COMP3010

Assignment 4

Chaz Lambrechtsen

45426317

Table of Contents

[2 Problem A 2](#_Toc55232382)

[2.1 Flow Network 2](#_Toc55232383)

[2.2 Question 1 2](#_Toc55232384)

[2.3 Question 2 2](#_Toc55232385)

[3 Problem B 2](#_Toc55232386)

[3.1 Flow Network 2](#_Toc55232387)

[3.2 Question 1 2](#_Toc55232388)

[4 Problem C 2](#_Toc55232389)

[4.1 Flow Network 2](#_Toc55232390)

[4.2 Question 1 3](#_Toc55232391)

[4.3 Question 2 3](#_Toc55232392)

[5 Problem D 3](#_Toc55232393)

[5.1 Flow Network 3](#_Toc55232394)

[5.2 Question 1 3](#_Toc55232395)

[6 Problem E 3](#_Toc55232396)

[6.1 Flow Network 3](#_Toc55232397)

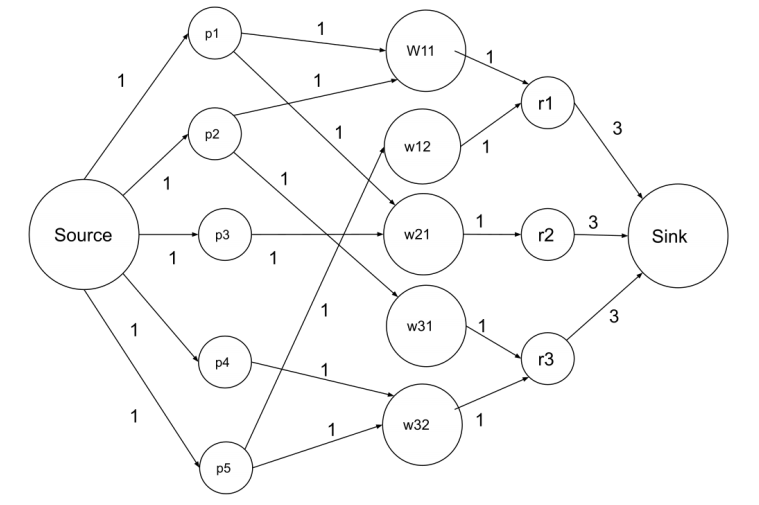
[6.2 Question 1 3](#_Toc55232398)

[6.3 Question 2 3](#_Toc55232399)

[6.4 Question 3 3](#_Toc55232400)

# Problem A

## Flow Network

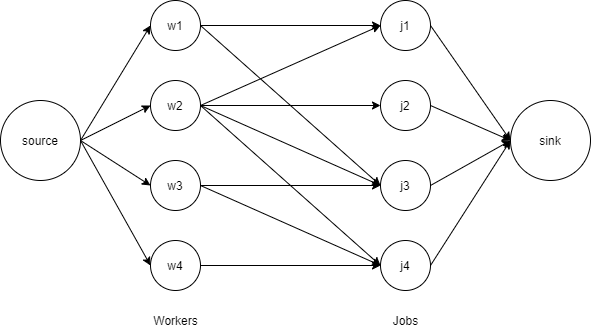


## Question 1

## Question 2

# Problem B

## Flow Network



## Question 1

Yes, it is possible to use the greedy approach to solve the job assignment problem, however it is not the most optimal solution.

Solving via reduction to maximum flow:

For example, using Ford-Fulkerson (Greedy Algorithm), all residual capacities and flows are 0-1; flow corresponds to edges in a matching F.

Residual Graph RG simplifies to:

If (w, j) ∉ F, then (w, j) is in RG.

If (w, j) ∈ F, then (w, j) is in RG.

* Augmenting path simplifies to:
* Edge from s to an unmatched node w ∈ Workers,
* Alternating sequence of unmatched and matched edges,
* Edge from unmatched node j ∈ Jobs to sink.

# Problem C

## Flow Network

## Question 1

## Question 2

# Problem D

## Flow Network

## Question 1

# Problem E

## Flow Network

## Question 1

## Question 2

## Question 3