

Sebastian Kamraj 235273
Tomasz Dybowski 235885
Prowadzący: mgr inż. Teodor Niżyński

Sterowanie procesami dyskretnymi

Algorytm NEH

Proc\Masz	I	II
1	1	4
2	2	2
3	3	8
4	10	1
5	5	7
6	2	7
7	2	2
8	4	4
9	5	5

4 procesy:

NEH execution time : 0.170 s; Optymalny Cmax algorytmu NEH to: 17 ; 2 1 3 4

Przegląd zupełny: 0.000545 s; Optymalny Cmax to 14; 1 3 2 4

5 procesów:

NEH execution time : 0.182 s; Optymalny Cmax algorytmu NEH to: 22; 1 2 3 5 4

Przegląd zupełny: 0.002417 s; Optymalny Cmax to 20; 1 4 3 2 5

6 procesów:

NEH execution time : 0.213 s; Optymalny Cmax algorytmu NEH to: 25; 6 1 2 3 5 4

Przegląd zupełny: 0.01764 s; Optymalny Cmax to 21; 1 4 6 5 3 2

7 procesów:

NEH execution time : 0.273 s; Optymalny Cmax algorytmu NEH to: 27; 6 1 7 2 3 5 4

Przegląd zupełny: 0.127022 s; Optymalny Cmax to 23; 1 4 7 6 3 2 5

8 procesów:

NEH execution time : 0.273 s; Optymalny Cmax algorytmu NEH to: 31; 6 1 7 2 3 8 5 4

Przegląd zupełny: 0.96246 s; Optymalny Cmax to 27; 1 2 3 4 6 5 7 8

9 procesów:

NEH execution time : 0.516 s; Optymalny Cmax algorytmu NEH to: 37; 6 7 2 1 8 9 3 5 4

Przegląd zupełny: 9.2562 s; Optymalny Cmax to 27; 1 2 3 4 6 5 8 7 9

NEH computational complexity $\rightarrow O(n^2)$

Przegląd zupełny $\rightarrow O(2^n)$

Porównując czasy wykonywania algorytmu NEH z przeglądem zupełnym jesteśmy w stanie zauważyć, że dla małej ilości procesów przegląd zupełny szybciej radzi sobie ze znalezieniem najbardziej optymalnego rozwiązania niż NEH, natomiast dla większej ilości procesów tj. powyżej 8 zauważamy znaczną różnicę w czasie wykonywania, w tym wypadku przegląd zupełny staje się wręcz bezwartościowy. Dzieje się tak, ponieważ złożoność obliczeniowa algorytmu NEHa wynosi n^2 , a przeglądu zupełnego 2^n . W celu poprawy czasu wykonywania programu zastosowaliśmy tablicę, która zapamiętuje konkretne stany między kolejnymi iteracjami.