Dana jest pojedyncza maszyna/procesor oraz zbiór $J=\{1,\ldots,j,\ldots n\}$ $n$ niepodzielnych zadań. Każde z zadań j charakteryzuje się czasem wykonywania $p_j$, terminem dostępności $r_j$ oraz czasem stygnięcia $q_j$. Należy tak uszeregować zadania, aby czas zakończenia wykonywania wszystkich zadań był jak najkrótszy (minimalizujemy tzw. długość uszeregowania – makespan).
Rozwiązanie może być zapisane jednoznacznie za pomocą permutacji $\pi=\{\pi(1), \pi(2),\ldots, \pi(j),\ldots, \pi(n)\}$, gdzie $\pi(j)$ oznacza zadanie uszeregowane na pozycji j w permutacji $\pi$.
Czas wykonywania zadania $\pi(j)$ to $p_{\pi(j)}$
Termin dostępności zadania $\pi(j)$ to $r_{\pi(j)}$
Czas stygnięcia zadania $\pi(j)$ to $q_{\pi(j)}$
Bez czasów stygnięcia:
Czas zakończenia wykonywania zadania $\pi(1)$ to $C_{\pi(1)}=r_{\pi(1)}+p_{\pi(1)}$
Ale czas zakończenia wykonywania kolejnego zadania to:
$C_{\pi(2)}=\max\{r_{\pi(2)}, C_{\pi(1)}\}+p_{\pi(2)}$

Dla zadania $\pi(j)$ mamy $C_{\pi(j)}=\max\{r_{\pi(j)}, C_{\pi(j-1)}\}+p_{\pi(j)}$

Ale co z ogonkami (czasami stygnięcia)???
Możemy zrobić sobie dodatkową tablicę, albo zmienną, gdzie przechowujemy najdłuższy do tej pory pełen czas wykonywania zadania (ze stygnięciem), gdzie ten pełen czas wykonywania danego zadania to:
Dla pierwszego: $\pi(1)$ to $C_{\pi(1)q}=r_{\pi(1)}+p_{\pi(1)}+q_{\pi(1)}$

A w ogólności
$C_{\pi(j)q}=\max\{r_{\{\pi(j)}, C_{\pi(j-1)}\}+p_{\pi(j)}+ q_{\pi(j)}$

I musimy przechowywać najdłuższy C_{π (j)}q, jaki do tej pory znaleźliśmy – to nasze kryterium