UWI

Design and Analysis of Algorithms

Tutorial 3

Brute Force Approach

It is important we learn to calculate the time efficiencies for simple brute force algorithms and to also examine the theories behind them as well as how to work out simple examples of each.

1. Solve the travelling salesman problem using the brute force approach for the instance shown in figure 1 where the starting city is **a**.

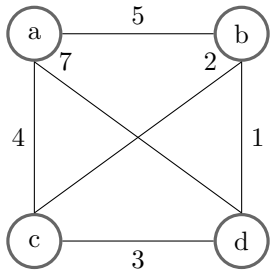

Figure 1: A small instance of the travelling salesman problem

2. Solve the 0/1 knapsack problem using the brute force approach for the instance shown in table 1 with a max knapsack capacity of **12**.

Item	Weight	Value
1	5	$20
2	3	$15
3	9	$30
4	4	$25

Table 1: A small instance of the 0/1 knapsack problem

3. Solve the assignment problem using the brute force approach for the instance shown in table 2.

	Job 1	Job 2	Job 3
Person 1	5	7	9
Person 2	3	8	5
Person 3	7	4	9

Table 2: A small instance of the assignment problem

4. Analyse the following Bubble Sort algorithm and calculate its time efficiency using summation rules.

for $i \leftarrow 0$ to n–2 **do**
  for $j \leftarrow 0$ to $n-2-i$ **do**
    if $A[j] > A[j+1]$ **then**

```
            swap A[j] and A[j + 1]
        end if
    end for
end for
```

5. Analyse the following Selection Sort algorithm and calculate its time efficiency using summation rules.

```
for i ← 0 to n–2 do
    min ← i
    for j ← i + 1 to n–1 do
        if A[j] < A[min] then
            min ← j
        end if
        swap A[i] and A[min]
    end for
end for
```