

Systemy operacyjne

Pamięć wirtualna

Dariusz Wawrzyniak



Systemy operacyjne



Plan wykładu

- Stronicowanie na żądanie
 - obsługa błędu strony
 - wymiana
- Problemy realizacji stronicowania na żądanie
 - problem wyboru ofiary
 - problem wznowiania rozkazów
- Algorytmy wymiany
 - algorytmy wymiany na żądanie
 - algorytmy wymiany ze sprowadzaniem na żądanie
 - algorytmy wstępnego stronicowania

Pamięć wirtualna (2)

Systemy operacyjne



Mechanizm stronicowania na żądanie

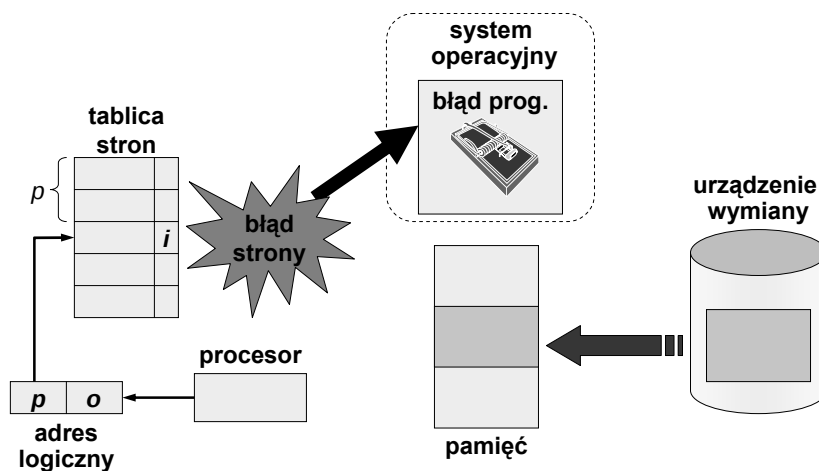
- Działanie mechanizmu: strony są sprowadzane do pamięci tylko wówczas, gdy jest to konieczne, czyli wówczas gdy następuje odniesienie do komórki o adresie, znajdującym się na tej stronie
- Wymaganie sprzętowe
 - tablica stron z bitem poprawności (ang. valid-invalid bit) dla każdej pozycji (dodatkowo z bitem modyfikacji i odniesienia),
 - mechanizm reakcji na odniesienie do strony niepoprawnej,
 - urządzenie wymiany (ang. swap device) — pamięć pomocnicza.

Pamięć wirtualna (3)

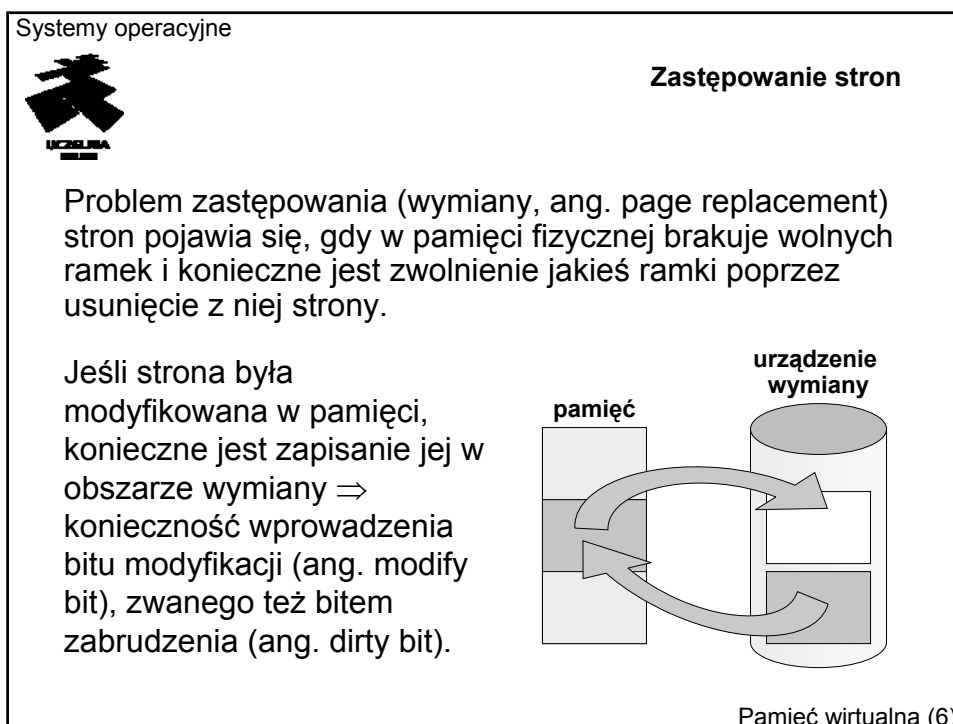
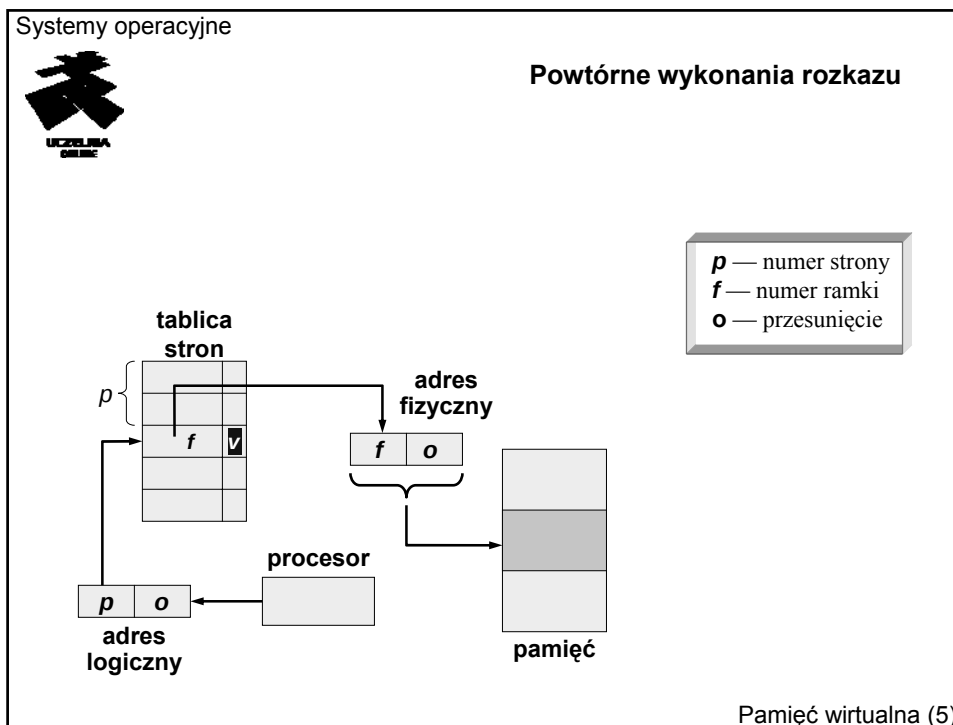
Systemy operacyjne



Obsługa błędu strony



Pamięć wirtualna (4)



Systemy operacyjne

**Koszt wymiany stron ⁽¹⁾**

- Całkowity koszt sprowadzenia stron w ogólnym przypadku dany jest wzorem:

$$\sum_t h(k_t)$$

gdzie:

- t — dyskretna chwila czasu, w której następuje odniesienie do pamięci (do strony)
- k_t — liczba stron sprowadzanych w chwili t ,
- $h(k)$ — koszt jednorazowego sprowadzenia grupy k stron, przy czym $h(0) = 0$, $h(1) = 1$.

Pamięć wirtualna (7)

Systemy operacyjne

**Koszt wymiany stron ⁽²⁾**

- W przypadku zastosowania dysku jako urządzenia wymiany, na czas sprowadzania wpływ mają:
 - T_w — czas oczekiwania (suma czasu oczekiwania w kolejce do urządzenia oraz czasu przygotowania urządzenia do transmisji)
 - T_{tr} — czas dostępu do danych
- Postać funkcji $h(k)$ dla dysku jest zatem następująca:
 $h(k) = T_w + k \cdot T_{tr}$
- Własność funkcji $h(k)$ przy założeniu, że $T_w > 0$ i $k > 0$:
 $h(k) < k \cdot (T_w + T_{tr}) \Rightarrow 1 \leq h(k) \leq k \cdot h(1)$

Pamięć wirtualna (8)

Systemy operacyjne

**Koszt wymiany stron ⁽³⁾**

- Koszt wymiany można aproksymować za pomocą parametrów FN (liczba wygenerowanych błędów strony) i TN (liczba transmisji stron).
- Całkowity czas oczekiwania na urządzenie (istotna składowa czasu wymiany) również jest zależny od liczby błędów strony.
- Wniosek: należy minimalizować liczbę błędów strony i tym samym redukować czas realizacji wymiany, czyli koszt.

Pamięć wirtualna (9)

Systemy operacyjne

**Problemy zastępowania stron**

- Problem wyboru ofiary — niewłaściwy wybór ramki-ofiary powoduje wzrost kosztu wymiany.
 - W skrajnym przypadku może dojść do zjawiska migotania, w przypadku którego często dochodzi do wystąpienia odniesienia do właśnie usuniętej strony.
- Problem wznowiania rozkazów — w przypadku wielokrotnego odniesienia do pamięci w jednym cyklu rozkazowym należy zapewnić, że wszystkie adresowane strony są jednocześnie dostępne w ramach w pamięci fizycznej.

Pamięć wirtualna (10)

Systemy operacyjne

**Problem wyboru ofiary**

- Zakładając, że przyszły ciąg odniesień do pamięci nie jest znany, na podstawie historii odniesień należy wybrać taką ramkę, do której prawdopodobieństwo odniesienia w przyszłości jest małe.
- Podstawowa własność programów, na podstawie której można szacować takie prawdopodobieństwo, nazywana jest lokalnością.

Pamięć wirtualna (11)

Systemy operacyjne

**Własność lokalności**

- Lokalność czasowa — tendencja procesów do generowania w stosunkowo długich przedziałach czasu odniesień do niewielkiego podzbioru stron wirtualnych zwanego zbiorem stron aktywnych.
 - Formalnie jest to tendencja procesu do generowania z dużym prawdopodobieństwem w przedziale czasu $(t, t + \tau)$ odniesień do stron adresowanych w przedziale czasu $(t - \tau, t)$.
- Lokalność przestrzenna — tendencja procesu do generowania z dużym prawdopodobieństwem kolejnych odniesień do stron o zbliżonych numerach (stron sąsiednich) lub stron o numerach skojarzonych w trakcie przetwarzania.

Pamięć wirtualna (12)

Systemy operacyjne

**Problem efektywności systemu z pamięcią wirtualną**

- Efektywność działania systemu pamięci wirtualnej zależy od precyzji identyfikacji zbioru stron aktywnych i możliwości utrzymania ich w pamięci fizycznej.
- Wobec braku a priori pełnego ciągu odniesień do stron wirtualnych identyfikacja takiego zbioru może wynikać z różnych przesłanek, czego skutkiem jest duża różnorodność algorytmów wymiany.

Pamięć wirtualna (13)

Systemy operacyjne

**Klasyfikacja algorytmów wymiany z względu na okoliczności sprowadzania i usuwania stron**

- Algorytmy wymiany na żądanie
 - sprowadzenia odbywa się na żądanie — strona sprowadzana jest dopiero wówczas, gdy następuje odniesienie do niej i nie ma jej w pamięci,
 - usuwanie odbywa się na żądanie — strona usuwana jest wówczas, gdy konieczne jest sprowadzenie innej strony w wyniku odniesienia i nie ma wolnej ramki
- Algorytmy wymiany ze sprowadzaniem na żądanie
 - tylko sprowadzanie odbywa się na żądanie
- Algorytmy wstępnego sprowadzania
 - sprowadzana jest strona żądana, a wraz z nią inne strony

Pamięć wirtualna (14)

Systemy operacyjne

**Klasyfikacja algorytmów wymiany ze względu na sposób zastępowania stron**

- Zastępowanie lokalne (ang. local replacement) — algorytm wymiany zastępuje tylko strony w ramach przydzielonych procesowi, który spowodował błąd strony.
- Zastępowanie globalne (ang. global replacement) — algorytm wymiany zastępuje strony znajdujące się w dostępnej puli ramek w całym systemie (w szczególności zatem usuwa strony innych procesów).

Pamięć wirtualna (15)

Systemy operacyjne

**Klasyfikacja algorytmów wymiany ze względu na przydział ramek dla procesów**

- Przydział statyczny — liczba ramek przydzielonych procesowi jest ustalona i nie ulega zmianie w trakcie przetwarzania.
- Przydział dynamiczny — liczba ramek przydzielonych procesowi może się zmienić w trakcie przetwarzania.

Pamięć wirtualna (16)

Systemy operacyjne

**Dobór liczby ramek**

- Minimalna liczba ramek — zdefiniowana przez architekturę komputera (zależna od maksymalnej liczby komórek adresowanych przez jeden rozkaz).
- Liczba ramek przydzielona dla procesu
 - podział równomierny (ang. equal allocation)
 - podział proporcjonalny (ang. proportional allocation)
 - przydział zależny od priorytetu procesu

Pamięć wirtualna (17)

Systemy operacyjne

**Algorytmy wymiany na żądanie**

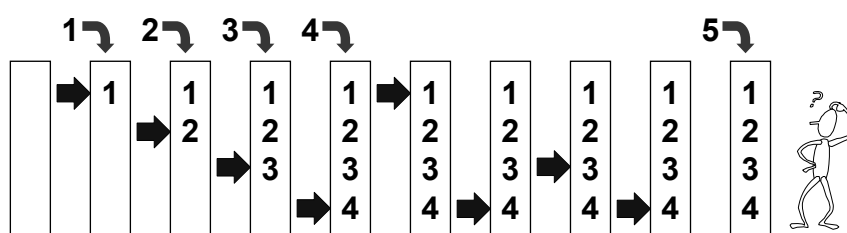
- MIN — zastępowana jest strona, która najdłużej nie będzie używana (optymalny w tej klasie)
- FIFO (ang. First In First Out) — zastępowana jest strona najstarsza (najwcześniej sprowadzona)
- LIFO (ang. Last In First Out) — zastępowana jest strona najmłodsza (najpóźniej sprowadzona)
- LRU (ang. Least Recently Used) — zastępowana jest najdawniej użyta strona (najdłużej nie używana)
- LFU (ang. Least Frequently Used) — zastępowana jest najrzadziej używana strona
- MFU (ang. Most Frequently Used) — zastępowana jest najczęściej używana strona

Pamięć wirtualna (18)

Systemy operacyjne

Przykład działania algorytmów wymiany na żądanie ⁽¹⁾

- W systemie pamięci wirtualnej są 4 ramki.
- Wszystkie ramki są początkowo puste
- W systemie pojawiają się następujące odniesień (odwołań) do stron: 1, 2, 3, 4, 1, 4, 3, 4, 5, 2, 1, 4, 3, 4

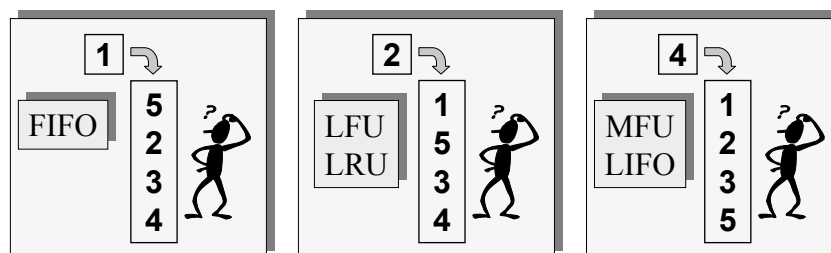


Pamięć wirtualna (19)

Systemy operacyjne

Przykład działania algorytmów wymiany na żądanie ⁽²⁾

Dalszy ciąg odniesień:
2, 1, 4, 3, 4



Pamięć wirtualna (20)

Systemy operacyjne



Przykład anomalii Belady'ego

Następujący ciąg odniesień obsługiwany jest zgodnie z algorytmem FIFO w systemie z 3, a następnie 4 ramkami:
1, 2, 3, 4, 1, 2, 5, 1, 2, 3, 4, 5

↓	↓	↓	↓	↓	↓	↓				↓	↓	
1	1	1	4	4	4	5	5	5	5	5	5	5
	2	2	2	1	1	1	1	1	3	3	3	3
		3	3	3	2	2	2	2	2	4	4	4
↓	↓	↓	↓				↓	↓	↓	↓	↓	↓
1	1	1	1	1	1	5	5	5	5	4	4	4
	2	2	2	2	2	2	1	1	1	1	5	5
		3	3	3	3	3	3	2	2	2	2	2
			4	4	4	4	4	4	3	3	3	3

Pamięć wirtualna (21)

Systemy operacyjne



Zagadnienia implementacyjne

- Implementacja algorytmu FIFO
- Implementacja algorytmu LRU
- Algorytmy przybliżające metodę LRU
 - algorytm dodatkowych bitów odwołań
 - algorytm drugiej szansy (FINUFO)
 - ulepszony algorytm drugiej szansy

Pamięć wirtualna (22)

Systemy operacyjne

**Implementacja algorytmu FIFO**

- Utrzymywanie listy numerów stron w kolejności ich sprowadzania do pamięci
- Umieszczanie numeru sprowadzanej strony na końcu listy
- Usuwanie z pamięci (i z listy) strony, której numer znajduje się na początku listy

Pamięć wirtualna (23)

Systemy operacyjne

**Implementacja algorytmu LRU**

- Licznik — przy każdym odniesieniu do pamięci zwiększana jest wartość pewnego licznika i wpisywana do odpowiedniej pozycji opisującej stronę w tablicy stron (lub w innej specjalnej strukturze systemu operacyjnego). Z pamięci usuwana jest wówczas strona z najmniejszą wartością tego licznika, co wymaga przejrzenia całej tablicy stron.
- Stos — numery stron, do których następuje odniesienie, odkładane są na szczycie stosu. Przed odłożeniem na szczycie numer strony musi być wydobyty ze środka stosu, czyli z miejsca, gdzie był ostatnio odłożony. W tej implementacji z pamięci usuwana jest strona, która jest na dnie stosu.

Pamięć wirtualna (24)

Systemy operacyjne



Algorytmy przybliżające metodę LRU

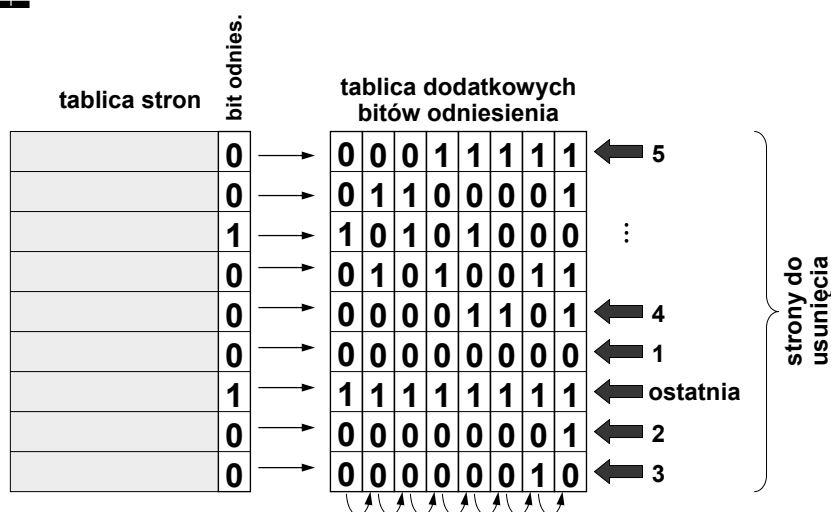
- Niezbędne wspomaganie sprzętowe:
 - bit odniesienia (ang. reference bit) — ustawiany, gdy następuje odniesienie do strony,
 - bit modyfikacji (ang. modify bit) — ustawiany, gdy następuje zapis na stronie.
- Algorytmy korzystające ze wspomaganie sprzętowego:
 - algorytm dodatkowych bitów odniesienia — wykorzystuje bit odniesienia,
 - algorytm drugiej szansy — wykorzystuje bit odniesienia,
 - ulepszony algorytm drugiej szansy — wykorzystuje bit odniesienia i bit modyfikacji.

Pamięć wirtualna (25)

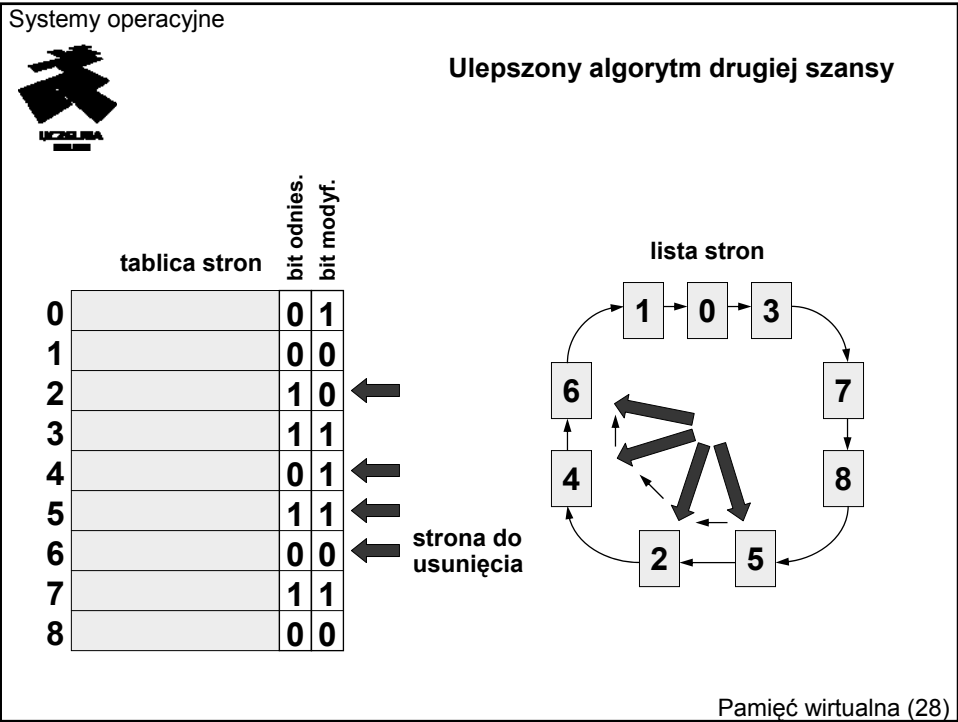
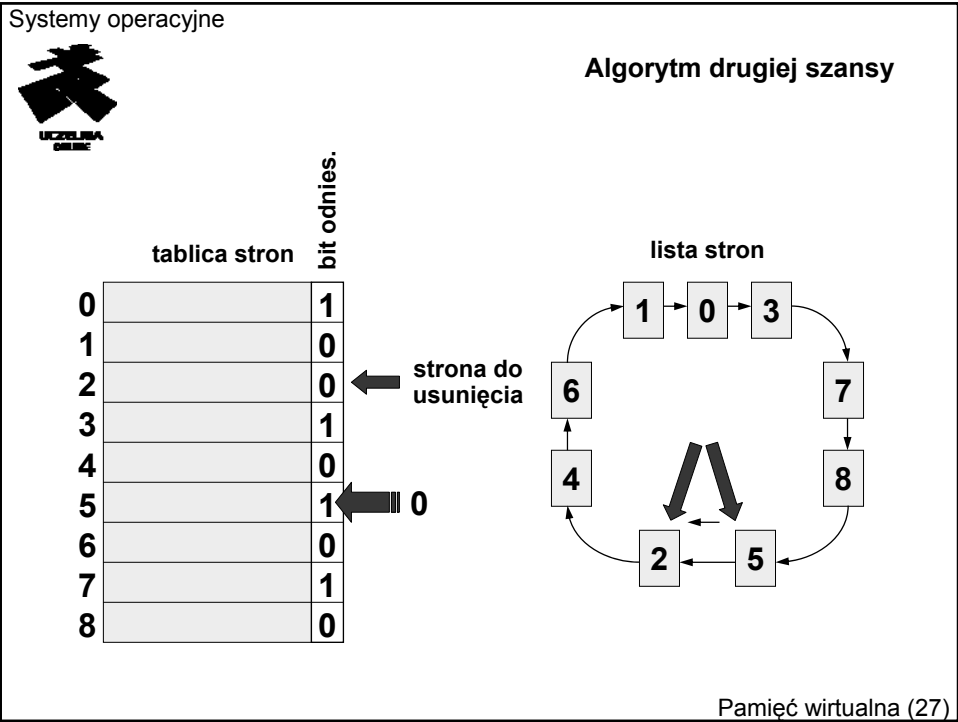
Systemy operacyjne



Algorytm dodatkowych bitów odniesienia



Pamięć wirtualna (26)



Systemy operacyjne

**Algorytmy ze sprowadzaniem na żądanie**

- VMIN — usuwane są strony, których koszt utrzymania w pamięci jest większy od kosztu ponownego sprowadzenia
- WS — usuwane są strony, do których nie było odniesień przez określony czas
- WSClock — przybliżona wersja algorytmu WS, oparta na bicie odniesienia
- PFF, VSWS — przydział i zwalnianie ramek procesów realizowane jest na podstawie częstości zgłaszania błędów strony

Pamięć wirtualna (29)

Systemy operacyjne

**Zbiór roboczy**

- Zbiór roboczy procesu (ang. working-set) — zbiór stron, które zostały zaadresowane w ciągu ostatnich τ odniesień do pamięci (w tzw. oknie zbioru roboczego)
- Okno zbioru roboczego (ang. working-set window) — zakres odniesień do pamięci, które adresują strony należące do zbioru roboczego
- Zbiór roboczy w chwili t przy rozmiarze okna τ oznaczony będzie jako $\mathcal{W}(t, \tau)$
- Algorytm zbioru roboczego usuwa z pamięci wszystkie strony, które nie należą do zbioru roboczego.

Pamięć wirtualna (30)

Systemy operacyjne



Przykład zbioru roboczego

1, 2, 4, 5, 7, 3, 2, 4, 4, 6, 7, 4, 1

$\tau = 4 \left\{ \begin{array}{l} \{3, 4, 5, 7\} = \mathcal{W}(6, 4) \\ \{2, 3, 5, 7\} = \mathcal{W}(7, 4) \end{array} \right.$

$\tau = 6 \left\{ \begin{array}{l} \{2, 3, 4, 6, 7\} = \mathcal{W}(11, 6) \\ \{2, 4, 6, 7\} = \mathcal{W}(12, 6) \end{array} \right.$

Pamięć wirtualna (31)

Systemy operacyjne



Przykład wymiany stron w oparciu o zbiór roboczy

Ciąg odniesień: 1, 2, 4, 5, 7, 3, 2, 4, 4, 6, 7, 4, 1

Rozmiar okna: 5

1 2 4 5 7 3 2 4 6 7 1

	1	1	1	1	1	3	3	3	3	3	7	7
		2	2	2	2	2	2	2	2	2	2	1
			4	4	4	4	4	4	4	4	4	4
				5	5	5	5	5	6	6	6	6
					7	7	7	7				

Pamięć wirtualna (32)

Systemy operacyjne

**Koncepcja identyfikacji zbioru roboczego**

- Dla każdej ramki utrzymywany jest wirtualny (mierzony odniesieniami do pamięci) czas ostatniego odniesienia do niej.
- Po każdym odniesieniu do strony zwiększana jest wartość licznika odniesień i wpisywana do odpowiedniej tablicy na pozycji odpowiadającej ramce, w której znajdują się adresowana strona.
- Do zbioru roboczego należą te strony, dla których różnica pomiędzy bieżącą wartością licznika odniesień, a wartością wpisaną w tablicy jest mniejsza lub równa rozmiarowi okna zbioru roboczego.

Pamięć wirtualna (33)

Systemy operacyjne

**Przybliżona realizacja koncepcji zbioru roboczego**

- Wspomaganie sprzętowe — bit odniesienia.
- Okresowe (wyznaczone przez czasomierz lub przez wystąpienie błędu strony) zwiększanie licznika reprezentującego upływ czasu wirtualnego oraz sprawdzanie bitu odniesienia dla każdej z ramek.
- Jeśli bit odniesienia jest ustawiony, to skasowanie bitu odniesienia i wpisanie bieżącej wartości licznika do odpowiedniej tablicy na pozycji odpowiadającej ramce.
- Jeśli bit odniesienia jest skasowany, to sprawdzenie różnicy pomiędzy bieżącą wartością licznika a wartością wpisaną na odpowiedniej pozycji w tablicy i usunięcie strony, gdy różnica jest większa niż rozmiar okna.

Pamięć wirtualna (34)

Systemy operacyjne

Algorytm WSClock ⁽¹⁾

- Dla każdej ramki (strony w ramce) utrzymywany jest wirtualny czas (przybliżony) ostatniego odniesienia.
- Wszystkie ramki pamięci (niezależnie od przynależności do procesu) powiązane są w cykl.
- W wyniku **wystąpienia błędu strony** sprawdzany jest bit odniesienia do strony wskazywanej jako kolejna do usunięcia.
- Jeśli bit odniesienia jest ustawiony, zostaje on skasowany, po czym następuje wskazanie następnej ramki w cyklu i sprawdzenie bitu odniesienia.

Pamięć wirtualna (35)

Systemy operacyjne

Algorytm WSClock ⁽²⁾

- Jeśli bit odniesienia jest skasowany, sprawdzana jest różnica pomiędzy wirtualnym czasem bieżącym, a czasem ostatniego odniesienia do wskazywanej strony:
 - jeśli różnica jest większa od rozmiaru okna zbioru roboczego, następuje wymiana strony w ramce,
 - w przeciwnym razie strona pozostaje i następuje wskazanie i sprawdzenie ramki następnej.
- W przypadku wykonania pełnego obiegu przez wskazówkę i stwierdzenia braku stron do wymiany następuje zawieszenie jakiegoś procesu.

Pamięć wirtualna (36)

Systemy operacyjne

Przykład działania algorytmu WSClock

czas bieżący: 51
rozmiar okna: 10

	tablica stron	bit odnies.	czas odnies.
0		1	39
1		0	30
2		0	22
3		1	50
4		0	45
5	0	1	48
6		0	33
7		1	49
8		0	41

strona do usunięcia

lista stron

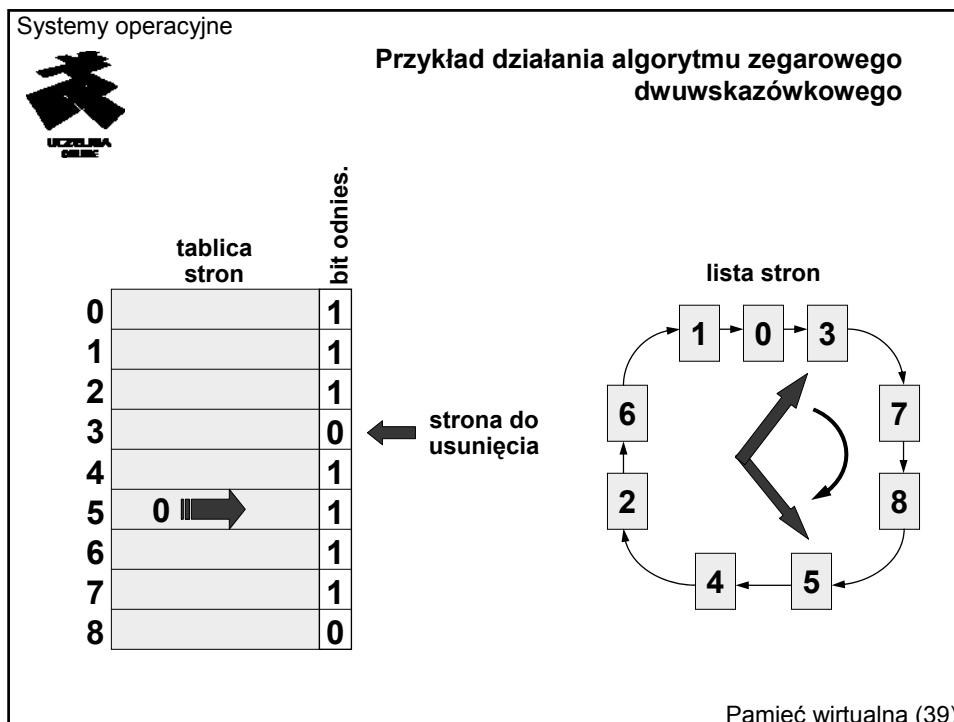
Pamięć wirtualna (37)

Systemy operacyjne

Algorytm zegarowy dwuwskazówkowy

- Wszystkie strony pamięci (niezależnie procesu) powiązane są w listę cykliczną.
- Lista przeglądana jest okresowo przez 2 wskazówki:
 - wiodąca (przednia) zeruje bit odniesienia,
 - zamykająca (tylna) wskazuje stronę do usunięcia.
- Jeśli bit odniesienia przed nadejściem tylnej wskazówki zostanie ponownie ustawiony, strona pozostaje w pamięci, w przeciwnym przypadku jest usuwana.
- Algorytm sterowany jest następującymi parametrami:
 - tempo przeglądania,
 - rozstaw wskazówek.

Pamięć wirtualna (38)



Systemy operacyjne

Algorytm PFF

- Jeśli częstotliwość błędów strony generowanych przez proces przekroczy ustalony poziom f_H , to dla stron tego procesu sprowadzanych do pamięci przydzielana są nowe ramki.
- Jeśli częstotliwość błędów strony spadnie poniżej ustalonego poziomu f_L , to
 - zwalniana jest jedna ramka procesu lub
 - zwalniane są wszystkie ramki ze stronami, do których nie było odniesienia od chwili wystąpienia ostatniego błędu strony w danym procesie.
- W szczególności $f_H = f_L$

Pamięć wirtualna (40)

Systemy operacyjne

**Implementacja algorytmu PFF — kontrola częstości błędów strony**

- Przy każdym błędzie strony generowanym przez proces zwiększany jest licznik błędów strony danego procesu oraz zerowane są bity odniesienia do jego stron.
- W określonych interwałach czasu, wyznaczanych przez czasomierz, sprawdzane są (a następnie zerowane) liczniki poszczególnych procesów.
- Jeśli wartość licznika jest większa od górnej granicy, to procesowi przydzielana jest dodatkowa ramka.
- Jeśli wartość licznika jest mniejsza od ustalonej dolnej granicy, to zwalniana jest:
 - jedna ramka lub
 - wszystkie ramki z wykasowanym bitem odniesienia.

Pamięć wirtualna (41)

Systemy operacyjne

**Implementacja algorytmu PFF — kontrola okresu pomiędzy błędami strony**

- Przy każdym błędzie strony sprawdzany jest czas jaki upłynął od poprzedniego błędu strony.
- Jeśli czas jest mniejszy od ustalonej wielkości T_{min} , to na potrzeby sprowadzanej strony przydzielana jest dodatkowa ramka.
- Jeśli czas jest większy od ustalonej wielkości T_{max} , to
 - zwalniana jest jedna ramka lub
 - zwalniane są wszystkie ramki z wykasowanym bitem odniesienia.
- Bit odniesienia dla stron pozostających w pamięci jest kasowany.

Pamięć wirtualna (42)

Systemy operacyjne

**Algorytm VSWS**

- Przy każdym błędzie strony zwiększany jest licznik błędów strony danego procesu — n .
- Co pewien okres czasu T_t , wynikający ze stanu wymiany, następuje kontrola odniesień do stron.
- W ramach kontroli wykonywane są następujące czynności:
 - strony, do których nie było odniesienia są usuwane (ich ramki są zwalniane),
 - bity odniesienia stron pozostających na następny okres (do których było odniesienie) są kasowane,
 - wartość licznika n jest zerowana.

Pamięć wirtualna (43)

Systemy operacyjne

**Wielkość interwału czasu dla algorytmu VSWS**

- Algorytm sterowany jest następującymi parametrami:
 - T_{max} — maksymalna wielkość interwału czasu,
 - T_{min} — minimalna wielkość interwału czasu,
 - n_{max} — maksymalna liczba błędów strony.
- Kontrola następuje w chwili t , po czasie T_t od momentu poprzedniej kontroli, ustalonym następująco:

Pamięć wirtualna (44)

Systemy operacyjne

**Techniki poprawy efektywności wymiany**

- Wymiana ramek zamiast wymiany stron — przyspieszenie wymiany brudnej strony
- Buforowanie usuniętych stron zamiast usuwania — mniej kosztowna obsługa błędu strony
- Czyszczenie stron — ułatwienie ewentualnej wymiany
- Utrzymywanie listy brudnych stron do usunięcia — opóźnienie zapisu stron w obszar wymiany

Pamięć wirtualna (45)

Systemy operacyjne

**Algorytmy wstępnego stronicowania**

- DPMIN — sprowadzane są wszystkie strony potrzebne w najbliższej przyszłości, które mieszczą się dostępnych ramkach
- OBL — sprowadzana jest strona żądana oraz strona następna (wg. numeracji w tablicy stron)
- SL — na podstawie wcześniejszych odniesień i błędów strony budowana jest tablica skojarzeń stron sprowadzanych po sobie i informacja taka wykorzystywana jest przez następny sprowadzaniu
- FDPA — podejście uwzględniające sugestie programisty co do stron potrzebnych w przyszłości

Pamięć wirtualna (46)

Systemy operacyjne



Algorytm SL

- System utrzymuje dodatkową tablicę $pred$, gdzie dla każdej strony pamiętana jest strona, sprowadzona po niej do pamięci. Początkowo $pred[i] = i + 1$.
- u jest zmienną przechowującą nr strony ostatnio sprowadzanej do pamięci w wyniku odniesienia do niej.
- W chwili wystąpienia błędu strony następuje sprowadzenie żądanej strony — p . Jeśli przy poprzedniej wymianie nie dokonano wstępnego sprowadzenia lub nie było odniesienia do strony wstępnie sprowadzonej do chwili obecnej $pred[u] := p$.
- Jeśli strona $q = pred[p]$ nie znajduje się w pamięci, to następuje również jej sprowadzenia.
- $u := p$

Pamięć wirtualna (47)

Systemy operacyjne

Algorytm FDPA ⁽¹⁾

- W programie występują dodatkowe instrukcje dla systemu operacyjnego:
 - $FREE(p)$ — strona p nie będzie wykorzystywana w niedalekiej przyszłości,
 - $PRE(p)$ — strona p będzie wkrótce wykorzystywana.
- Wykonanie takiej instrukcji skutkuje ustawieniem odpowiednich bitów dla strony:
 - bitu usunięcia — D ,
 - bitu wstępnego sprowadzenia — P .

Pamięć wirtualna (48)

Systemy operacyjne

**Algorytm FDPA⁽²⁾**

- W przypadku wystąpienia błędu strony sprowadzana jest strona żądana, ewentualnie wymieniana z jakąś stroną z ustawionym bitem D . Jeśli nie ma takiej strony stosowany jest inny algorytm usuwania np. LRU, z pominięciem stron, dla których ustawiony jest bit P .
- Jeśli w systemie są wolne ramki lub mogą zostać zwolnione przez usunięcie stron z ustawionym bitem D , to sprowadza się tyle stron z ustawionym bitem P , na ile pozwala potencjalna liczba wolnych ramek.
- W przypadku odniesienia do strony z ustawionym bitem P , bit ten jest kasowany.

Pamięć wirtualna (49)

Systemy operacyjne

**Segmentacja w systemie pamięci wirtualnej**

- Segmentacja na żądanie — wymiana segmentów pomiędzy pamięcią pierwszego i drugiego rzędu
- Skomplikowana wymiana — zróżnicowany rozmiar segmentów
- Mniejsza przewidywalność czasu dostępu — zróżnicowany czas przesyłania zawartości segmentu

Pamięć wirtualna (50)