

Michał Ryszka [REDACTED]

Tomasz Czaplicki [REDACTED]

Michał Malinowski [REDACTED]

Obiekty Internetu Rzeczy

Dokumentacja Projektu

1. Architektura oprogramowania

Komponenty:

- Obir.ino – kod źródłowy
- OBIR_coap_server.cpp – biblioteka implementująca protokół CoAP
- OBIR_coap_server.h – biblioteka implementująca protokół CoAP
- platforma EBSimUnoEth – używana do uruchomienia serwera
- klient CoAP (wtyczka Copper dla przeglądarki Mozilla Firefox)
- biblioteki Arduino
- biblioteki laboratoryjne z przedmiotu OBIR

Interfejsy:

- klient CoAP (wtyczka Copper)
- serwer uruchomiony za pomocą platformy EBSimUnoEth na podstawie utworzonego pliku .hex stworzonego w wyniku kompilacji kodu źródłowego

Przyjęte rozwiązania programistyczne:

- obliczanie wierzchołków centralnych - algorytm Dijkstr'y
 - o przyjęto, że każda dodana krawędź posiada taką samą wagę
 - o wykorzystano kod stworzony w ramach przedmiotu AISDE (funkcje `minDistance()` i `Dijkstra()`)
- sortowanie wierzchołków - algorytm Bubble Sort
- można dodawać wierzchołki z zakresu 0-9; maksymalna liczba krawędzi wynosi 10 (parametry definiowane w kodzie źródłowym)

Sposób testowania systemu:

Działanie systemu przetestowaliśmy poprzez wykonywanie na nim różnych operacji - uruchamianie wszystkich dostępnych funkcjonalności dla każdego z zasobów.

Algorytm znajdowania centralnych wierzchołków został dodatkowo dogłębnie przetestowany przed użyciem go w kodzie źródłowym – debugowanie kodu w niezależnym środowisku C++.

2. Lista funkcjonalności¹

Obsługa wiadomości:

- NON GET – 1, 2
- NON PUT – 2, 3

Obsługa opcji:

- Content-Format – 10, 5
- Uri-Path – 1, 1
- Accept – 10, 5
- Obsługa tokena i MID – 1, 1
- Observe – 8, 7
- eTag – 9, 6

Funkcjonalność grafu:

- GET: pobranie wszystkich krawędzi – 3, 4
- GET: pobranie wszystkich centrów (wierzchołków centralnych) grafu – 4, 5
- PUT: dodanie nowej krawędzi do grafu (Zakłada się, że na każdym etapie dodawania krawędzi graf zachowuje spójność) – 2, 3
- gdy wyczerpie się pamięć przydzielona na krawędzie, serwer powinien odesłać odpowiedni kod błędu – 5, 17

Metryki:

- GET: GetNumber - pobranie liczby wysłanych żądań GET – 6, 18
- GET: PutNumber - pobranie liczby wysłanych żądań PUT - zasób obserwowalny – 7, 18
- GET: SendPackets - suma wysłanych pakietów – 1, 18

Zasoby:

- CentralVertices – zasób wylicza centralne wierzchołki. Obsługuje tylko żądanie GET, dla innych kodów żądania zwraca: 4.05 Method Not Allowed. W przypadku braku krawędzi w bazie otrzymamy odpowiedź 4.04 Not Found. Obsługuje następujące opcje ACCEPT: text/plain, application/link-format, application/json.
- Edges – zasób pozwala na dodanie nowych krawędzi i zwraca wszystkie dodane krawędzie. Gdy jest pusty lub wykonamy nieobsługiwane żądanie (czyli POST lub DELETE) odpowie w analogiczny sposób jak zasób CentralVertices. W przypadku przepełnienia maksymalnej liczby krawędzi wyśle odpowiedź 5.00 Internal Server Error. Dla błędnie wprowadzonych danych: utworzenie krawędzi dla jednego wierzchołka, dodanie istniejącej krawędzi, wprowadzenie błędnie danych, przerwanie spójności grafu zwróci błąd 4.05 Method Not Allowed. Obsługuje te same opcje ACCEPT co zasób CentralVertices. Wspiera opcje ETag. Dla operacji PUT należy wprowadzać dane w formacie: „X,Y”, gdzie X i Y to numery wierzchołków w zakresie 0-9.
- GetNumber – zasób zwraca ilość wysyłanych odpowiedzi na żądania GET. Obsługuje tylko żądanie GET, dla innych kodów żądania zwraca: 4.05 Method Not Allowed.

¹Cyfry oznaczone kolorem niebieskim znajdujące się po myślnikach oznaczają podpunkty testów, dotyczące wybranej funkcjonalności, a cyfry oznaczone kolorem czerwonym znajdujące się po myślnikach oznaczają podpunkty scenariusza, dotyczące wybranej funkcjonalności.

- PutNumber – zasób zwraca ilość wysyłanych odpowiedzi na żądania PUT. Obsługuje tylko żądanie GET, dla innych kodów żądania zwraca: 4.05 Method Not Allowed. Zasób ten może być obserwowany.
- SendPackets – zasób zwraca ilość wysyłanych pakietów. Obsługuje tylko żądanie GET, dla innych kodów żądania zwraca: 4.05 Method Not Allowed.
- .well-know/core – zwraca pozostałe zasoby w formacie application/link-format. Obsługuje tylko żądanie GET, dla innych kodów żądania zwraca: 4.05 Method Not Allowed.

3. Wykonane testy

1. Wysłanie żądania NON-GET do zasobu *SendPackets* zwracającego ilość wysłanych pakietów przez serwer:

| Time | CoAP Message | MID | Token | Options | Payload |
|----------|------------------|-------|--------|--|---------|
| 22:05:17 | NON-GET | 32097 | 0xCE33 | Uri-Path: SendPackets, Accept: 50, Block2: 0/0/128 | |
| 22:05:17 | NON-2.05 Content | 32098 | 0xCE33 | Content-Format: 0 | 1 |
| 22:07:02 | NON-GET | 32098 | 0xCE33 | Uri-Path: SendPackets, Accept: 50, Block2: 0/0/128 | |
| 22:07:02 | NON-2.05 Content | 32099 | 0xCE33 | Content-Format: 0 | 2 |
| 22:07:03 | NON-GET | 32099 | 0xCE33 | Uri-Path: SendPackets, Accept: 50, Block2: 0/0/128 | |
| 22:07:03 | NON-2.05 Content | 32100 | 0xCE33 | Content-Format: 0 | 3 |

2. Wysłanie żądania NON-PUT do zasobu *Edges* przechowującego krawędzie w celu dodania nowych krawędzi:

| Time | CoAP Message | MID | Token | Options | Payload |
|----------|------------------|-------|--------|-----------------------------|---------|
| 22:13:22 | NON-PUT | 38339 | 0xCE33 | Uri-Path: Edges, Accept: 50 | 1,2 |
| 22:13:22 | NON-2.01 Created | 38340 | 0xCE33 | Content-Format: 0 | |
| 22:13:27 | NON-PUT | 38340 | 0xCE33 | Uri-Path: Edges, Accept: 50 | 1,3 |
| 22:13:27 | NON-2.01 Created | 38341 | 0xCE33 | Content-Format: 0 | |
| 22:13:31 | NON-PUT | 38341 | 0xCE33 | Uri-Path: Edges, Accept: 50 | 1,4 |
| 22:13:31 | NON-2.01 Created | 38342 | 0xCE33 | Content-Format: 0 | |

3. Wysłanie żądania NON-GET do zasobu *Edges* by otrzymać listę wszystkich przechowywanych krawędzi:

| Time | CoAP Message | MID | Token | Options | Payload |
|----------|------------------|-------|--------|----------------------------------|-------------|
| 22:19:17 | NON-GET | 44586 | 0xCE33 | Uri-Path: Edges, Block2: 0/0/128 | |
| 22:19:17 | NON-2.05 Content | 44587 | 0xCE33 | Etag: 0x6101, Content-Format: 0 | 1-2 |
| 22:19:29 | NON-GET | 44587 | 0xCE33 | Uri-Path: Edges, Block2: 0/0/128 | |
| 22:19:29 | NON-2.05 Content | 44588 | 0xCE33 | Etag: 0x6201, Content-Format: 0 | 1-2,1-3 |
| 22:19:41 | NON-GET | 44588 | 0xCE33 | Uri-Path: Edges, Block2: 0/0/128 | |
| 22:19:41 | NON-2.05 Content | 44589 | 0xCE33 | Etag: 0x6301, Content-Format: 0 | 1-2,1-3,1-4 |

4. Wysłanie żądania NON-GET do zasobu *CentralVertices* by otrzymać listę wszystkich wierzchołków centralnych:

| Time | CoAP Message | MID | Token | Options | Payload |
|----------|------------------|-------|--------|--|---------|
| 22:22:05 | NON-GET | 10390 | 0xCE33 | Uri-Path: CentralVertices, Block2: 0/0/128 | |
| 22:22:05 | NON-2.05 Content | 10391 | 0xCE33 | Content-Format: 0 | 1,2 |
| 22:22:21 | NON-GET | 10391 | 0xCE33 | Uri-Path: CentralVertices, Block2: 0/0/128 | |
| 22:22:21 | NON-2.05 Content | 10392 | 0xCE33 | Content-Format: 0 | 2 |
| 22:22:40 | NON-GET | 10392 | 0xCE33 | Uri-Path: CentralVertices, Block2: 0/0/128 | |
| 22:22:40 | NON-2.05 Content | 10393 | 0xCE33 | Content-Format: 0 | 2,3 |

5. Wysłanie żądań NON-PUT do zasobu *Edges* przechowującego krawędzie aż do zapelnienia pamięci przeznaczonej na krawędzie:

| Time | CoAP Message | MID | Token | Options | Payload |
|----------|--------------------------------|-------|--------|-------------------|---------|
| 22:26:36 | NON-2.01 Created | 20095 | 0xCE33 | Content-Format: 0 | |
| 22:26:42 | NON-PUT | 20095 | 0xCE33 | Uri-Path: Edges | 4,8 |
| 22:26:42 | NON-2.01 Created | 20096 | 0xCE33 | Content-Format: 0 | |
| 22:26:47 | NON-PUT | 20096 | 0xCE33 | Uri-Path: Edges | 4,9 |
| 22:26:47 | NON-2.01 Created | 20097 | 0xCE33 | Content-Format: 0 | |
| 22:26:55 | NON-PUT | 20097 | 0xCE33 | Uri-Path: Edges | 5,9 |
| 22:26:55 | NON-5.00 Internal Server Error | 20098 | 0xCE33 | Content-Format: 0 | |

6. Wysłanie żądania NON-GET do zasobu *GetNumber* zwracającego ilość odebranych żądań typu GET:

| CoAP Message Log | | | | | |
|------------------|------------------|-------|--------|--------------------------------------|---------|
| Time | CoAP Message | MID | Token | Options | Payload |
| 22:34:04 | NON-GET | 28164 | 0xCE33 | Uri-Path: GetNumber, Block2: 0/0/128 | |
| 22:34:04 | NON-2.05 Content | 28165 | 0xCE33 | Content-Format: 0 | 9 |
| 22:34:07 | NON-GET | 28165 | 0xCE33 | Uri-Path: GetNumber, Block2: 0/0/128 | |
| 22:34:07 | NON-2.05 Content | 28166 | 0xCE33 | Content-Format: 0 | 10 |
| 22:34:08 | NON-GET | 28166 | 0xCE33 | Uri-Path: GetNumber, Block2: 0/0/128 | |
| 22:34:08 | NON-2.05 Content | 28167 | 0xCE33 | Content-Format: 0 | 11 |

7. Wysłanie żądania NON-GET do zasobu *PutNumber* zwracającego ilość odebranych żądań typu PUT:

| CoAP Message Log | | | | | |
|------------------|------------------|-------|--------|--------------------------------------|---------|
| Time | CoAP Message | MID | Token | Options | Payload |
| 22:38:26 | NON-GET | 32081 | 0xCE33 | Uri-Path: PutNumber, Block2: 0/0/128 | |
| 22:38:26 | NON-2.05 Content | 32082 | 0xCE33 | Content-Format: 0 | 0 |
| 22:38:32 | NON-GET | 32082 | 0xCE33 | Uri-Path: PutNumber, Block2: 0/0/128 | |
| 22:38:32 | NON-2.05 Content | 32083 | 0xCE33 | Content-Format: 0 | 1 |
| 22:38:40 | NON-GET | 32083 | 0xCE33 | Uri-Path: PutNumber, Block2: 0/0/128 | |
| 22:38:40 | NON-2.05 Content | 32084 | 0xCE33 | Content-Format: 0 | 2 |

8. Wysłanie żądania NON-GET do zasobu *PutNumber* z opcją Observe (rozpoczęcie obserwowania zasobu).

| CoAP Message Log | | | | | |
|------------------|------------------|-----------|--------|---|---------|
| Time | CoAP Message | MID | Token | Options | Payload |
| 22:05:11 | CON-GET | 14846 (0) | 0xFE5D | Observe: 0, Uri-Path: PutNumber, Accept: 50, Block2: 0/0/64 | |
| 22:05:11 | NON-2.05 Content | 14847 | 0xFE5D | Etag: 0x08, Observe: 11, Content-Format: 0, Max-Age: 15 | 0 |
| 22:05:15 | NON-2.03 Valid | 14848 | 0xFE5D | Etag: 0x08, Observe: 12, Content-Format: 0, Max-Age: 15 | |
| 22:05:19 | NON-2.03 Valid | 14849 | 0xFE5D | Etag: 0x08, Observe: 13, Content-Format: 0, Max-Age: 15 | |

9. Wysłanie żądań NON-GET do zasobu *Edges* :

| CoAP Message Log | | | | | |
|------------------|------------------|-----|--------|-----------------------------------|---------|
| Time | CoAP Message | MID | Token | Options | Payload |
| 22:42:21 | NON-GET | 670 | 0xCE33 | Uri-Path: Edges, Block2: 0/0/128 | |
| 22:42:21 | NON-2.05 Content | 671 | 0xCE33 | Etag: 0xD505F8, Content-Format: 0 | 3-8,3-2 |
| 22:42:24 | NON-GET | 671 | 0xCE33 | Uri-Path: Edges, Block2: 0/0/128 | |
| 22:42:24 | NON-2.03 Valid | 672 | 0xCE33 | Etag: 0xD505F8, Content-Format: 0 | |
| 22:42:25 | NON-GET | 672 | 0xCE33 | Uri-Path: Edges, Block2: 0/0/128 | |
| 22:42:25 | NON-2.03 Valid | 673 | 0xCE33 | Etag: 0xD505F8, Content-Format: 0 | |

10. Wysłanie żądań NON-GET do zasobu *CentralVertices* z różnymi opcjami Accept :

| CoAP Message Log | | | | | |
|------------------|------------------|-------|--------|--|---------------------------|
| Time | CoAP Message | MID | Token | Options | Payload |
| 22:44:34 | NON-GET | 48209 | 0xCE33 | Uri-Path: CentralVertices, Accept: 0, Block2: 0/0/128 | |
| 22:44:34 | NON-2.05 Content | 48210 | 0xCE33 | Content-Format: 0 | 3 |
| 22:44:43 | NON-GET | 48210 | 0xCE33 | Uri-Path: CentralVertices, Accept: 40, Block2: 0/0/128 | |
| 22:44:43 | NON-2.05 Content | 48211 | 0xCE33 | Content-Format: 40 | </3>; |
| 22:44:49 | NON-GET | 48211 | 0xCE33 | Uri-Path: CentralVertices, Accept: 50, Block2: 0/0/128 | |
| 22:44:49 | NON-2.05 Content | 48212 | 0xCE33 | Content-Format: 50 | {"Central Vertices": "3"} |

192.168.1.56:5683

- .well-known
- core
- CentralVertices**
- Edges
- GetNumber
- PutNumber
- SendPackets

| Header | Value | Option | Value |
|--------|--------------|----------------|------------------|
| Type | ACK | Content-Format | application/json |
| Code | 2.05 Content | | |
| MID | 9984 | | |
| Token | 0xFFFF | | |

Payload (25)

Incoming Rendered Outgoing

```
{
  Central Vertices: 9
}
```

11. Po włączeniu opcji “Observe” dla metryki PutNumber, pierwszej wartości (w tym przypadku 0) nadawany jest eTag. Wraz z przesyłaniem kolejnych pakietów, przesyłany jest tylko eTag.

| CoAP Message Log | | | | | |
|------------------|------------------|-------|--------|---|---------|
| Time | CoAP Message | MID | Token | Options | Payload |
| 22:06:32 | NON-2.03 Valid | 14869 | 0xFE5D | Etag: 0x08, Observe: 33, Content-Format: 0, Max-Age: 15 | |
| 22:06:35 | NON-2.03 Valid | 14870 | 0xFE5D | Etag: 0x08, Observe: 34, Content-Format: 0, Max-Age: 15 | |
| 22:06:38 | NON-2.05 Content | 1 | 0xFE5D | Observe: 35, Content-Format: 0, Max-Age: 15 | 1 |
| 22:06:41 | NON-2.05 Content | 2 | 0xFE5D | Observe: 36, Content-Format: 0, Max-Age: 15 | 1 |
| 22:06:45 | NON-2.05 Content | 3 | 0xFE5D | Observe: 37, Content-Format: 0, Max-Age: 15 | 1 |
| 22:06:48 | NON-2.05 Content | 4 | 0xFE5D | Etag: 0x2801, Observe: 38, Content-Format: 0, Max-Age: 15 | 1 |
| 22:06:51 | NON-2.03 Valid | 5 | 0xFE5D | Etag: 0x2801, Observe: 39, Content-Format: 0, Max-Age: 15 | |

W momencie zmiany wartości parametru PutNumber, kolejne pakiety są przesyłane z pełnym payload’em. W przypadku gdy kolejne 4 pakiety przyjdą z tym samym payload’em, nadawany jest nowy eTag.

12. Opcja Accept i Content-format:

Działanie opcji Accept na przykładzie:

A) GET funkcji Edges:

192.168.1.56:5683
well-known
core
CentralVertices
Edges
GetNumber
PutNumber
SendPackets

| Header | Value | Option | Value |
|--------|--------------|----------------|------------------|
| Type | ACK | Etag | 0x4806 |
| Code | 2.05 Content | Content-Format | application/json |
| MID | 16145 | | |
| Token | 0xFFFF | | |

Payload (15)
Incoming Rendered Outgoing

```
{
  Edges: 9-5
}
```

B) GET funkcji CentralVertices:

192.168.1.56:5683
well-known
core
CentralVertices
Edges
GetNumber
PutNumber
SendPackets

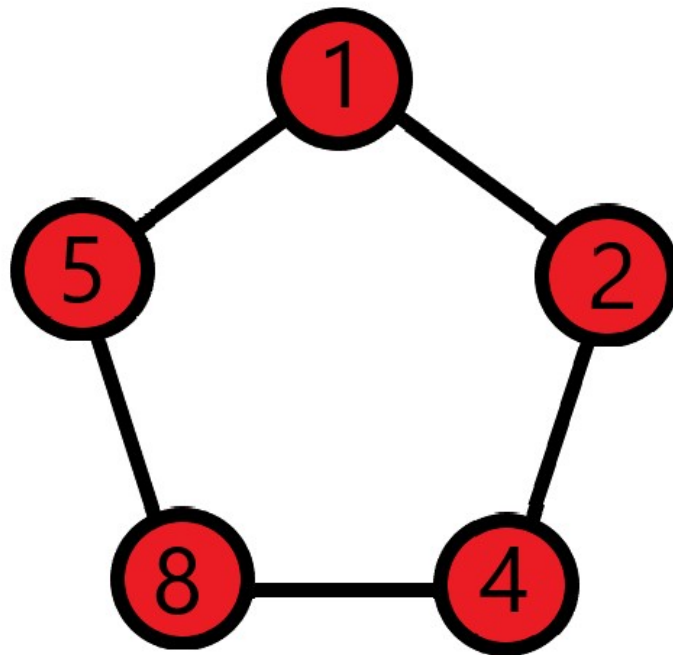
| Header | Value | Option | Value |
|--------|--------------|----------------|------------------|
| Type | ACK | Content-Format | application/json |
| Code | 2.05 Content | | |
| MID | 9984 | | |
| Token | 0xFFFF | | |

Payload (25)
Incoming Rendered Outgoing

```
{
  Central Vertices: 9
}
```

4. Skrypt demonstracji

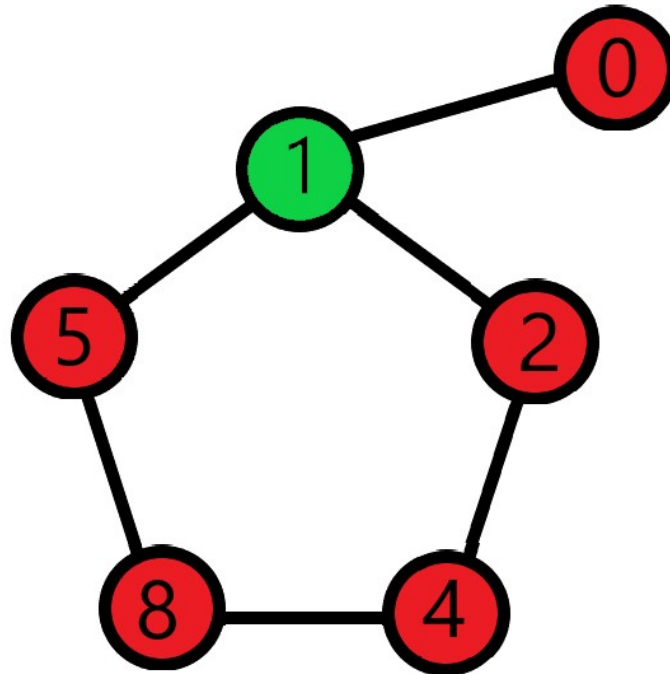
1. Użycie funkcji Discover – znalezienie wszystkich zasobów
2. Wykonanie funkcji GET dla wszystkich zasobów
3. Dodanie krawędzi 1-2 (Edges->PUT)
4. Wyświetlenie dostępnych krawędzi (Edges ->GET)
5. Wyświetlenie wierzchołków wewnętrznych we wszystkich formatach ACCEPT (CentralVertices-> GET)
6. Pokazanie funkcji Etag dla Edges (Edges->GET)
7. Uruchomienie obserwatora dla zasobu PutNumber
8. Dodawanie krawędzi (Edges->PUT)
Utworzenie ścieżek: 1-2, 2-4, 4-8, 8-5, 5-1. Każdy z wierzchołków jest wierzchołkiem centralnym.



9. Pokazanie kodów błędów dla zasobu Edges (Edges->PUT):
 - „6-5” – niespójny graf;
 - „5-5”- takie same wierzchołki;
 - „5.5” – złe formatowanie;
 - „1-2” – krawędź już istnieje
10. Pokazanie wierzchołków centralnych (CentralVertices->GET)
11. Pokazanie działanie obserwatora, a następnie jego zamknięcie
12. Pokazanie, że niezaimplementowane funkcje zwracają odpowiedni kod błędu

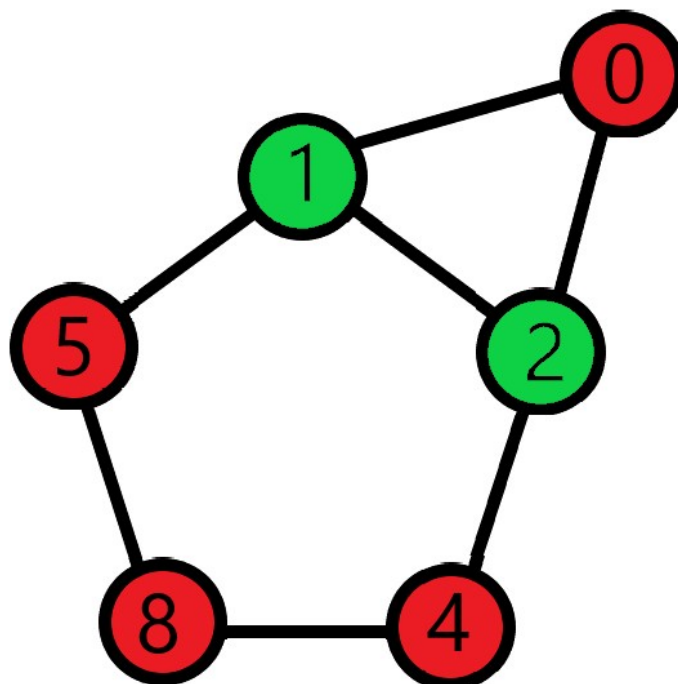
13. Dodanie krawędzi (Edges->PUT) i wyświetlenie wierzchołków wewnętrznych (CentralVertices->GET)

Dodanie ścieżki 1-0 powoduje, że wierzchołek nr 1 staje się wierzchołkiem wewnętrznym (wierzchołki wewnętrzne są zaznaczone na zielono).

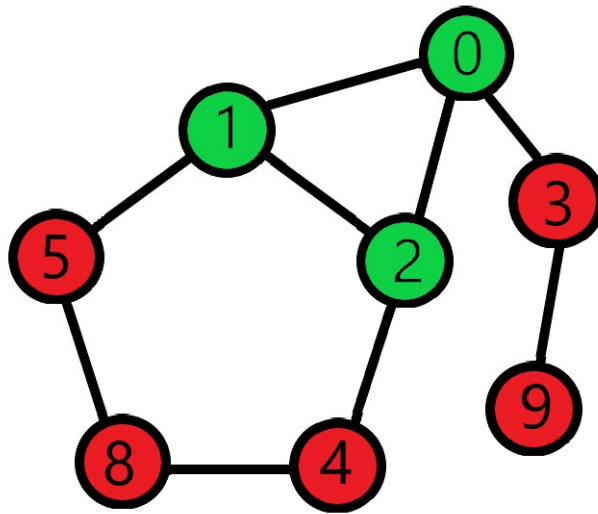


14. Dodanie krawędzi (Edges->PUT) i wyświetlenie wierzchołków wewnętrznych (CentralVertices->GET)

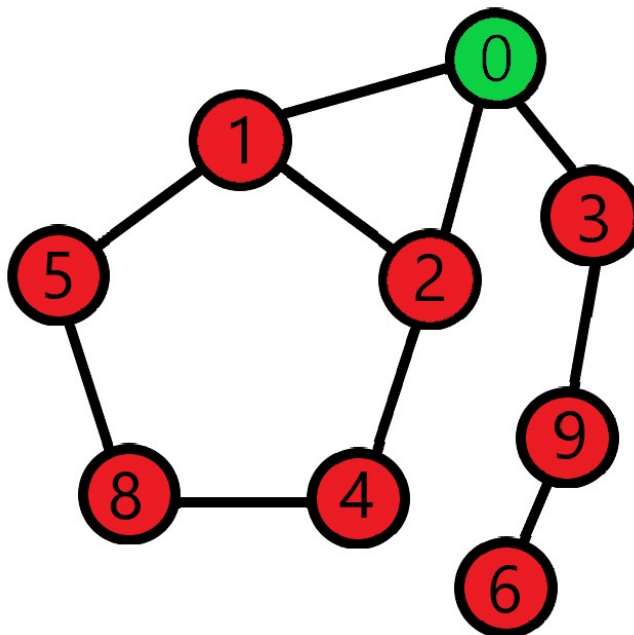
Utworzenie ścieżki 0-2.



15. Dodanie krawędzi (Edges->PUT) i wyświetlenie wierzchołków wewnętrznych (CentralVertices->GET)
Dodanie ścieżek 0-3 i 3-9.



16. Dodanie krawędzi (Edges->PUT) i wyświetlenie wierzchołków wewnętrznych (CentralVertices->GET)
Dodanie ścieżki 9-6.



17. Dodanie krawędzi (Edges->PUT) – pokazanie błędu wynikającego z braku pamięci

18. Wywołanie funkcji GET dla każdego zasobu

19. Wywołanie funkcji GET dla zasobu GetNumber kilka razy w celu ukazania zmieniania się wartości metryki