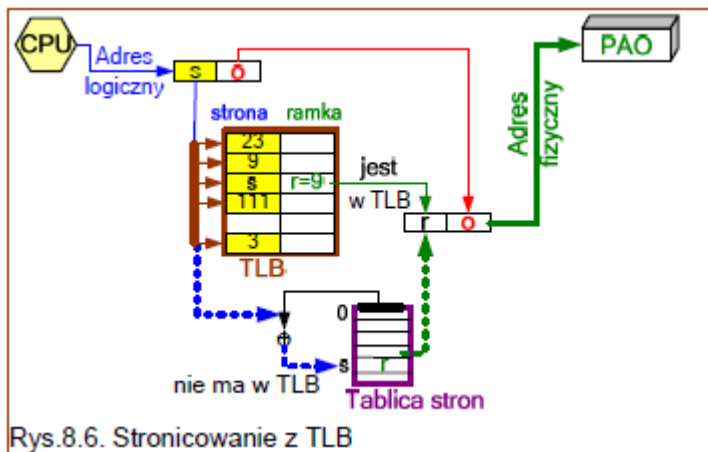


## 1. Tablica Stron i Odwrócona Tablica Stron – omówić różnice

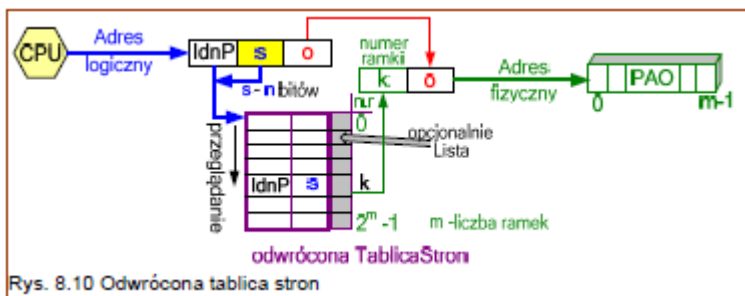
Tablica stron zawiera jedną pozycję (adres ramki) dla każdej strony wirtualnej procesu. Jest ona uporządkowana wg. adresów wirtualnych, więc adres fizyczny jest dostępny natychmiast. Używają one PAO (pamięci operacyjnej) tylko do pokazania jak użytkowany jest inny blok PAO.

Odwrócona tablica stron ma po jednej pozycji dla każdej ramki PAO odwzorowującej stronę. Indeksuje się ona numerem ramki, a nie na podstawie wirtualnego numeru strony. Taka struktura zmniejsza rozmiar pamięci potrzebnej do pamiętania wszystkich TablicStron, zwiększa jednak czas przeszukiwania Odwróconej Tablicy stron przy odwołaniu do strony, gdyż jest ona uporządkowana wg. Adresów fizycznych.

### Tablica Stron



### Odwrócona Tablica Stron



2.

3. Podać metodę wyznaczania efektywnego czasu dostępu do PAO i omówić jego składniki

Efektywny czas dostępu do PAO jest określany na podstawie:

- a - prawdopodobieństwa odnalezienia numerów stron w rejestrach asocjacyjnych.
- b- czasu potrzebnego na przeglądnięcie rejestrów asocjacyjnych
- c - czasu potrzebnego na dostęp do PAO
- d - czasu na sięgnięcie do Tablicy Stron i numer ramki (w przypadku NIE znalezienia numerów stron w rejestrach asocjacyjnych)

Efektywny czas dostępu do PAO oblicza się następująco

$$a * (b + c) + (1.0 - a) * (b + c + d)$$


jeśli więc  $a = 80\%$ ,  $b = 20 \text{ ns}$ ,  $c = d = 100 \text{ ns}$ , to wtedy:

$$0,8 * (20 \text{ ns} + 100 \text{ ns}) + (1,0 - 0,8) * (20 \text{ ns} + 100 \text{ ns} + 100 \text{ ns}) = 96 \text{ ns} + 44 \text{ ns} = 140 \text{ ns}$$

#### 4. Omówić metody identyfikacji obszarów wolnych na dysku

##### a) Wektor Bitowy

wektor bitowy implementuje liste wolnych obszarów na dysku. Każdy blok reprezentuje 1 bit. Dla bloku wolnego bit ma wartość 1, dla przydzielonego wartość 0. Metoda sprawdza kolejne słowa (8 bitów) w mapie bitowej (ciągu 0 i 1) czy są równe 0. Następnie w pierwszym niezerowym sioie szuka pierwszego bitu 1, który określa początek wolnego miejsca w pamięci.

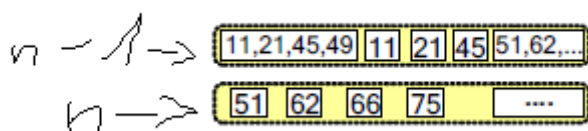
Wygląd mapy wolnych obszarów: 

##### b) Lista Wiązana

Można powiązać ze sobą wszystkie wolne bloki dyskowe i przechowywać wskaźnik do pierwszego wolnego bloku na dysku. Metoda nie jest wydajna, ponieważ przeglądanie listy wymaga odczytanie każdego bloku pamięci. Przeglądanie listy nie jest jednak wykonywane często, ponieważ SO najczęściej potrzebuje pierwszego wolnego bloku aby przydzielić go plikowi, więc pobiera pierwszy blok z listy.

##### c) Grupowanie

Można przechowywać adresy  $n$  wolnych bloków w pierwszym wolnym bloku. Pierwsze bloki ( $n-1$ ) to bloki rzeczywiście wolne. Ostatni z nich zawiera adresy kolejnych  $n$  wolnych bloków. Metoda skutecznie odnajduje adresy przy dużej licznie wolnych bloków.



##### d) Zliczanie

Zamiast wykazu  $n$  wolnych adresó można przechowywać adres pierwszego wolnego bloku pamięci, oraz liczbę  $m$  wolnych bloków następujących bezpośrednio po nim. Każda pozycja na takim wykazie składa się z adresu dyskowego i licznika. Każdy wpis zajmuje więcej miejsca niz zwykły adres dyskowy, lecz cały wykaz będzie krótszy jeśli licznik  $> 1$

11. Przydziałowi CPU moze towarzyszyć "efekt konwoju", wyjaśnić zjawisko i okoliczności powstawania.

Efekt konwoju towarzyszy przydziałowi CPU wtedy kiedy dla dużego procesu  $P_x$  wykorzystującego duży przydział CPU istnieje kilka mniejszych procesów  $P_{y1}, P_{y2}, P_{yn}$ , które mają mniejszy przydział. W czasie kiedy  $P_x$  uzyskuje przydział CPU,  $P_y$  kończą swoje operacje  $W_e/W_y$  i przechodzą do kolejki procesów Gotowych. Po pewnym czasie  $P_x$  kończy faze CPU i rozpoczyna  $W_e/W_y$ , a w tym czasie mniejsze procesy wykonują szybko faze CPU. Sytuacja następnie się powtarza po kolejnym przejściu  $P_x$  do fazy CPU. W ten sposób zmniejszono wykorzystanie zasobów, azizeli pozwolono by pracować najpierw krótszym procesom.

6. Przedstawić algorytm synchronizacji procesów, wykorzystujący środki sprzętowe

TAKI CHUJ

10.