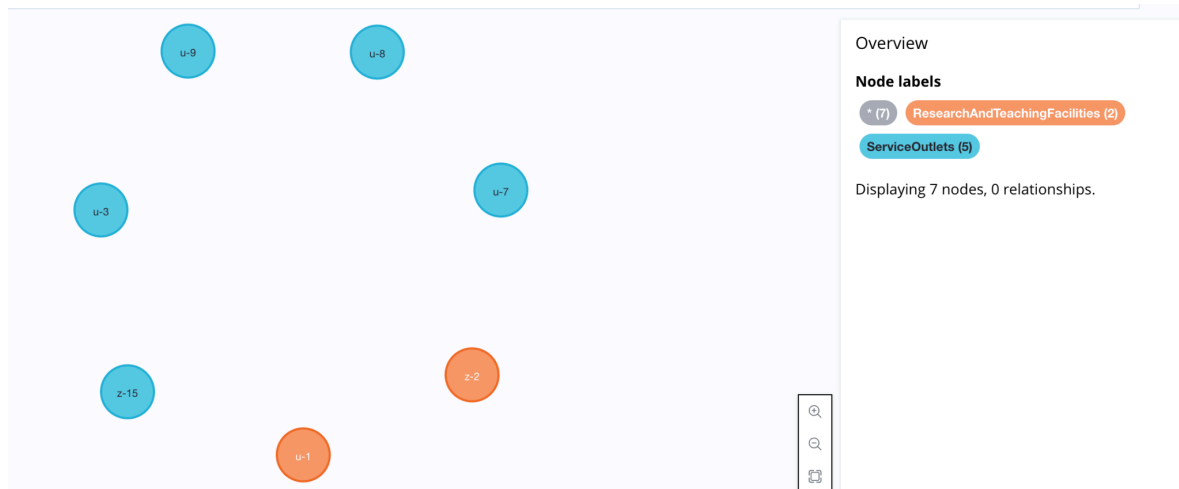


Zaawansowane technologie bazodanowe
Mikołaj Wielgos
Lab3

1. czy występują samotne budynki, tj. takie, które nie są połączone z innymi budynkami,

```
MATCH (b)
WHERE NOT
(b)-[:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_1|INSIDE_WALK_FLOOR_ALL]-( )
AND NOT
()-[:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_1|INSIDE_WALK_FLOOR_ALL]-(b)
AND NOT b:Faculty
RETURN b
```



b
(:ResearchAndTeachingFacilities {id: "u-1"})
(:ResearchAndTeachingFacilities {id: "z-2"})
(:ServiceOutlets {id: "u-7"})
(:ServiceOutlets {id: "u-8"})
(:ServiceOutlets {id: "u-9"})
(:ServiceOutlets {id: "u-3"})
(:ServiceOutlets {id: "z-15"})

2. ile jest budynków typu service,

```
MATCH (s:ServiceOutlets)
RETURN COUNT(s) as serviceCount
```

serviceCount
10

3. z jakimi budynkami i na jakich piętrach połączony jest A-1,

```
MATCH (a1 {id: "a-1"})-[r:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_ALL]→(b)
RETURN b.id as connected_building,
CASE
WHEN type(r) = 'INSIDE_WALK_FLOOR_0' THEN 'Parter'
WHEN type(r) = 'INSIDE_WALK_FLOOR_ALL' THEN 'Wszystkie piętra'
END as connection_type
```

connected_building	connection_type
"c-1"	"Parter"
"a-0"	"Parter"
"h-a1"	"Parter"
"h-a1"	"Wszystkie piętra"
"a-2"	"Parter"
"a-2"	"Wszystkie piętra"

4. jak dojść z Admission Centre do Faculty of Mechanical Engineering and Robotics bez opuszczania budynków,

```
MATCH (start) -[:BELONGS_TO_FACULTY]→(:Faculty {id: 31}),
(end) -[:BELONGS_TO_FACULTY]→(:Faculty {id: 6}),
path =
shortestPath((start) -[:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_1|INSIDE_WALK_FLOOR_ALL*1..9]→(end))
RETURN path
```

(no changes, no records)

Natomiast dodając możliwość przechodzenia między pobliskimi budynkami wychodząc na zewnątrz:

```
MATCH (start) -[:BELONGS_TO_FACULTY]→(:Faculty {id: 31}),
(end) -[:BELONGS_TO_FACULTY]→(:Faculty {id: 6}),
path =
shortestPath((start) -[:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_1|OUSTIDE_WALK|INSIDE_WALK_FLOOR_ALL*1..9]→(end))
RETURN path
```

path
(:ResearchAndTeachingFacilities {id: "u-2"})-[:INSIDE_WALK_FLOOR_ALL]->(:ResearchAndTeachingFacilities {id: "a-3"})-[:INSIDE_WALK_FLOOR_ALL]->(:ResearchAndTeachingFacilities {id: "a-4"})-[:INSIDE_WALK_FLOOR_0]->(:ResearchAndTeachingFacilities {id: "c-4"})-[:OUSTIDE_WALK]->(:ResearchAndTeachingFacilities {id: "b-1"})-[:INSIDE_WALK_FLOOR_ALL]->(:ResearchAndTeachingFacilities {id: "b-2"})

5. jaka jest najkrótsza droga (najmniejsza liczba odwiedzanych budynków) z parteru C-3 do wejścia w A-0?

```
MATCH path = shortestPath(
(start {id: "c-3"})-[:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_ALL*1..10]→(end
{id: "a-0"})
)
RETURN path, length(path) as numberOfBuildings
```

path	numberOfBuildings
(:ResearchAndTeachingFacilities {id: "c-3"})-[:INSIDE_WALK_FLOOR_ALL]->(:ResearchAndTeachingFacilities {id: "c-2"})-[:INSIDE_WALK_FLOOR_0]->(:ResearchAndTeachingFacilities {id: "c-1"})-[:INSIDE_WALK_FLOOR_0]->(:ResearchAndTeachingFacilities {id: "a-1"})-[:INSIDE_WALK_FLOOR_0]->(:ResearchAndTeachingFacilities {id: "a-0"})	4

6. znajdź budynki, które są połączone z trzema innymi budynkami (sąsiadują ze sobą i można się do nich dostać nie wychodząc na zewnątrz).

```
MATCH
(b)-[r:INSIDE_WALK_FLOOR_0|INSIDE_WALK_FLOOR_1|INSIDE_WALK_FLOOR_ALL]→(neighbor)
WITH b, COUNT(DISTINCT neighbor) as connections
WHERE connections = 3
RETURN b.id as building, connections
```

building	connections
"c-7"	3
"c-6"	3
"a-4"	3

Wynik jest niepełny ze względu na braku zamodelowania części budynków.