

PRZYKŁADOWE WEJŚCIE – ZESTAW 1.

Ad. 1, 2: Reprezentacje grafów prostych + wizualizacja

Dane wejściowe: lista sąsiedztwa/macierz sąsiedztwa/macierz incydencji. Program powinien potrafić zinterpretować na wejściu każdą z trzech reprezentacji (jedno wywołanie = jedna reprezentacja na wejściu). *Oczywiście format zapisu listy/macierzy nie musi być identyczny z zaprezentowanym poniżej.*

Wyjście programu: pozostałe dwie reprezentacje oraz rysunek z wizualizacją z 2. zadania.

Przykład: dla danych wejściowych z listingu 1. program powinien dać wyniki analogiczne do listingów 2., 3. oraz rys. 1.

```
1 0 1 0 0 1 1 0 0 0 0 0 0
2 1 0 1 0 0 1 0 0 0 0 0 0
3 0 1 0 1 1 0 0 0 0 0 0 1
4 0 0 1 0 0 0 0 1 1 0 1 0
5 1 0 1 0 0 0 1 0 1 0 0 0
6 1 1 0 0 0 0 1 0 0 0 0 0
7 0 0 0 0 1 1 0 1 0 0 0 0
8 0 0 0 1 0 0 1 0 1 0 0 1
9 0 0 0 1 1 0 0 1 0 1 0 0
10 0 0 0 0 0 0 0 0 1 0 0 0
11 0 0 0 1 0 0 0 0 0 0 0 0
12 0 0 1 0 0 0 0 1 0 0 0 0
```

Listing 1: Dane **wejściowe**: macierz sąsiedztwa

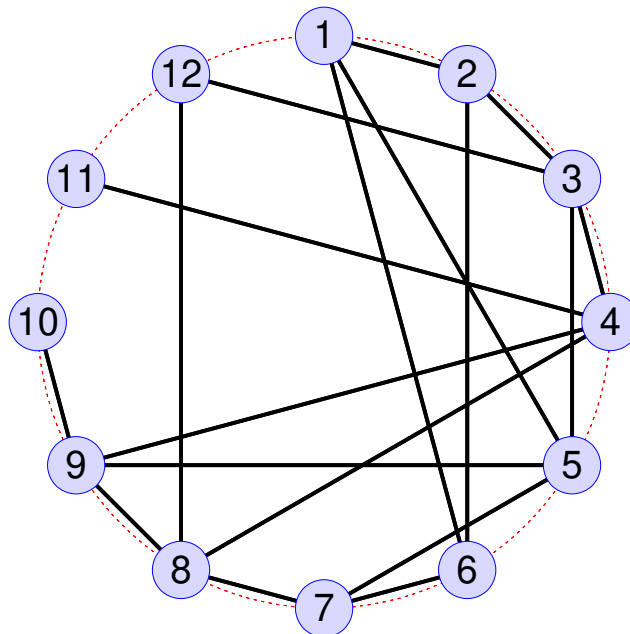
```
1 1. 2 5 6
2 2. 1 3 6
3 3. 2 4 5 12
4 4. 3 8 9 11
5 5. 1 3 7 9
6 6. 1 2 7
7 7. 5 6 8
8 8. 4 7 9 12
9 9. 4 5 8 10
10 10. 9
11 11. 4
12 12. 3 8
```

Listing 2: Dane **wyjściowe**: lista sąsiedztwa dla przykładu z listingu 1.

```
1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0
2 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
3 0 1 1 0 0 0 0 0 1 0 0 0 0 1 0 0 0
4 0 0 1 1 0 0 0 0 0 0 0 0 1 0 1 0 0
5 0 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0
6 0 0 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0
7 0 0 0 0 0 1 1 0 0 0 0 1 0 0 0 0 0
8 0 0 0 0 1 1 0 0 0 0 0 0 0 1 0 0 0
9 0 0 0 1 1 0 0 0 0 0 1 0 0 0 0 0 1
```

10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1

Listing 3: Dane **wyjściowe**: macierz incydencji dla przykładu z listingu 1. (kolejność kolumn: dowolna)



Rysunek 1: Dane **wyjściowe**: wizualizacja grafu dla przykładu z listingu 1.

Ad. 3: Generowanie grafów losowych

(a) Model $G(n, l)$

Dane wejściowe:

- n – liczba wierzchołków grafu,
- l – liczba krawędzi grafu.

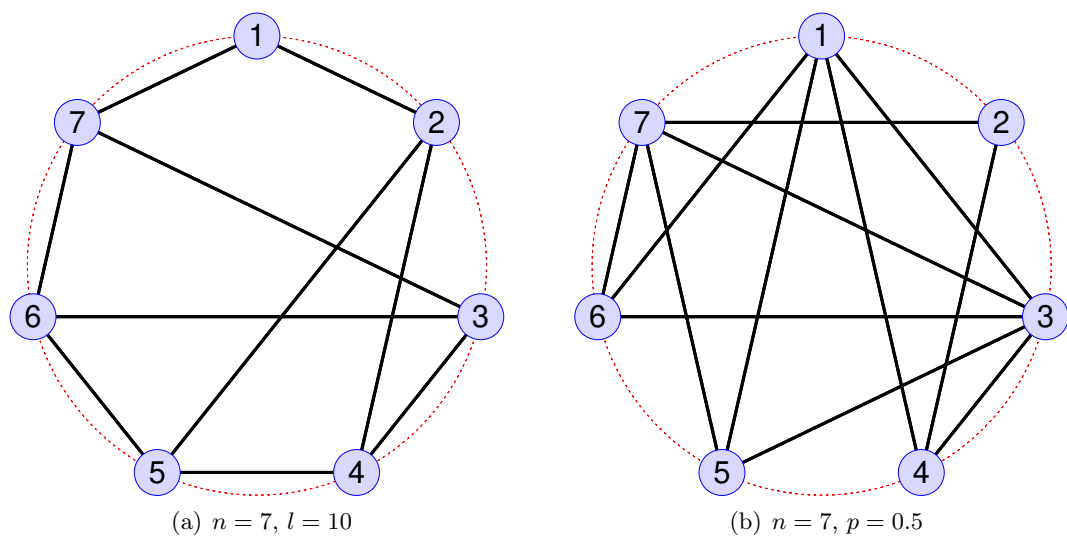
Wyjście programu: wylosowany graf przedstawiony w dowolnej postaci (np. jedna z reprezentacji z 1. zadania lub obrazek z wizualizacją, jak np. na rys. 2(a)).

(b) Model $G(n, p)$

Dane wejściowe:

- n – liczba wierzchołków grafu,
- p – prawdopodobieństwo, że pomiędzy dwoma wierzchołkami istnieje krawędź.

Wyjście programu: wylosowany graf przedstawiony w dowolnej postaci (np. jedna z reprezentacji z 1. zadania lub obrazek z wizualizacją, jak np. na rys. 2(b)).



Rysunek 2: Dane **wyjściowe**: wizualizacja grafu dla przykładowych parametrów, podanych w podpisach rysunków (a) i (b)