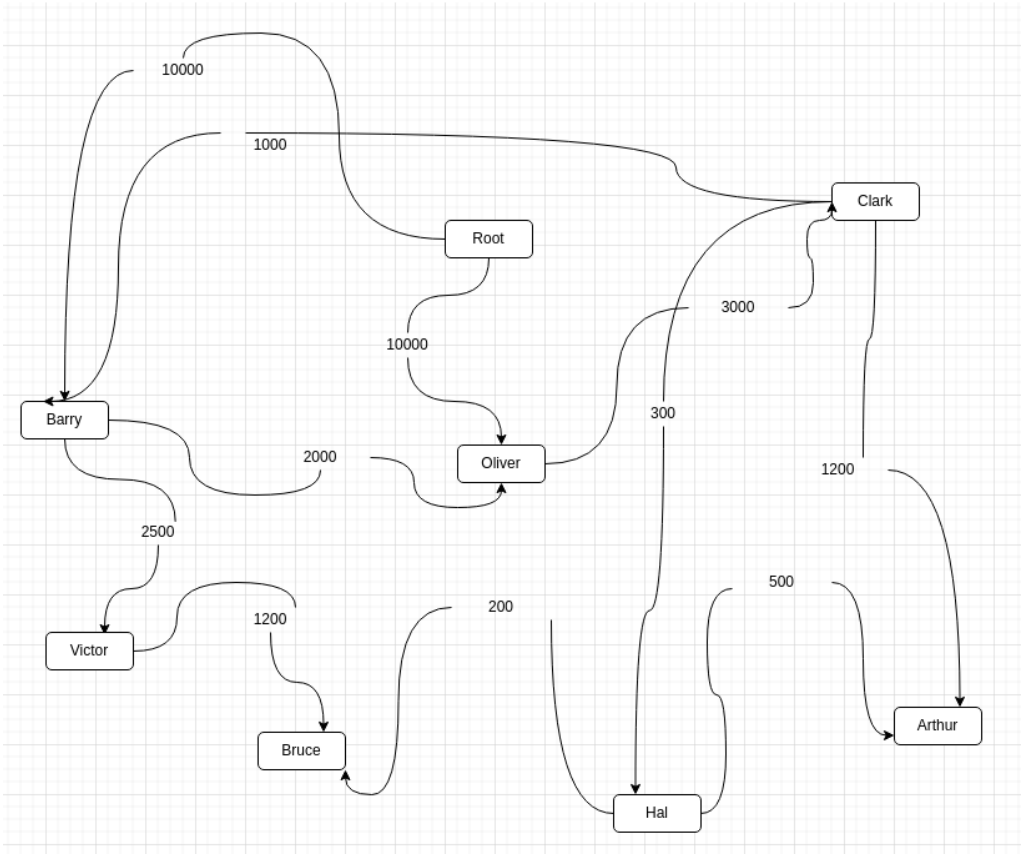
NAME : ANKIT KUMAR SAHU

REG NO. : 20BAI1005

LAB 5

A* SEARCH

- **Transaction History of a cryptocurrency**



```
def A_star(start, dest):
        dist_from_start = {}
        open_set = set(start)
        closed_set = set()
        parents = {}
        dist_from_start[start] = 0
        parents[start] = start

        while len(open_set) > 0:
            n = None

            for v in open_set:
                if n == None or dist_from_start[v]+heuristic(v) < dist_from_start[n]+heuristic(n):
                    n = v

            if n == dest or Graph_nodes[n] == None:
                pass
            else:
                for (m, weight) in get_neighbors(n):
                    if m not in open_set and m not in closed_set:
                        open_set.add(m)
                        parents[m] = n
                        dist_from_start[m] = dist_from_start[n]+weight

                    else:
                        if dist_from_start[m] > dist_from_start[n]+weight:
                            dist_from_start[m] = dist_from_start[n] + weight
                            parents[m] = n

                            if m in closed_set:
                                closed_set.remove(m)
                                open_set.add(m)

            if n == None:
                print('Path not found')
                return None

            if n == dest:
                path = []

                while parents[n] != n:
                    path.append(n)
                    n = parents[n]

                path.append(start)
                path.reverse()
                print('Path : {}'.format(path))
                return path


            open_set.remove(n)
            closed_set.add(n)

        print('Path not found')
        return None
```

```
def get_neighbors(v):
    if v in Graph_nodes:
        return Graph_nodes[v]
    else:
        return None
def heuristic(n):
        H_dist = {
            'R': 10000,
            'O': 8000,
            'C': 1200,
            'B': 3000,
            'V': 1000,
            'B': 1200,
            'H': 300,
            'A': 0,
        }

        return H_dist[n]

Graph_nodes = {
    'R': [('B', 10000), ('O', 10000)],
    'O': [('C', 3000)],
    'B': [('O', 2000),('V', 2500)],
    'V': [('B', 1200)],
    'A': None,
    'C': [('H', 3000), ('A', 1200),('B', 1000)],
    'H': [('B', 200),('A', 500)],
    'B': None,
}

A_star('R', 'A')
```

```
Path : ['R', 'O', 'C', 'A']

['R', 'O', 'C', 'A']
```