

Jak system wykonuje zadania?

- Przetwarzanie wsadowe, monitor prosty, praca pośrednia
- Buforowanie, Spooling
- Wieloprogramowość
- Wielozadaniowość, wielodostępowość, systemy z podziałem czasu
- Wieloprocessorowość, systemy HPC
- Systemy rozproszone
- Systemy czasu rzeczywistego

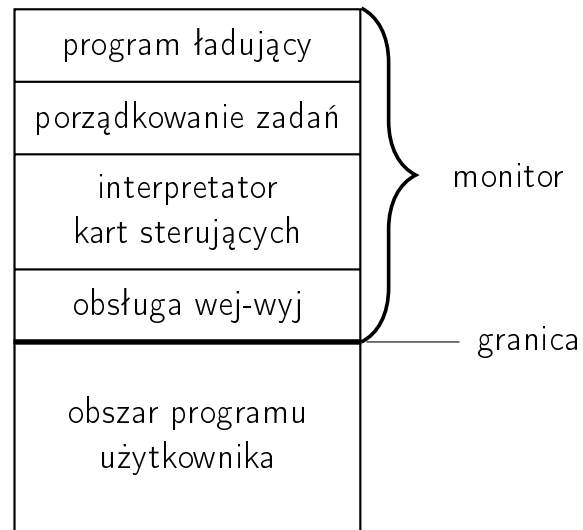
Monitor prosty

W czasie instalowania taśm jednostka centralna była bezczynna. Komputery były drogie i ich czas był cenny.

Rozwiązanie:

- zatrudnienie profesjonalnych operatorów
- **przetwarzanie wsadowe** (*batch processing*) - zadania gromadzone (przez operatora) w postaci wsadów są kolejno wykonywane, bez interakcji użytkownika
- automatyczne porządkowanie zadań (*automatic job sequencing*)
- **monitor rezydujący** – stale obecny w pamięci, automatyczne przekazywanie sterowania od zadania do zadania; karty sterowania zadaniami (JCL, *Job Control Language*)

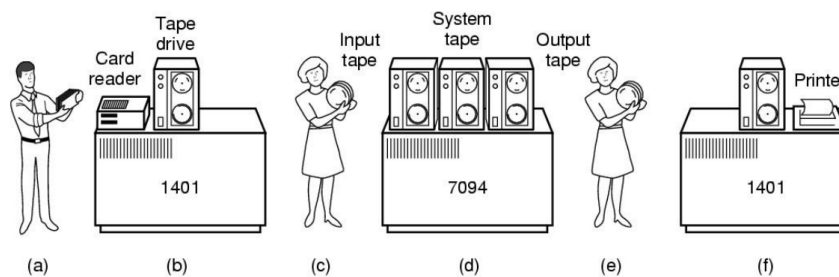
Monitor rezydujący w pamięci



- ochrona pamięci - program nie może ingerować w obszar pamięci monitora
- uprzywilejowane instrukcje - pewne instrukcje mogą być wykonywane wyłącznie przez monitor (tryb jądra)

Praca pośrednia (off-line)

- Wolne operacje wej-wyj wykonywane przez mniejsze, satelitarne minikomputery.
- Jednostka główna czyta dane z taśmy magnetycznej i umieszcza wyniki na taśmie magnetycznej.
- Obsługa czytników kart i drukarek wierszowych w trybie pośrednim (*off-line*).
- Niezależność od urządzeń wej-wyj – programy pisane z myślą o korzystaniu z logicznych, a nie fizycznych urządzeń peryferyjnych.



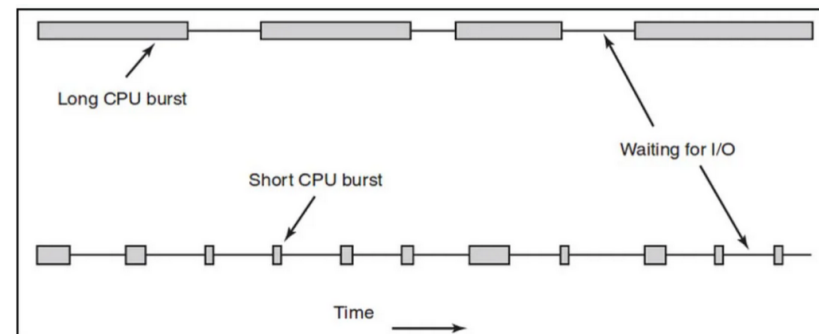
Buforowanie

Buforowanie jest metodą usprawniającą wykonywanie obliczeń i operacji wej-wyj dla danego zadania

- dane są tymczasowo przechowywane w pamięci (buforze) przed ich dalszym przetwarzaniem lub przesyłaniem
- buforowanie zmniejsza liczbę operacji I/O, przyspiesza dostęp do danych
- procesor może pracować nad innymi zadaniami, podczas gdy dane są ładowane do bufora. To pozwala na lepsze wykorzystanie zasobów procesora i zmniejsza czas oczekiwania na dane.

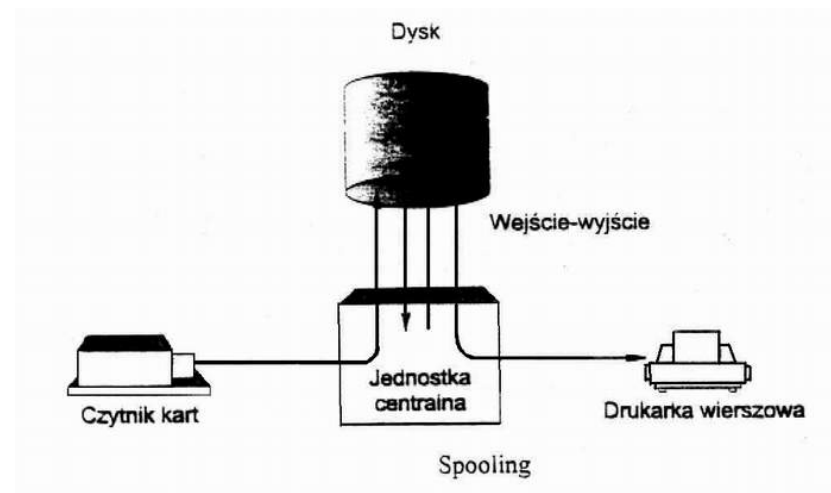
Zadania ograniczone CPU i ograniczone I/O

- Zadania ograniczone operacjami wej-wyj (*I/O bound*)
 - prędkość przetwarzania zależna od szybkości urządzeń wej-wyj.
 - buforowanie może zwiększyć przepustowość poprzez grupowanie wielu operacji I/O
- Zadania ograniczone procesorem (*CPU bound*)
 - prędkość przetwarzania ograniczona przez szybkość CPU (bufor wejściowy zawsze pełny, bufor wyjściowy zawsze pusty)
 - buforowanie zwiększa wydajność gromadząc często używane dane i zmniejszając czas dostępu



Spooling

- **spooling** (*Simultaneous Peripheral Operation On-Line*) jednoczesna bezpośrednia praca urządzeń
- dane są przechowywane w plikach na dysku w celu ich późniejszego przetwarzania przez urządzenia peryferyjne (np. drukarki)
- spooling umożliwia równoczesne wykonywanie operacji wej-wyj jednego zadania i obliczeń dla innych zadań znajdujących się we wsadzie
- spooling wytwarza pulę zadań do wykonania i umożliwia planowanie zadań i wieloprogramowość



Wieloprogramowość

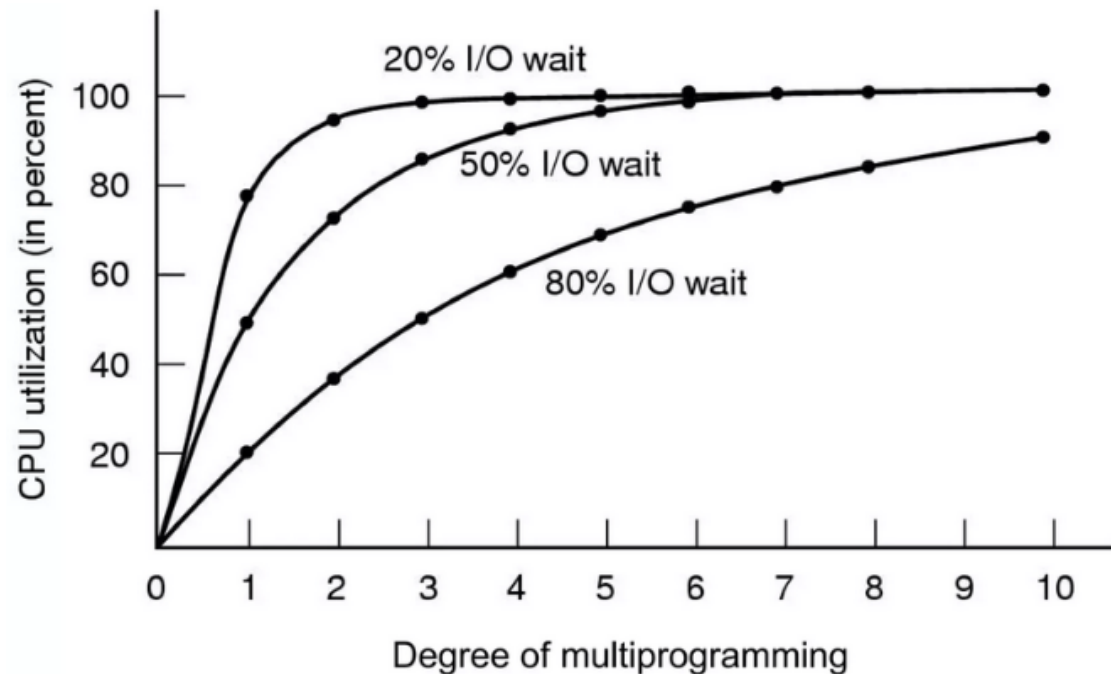
- praca pośrednia, buforowanie, spooling mają swoje ograniczenia. Jeden użytkownik nie jest w stanie angażować stale urządzeń wej-wyj i jednostki centralnej!
- **wieloprogramowość** - w pamięci rezyduje więcej niż jeden program użytkownika, w momencie zablokowania (operacja I/O) CPU jest przełączany do innego zadania
- realizacja wieloprogramowości wymaga skomplikowanego systemu operacyjnego: ochrona zadań między sobą i planowanie przydziału procesora, kolejki, partycje o stałych lub zmiennych wielkościach



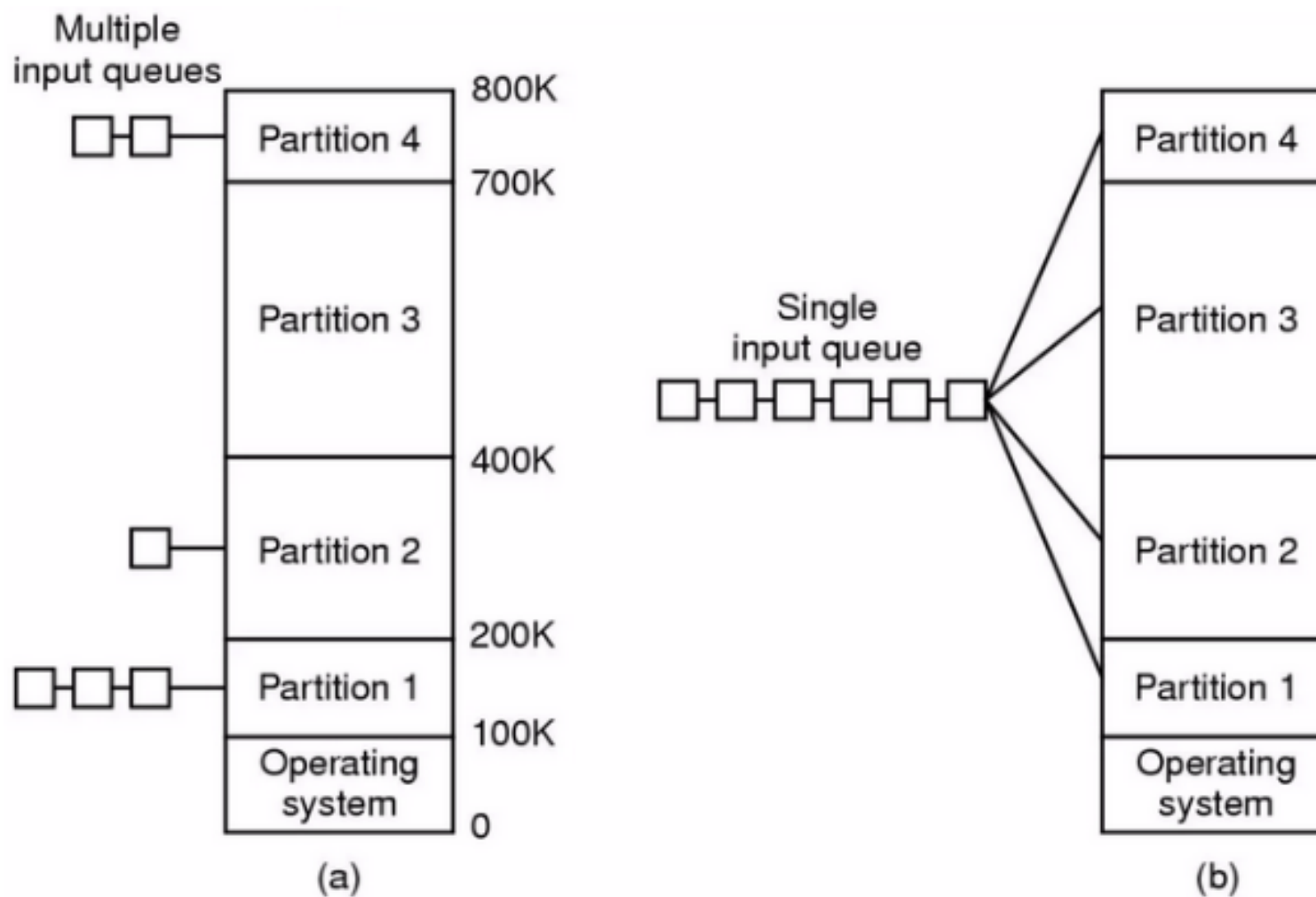
Wieloprogramowość: wykorzystanie CPU

Prosty model wykorzystania CPU:

- n -procesów, każdy spędza część p swego czasu w stanie I/O
- p^n – prawdopodobieństwo, że n -procesów będzie w stanie I/O
- wykorzystanie CPU = $1 - p^n$



Wieloprogramowość: wykorzystanie pamięci



Wieloprogramowość ze stałym podziałem pamięci

a.) Osobne kolejki zadań do każdej partycji

- zadanie trafia do kolejki do najmniejszej pasującej partycji
- duże partycje mogą pozostawać puste, nawet gdy są zadania w innych kolejkach, które mogłyby się zmieścić
- małe zadania muszą czekać

b.) Wspólna kolejka do wszystkich segmentów:

- zadanie z początku kolejki trafia do pierwszej wolnej partycji
- może powodować marnowanie pamięci na małe zadania
- potrzebny alg. wyszukiwania zadań w kolejce pasujących do wolnego obszaru - ale to dyskryminuje małe zadania

Wieloprogramowość z dynamicznym podziałem pamięci

- obszary tworzone dynamicznie dopasowane do rozmiaru zadania
- brak fragmentacji wewnętrznej - efektywne wykorzystanie pamięci RAM
- konieczność wykonywania upakowań (defragmentacji) w celu uniknięcia fragmentacji zewnętrznej (niewykorzystane obszary pomiędzy zaalokowanymi blokami), co zwiększa użycie CPU

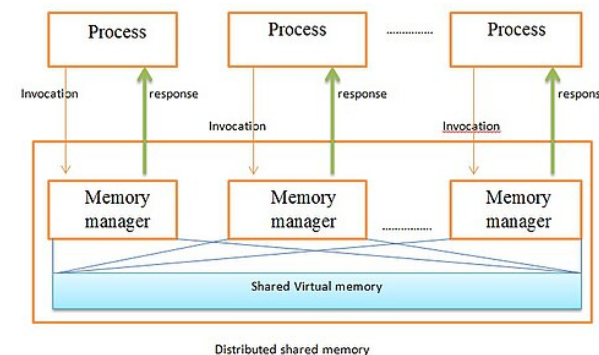
Obecnie stosowane rozwiązanie: pamięć wirtualna, segmentacja o stałym rozmiarze i mechanizm stronicowania

Systemy z podziałem czasu

- praca wsadowa a praca interakcyjna
- wieloprogramowość → **wielozadaniowość**, wykonywanie wielu procesów w określonym przedziale czasu, wymaga mechanizmu podziału czasu
- system z podziałem czasu – wielu użytkowników dzieli jeden komputer, CPU jest przydzielane procesom w małych porcjach czasu (kwantach), gdy kwant się kończy to CPU przedzielane innemu procesowi
- systemy z podziałem czasu są skomplikowane, gdyż wymagają:
 - realizacji mechanizmów działań współbieżnych
 - zarządzania pamięcią
 - ochrony pamięci
 - planowania przydziału CPU
 - administrowania pamięcią dyskową
 - systemu plików dostępnych bezpośrednio

Systemy HPC (*High Performance Computing*)

- **wieloprocusowość** (*multiprocessing*) - systemy posiadające więcej niż jeden procesor pozwalające na równoczesne przetwarzanie zadań
- **SMP** (*Symmetric MultiProcessing*)
 - współdzielona pamięć, I/O
 - wspólny system operacyjny
 - jednostki CPU są równoważne
- **DSM** (*Distributed Shared Memory*)
 - rozdzielone fizycznie pamięci są adresowane wspólną przestrzenią adresową
 - tańszy niż SMP i dobrze się skaluje
 - pozwala na obsługę dużych baz danych
 - wolniejszy dostęp do danych od SMP



Systemy HPC (*High Performance Computing*)

- **MPP** (*Massively Parallel Processor*)
 - setki/tysiące procesorów, rozproszona pamięć
 - każdy CPU posiada własny system operacyjny, własną pamięć
 - podsystemy komunikują łączami o dużej szybkości
- **klastry obliczeniowe**
 - zrównoleglenie przez węzły (liczba węzłów > liczba CPU)
- **konstelacje obliczeniowe**
 - zrównoleglenie przez wielorpoceowość (liczba CPU > liczba węzłów)

Zalety systemów wieloprocessorowych

- podział zasobów
- przyspieszenie obliczeń
- niezawodność
- komunikacja

Rygorystyczne systemy czasu rzeczywistego

- sterowniki urządzeń o ściśle określonym zastosowaniu: nadzorowanie procesów produkcyjnych, eksperymentów naukowych, kierowanie sygnalizacją świetlną, autopilot, itp.
- działanie podlega ostrym rygorom czasowym, nawet kosztem efektywnego wykorzystania systemu

Łagodne systemy czasu rzeczywistego

- działanie podlega złagodzonym rygorom czasowym
- niespełnienie wymagań czasowych nie jest niebezpieczne ale powoduje pogorszenie jakości usług świadczonych przez system
- zastosowania: techniki multimedialne, tworzenie wirtualnej rzeczywistości, urządzenia zdolne do samodzielnej eksploracji