

Algorytm Hu

minimalizacja długości uszeregowania dla procesorów identycznych zadania zależne, niepodzielne

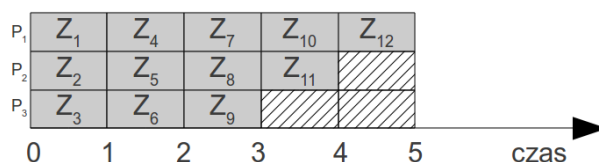
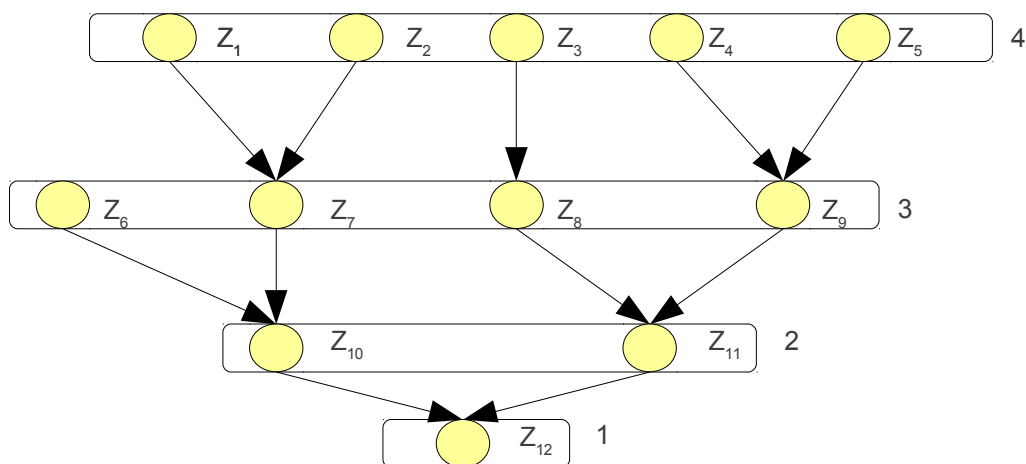
Założenia:

- wszystkie zadania mają równe czasy wykonania
- zbiór zadań tworzy digraf typu drzewo

Zacznijmy od podania algorytmu dla antydrzewa. Algorytm ten składa się z następujących kroków.

m - liczba procesorów
 n - liczba zadań

1. Określ poziomy zadań.
2. Jeśli liczba zadań bez poprzedników jest mniejsza lub równa m , to przydziel tym zadaniom procesory i przejdź do kroku 3. W przeciwnym przypadku wybierz spośród nich m zadań o najwyższych poziomach i przydziel im procesory (spośród zadań o równym poziomie wybór jest dowolny)
3. Usuń wybrane zadania z grafu. Powtarzaj krok 2 dopóty, dopóki wszystkie zadania nie zostaną wykonane.



Przykład zastosowania algorytmu w przypadku trzech procesorów

Złożoność tego algorytmu jest $O(n)$, co wynika z liczby łuków w grafie typu drzewa. Aby wykorzystać powyższy algorytm w przypadku grafu typu lasu składającego się z antydrzew, należy do takiego grafu skonstruować odpowiadający mu graf typu drzewa przez dodanie zadania pustego będącego następnikiem wszystkich zadań w grafie. Aby rozwiązać problem szeregowania w przypadku grafu typu drzewa należy w danym grafie odwrócić zwroty wszystkich łuków i zastosować algorytm, a otrzymane uszeregowanie czytać od tyłu do przodu.