

---

---

# ALGORYTMY I STRUKTURY DANYCH

## PROGRAMOWANIE DYNAMICZNE

Instytut Informatyki Uniwersytetu Wrocławskiego

Paweł Rzechonek

---

---

1. Dany jest graf prosty ważony  $G = (V, E)$ , w którym każda krawędź ma określoną nieujemną wagę. Zaprezentuj i przeanalizuj złożoność obliczeniową *algorytmu Floyda* opracowanego w 1962 roku, rozwiązującego problem najtańszych dróg pomiędzy wszystkimi parami wierzchołków. Koszt drogi w takim grafie to suma wag krawędzi na ścieżce łączącej parę wierzchołków.

**Uwaga:** Przyjmij, że graf jest pamiętany w postaci macierzy sąsiedztwa. W macierzy tej podane są nieujemne koszty krawędzi (bezpośredniej drogi pomiędzy wierzchołkami) lub wartość  $+\infty$  jeśli nie ma krawędzi łączącej dwa wierzchołki (na głównej przekątnej są zera).

---

```
(1)  function shortest-paths (integer E[n][n]) ↦ integer[n][n]
(2)  {
(3)      integer D[n][n];
(4)      integer D ← E;
(5)      for k = 0 .. n - 1 do
(6)          for i = 0 .. n - 1 do
(7)              for j = 0 .. n - 1 do
(8)                  Di,j ← min(Di,j, Di,k + Dk,j);
(9)      return D;
(10) }
```

---

2. Wynikiem działania algorytmu Floyda jest kwadratowa macierz z kosztami najtańszych ścieżek pomiędzy wszystkimi parami wierzchołków. Jak na podstawie macierzy sąsiedztwa i macierzy najtańszych ścieżek wyznaczyć najtańszą ścieżkę pomiędzy zadaną parą wierzchołków?
3. Jak można wykorzystać algorytm Floyda do wyznaczenia najkrótszych ścieżek (chodzi o liczbę krawędzi na ścieżce) pomiędzy wszystkimi parami wierzchołków w grafie prostym bez wag?
4. Czy algorytm Floyda można wykorzystać do wyznaczenia najtańszych dróg pomiędzy wszystkimi parami wierzchołków w grafie prostym ważonym, jeśli dopuścimy ujemne wagi krawędzi?