای برنامه‌ها برای جلوگیری از race condition است.
- ☒ c. تمامی کنترل‌ها و همگام‌سازی بایستی توسط خود برنامه‌ها انجام شود (با استفاده از امکانات سیستم عامل).
- ☐ d. امکان رخداد race condition در ارتباط از طریق حافظه مشترک وجود ندارد، بنابراین نیازی به همگام‌سازی وجود ندارد.



پاسخ شما صحیح می‌باشد

پاسخ درست »

تمامی کنترل‌ها و همگام‌سازی بایستی توسط خود برنامه‌ها انجام شود (با استفاده از امکانات سیستم عامل). « است.

سؤال 17

درست

نمره 2.00 از 2.00

۳ علامت زدن
سؤال

کدام مورد در ارتباط با فراخوانی سیستمی fork صحیح نیست؟ (۳ دقیقه)

a. در اثر این فراخوانی، پردازش فرزند یک کپی از کد پردازش پدر دریافت می‌کند.

b. در اثر fork یک کپی از جدولی که اطلاعات فایل‌های باز در پردازش پدر را در خود نگه می‌دارد، برای پردازش فرزند ایجاد می‌شود.

c. فراخوانی سیستمی fork پردازش پدر را متوقف می‌کند تا اینکه اجرای پردازش فرزند تمام شود.

d. فراخوانی سیستمی fork یک کپی از داده‌های پردازش پدر را برای پردازش فرزند ایجاد می‌کند.



پاسخ شما صحیح می‌باشد

پاسخ درست »

فراخوانی سیستمی fork پردازش پدر را متوقف می‌کند تا اینکه اجرای پردازش فرزند تمام شود. « است.

سوال 18

درست

نمره 1.00 از 1.00

۳ علامت زدن

سوال

برنامه‌ای که به همراه سیستم عامل بر روی سیستم کامپیوتری نصب شده ولی لزوماً بخشی از هسته سیستم عامل نیست، چه نامیده می‌شود؟ (۱.۵ دقیقه)

a. Application program ☐

b. Middleware ☐

c. Software framework ☐

d. System program ☒



پاسخ شما صحیح می باشد

پاسخ درست »

System program « است.

سوال 19

درست

نمره 1.00 از 1.00

۳ علامت ردن

سوال

یک برنامه چند-نخی (multi-threaded) همانند یک برنامه یک-نخی (single-threaded)، نیاز به یک program counter دارد. (۱.۵ دقیقه)
توضیحات: PC آدرس دستور بعدی که برنامه باید اجرا کند را مشخص می‌کند. (۱.۵ دقیقه)

یک گزینه را انتخاب کنید:

☐ صحیح

☒ غلط ✓

پاسخ درست گزینه «غلط» است.

سوال 20

درست

نمره 2.00 از 2.00

علامت زدن

سوال

در روش انتقال پیام مابین پردازنده‌ها از طریق shared memory، **ماکزیمم** طول یک پیام چقدر است؟ (۳ دقیقه)

a. 128 bytes ☐

b. هیچکدام ☒

c. 512 bytes ☐

d. 1024 bytes ☐

e. 32 bytes ☐

پاسخ شما صحیح می باشد

پاسخ درست »

هیچکدام « است.

