Marcin Różański 293145

**Problem:**

Kręgielnia

Dla kręgielni należy opracować algorytm rezerwacji torów. Rezerwacja określa, na jak długo potrzebny będzie pojedynczy tor (których w kręgielni jest n) oraz przedział czasowy, w którym musi być ona zrealizowana. Zbiór rezerwacji dla danego dnia jest znany z góry - zadanie polega na ułożeniu planu obłożenia torów maksymalizującego ich użycie.

Należy porównać rozwiązanie dokładne z heurystyką.

**Przyjęte założenia:**

· Przyjmujemy, że kręgielnia jest czynna w przedziale czasowym zaczynającym i kończącym w momencie minięcia kwadransa. Rezerwacje mogą być realizowanie wyłącznie w tym przedziale.

· Czas, w którym kręgielnia jest czynna podzielony jest na okresy czasowe(przyjmujemy każdy po 15 min -> 8:00-8:15; 8:15-8:30 itd.). Długość rezerwacji musi być wielokrotnością okresu czasowego, a jej przedział pokrywać się z początkiem/zakończeniem okresów czasowych. Realizacja rezerwacji również ma się pokrywać z okresami czasowymi(nie może się zaczynać w jego trakcie).

· Złożoność obliczeniowa zależy od ilości rezerwacji(n) i ilości torów(t).

**Algorytm:**

(1-3 szukanie zakresu o największym prawdopodobieństwie bycia wolnym z zakresów zawartych w przedziale rezerwacji; 4-6 układanie planu):

Struktury:

Rezerwacja – obiekt posiadający start, koniec, długość i wybrany start

Rezerwacje – lista rezerwacji

Obciążenia[] – tablica o długości równej ilości okresów czasowych początkowo wypełniona 0

Plan[][] – tablica dwuwymiarowa z 1 parametrem równym ilości torów i drugim równym ilości okresów

Plan tymczasowy [] – tablica o długości równej ilości okresów czasowych

1. Obliczenie ”obciążenia” okresów czasowych (złożoność – n):

Dla każdego elementu z rezerwacje:

Dla Obciążenia[i] gdzie i >= start rezerwacji oraz i < koniec rezerwacji Obciążenia[i] := Obciążenia[i] + 1

2. Uszeregować rezerwacje według luzu rosnąco (złożoność – n \* log(n)).

3. Wyznaczyć najmniej obciążone zakresy czasowe(złożoność – n):

Dla każdego elementu z rezerwacje:

min := 0, wybanyStart := 0

· Dla każdego możliwego zakresu czasowego realizacji rezerwacji:

· suma := SUMA(Obciążenia[i] gdzie i >= start rezerwacji oraz i < długość rezerwacji)

Jeżeli pierwszy zakres lub suma < min to

min := suma wybanyStart := i

Dla Obciążenia[i] gdzie i >= start rezerwacji oraz i < wybranyStart lub i > wybranyStart + długość rezerwacji oraz i < koniec rezerwacji

Obciążenia[i] := Obciążenia[i] - 1

4. Uszeregowanie rezerwacji według długości realizacji malejąco (złożoność – n \* log(n)).

5. Utworzyć zestawy rezerwacji zapewniające maksymalne obciążenie torów (złożoność – n2 \* t + n):

Dla każdego toru(t):

max := 0

Dla każdej rezerwacji(1):

Jeśli rezerwacja użyta kontynuuj

Dodaj rezerwacje do planu tymczasowego

Dla każdej rezerwacji(2):

Jeśli nie użyta oraz rezerwacja(2) != rezerwacja(1) to spróbuj dodać do planu tymczasowego(\*)

długość := 0

Dla każdego elementu(i) z planu tymczasowego

Jeśli plan tymczasowy[i] != 0 to długość := długości + 1

Jeśli długość > max plan[t] = plan tymczasowy

Wyczyść plan tymczasowy

Dla każdej rezerwacji z plan[t]

rezerwacja użyta := prawda

6. Krok korygujący(złożoność – n \* t)

Dla każdej rezerwacji gdzie nie użyta:

Dla każdego toru[t]:

Spróbuj dodać rezerwacje do plan[t] w każdym możliwym zakresie(\*\*)

(\*) SUMA(od plan tymczasowy[wybrany start] do plan tymczasowy[wybrany start + długość rezerwacji] == 0

(\*\*) – zakres początkowy, a nie wybrany w punkcie 3

Sumaryczna złożoność: n2 \* t + n \* t + 2 \* n \* log(n) + 2 \* n

O(T(n, i)) = n2 \* t

**Generacja danych:**

Zadaniem generatora jest

Stwórz tabelę list o wielkości równej ilości torów

W każdym torze dodaj wartość równą ilości okresów czasowych.

Dla i := 0 do i = liczba rezerwacji

Wylosuj tor(t)

Wylosuj element z tabeli(t)

Jeśli wartość elementu == 1 kontynuuj

Podziel losowo wartość na 2 i dodaj do tabeli(t)

i := i + 1

Dla każdego elementu tabeli(element)

Dodaj do rezerwacji element

Losowo rozszerz zakres elementu z obu stron(przed i po)

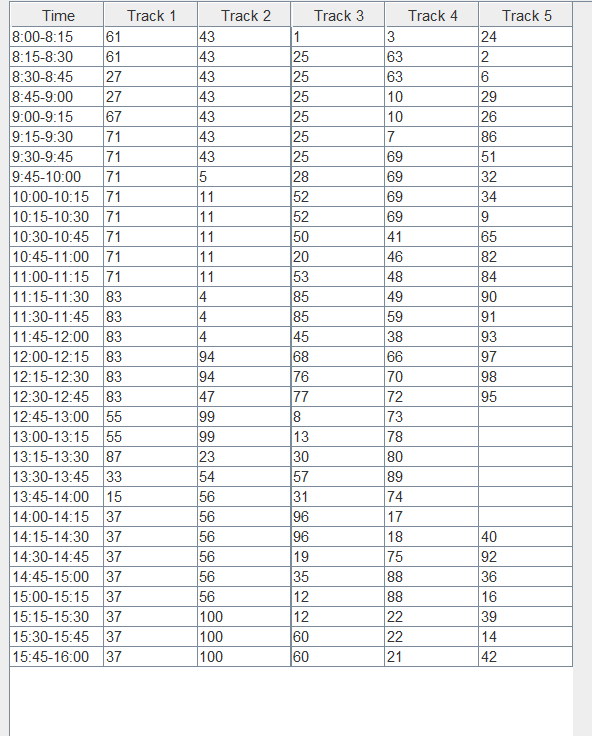
Wymagania:

Liczba rezerwacji >= liczba torów

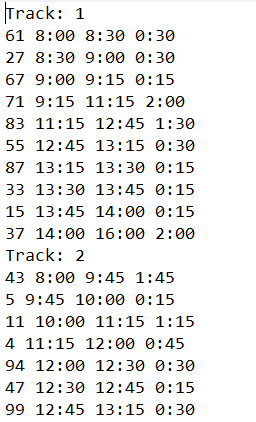
Liczba torów \* ilość okresów czasowych <= Liczba rezerwacji

**Wyjście:**

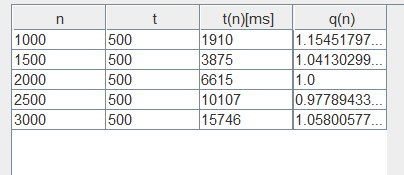
Przykładowa tabela dla m1/m2:



Przykładowy plik result.txt dla m1/m2:



Przykładowa tabela dla m3:



**Rozwiązanie dokładne:**

Złożoność rozwiązania dokładnego: n!\*tn\*96n

n! – próba wstawiania po kolei

tn – wstawianie do różnych torów

96n – na różne przedziały czasowe(mocno pesymistyczne założenie)

Przykładowe wyniki:

n = 6, t = 3 96->20 - 32301ms

Wynik dla heurystyki: 11ms

Jak dla każdego problemu NP-trudnego czas wykonania rośnie w zawrotnym tempie nawet przy niskich wartościach. Wykorzystywanie algorytmu dokładnego do większych danych zwyczajnie nie ma sensu.