Temat: Algorytmy ścieżkowe na grafach z wagami. Algorytm Forda-Bellmana, Dijkstry, Floyda

1. Algorytm Forda - Bellmana

WP: Graf G - zorientowany, z wagami, bez cykli o ujemnej długości, ustalony wierzchołek startowy s.

WK: Minimalne odległości od s do pozostałych wierzchołków grafu.

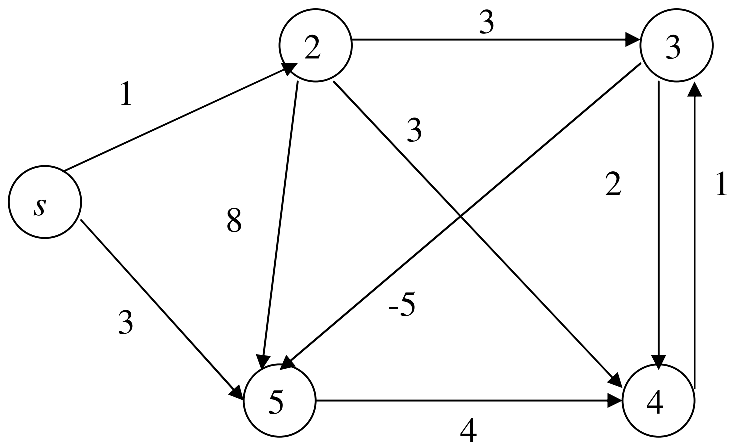
A – macierz sąsiedztwa grafu G
D – macierz wynikowa

$$D[v] = \begin{cases} 0 \quad \text{dla } v = s \\ \text{minimalna suma wag na ścieżce od } s \text{ do } v\text{, o ile taka ścieżka istnieje} \\ \infty \quad \text{gdy ścieżka z } s \text{ do } v \text{ nie istnieje} \end{cases}$$

<u>Algorytm</u>

```
for (i=1; i<=n; i++)
    if  (i = s)  D[i]=0; else D[i]=A[s,i];
for (k=1; k<=n-2; k++)
      for (j=1;j<=n; j++)
          if (j!=s)
             for (i=1;i<=n; i++)
                 D[j]=min(D[j],D[i]+A[i,j]);
```

Przykład 1



$$A = \begin{bmatrix} \infty & 1 & \infty & \infty & 3 \\ \infty & \infty & 3 & 3 & 8 \\ \infty & \infty & \infty & 2 & -5 \\ \infty & \infty & 1 & \infty & \infty \\ \infty & \infty & \infty & 4 & \infty \end{bmatrix}$$

k	$D[s]$	$D[2]$	$D[3]$	$D[4]$	$D[5]$
	0	1	∞	∞	3
1	0	1	4	4	3
2	0	1	4	3	-1
3	0	1	4	3	-1

Złożoność czasowa algorytmu: $\Theta(n^3)$

2. Algorytm Dijkstry

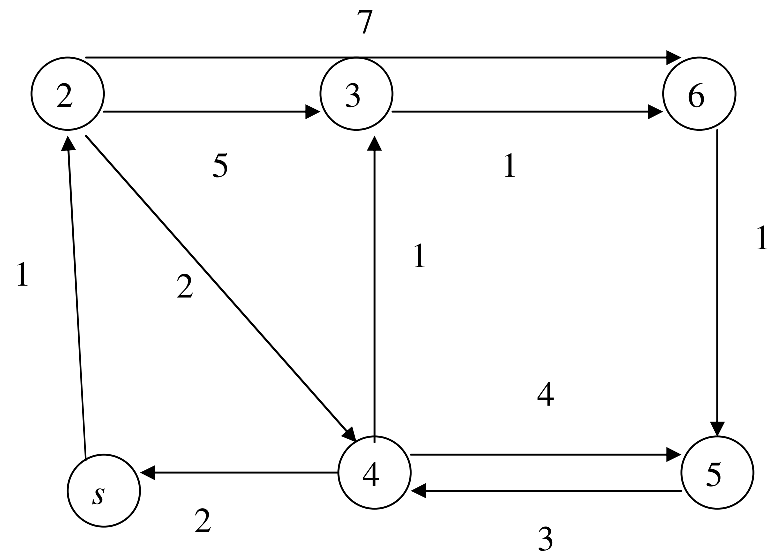
WP: Graf *G* – zorientowany, z nieujemnymi wagami, ustalony wierzchołek startowy *s*

WK: Minimalne odległości od *s* do pozostałych wierzchołków grafu

Algorytm

```
T ={1, 2, ..., n};
 for (i=1; i<=n; i++)
          if  (i == s)  D[i]=0; else D[i]=A[s,i];
  T={1,2, ..., n}-{s};
  while (T!=∅)
        {
          u="dowolny wierzchołek r ∈ T taki, że
               D[r]=min{D[p]:p∈ T}";
         T=T-{u};
         for (v=1; v<=n; v++)
             if v ∈  T  then
                    D[v]=min(D[v],D[u]+A[u,v]);
         }
```

Przykład 2



Stan zbioru T oraz tablicy D po kolejnych iteracjach pętli while

1. Przed pierwszym krokiem pętli
   $T= \{2, 3, 4, 5, 6\}$, $D=\{0, 1, \infty, \infty, \infty, \infty \}$

2. I krok pętli
   $u= 2$, $T=\{3, 4, 5, 6\}$, $D= \{0, 1, 6, 3, \infty, 8 \}$

3. II krok pętli
   $u= 4$, $T=\{3, 5, 6\}$, $D= \{0, 1, 4, 3, 7, 8 \}$

4. III krok pętli
   $u= 3$, $T=\{ 5, 6\}$, $D= \{0, 1, 4, 3, 7, 5 \}$

5. IV krok pętli
   $u= 6$, $T=\{5\}$, $D= \{0, 1, 4, 3, 6, 5\}$

6. V krok pętli
   $u=5$, $T=\emptyset$, $D= \{0, 1, 4, 3, 6, 5\}$

Złożoność czasowa: $\Theta(n^2)$.

Gdybyśmy zastosowali w algorytmie Dijkstry kolejkę priorytetową, to koszt tego algorytmy spadłby do O(*nlogn*).

3. Algorytm znajdowania najkrótszej drogi z ustalonego wierzchołka do pozostałych wierzchołków grafu, na podstawie tablicy minimalnych odległości

WP: Odległości minimalne z ustalonego wierzchołka s do pozostałych wierzchołków grafu, zapamiętane w tablicy D (wynik poprzednich algorytmów: Forda-Bellmana albo Dijkstry) oraz tablica A wag grafu. Ustalony wierzchołek t grafu różny od s.

WK: Stos określający najkrótszą ścieżkę z s do t.

Algorytm

```
S=∅; push(t, S);
v =t ;
while (v != s)
   {
     u="wierzchołek, dla którego D[v]=D[u]+A[u,v]";
     push(u, S);
     v = u;
   }
```

Przykład 3

Niech $t = 5$, a $s = 1$

$$A = \begin{bmatrix} \infty & 1 & \infty & \infty & 3 \\ \infty & \infty & 3 & 3 & 8 \\ \infty & \infty & \infty & 2 & -5 \\ \infty & \infty & 1 & \infty & \infty \\ \infty & \infty & \infty & 4 & \infty \end{bmatrix}$$

D: [0, 1, 4, 3, -1]

S:

1
2
3
5

Złożoność czasowa: $\Theta(n^2)$

4. Algorytm wyznaczania najmniejszych odległości między każdą parą wierzchołków grafu - metoda Floyda

WP : Graf G – zorientowany, z wagami, bez cykli ujemnej długości

WK: Odległości minimalne między wszystkimi parami wierzchołków grafu

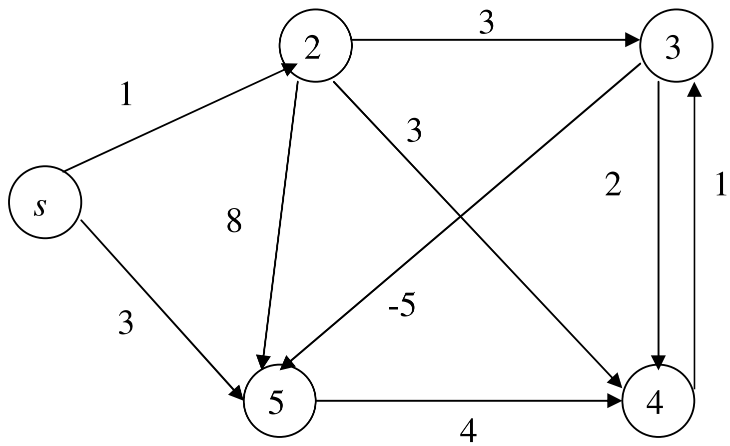
Tablica wynikowa

$$D[i, j] = \begin{cases} \text{minimalna suma wag na ścieżce od } i \text{ do } j\text{, o ile taka ścieżka istnieje} \\ \infty \text{ gdy ścieżka z } j \text{ do } j \text{ nie istnieje} \end{cases}$$

<u>Algorytm</u>

```
for (i=1;i<=n; i++)
      for (j=1; j<=n; j++) D[i, j]=A[i, j];
for (i=1; i<=n; i++)  D[i, j]:=0;
for (m=1; m<=n; m++)
   for (i=1; i<=n; i++)
        for (j=1; j<=n; j++)
                 D[i, j]=min(D[i,j],D[i,m]+A[m,j]);
```

Przykład 4



$$A = \begin{bmatrix} \infty & 1 & \infty & \infty & 3 \\ \infty & \infty & 3 & 3 & 8 \\ \infty & \infty & \infty & 2 & -5 \\ \infty & \infty & 1 & \infty & \infty \\ \infty & \infty & \infty & 4 & \infty \end{bmatrix}$$

$$D = \begin{bmatrix} 0 & 1 & 4 & 3 & -1 \\ \infty & 0 & 3 & 2 & -2 \\ \infty & \infty & 0 & 1 & -1 \\ \infty & \infty & 2 & 0 & -3 \\ \infty & \infty & 5 & 4 & 0 \end{bmatrix}$$

Złożoność czasowa algorytmu: $\Theta(n^3)$