

Systemy operacyjne

Ćwiczenia 7

Należy przygotować się do zajęć czytając następujące rozdziały książek:

- Stallings: 8.1 – 8.3
- Tanenbaum: 3.3 – 3.7
- Silberschatz: 9.1 – 9.9

Uwaga: Deklarując zadanie wymagające podania pseudokodu, należy podejść do tablicy ze spisaniem na kartce algorytmem. Wymyślanie przy tablicy będzie kończyło się przyznaniem **punktu motywującego do cięższej pracy™**.

Zadanie 1

Wyjaśnij czym różni się **strona** od **ramki**. Znajdź informacje dotyczące **bitów pomocniczych w deskryptorach stron** i **deskryptorach katalogów stron** dla architektury x86. Które z nich dotyczą:

- sposobu używania pamięci podręcznej,
- wspomagają algorytmy zarządzania pamięcią wirtualną,
- określają uprawnienia dostępu (włączając w to tryb pracy procesora).

Opisz dokładnie strukturę **PTE** i **PDE** oraz znaczenie wszystkich bitów.

Zadanie 2

Rozważmy **procedurę obsługi braku strony**, do której przekazywane jest sterowanie w wyniku niewłaściwego odwołania do pamięci. Jakie dane musi otrzymać ta funkcja by prawidłowo wykonywać swoją pracę? Zaproponuj pseudokod procedury umożliwiającej **stronicowanie na żądanie** z uwzględnieniem uprawnień dostępu do stron. Wykorzystaj następujący przykładowy interfejs:

- `page_t *find_pte(void *virt_addr);`
- `void error(const char *msg);`
- `frame_t *alloc_frame();`
- `void read_block(frame_t *frame, int block_addr);`
- `void resume(void *pc);`

Czy oprócz tablicy stron musisz zarządzać jakąś dodatkową strukturą danych?

Zadanie 3

Pokaż w pseudokodzie jak należy rozszerzyć procedurę obsługi braku stron z poprzedniego zadania celem implementacji **wymiany** i **mapowania plików** na pamięć. Czym różni się **pomniejszy błąd strony** od **głównego błędu strony**? Załóż, że posiadasz wiele plików (każdy z unikalnym identyfikatorem *i-node*) i jedną **partycję wymiany**. Zaproponuj prosty interfejs (zbiór funkcji) podobny do tego z poprzedniego zadania. Jeśli wprowadzono dodatkowe struktury danych – opisz je i uzasadnij ich niezbędność.

Zadanie 4

Wyjaśnij zasadę działania mechanizmu **kopiowania przy zapisie** w systemach ze stronicowaniem. Jakie korzyści przynosi? Do czego używa się go w praktyce? Pokaż w pseudokodzie jak rozszerzyć procedurę obsługi braku stron. Zaproponuj prosty abstrakcyjny interfejs. Jeśli wprowadzono dodatkowe struktury danych lub rozszerzono istniejące – opisz je i uzasadnij ich niezbędną zmian.

Zadanie 5

Rozważmy cztery klasy polityk zarządzania pamięcią wirtualną. Wyjaśnij kiedy mają zastosowanie **polityka ładowania, przydziału miejsca, zastępowania i usuwania**? Poznaliśmy już stronicowanie na żądanie. Czym różni się od niego **stronicowanie wstępne** i kiedy ma sens?

Zadanie 6

Wykaż, że zasada lokalności dotyczy również pamięci wirtualnej opartej na stronicowaniu. Wyjaśnij pojęcia **zbioru roboczego** i **zestawu rezydentnego**. Jak te zbiory zmieniają się w czasie działania procesu i jak to się ma do zasady lokalności? Znajdź i opisz wykres zależności braków stron od rozmiaru ramki dla typowego programu. Czemu w systemach z bardzo małymi ramkami występuje (statystycznie) mniej braków stron niż w systemach z ramkami o średniej wielkości? Załóż, że ilość dostępnych stron rośnie odwrotnie proporcjonalnie do ich wielkości.

Zadanie 7

Rozważmy system ze **stałym przydziałem zbioru rezydentnego**. Pamięć fizyczna składa się z 4 ramek, a pamięć wirtualna z 8 stron. Na maszynie wykonuje się jeden proces i generuje następujący ciąg dostępów do stron: 7 0 1 2 0 3 0 4 2 3 0 3 2. Pokaż działanie następujące algorytmy zastępowania ramek: **FIFO**, **LRU**, **CLOCK**. Jak te algorytmy korzystają z pomocniczych bitów w deskryptorach stron? Który z nich generuje najmniej **braków stron**? Załóż, że zastąpienie ramki zachodzi dopiero w momencie zapełnienia całej pamięci fizycznej.

Zadanie 8

Rozważmy **buforowanie stron** – tj. jeśli strony zostały oznaczone do zastąpienia w wyniku stosowania konkretnej strategii (np. **FIFO**), to nie są usuwane od razu, ale trzymają się przez jakiś czas w pamięci operacyjnej. Wymień zalety tego rozwiązania w kontekście lokalności i zarządzania zbiorem roboczym, oraz polityki usuwania stron. Jak z buforowania stron może skorzystać algorytm zarządzający **zbiorem rezydentnym zmiennego rozmiaru**, działający w obrębie całego systemu? Do czego służy **przypinanie stron** w pamięci operacyjnej?

Zadanie 9

Rozważmy system ze stronicowaniem na żądanie, w którym stopień wieloprogramowości wynosi 4 (programy). W pewnym momencie czasu w systemie zbadano procent wykorzystania procesora i dyskowej pamięci wirtualnej. Dla każdego z poniższych pomiarów odpowiedz, czy należy zwiększyć / zmniejszyć stopień wieloprogramowości i czy zastosowanie stronicowania nie przyczyniło się do spadku wydajności systemu:

- CPU: 13%, IO: 97%,
- CPU: 87%, IO: 3%,
- CPU: 13%, IO: 3%.

Wyjaśnij pojęcie **szamotania**. Wymień kilka możliwych zachowań systemu operacyjnego celem obniżenia stopnia wieloprogramowości po wykryciu tego zjawiska?