



INFORMATICS
INSTITUTE OF
TECHNOLOGY

UNIVERSITY OF
WESTMINSTER[®]

Informatics Institute of Technology

Department of computing

(B.Eng.) Software Engineering

Module: 5SENG003C.2

Algorithms: Theory, Design, and Implementation

COURSEWORK REPORT

Student ID -: 20200649

Student UoW ID -: w1866979

Student Name -: R.M.S.J. Bandara

Data Structure used in this course work.

Array Lists

An array list is dynamic array. This indicates that when data was added, the array's size rose. This data structure has added, remove, and get data from array operations. Array List is mainly use in this course work.

Set with LinkedHashSet

This is utilized to directly store distinct elements without filtering. Only unique values will be stored here, so even if there are duplicates, they will only be added once. This is used in the course work to read the vertices and edges from the file

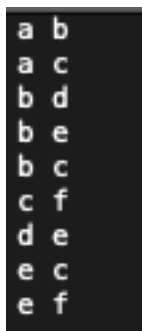
Algorithm used

Finding the sink and removing each sink until the graph is empty will show if the graph is acyclic or not if all the vertices are eliminated in this implementation.

Using depth first search to obtain the cycle's vertices, a second graph was created to depict the cycle.

Run of the Algorithm Sample 1

Edges



```
a b
a c
b d
b e
b c
c f
d e
e c
e f
```

Output

```
---Graph does not have a cycle---  
  
Removed vertices : f,c,e,d,b,a  
  
---Graph is an Acyclic---
```

Sample 2

Edges

```
a b  
a c  
b d  
b e  
c b  
c f  
d e  
e c  
e f
```

Output

```
c --> b  
b --> d  
d --> e  
e --> c  
  
---Graph does have a cycle---  
  
---Graph is not an Acyclic---
```

A performance analysis

It will take $O(n)$ time to read the input file line by line, where n is the number of lines. The quantity of edges and vertices will affect how well this performs. The quantity of vertices and edges affects every aspect of performance, from graph construction to depth first search. Work your way through each vertex and edge to find the cycle program. Check for edges in each vertex while you search for the sink program. This means that if the input is enormous (for example, a big graph), the application will run slower. The program will execute rapidly if the input is small (the graph is tiny).