

Architektury systemów komputerowych

Lista 11

$$x_{11} = 7 \text{ (minimum na bdb)}$$

1. Rozważmy system pamięci z pamięcią główną o pojemności 2^{12} bajtów, adresowaną bajtowo. Pamięć cache składa się z czterech 16-bajtowych wierszy. Przedstaw logiczny podział adresu w pamięci głównej, przy założeniu, że:
 - (a) cache używa mapowania bezpośredniego,
 - (b) cache jest sekcyjno-skojarzeniowy i ma dwie sekcje,
 - (c) cache jest w pełni skojarzeniowy.
2. W tym zadaniu zakładamy czterogigabajtową przestrzeń adresową i adresowanie bajtowe. Ilu bitów potrzebujemy do fizycznej realizacji pamięci cache w dwóch poniższych wariantach? Należy policzyć bity potrzebne na przechowywane dane oraz na znaczniki. Proszę pominąć ewentualne dodatkowe bity (np. tzw. bity aktualności danych). W obu wariantach mamy 32-bajtowe bloki, a cache przechowuje 64 kilobajty danych.
 - (a) mapowanie bezpośrednie
 - (b) mapowanie sekcyjno-skojarzeniowe 4-drożne (sekcje czterowierszowe).
3. Rozważamy pamięć podręczną z mapowaniem sekcyjno-skojarzeniowym, wykorzystującą algorytm wymiany LRU (*least recently used* – najdawniej używany). Rozmiar sekcji – 4 bloki. Do każdego bloku dodane są dwa bity, używane do obliczenia, który blok był ostatnio używany. Licznik działa następująco:
 - kiedy pojawi się „chybienie”, to blok z licznikiem ustawiony na 0 jest podmieniany, a nowo wstawiony blok dostaje licznik 3; pozostałe bloki mają liczniki zmniejszane o 1;
 - kiedy pojawi się „trafienie”, to licznik bloku, do którego się odwołujemy ustawiamy na 3 i wprowadzmy konieczne modyfikacje w pozostałych licznikach.

Jeśli w pamięci cache znajdują się cztery bloki A, B, C, D z licznikami ustawionymi odpowiednio na 0, 1, 2, 3, to które bloki pozostaną w zbiorze i jakie będą wartości ich liczników po sekwencji odwołań: E, B, E, D, A, E?
4. W wiele procesorach, algorytmem wymiany bloków nie jest dokładny algorytm LRU (jak w zadaniu poprzednim), ale jego przybliżenie. Opiszemy teraz taki przybliżony algorytm LRU (*pseudo-LRU*) dla pamięci 4-drożnej: z każdą sekcją (oznaczymy jej wiersze jako L1, L2, L3, L4) są skojarzone trzy bity (zwracam uwagę, że bity są skojarzone z całą sekcją a nie z pojedynczym wierszem): B0, B1, B2. Bit B0 określa która para wierszy (L0-L1, czy L2-L3) była używana dawniej, bity B1 i B2 wskazują, które wiersze w poszczególnych parach były wykorzystywane dawniej. Algorytm, w zależności od B0, wyrzuca wiersz wskazywany przez B1 lub B2.
 - (a) Pokaż, że algorytm stanowi tylko przybliżenie prawdziwego algorytmu LRU (wskaz sekwencję odwołań, która rozróżnia te dwa algorytmy).
 - (b) Wykaż, że prawdziwy algorytm LRU wymaga co najmniej 5 bitów na jeden zbiór.
5. Rozważmy pamięć cache o pojemności 1 kilobajta, z mapowaniem bezpośrednim i blokiem o rozmiarze 8 bajtów.
 - (a) Jaki jest współczynnik chybień przy następującej sekwencji odwołań (adresy podane są szesnastkowo): 0x0, 0x8, 0x10, 0x18, 0x20, 0x28?
 - (b) Jak zmieni się współczynnik chybień przy powyższej sekwencji odwołań jeśli
 - i. zwiększymy rozmiar bloku do 16,

- ii. zastąpimy mapowanie bezpośrednie mapowaniem sekcyjno-skojarzeniowym, dwudrożnym (dwa wiersze na sekcję).
6. Rozważmy system pamięci zbudowany z jednopoziomowej pamięci cache oraz pamięci głównej. Pamięć cache jest sekcyjno-skojarzeniowa, składa się z czterech dwuwierszowych sekcji, algorytm wymiany to LRU. Rozmiar bloku (wiersza) to cztery słowa. Pamięć główna mieści 1024 słowa. Adresowalną jednostką jest słowo. Zakładamy, że czas odczytu z pamięci cache to 5 ns, natomiast czas przeniesienia całego bloku z pamięci głównej do cache wynosi 95 ns. Przyjmij, że w przypadku chybienia w pamięci cache najpierw wczytywany jest cały blok, a potem jeszcze odpowiednie słowo przenoszone jest do procesora (trwa to zatem w sumie 100 ns.).
- Przedstaw format adresu w tym systemie (podziel adres na logiczne fragmenty, podaj ich długości w bitach)
 - Rozważ wykonanie programu, który odczytuje kolejno słowa 2, 3, 4, 5, ..., 39, a następnie wykonuje pięciokrotnie pętlę odczytującą słowa 40, 41, ..., 47. Ile chybień w pamięci cache wystąpi podczas wykonywania tego programu? Jaki jest średni czas odczytu słowa w tym programie (wystarczy przedstawić wzór, nie trzeba koniecznie wyliczać dokładnej wartości)? Zakładamy, że przed wykonaniem programu pamięć cache jest pusta.
7. Rozważmy system komputerowy z pamięcią główną o pojemności 2^{15} słów 16-bitowych. Ma także pamięć podręczną 2^{12} słów podzieloną na 4-wierszowe sekcje z 64 słowami w każdym wierszu. Załóż, że pamięć podręczna jest na początku pusta. Procesor pobiera słowa z lokalacji 0, 1, 2, ..., 4351 w tym właśnie porządku. Następnie powtarza tę sekwencję pobierania jeszcze 9 razy. Pamięć podręczna jest 10 razy szybsza niż pamięć główna. Oszacuj korzyść wynikającą z zastosowania pamięci podręcznej. Przyjmij, że wymiana bloków realizowana zgodnie z (dokładnym) algorytmem LRU.
8. Rozważmy dwie realizacje pamięci cache:
- z mapowaniem bezpośrednim
 - z pełnym mapowaniem skojarzeniowym i algorytmem wymiany LRU (*least recently used*).
- Czy można skonstruować sekwencję odwołań, w której pierwsze rozwiązanie da większą liczbę trafień niż drugie? Skonstruuj taką sekwencję lub uzasadnij, że jest to niemożliwe.
9. Wyjaśnij na czym polegają strategie zapisu *write-back/write-through* oraz *write-allocate/write-no-allocate*.

Emanuel Kieroński