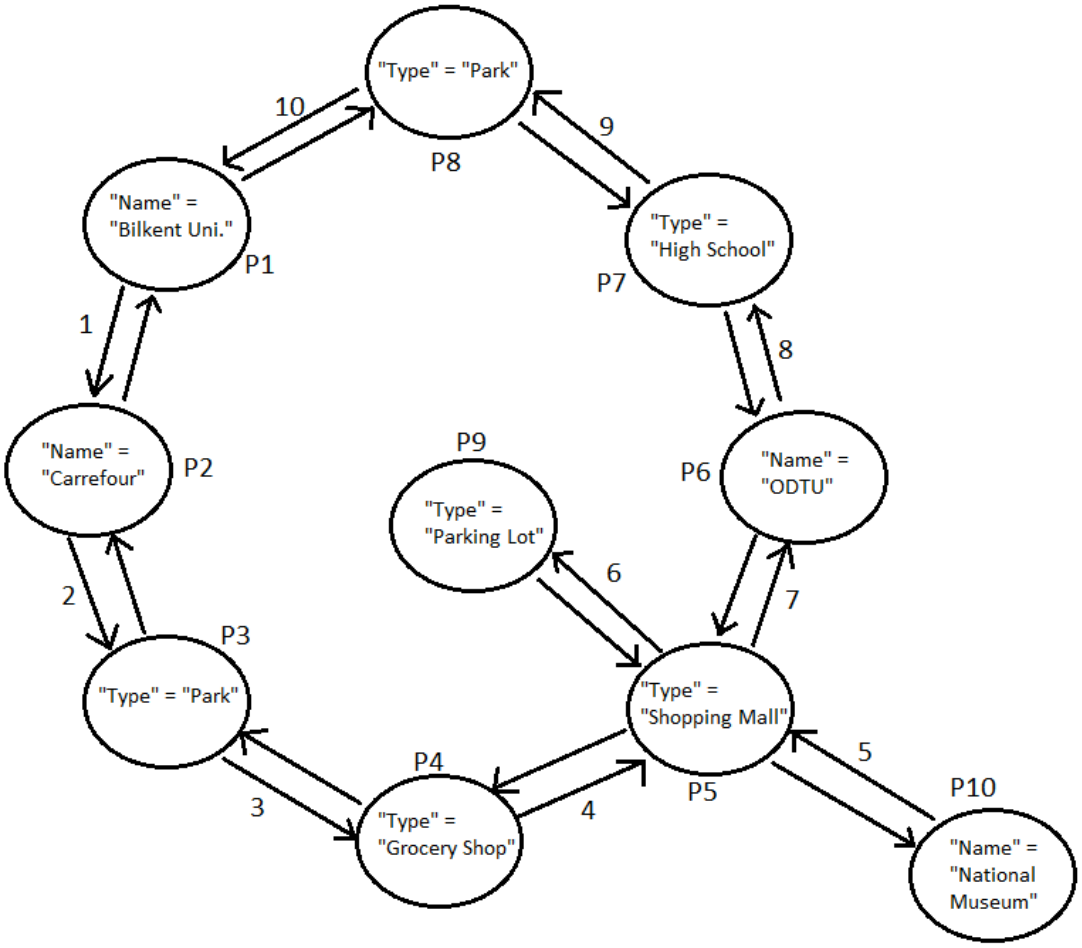
Two Way Street Network Definition Sample:

Sample Code:

```
sn2 = SN(=>);           //create a two way street network

point1 = Point();       //create a new point

//add properties with string values to point1.
point1.addProperty(Property("Name", "Bilkent University"));
point1.addProperty(Property("Type", "University"));

//We will demonstrate how to add a List as a property Value
buildingList = List();
            //This will be: {'A Building', 'EA Building', 'B
            Building', 'T Building', 'EE Building'}
buildingList.append("A Building");
buildingList.append("EA Building");
buildingList.append("B Building");
buildingList.append("T Building");
buildingList.append("EE Building");

point1.addProperty(Property("Buildings", buildingList));

point2 = Point();
point2.addProperty(Property("Name", "Carrefour"));

//We will demonstrate how to add a Map as a property value.
//The map will also have lists in it.
saleOptionsMap = Map();
//This will be: {{'Product Types': 'Technology', 'Food'},
//{'Payment Options': 'Cash', 'Visa', 'Mastercard'}}
list1 = List();   //will contain the list of product-types
list1.append("Technology");
list1.append("Food");
list2 = List();   //will contain the list of payment-options
list2.append("Cash");
list2.append("Visa");
list2.append("Mastercard");
saleOptionsMap.add("Product types", list1)
saleOptionsMap.add("Payment methods", list2);
point1.addProperty("Sells", saleOptionsMap);
//add this map as the value of the property


//Let us create a street which we can pass in 30 minutes
//from Bilkent University to Carrefour
street1 = Street(30, point1, point2);

//Let us create some points and streets
point3 = Point();
point4 = Point();
point5 = Point();
point6 = Point();
point7 = Point();
point8 = Point();
point9 = Point();
point10 = Point();
```

```
point2.addProperty(Property("Type", "Grocery Shop"));
point3.addProperty(Property("Type", "Park"));
point4.addProperty(Property("Type", "Grocery Shop"));
point5.addProperty(Property("Type", "Shopping Mall"));
point5.addProperty(Property("Payment-Options", "Cash");
point6.addProperty(Property("Name", "ODTU"));
point7.addProperty(Property("Type", "High School"));
point8.addProperty(Property("Type", "Park"));
point9.addProperty(Property("Type", "Parking Lot"));
point10.addProperty(Property("Name", "National Museum"));

//Let us remove a property
point8.removeProperty(0);     //removes property at index 0.
point8.addProperty(Property("Type", "Park"));
//add new property

street2 = Street(5, point2, point3);
street3 = Street(13, point3, point4);
street4 = Street(20, point4, point5);
street5 = Street(25, point5, point10);
street6 = Street(15, point5, point9);
street7 = Street(15, point5, point6);
street8 = Street(25, point6, point7);
street9 = Street(30, point7, point8);s
street10 = Street(15, point8, point1);

//Let us add a delay to a street and remove it
street1.addDelay(10, "Presidential pass");
//10 minute delay due to special purpose
street1.removeDelay(0); //Let us remove this delay

//Let us close a street and re-open it
street1.closed(Time(10, 30), Time(15,25), "Traffic accident");
//Street is closed temporarily from 10:30 to 15:25 due to
//traffic accident
street1.opened()

//Add these streets to our street network
sn2.addStreet(street1);
sn2.addStreet(street2);
sn2.addStreet(street3);
sn2.addStreet(street4);
sn2.addStreet(street5);
sn2.addStreet(street6);
sn2.addStreet(street7);
sn2.addStreet(street8);
sn2.addStreet(street9);
sn2.addStreet(street10);
```

Graph for This Code:



Query Examples:

```
predicate1 = (“Type” == “Park”);
predicate2 = (“Name” == “Bilkent University”)
predicate3 = (“Type” == “Grocery Shop”)
predicate4 = (“Name” == “Carrefour”)
predicate5 = (“Payment Options” == “Cash”)
predicate6 = (“Name” == vary(x))
predicate7 = (“Name” == vary(y))
predicate8 = (“Name” == “National Museum”)

predicate9 = (predicate6)alter(predicate7)
predicate10 = (predicate1)concat(predicate2)concat(predicate8)
predicate11 = ((predicate4)concat(predicate7))rep(2)
predicate12 = (predicate3)rep(2)concat(predicate8)
predicate13 = (predicate12)alter(predicate11)

sn2.findRoute(predicate9);
sn2.findRoute(predicate10);
sn2.findRoute(predicate11);
sn2.findRoute(predicate12);
sn2.findRoute(predicate13);
```