**Przedmiot:** Sterowanie Procesami Dyskretnymi - laboratorium
**Imię i nazwisko:** Igor Jewiarz 263478
**Termin zajęć:** środa 18:55
**Data oddania:** 17.04.2023

# *Projekt 4: Algorytm Schrage z i bez podziału zadań*

## 1 Opis problemu

Problem dotycz omawianego już zagadnienia RPQ $(1|r_i, q_i|C_{max})$. Każde zadanie ma trzy parametry: czas dostarczenia, czas trwania oraz czas stygnięcia. Rozwiązaniem problemu jest znalezienie taiej permutacji zadań, dzięki której otrzyma się jak najmniejszy całkowity czas skończenia wszystkich zadań.

$$(r_i, p_i, q_i)$$

- $r_i$ - czas dostarczenia
- $p_i$ - czas trwania
- $q_i$ - czas stygnięcia

## 2 Algorytm Schrage

### 2.1 Bez wywłaszczeń

Algorytm Schrege jest jednym z algorytmów, rozwiązujący problem. Używa on dwóch zbiorów:

- Zbiór N - znajdują się w nim nieuszeregowane jeszcze zadania
- zbiór G - znajdują się w nim dostarczone (gotowe do realizacji) zadania

Algorytm wyznacza z puli dostępnych zadań te o największym czasie stygnięcia i podaje je na maszynę. Proces powtarza się aż do przejścia wszystkich zadań przez maszynę.

1. $t = 0, k = 0, C_{\max} = 0, G = \varnothing, N = \{1,2,\ldots,n\}$,
2. **Dopóki** $((G \neq \varnothing)$ **lub** $(N \neq \varnothing))$ **wykonaj**
3. **Dopóki** $((N \neq \varnothing)$ **oraz** $(\min_{j \in N} r_j \leq t))$ **wykonaj**
4. $e = \arg\min_{j \in N} r_j$, $G = G \cup \{e\}$, $N = N \setminus \{e\}$.
5. **Jeżeli** $G = \varnothing$ **wykonaj**
6. $t = \min_{j \in N} r_j$, **idź do** 3.
7. $e = \arg\max_{j \in G} q_j$, $G = G \setminus \{e\}$,
8. $k = k+1$, $\pi(k) = e$, $t = t + p_e$, $C_{\max} = \max(C_{\max}, t + q_e)$.

Rysunek 1: Pseudoko bez wywłaszczeń

Gdzie:

- $\pi$ - permutacja wykonania zadań na maszynie
- $C_{max}$ - całkowity czas trwania zadań
- $t$ - czas
- $k$ - pozycja w permutacji$\pi$
- $N$ - zbiór zadań nieuszeregowanych
- $G$ - zbiór zadań gotowych do realizacji

### 2.2 Z wywałaszczeniami

Ten wariant algorytmu korzysta z wywałaszczeń. W momencie, w którym pojawia się zadanie gotowe do realizacji z większym czasem stygięcia $q_i$, zadanie jest przerywane i zwracane do zbioru zadań gotowych z odpowiednio mniejszym czasem wykonywania. Następnie wykonywane jest dostarczone zadanie z większym priorytetem.

1. $t = 0,\ C_{\max} = 0,\ G = \varnothing,\ N = \{1,2,\ldots,n\},\ l=0,\ q_0=\infty.$
2. **Dopóki** $((G \neq \varnothing)$ **lub** $(N \neq \varnothing))$ **wykonaj**
3. **Dopóki** $((N \neq \varnothing)$ **oraz** $(\min_{j \in N} r_j \leq t))$ **wykonaj**
4. $e = \arg\min_{j \in N} r_j,\ G = G \cup \{e\},\ N = N \setminus \{e\},$
5. **jeżeli** $q_e > q_l$ **to** $p_l = t - r_e,\ t = r_e,$ **jeżeli** $p_l > 0$ **to** $G = G \cup \{l\}.$
6. **Jeżeli** $G = \varnothing$ **wykonaj**
7. $t = \min_{j \in N} r_j,$ **idź do** 3.
8. $e = \arg\max_{j \in G} q_j,\ G = G \setminus \{e\},$
9. $l = e,\ t = t + p_e,\ C_{\max} = \max(C_{\max}, t + q_e).$

Rysunek 2: Pseudoko z wywłaszczeniami

Gdzie dodatkowo:

- $l = 0$ - zadanie zerowe
- $q_o = \infty$ - czas stygnięcia zadania zerowego - zapobiega przerwaniu pierwszego zadania przez algorytm

## 3 Wnioski

1. Algorytm Schrage z wywłaszczeniami daje lepsze wyniki niż bez wywłaszczeń.
2. Modyfikacja podstawowego Algorytmu Schrage'a w celu dodania podziału zadań nie sprawia większych problemów. Powinna być więc stosowana w przypadkach, w których zadania mogą być przerwane.