

Ukázka zkuškové písemka OSY

Jméno a příjmení:

.....

Odpovězte na otázky zaškrtnutím příslušného políčka. Otázky označené znakem ♣ mohou mít více než jednu správnou odpověď. U otázek se slovní odpovědí nezaškrťte políčka na šedém pozadí.

Otázka 1 ♣ Software šířený pod licencí GPLv2:

- není možné prodávat za peníze
- nesmí se používat s komerčními OS (např. MS Windows)
- musí být šířen společně se zdrojovými kódy
- je možné použít v proprietárním SW, ale pouze po odstranění všech hlášek "copyright"
- Nic z výše uvedeného není správně

Otázka 2 ♣ Která následující tvrzení jsou pravdivá?

- Při použití mutexu může vždy nastat deadlock – vzájemné uváznutí
- Mutex lze použít pro správný přístup ke sdíleným datům
- Mutex zaručuje, že nenastane deadlock – vzájemné uváznutí
- Mutex lze použít pro synchronizaci paralelně běžících vláken
- Nic z výše uvedeného není správně

Otázka 3 ♣ Vloni zveřejněná zranitelnost CPU zvaná Meltdown:

- způsobí, že při vykonání škodlivého kódu se procesor přehřeje a dojde k jeho zničení
- je to chyba HW a OS s tím nemůže nic dělat
- umožňuje číst data i ze stránek, ke kterým nemá uživatelský proces přístup
- je jen další mediální bublina
- vyskytuje se pouze u procesorů od Intelu
- Nic z výše uvedeného není správně

Otázka 4 Mějme následující program:

```
int a[2];
pthread_mutex_t mutex[2];

void *fce1(void *n) {
    int num=(int*)n;
    for (int i = 0; i < 150; i++) {
        pthread_mutex_lock(&mutex[0]);
        pthread_mutex_lock(&mutex[1]);
        a[num] += a[1-num];
        pthread_mutex_unlock(&mutex[1]);
        pthread_mutex_unlock(&mutex[0]);
    }
    pthread_exit(NULL);
}

void *fce2(void *n) {
    int num=(int*)n;
    for (int i = 0; i < 150; i++) {
        pthread_mutex_lock(&mutex[1]);
        pthread_mutex_lock(&mutex[0]);
        a[num] += a[1-num];
        pthread_mutex_unlock(&mutex[0]);
        pthread_mutex_unlock(&mutex[1]);
    }
    pthread_exit(NULL);
}

int main()
{
    pthread_t tid[2];
    a[0]=0; a[1]=1;
    pthread_mutex_init(&mutex[0], NULL);
    pthread_mutex_init(&mutex[1], NULL);
    pthread_create(&tid[0], NULL, fce1, NULL);
    pthread_create(&tid[1], NULL, fce2, NULL);
    pthread_join(tid[0], NULL);
    pthread_join(tid[1], NULL);
    return 0;
}
```

- program neskončí, obsahuje nekonečnou smyčku
- program vždy bez problémů skončí
- program někdy skončí, někdy uvázne v deadlocku
- program vždy uvázne v deadlocku

Otázka 5 ♣ Stárnutí (starvation) je problémem, který hrozí plánovacím algoritmům. Které algoritmy ohrožuje stárnutí:

- | | |
|---|--|
| <input checked="" type="checkbox"/> prioritní plánování | <input type="checkbox"/> spuštění |
| <input checked="" type="checkbox"/> nejkratší proces první – shortes process next | <input type="checkbox"/> cyklické plánování – round robin |
| <input type="checkbox"/> First Come First Served – podle pořadí | <input type="checkbox"/> Nic z výše uvedeného není správně |

Otázka 6 ♣ V tabulce stránek 32bitového systému x86 je pro virtuální adresu 0x12345678 uvedena hodnota 0xCCCCCC005.

```
enum {
    PRESENT = 1<<0,
    RW      = 1<<1,
    USER    = 1<<2,
    ACCESS  = 1<<5,
    DIRTY   = 1<<6,
};
```

Které z následujících tvrzení je pravdivé:

- Data z této virtuální adresy jsou v RAM na adrese 0xCCCCCC678
- Jádro OS může z této stránky číst
- Uživatelský proces může z této stránky číst
- Data z této virtuální adresy jsou v RAM na adrese 0xCCCCCC123
- Data z této virtuální adresy jsou v RAM na adrese 0xCCCCCC003
- Uživatelský proces může do této stránky zapisovat
- Nic z výše uvedeného není správně

Otázka 7 Předpokládejte, že nedojde k chybě při spuštění následujícího programu:

```
int main() {
    int f=fork();
    f=fork();
    f=fork();
    printf("%d\n", f);
    return 0;
}
```

Jaký bude výstup tohoto programu?

- dvě nenulová čísla a dvě 0
- čtyři nenulová čísla a dvě 0
- osm nenulových čísel
- osm 0
- čtyři nenulová čísla a čtyři 0
- šest nenulových čísel a dvě 0
- tři nenulová čísla a jedna 0

Otázka 8 ♣ Hardwarově asistovaná virtualizace

- zrychluje běh virtuálního uživatelského režimu.
- odstraňuje nutnost emulovat hardware mechanismem trap-and-emulate.
- funguje jedině na 32bitovém systému.
- přidává další vrstvu stránkovacích tabulek.
- zrychluje vykonávání některých privilegovaných instrukcí.
- Nic z výše uvedeného není správně

Otázka 9 ♣ Jaký je vztah pojmů „page fault“ (výpadek stránky) a „segmentation fault“

- Page fault běžně nastává během běhu většiny programů
- Dva různé pojmy pro stejnou věc
- Page fault je vždy důsledek segmentation fault
- Page fault se používá k implementaci copy-on-write
- Segmentation fault běžně nastává během běhu většiny programů
- Segmentation fault je vždy důsledek page fault
- Nic z výše uvedeného není správně

Otázka 10 ♣ K zajištění konzistence souborového systému po náhlém vypnutí či pádu systému

- je potřeba provádět všechny související modifikace souborového systému atomicky
- je potřeba vybavit počítač záložním zdrojem (tzv. UPS).
- je vždy potřeba kompletní kontrola souborového systému po následném zapnutí počítače.
- lze použít metodu, kdy se popis potřebných modifikací nejprve uloží do speciální oblasti na disku
- je potřeba používat RAID1 a vyšší.
- Nic z výše uvedeného není správně

Otázka 11 Uvažujte následující příkaz BASHe:

```
l=`ps`
```

Který z následujících příkazů vytiskne na každou řádku jeden proces?

- echo \$l echo '\$l' echo \${l} echo "\$l" echo (\$l)

Otázka 12 Vlákna na jedno-procesorovém počítači (bez hyper-threadingu):

- mohou běžet paralelně
- mohou běžet paralelně pokud nikdy nepoužívají sdílené proměnné
- nemohou běžet paralelně nikdy
- mohou běžet paralelně pouze pokud použijí mutex

Otázka 13 Co provádí následující příkaz? `tr -d 't'`

- Odstraní znaky t ze standardního vstupu
- Zkopíruje standardní vstup beze změny
- Odstraní tabulATORY ze standardního vstupu
- Změní všechny znaky d na znak t

Otázka 14 Provedení funkce z jádra operačního systému na architektuře x86 lze vyvolat:

- instrukcí call na adresu služby jádra
- instrukcí int s registrem obsahujícím číslo služby jádra
- instrukcí nop s registrem obsahujícím číslo služby jádra
- instrukcí jmp na adresu služby jádra

Otázka 15 Popište, jak probíhá odesílání dat z aplikace na síť. Co se s daty děje a jakou roli v tom hraje jádro OS?

0 0.5 1 1.5 2

.....

.....

.....

.....

.....

CORRECTED