

Zarządzanie pamięcią wirtualną metodą segmentowania

Kamil Osak

Zarządzanie pamięcią wirtualną metodą segmentowania

Krótki opis mechanizmu:

1. W momencie, gdy procesor zapyta pamięć wirtualną o konkretny program to pamięć wirtualna przeszukuje swoje segmenty w celu odnalezienia.
2. Jeśli program nie znajduje się w pamięci wirtualnej generowane jest przerwanie systemowe, powiadamiające system operacyjny o konieczności jego załadowania - w trakcie może okazać się konieczność przeniesienia innych segmentów do pamięci operacyjnej lub na dysk.
3. Załadowany program zostaje zwrócony.

Przykład 1

Problem: Użytkownik tworzy plik o nazwie 4-literowej np.: plik.txt

- a. procesor otrzymuje rozkaz wykonania pliku,
- b. odwołanie do pamięci wirtualnej,
- c. pamięć wirtualna znajduje rozkaz na swojej liście rozkazów,
- d. pamięć wirtualna zwraca instrukcje dla rozkazu

Przykład 2

Problem: Użytkownik tworzy katalog o nazwie 4-literowej np.: home

- a. procesor otrzymuje rozkaz wykonania katalogu,
- b. odwołanie do pamięci wirtualnej,
- c. pamięć wirtualna nie znajduje rozkazu na swojej liście. Generowane zostaje przerwanie informujące o konieczności załadowania rozkazu (jeśli zabraknie miejsca na segment zostanie zwolniony inny segment),
- d. pamięć wirtualna zwraca instrukcje dla rozkazu.

Tablica segmentów

Procesor odczytuje deskryptor segmentu z tablicy segmentów, który zawiera flagę informującą, czy dany segment znajduje się aktualnie w pamięci czy nie, adres początku segmentu oraz jego długość.

Adres segmentu

Adres logiczny danego segmentu składa się z dwóch części: numeru segmentu oraz przesunięcia w obrębie tego segmentu.

Zwalnianie segmentu

W implementacji modułu będą używane tylko 2 segmenty - pierwszy do programów, a drugi do rozkazów. Segment zostanie zwolniony, jeżeli znajdzie potrzeba zapisania w jego miejsce innego. Wówczas zmieni się rozmiar danego segmentu - rozmiar zależy od rozkazu wpisanego w segment.

Podstawowe metody

- a. `vm_find(numer_segmentu)` - metoda, która zwraca instrukcje dla utworzonego i uruchomionego procesu. Pamięć wirtualna w pierwszej kolejności przeszuka swoje segmenty, a dopiero następnie pamięć operacyjną i pamięć fizyczną,
- b. `vm_erase()` - metoda, która pozwala usunąć zawartość pamięci wirtualnej,
- c. `vm_display()` - metoda, która wyświetla aktualny stan pamięci wirtualnej.