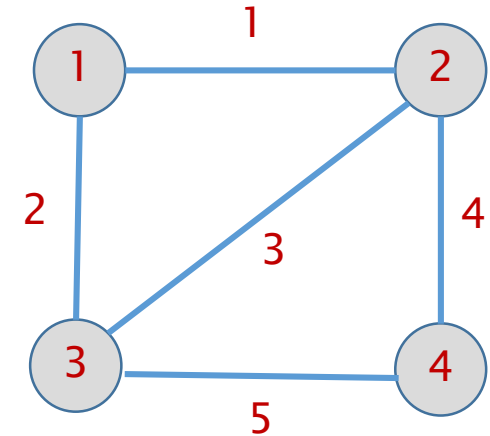
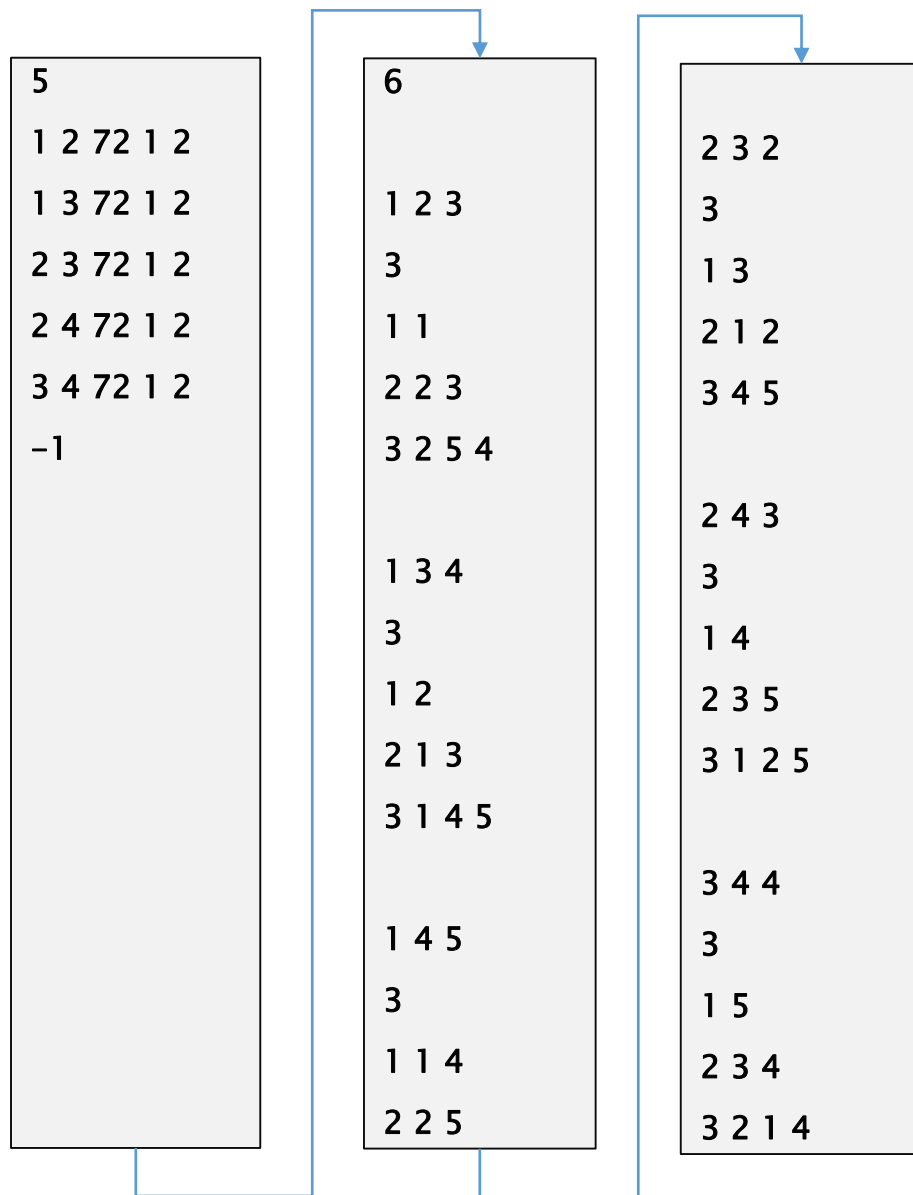


## ćwiczenia - format MP2k (net4.txt)



$h(d)$ : 3, 4, 5, 2, 3, 4

$d$ : (1,2), (1,3), (1,4), (2,3), (2,4), (3,4)

### routes

$P(1,1) = \{1\}$ ,  $P(1,2) = \{2,3\}$ ,  $P(1,3) = \{2,4,5\}$

$P(2,1) = \{2\}$ ,  $P(2,2) = \{1,3\}$ ,  $P(2,3) = \{1,4,5\}$

$P(3,1) = \{1,4\}$ ,  $P(3,2) = \{2,5\}$

$P(4,1) = \{3\}$ ,  $P(4,2) = \{1,2\}$ ,  $P(3,3) = \{4,5\}$

$P(5,1) = \{4\}$ ,  $P(5,2) = \{3,5\}$ ,  $P(5,3) = \{1,2,5\}$

$P(6,1) = \{5\}$ ,  $P(6,2) = \{3,4\}$ ,  $P(6,3) = \{1,2,4\}$

# ćwiczenia - format MP2k - BNF (Backus-Naur Form)

```
<network> ::= <links><EOL><separator><EOL><demands>
```

```
<separator> ::= "-1"
```

```
<links> ::= <number of links><EOL><link list>
```

```
<number of links> ::= <integer>
```

```
<linkList> := <link>[<EOL><link>]*
```

```
<link> ::= <start node> <end node> <number of fibre pairs in cable> <fibre pair cost> <number of lambdas in fibre>
```

```
<start node> ::= <node id>
```

```
<end node> ::= <node id>
```

```
<number of fibre pairs in cable> ::= <integer>
```

```
<fibre cost> ::= <float>
```

```
<number of lambdas in fibre> ::= <integer>
```

```
<node id> ::= <integer>
```

5

1 2 72 1 2

1 3 72 1 2

2 3 72 1 2

2 4 72 1 2

3 4 72 1 2

-1

*Notacja ta jest powszechnie używana w informatyce do zapisu składni (syntaktyki) języków programowania i protokołów komunikacyjnych. Została wymyślona przez Johna Backusa w latach 50. w czasie prac nad językiem Fortran, a następnie zmodyfikowana przez Petera Naura i użyta do zdefiniowania składni języka Algol (z Wikipedii).*

## ćwiczenia - format MP2k - BNF (Backus-Naur Form) - 2

`<demands> ::= <number of demands><EOL><demand list>`

`<number of demands> ::= <integer>`

`<demand list> ::= <demand>[<EOL><demand>]*`

`<demand> ::= <start node> <end node> <demand volume><EOL><demand paths>`

`<demand volume> ::= <integer>`

`<demand paths> ::= <number of demand paths><EOL><demand path list>`

`<number of demand paths> ::= <integer>`

`<demand path list> ::= <demand path>[<EOL><demand path>]*`

`<demand path> ::= <demand path id> <link list><EOL>`

`<demand path id> ::= <integer>`

`<link list> ::= <link id>[ <link id>]*`

`<link id> ::= <integer>`

