Grafowe Modelowanie Systemów (w), kolokwium 1, 13.06.2006. $\Sigma =$ ☐

Wypełnij **drukowanymi** literami:

Imię	Nazwisko	Nr indeksu

Uwagi:

1. W każdym zadaniu podano liczbę punktów za każdą poprawną odpowiedź. Jeśli odpowiedź nie jest poprawna, to liczba otrzymanych punktów wynosi 0.
2. Czas pisania **105 min**.
3. Maksymalna liczba punktów do zdobycia wynosi **100**.

☐ **1.** (18 pkt.) Dana jest macierz sąsiedztwa

$$W = \begin{bmatrix} 0 & 10 & \infty & \infty & \infty & 6 \\ \infty & 0 & 7 & \infty & \infty & -4 \\ \infty & \infty & 0 & 3 & -2 & \infty \\ \infty & \infty & \infty & 0 & \infty & \infty \\ \infty & \infty & \infty & -1 & 0 & 8 \\ \infty & \infty & -3 & \infty & \infty & 0 \end{bmatrix}.$$

Kolejne wiersze (kolumny) macierzy odpowiadają wierzchołkom $1, 2, 3, 4, 5, 6$. Funkcja h znaleziona podczas działania algorytmu Johnsona ma postać:

	1	2	3	4	5	6	
funkcja h:							$(6 \cdot 1$ pkt.)

Wiersz numer 1 w tablicy d bezpośrednio po wykonaniu algorytmu Dijkstry ma postać:

	1	2	3	4	5	6	
wiersz $d[1, *]$:							$(6 \cdot 1$ pkt.)

Po dokonaniu przekształcenia odwrotnego, zwracany wiersz nr 1 w tablicy d to:

	1	2	3	4	5	6	
wiersz $d[1, *]$:							$(6 \cdot 1$ pkt.)

☐ **2.** (20 pkt.) Dana jest macierz sąsiedztwa

$$W = \begin{bmatrix} 0 & 13 & 2 & 5 & 5 \\ 13 & 0 & 17 & 8 & 9 \\ 2 & 17 & 0 & 10 & 7 \\ 5 & 8 & 10 & 0 & 11 \\ 5 & 9 & 7 & 11 & 0 \end{bmatrix}.$$

Przeprowadź symulację jednej iteracji algorytmu 2-OPT (szukanie cykli komiwojażera). Cyklem początkowym jest cykl zawierający krawędzie $\{a, b\}, \{b, c\}, \{c, d\}, \{d, e\}, \{a, e\}$.

Wypełnij poniższą tabelę (3 punkty za każdy wiersz) tak, aby:

- pierwsza kolumna zawierała parę usuwanych krawędzi z cyklu;
- druga kolumna zawierała parę krawędzi uzupełniających cykl;
- trzecia kolumna zawierała długość nowego cyku.

usuwane krawędzie	nowe krawędzie	długość

Nowym cyklem bieżącym, obliczonym podczas pierwszej iteracji jest:

(5 pkt.)

3. (19 pkt.) Dany jest graf G taki, że $V(G) = \{a, b, c, d, e, f, g\}$, $E(G) = \{\{a, b\}, \{a, c\}, \{b, c\}, \{b, d\}, \{b, e\}, \{c, e\}, \{d, e\}, \{d, f\}, \{e, f\}, \{e, g\}, \{f, g\}\}$. Dana jest również funkcja wagowa w taka, że $w(\{a, b\}) = 1, w(\{a, c\}) = 5, w(\{b, c\}) = 9, w(\{b, d\}) = 3, w(\{b, e\}) = 12, w(\{c, e\}) = 15, w(\{d, e\}) = 4, w(\{d, f\}) = 7, w(\{e, f\}) = 9, w(\{e, g\}) = 8, w(\{f, g\}) = 3$.

Podaj krawędzie w takiej kolejności, w jakiej były wybierane przez algorytm Kruskala podczas tworzenia drzewa spinającego:

(8 pkt.)

Suma wag krawędzi tworzących dowolne drzewo spinające o minimalnej sumie wag dla tego grafu wynosi: (5 pkt.)

Ile jest różnych drzew spinających o minimalnej sumie wag? (3 pkt.)

Ile jest różnych drzew spinających o minimalnej sumie wag może znaleźć powyższy algorytm?

(3 pkt.)

4. (23 pkt.) Niech będzie dany graf prosty G taki, że $V(G) = \{a, b, c, d, e, f, g, h\}$ oraz $E(G) = \{\{a, b\}, \{b, d\}, \{d, e\}, \{b, c\}, \{d, c\}, \{a, e\}, \{d, f\}, \{d, h\}, \{f, g\}, \{g, h\}\}$. Mamy dane drzewo spinające T takie, że $E(T) = \{\{a, e\}, \{b, d\}, \{d, e\}, \{d, c\}, \{d, f\}, \{d, h\}, \{g, h\}\}$.

Przeprowadź symulację algorytmu szukania drzewa spinającego o minimalnym stopniu. Wypisz taki przebieg algorytmu, gdzie liczba iteracji jest największa. W każdym wierszu podaj krawędzie nowego drzewa spinającego otrzymanego w danej iteracji oraz jego stopień. Jeśli liczba iteracji jest mniejsza niż 3, to w pozostałych wierszach wpisz 'KONIEC'.

Nr iteracji	Drzewo spinające	Stopień
1		
2		
3		

(4 pkt. za wiersz)

Jaka jest minimalna liczba iteracji algorytmu? (4 pkt.)

Jaki jest stopień drzewa spinającego o minimalnym stopniu? (4 pkt.)

Czy dla każdego grafu G istnieje taki przebieg algorytmu, który prowadzi do znalezienia optymalnego rozwiązania? (wpisz "TAK" lub "NIE") (3 pkt.)

5. (20 pkt.) Przez kolejność BFS(x) rozumiemy kolejność w jakiej wierzchołki były oznaczane, zakładając, że wierzchołkiem startowym był x, natomiast kolejność DFS(x) to kolejność w jakiej wierzchołki były odwiedzane (zgodnie z opisem na wykładzie). Dla grafu z poprzedniego zadania podaj kolejność:

BFS(d): (10 pkt.)

DFS(h): (10 pkt.)