

Systemy operacyjne (zaawansowane)

Lista zadań nr 8

Na zajęcia 14 grudnia 2016

Należy przygotować się do zajęć czytając następujące rozdziały książek:

- Tanenbaum (wydanie czwarte): 3.3 – 3.6
- Stallings (wydanie siódme): 8.1 – 8.2

UWAGA! W trakcie prezentacji rozwiązań należy zdefiniować i wyjaśnić pojęcia, które zostały oznaczone **wytłuszczoną** czcionką.

Zadanie 1. Procesory MIPS nie definiują, a zatem nie zarządzają, **bitami monitorowania dostępu** we wpisach tablicy stron. Co więcej nie oferują również sprzętowego **przeglądania tablicy stron**, więc jądro systemu zarządza ręcznie zawartością **bufora translacji adresów**.

Wpisy tablicy stron mają **bity uprawnień**: *bit modyfikacji* (ang. *Dirty*) i *bit ważności* (ang. *Valid*). Procesor nie zmienia samodzielnie wartości tych bitów. Instrukcje generują odrębne wyjątki procesora przy **chybieniu w TLB** (ang. *TLB refill exception*); przy próbie dostępu, gdy $V=0$ (ang. *TLB invalid exception*); przy próbie modyfikacji strony, gdy $D=0$ (ang. *TLB modified exception*). Jak w takiej sytuacji jądro może odtworzyć bity referenced i modified?

Zadanie 2. Jak wspomniano na wykładzie **wielopoziomowa tablica stron** nie jest jedynym sposobem opisu **wirtualnej przestrzeni adresowej**. Przedstaw mechanizm **odwróconej tablicy stron** (ang. *inverted page table*). Wytłumacz jak przebiega proces tłumaczenia **adresu wirtualnego** na **adres fizyczny**. Jakie są wady i zalety takiej organizacji wpisów stron?

Podpowiedź: Porównanie najlepiej zrobić na podstawie §31.1.5 książki „Memory Systems: Cache, DRAM, Disk”.

Zadanie 3. Dla podanego poniżej fragment wydruku polecenia `«x86info -a»` podaj **zasięg bufora translacji adresów** pierwszego i drugiego poziomu.

```
L1 Data TLB: 4KiB pages, 16 entries
L1 Data TLB: 4MiB pages, 4 entries
L2 Data TLB: 4KiB pages, 256 entries
L2 Data TLB: 4MiB pages, 32 entries
```

Z jakich przyczyn producenci procesorów wprowadzili **duże strony** (ang. *superpages*, *huge pages*)? W jaki sposób zarządzać przydziałem dużych stron dla procesów? Jakie problemy są z tym związane?

Podpowiedź: Przeczytaj §2 publikacji “[Practical, transparent operating system support for superpages¹](https://people.mpi-sws.org/~druschel/publications/superpages.pdf)”.

Zadanie 4 (P). Wydrukuj informacje dotyczące przestrzeni adresowej procesów użytkownika:

```
cat /proc/*/status
```

Co opisują pola `VmPeak` i `VmSize`? Jak znaleźć rozmiar **zbioru rezydentnego**? Ile ramek przeznaczono na **pamięć anonimową**, a ile na **odzworowanie plików w pamięć**? Jakiego rozmiaru są segmenty danych, kodu i stosu? Ile pamięci jądro zużywa na tablice stron opisujących przestrzeń adresową procesu?

¹<https://people.mpi-sws.org/~druschel/publications/superpages.pdf>

Zadanie 5. Rozważmy system ze stałym rozmiarem **zbioru rezydentnego**. Mamy pamięć fizyczną składającą się z 4 **ramek** oraz pamięć wirtualną złożoną z 8 **stron**. Startujemy z pustą pamięcią fizyczną. Jedyny wykonujący się proces generuje następujący ciąg dostępów do stron:

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2

Zaprezentuj działanie następujących **algorytmów zastępowania stron**: OPT, FIFO, LRU, CLOCK. Który z nich generuje najmniej **błędów stron**? Załóż, że zastąpienie ramki zachodzi dopiero w momencie zapełnienia całej pamięci fizycznej.

Zadanie 6. Jak polityki zarządzania pamięcią wirtualną mogą wykorzystywać **przestrzenną lokalność odwołań** do pamięci? Jakie kryteria może stosować **demon stronicowania** (ang. *pageout daemon*) określając rozsądną wielkość zbioru rezydentnego dla procesu? Wyjaśnij przyczyny powstawania zjawiska **szamotania**. W jaki sposób można to zjawisko wykryć i jak mu zapobiegać?

Zadanie 7 (P). Wyświetl zmienne jądra Linuksa związane z zarządzaniem pamięcią wirtualną:

```
sysctl -a -r ^vm |& sort
```

Na podstawie [dokumentacji](#)² wyjaśnij znaczenie zmiennych: `dirty_ratio`, `overcommit_memory`, `oom_kill_allocating_task`, `swappiness`, `watermark_scale_factor`.

Zadanie 8. W jaki sposób jądro Linuksa reaguje na wyczerpanie pamięci fizycznej? Na podstawie opisu [Out-of-Memory Killer](#)³ omów kryteria, którymi kieruje się jądro Linuksa w odzyskiwaniu pamięci fizycznej w krytycznych sytuacjach.

²<https://www.kernel.org/doc/Documentation/sysctl/vm.txt>

³https://linux-mm.org/OOM_Killer