

Systemy komputerowe

Lista zadań nr 11

Na ćwiczenia 25. i 26. maja 2022

Zadanie 1. Wyjaśnij różnice pomiędzy planowaniem wywłaszczającym (ang. *preemptive scheduling*) a planowaniem niewywłaszczającym (ang. *non-preemptive scheduling*). Posłuż się diagramem stanów procesu. Zdefiniuj pojęcia: proces ograniczony przez dostęp do procesora (ang. *CPU-bound process*), proces ograniczony przez wejście-wyjście (ang. *Input-Output bound process*), proces interaktywny, proces wsadowy (ang. *batch process*). Planowanie wywłaszczające jest trudniejsze w implementacji niż niewywłaszczające, a jednak dominuje we współczesnych systemach operacyjnych. Dlaczego?

Zadanie 2. Zdefiniuj miary efektywności algorytmów planowania procesów i uzasadnij ich sensowność. Dla każdej z nich zastanów się, czy każdy algorytm optymalizujący tą miarę optymalizuje również pozostałe.

Zadanie 3. Rozważmy system w którym pracują procesy o podanych w tabelce długościach faz procesora oraz priorytetach (większa liczba oznacza wyższy priorytet).

<u>Process</u>	<u>Burst Time</u>	<u>Priority</u>
P_1	2	2
P_2	1	1
P_3	8	4
P_4	4	2
P_5	5	3

Procesy zostały wprowadzone do systemu w chwili 0 w kolejności P_1, P_2, P_3, P_4, P_5 .

- Narysuj cztery diagramy Gantta, ilustrujące wykonanie tych procesów przy użyciu algorytmów planowania: FCFS, SJF, priorytetowego bez wywłaszczeń oraz karuzelowego dla $Q = 2$.
- Jaki jest czas cyklu przetwarzania (ang. *turnaround time*) dla czterech algorytmów z poprzedniego punktu?

- c) Oblicz czas oczekiwania dla każdego procesu oraz średni czas oczekiwania w każdym z wariantów z punktu a)

Zadanie 4. Odpowiedz na poniższe pytania dotyczące planowania karuzelowego (ang. *round-robin scheduling*).

1. W jaki sposób można zasymulować planowanie priorytetowe w tym algorytmie? Pamiętaj, że kwant czasu ma ustaloną, niezmienną długość.
2. Załóżmy, że średnia długość fazy procesora (ang. *CPU burst time*) dla procesów w systemie wynosi T , czas zmiany kontekstu to S , a rozmiar kwantu czasu to Q . Oszacuj efektywność wykorzystania procesora dla: a) $Q = \infty$, b) $Q > T$, c) $S < Q < T$, d) $Q = S$, e) Q bliskie 0. Pamiętaj, że czas poświęcony na zmianę kontekstu jest z punktu widzenia efektywności wykorzystania procesora stracony – w tym czasie nie wykonuje się żaden proces.

Zadanie 5. Załóżmy tym razem, że długość kwantu czasu w algorytmie karuzelowym można dowolnie zmieniać podczas działania systemu. Co można powiedzieć o algorytmie, w którym domyślną wartością kwantu jest Q_0 , a każdy proces którego bieżąca faza procesora miała długość P , gdzie $P < Q_0$, będzie w momencie kolejnego wyboru go przez algorytm uruchomiony z kwantem czasu $Q_0 - P$? W tym algorytmie dla $P = Q_0$, długością następnego kwantu czasu dla procesu będzie po prostu Q_0 .

Zadanie 6. Zdefiniuj przybliżenie algorytmu SJF używające techniki wykładniczego uśredniania oraz jego wariant wyłuszczający – SRT. W jaki sposób ten algorytm będzie się zachowywał w następujących przypadkach:

- a) $\alpha = 0$ i $\tau_0 = 100$ milisekund
- b) $\alpha = 0.99$ i $\tau_0 = 10$ milisekund

Zadanie 7. Zdefiniuj algorytm planowania procesów z użyciem wielopoziomowych kolejek ze sprzężeniem zwrotnym (ang. *multilevel feedback queues*). W jaki sposób taki algorytm obsługuje różne klasy procesów (interaktywne, czasu rzeczywistego, wsadowe)? Czemu służą reguły promocji i degradacji procesów pomiędzy kolejkami?

Zadanie 8 (2pkt). Zaprezentuj klasyczny Uniksowy algorytm planowania procesów.

Wskazówka: Przeczytaj podrozdziały 8.1.1 – 8.1.4 książki "Maurice J. Bach. 1986. The design of the UNIX operating system. Prentice-Hall, Inc., USA" zamieszczonej w SKOSie.

Zadanie 9. Zaprezentuj algorytm planowania ze sprawiedliwym podziałem (ang. *fair share scheduler*), będący wariantem algorytmu z poprzedniego zadania.

Wskazówka: Przeczytaj podrozdział 8.1.5 książki wymienionej we wskazówce do zadania 8.