COMP2521 23T3

Algorithm

Pseudocode

Example

raul rilluli

vertex Set

Allalysis

Algorithms

Appendix

COMP2521 23T3 Dijkstra's Algorithm

Kevin Luxa cs2521@cse.unsw.edu.au

shortest path dijkstra's algorithm

Pseudocod

Example

Path Findi

Vertex Se

Analysi

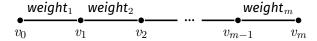
Algorithms

Appendix

In a weighted graph...

A path is a sequence of edges connected end-to-end

$$(v_0,v_1,w_1)$$
 , (v_1,v_2,w_2) , ... , (v_{m-1},v_m,w_m)



The cost of a path is the sum of edge weights along the path

The shortest path between two vertices s and t is the path from s to t with minimum cost

Pseudocoo

Example

Patri Findi

Vertex Set

Analysi

Other Algorithm

Appendi

Variations on shortest path problem:

- Source-target shortest path
 - ullet Shortest path from source vertex s to target vertex t
- Single-source shortest path
 - Shortest path from source vertex s to all other vertices
- All-pairs shortest path
 - Shortest path between all pairs of source and target vertices

Pseudocoo

Example

Path Findir

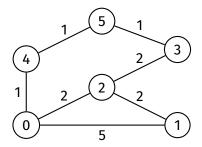
Vertex Se

AllatySi

Algorithms

Appendix

In a weighted graph, the shortest path between two vertices is not necessarily the path with the least number of edges



Dijkstra's Algorithm

lgorithm

seudocod

Example

i deii i iiidii

vertex 36

Analysis

Algorithms

Appendix

Dijkstra's algorithm is used to find the shortest path in a weighted graph with non-negative weights

. . .

Pseudocod

Patri Findir

Δnalvsis

Other Algorithms

Appendi

Data structures used in Dijkstra's algorithm:

- Distance array (dist)
 - To keep track of shortest currently known distance to each vertex
- Predecessor array (pred)
 - Same purpose as in BFS/DFS
 - To keep track of the predecessor of each vertex on the shortest currently known path to that vertex
 - Used to construct the shortest path
- Set of vertices
 - Stores unexplored vertices

- Create and initialise data structures
 - Initialise distance array to infinity
 - In C, can use INT MAX (from <limits.h>)
 - Initialise predecessor array to -1
 - Initialise set of vertices to contain all vertices
- Set distance of starting vertex to 0
- While set of vertices is not empty:
 - Remove vertex from vertex set with smallest distance in distance array
 - Let this vertex be v
 - **2 Explore** v that is, for each edge v-w:
 - ullet Check if using this edge gives a shorter path to w
 - If so, update w's distance and predecessor this is called edge relaxation

Edge relaxation

Pseudoco

Example

Patri Findii

Analycic

Other

..

Appendix

During Dijkstra's algorithm, the dist and pred arrays:

- contain data about the shortest path discovered so far
- need to be updated if a shorter path to some vertex is found
 - this is done via edge relaxation

Algorithm Edge relaxation

Pseudocode

rseudocode

.

i atii i iiiaii

Analycic

Other

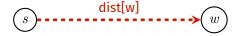
Suppose we are considering edge (v, w, weight).

We have the following data:

ullet dist[v] - length of shortest known path from s to v



ullet dist[w] - length of shortest known path from s to w



In edge relaxation, we take the shortest known path from s to v and extend it using edge (v, w, weight) to create a new path from s to w.



Edge relaxation

Example

Path Findir

Vertex Set

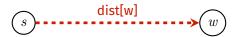
Analysis

Algorithms

A m m a m ali

Now we have two paths from s to w:

Shortest known path:



• New path via v:



If the new path is shorter, then we update dist[w] and pred[w].

```
if dist[v] + weight < dist[w]:
    dist[w] = dist[v] + weight
    pred[w] = v</pre>
```

Edge Relaxation

lgorithm

Edge relaxation

rseudoco

- 1

Dath Findin

...

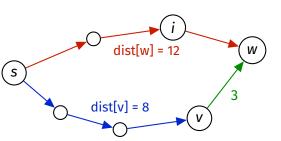
Analycic

Other

Algorithm

Appendix

Before relaxation along (v, w, 3)



	•••	[v]	•••	[w]
dist	:	8	:	12
pred		•••	•••	i

Edge Relaxation

Edge relaxation

Pseudoco

Example

Path Findir

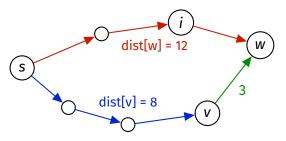
.

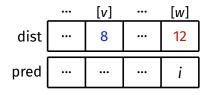
Analysis

Other Algorithms

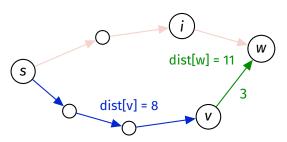
Appendi

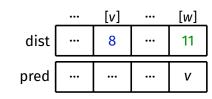
Before relaxation along (v, w, 3)





After relaxation along (v, w, 3)





```
Pseudocode
```

Eva man La

Path Findi

dijkstraSSSP(G, src):

Vertex Set

Analysis

Other Algorithm:

Appendi

```
Inputs: graph G, source vertex src

create dist array, initialised to ∞
create pred array, initialised to -1
create vSet containing all vertices of G

dist[src] = 0
while vSet is not empty:
    find vertex v in vSet such that dist[v] is minimal remove v from vSet
    for each edge (v, w, weight) in G:
        relax along (v, w weight)
```

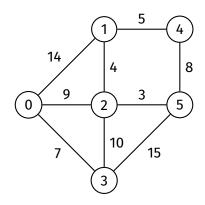
seudocod

Example

Algorithms

Appendix

Dijkstra's algorithm starting at 0



Algoritiiii

Pseudocode

Example

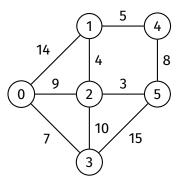
Path Findin

Analysis

Other Algorithms

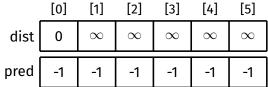
Appendi

Initialisation



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)



Pseudocode

Example

Path Finding

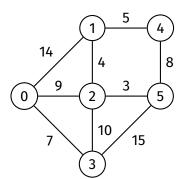
Vortov Sot

Analysis

Other Algorithms

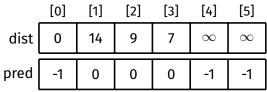
Appendi





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)



Pseudocode

Example

Path Findin

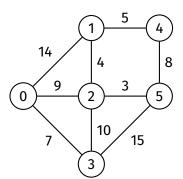
Vartov Cat

Analysis

Other Algorithms

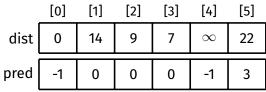
Appendi

After second iteration (v = 3)



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)



Pseudocod

Example

Path Finding

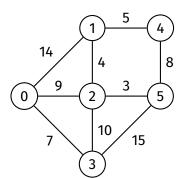
VCICCXO

Analysis

Algorithms

Appendi

After third iteration (v = 2)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)

	[0]	[1]	[2]	[3]	[4]	[5]
dist	0	13	9	7	∞	12
pred	-1	2	0	0	-1	2

Pseudocod

Example

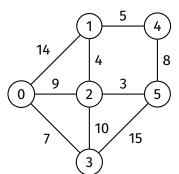
Path Findin

Analysis

Other Algorithm

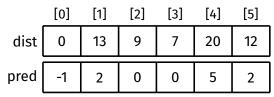
Appendi





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)



Pseudocode

Example

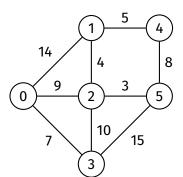
Path Findin

Analysis

Other Algorithm:

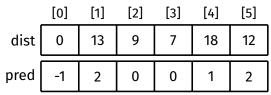
Appendi

After fifth iteration (v = 1)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)



Pseudocode

Example

Path Findin

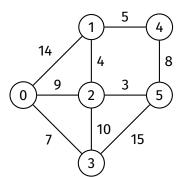
Analysis

Other

Algorithms

Appendi





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)

	[0]	[1]	[2]	[3]	[4]	[5]
dist	0	13	9	7	18	12
pred	-1	2	0	0	1	2

gorithm

Pseudocod

Example

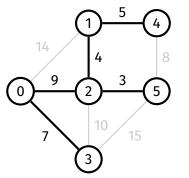
Path Findin

Analysis

Algorithm:

Appendi





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal remove v from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)

	[0]	[1]	[2]	[3]	[4]	[5]
dist	0	13	9	7	18	12
pred	-1	2	0	0	1	2

gorithn

eudoco

Example

Path Finding

VCICCXO

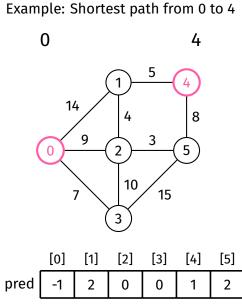
Othor

Algorithms

Appendi

The shortest path from the source vertex to any other vertex can be constructed by tracing backwards through the predecessor array (like for BFS)

Example



guiitii

Exampl

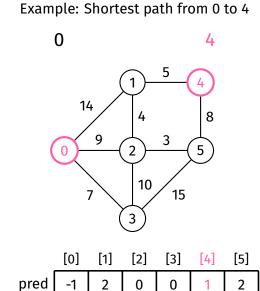
Path Fin Example

Vertex 9

Analysis

Other Algorithms

Appendix



gontin

Exampl

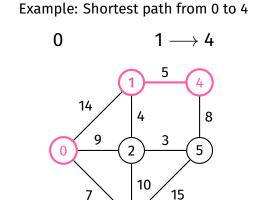
Path Fir

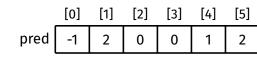
Example

Analysis

Other Algorithms

Appendix





3

Algoritii

Exampl

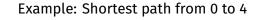
Path Fir Example

vertex

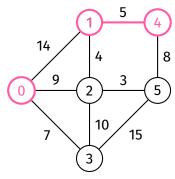
Allalysis

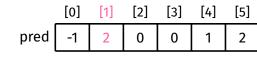
Algorithms

Appendi









,

Exampl

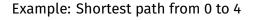
Path Fir

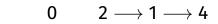
Example

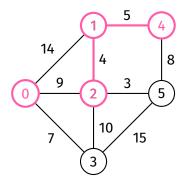
Analysis

Other

Appendix







	[0]	[1]	[2]	[3]	[4]	[5]
pred	-1	2	0	0	1	2

.gorithi

Pseudoc

Exampl

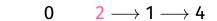
Example

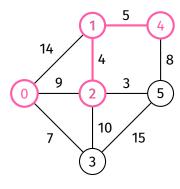
Analysis

Other

Appendix

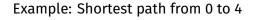
Example: Shortest path from 0 to 4

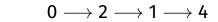


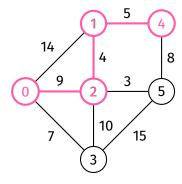


[0] [1] [2] [3] [4] [5]
pred -1 2 0 0 1 2

Appendix







	[0]	[1]	[2]	[3]	[4]	[5]
pred	-1	2	0	0	1	2

gorithm

eudoco

Example

Path Find

Example

Analysis

Other

Appendix

How to find shortest path between two other vertices (neither of which are the source vertex)?

Generally, you will need to rerun Dijkstra's algorithm from one of these vertices.

seudoco

Example

Path Findi

Vertex Set

Analysis

Algorithms

Appendix

How to implement the vSet? Different methods...

- 1. Visited array
- 2. Explicit array/list of vertices
 - 3. Priority queue

Vertex Set

Visited array implementation

Vertex Set

Visited array implementation:

- Similar to visited array in BFS/DFS
- Array of V booleans, initialised to false
- After exploring vertex v, set visited[v] to true
- At the start of each iteration, find vertex v such that visited [v] is false and dist[v] is minimal $\Rightarrow O(V)$

orithn

udoco

LAGIIIPIG

Vertex Set

Allatysis

Algorithms

Append

Array/list of vertices implementation:

- Store all vertices in an array/linked list
- After exploring vertex v, remove v from array/linked list
- At the start of each iteration, find vertex in array/list such that dist[v] is minimal $\Rightarrow O(V)$

Pseudocod

Path Findi

Vertex Set

Analysi

Other Algorithm

Appendix

Priority queue implementation:

- A priority queue is an ADT...
 - · where each item has a priority
 - with two main operations:
 - Insert: insert item with priority
 - Delete: remove item with highest priority
- Use priority queue to store vertices, use distance to vertex as priority (smaller distance = higher priority)
- A good priority queue implementation has $O(\log n)$ insert and delete

Priority queues will be discussed next week.

Analysis Correctness

Algoritiiii

Dath Fin

Vertex Se

Correctness

Other Algorithm

Appendi

Proof by induction.

Aim is to prove that before and after each iteration:

- \bullet For all explored nodes s, dist[s] is shortest distance from source to s
- ② For all unexplored nodes t, dist[t] is shortest distance from source to t via explored nodes only

Ultimately, all nodes are explored, so by **1**:

ullet For all nodes v, dist[v] is the shortest distance from source to v

Algorithm

rseudoco

Example

Vertex Se

Analysis Correctness

Other

Algorithm

Annendix

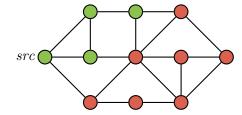
Base case:

- Start of first iteration
 - 1 holds, as there are no explored nodes
 - 2 holds, because
 - dist[source] = 0
 - For all other nodes t, $dist[t] = \infty$

Correctness

Induction step:

Assume that 1 and 2 hold at the start of an iteration



Algoritiiii

LXumpte

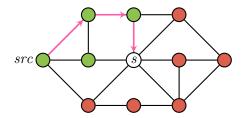
vertex se

Correctness

Other Algorithm

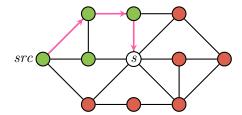
Appendix

- Assume that and hold at the start of an iteration
- Let s be an unexplored node with minimum distance



Correctness

- Assume that and and hold at the start of an iteration
- Let s be an unexplored node with minimum distance
- We claim that dist[s] is the shortest distance from source to s



Algorithm

Example

Path Findi

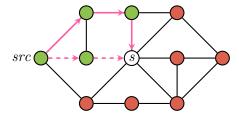
Vertex Se

Analysis Correctness

Other Algorithm

Appendix

- Assume that 1 and 2 hold at the start of an iteration
- Let s be an unexplored node with minimum distance
- We claim that dist[s] is the shortest distance from source to s
 - If there is a shorter path to s via explored nodes only, then dist[s] would have been updated when exploring the predecessor of s on that path



Algorithm

LAdiliple

Patri Findi

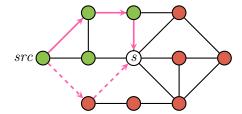
Vertex Se

Correctness

Other Algorithm

Appendix

- Assume that and hold at the start of an iteration
- Let s be an unexplored node with minimum distance
- We claim that dist[s] is the shortest distance from source to s
 - If there is a shorter path to s via explored nodes only, then dist[s] would have been updated when exploring the predecessor of s on that path
 - If there is a shorter path to s via an unexplored node u, then dist[u] <
 dist[s], which is a contradiction, since s has minimum distance out of all
 unexplored nodes



vertex 3e

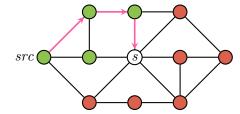
Correctness

Other

Annendix

Induction step (continued):

dist[s] is the shortest distance from source to s



Algoritiiii

LAdilipte

Path Find

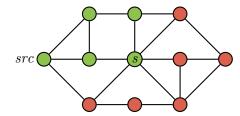
Vertex Se

Analysis

Time comple

Annondi

- ullet dist[s] is the shortest distance from source to s
- After exploring *s*:



Algorithm

. .

.

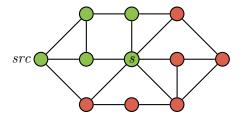
Vertex Se

Correctness

Other Algorithm:

Appendi

- dist[s] is the shortest distance from source to s
- After exploring s:
 - 1 still holds for s, since dist[s] is not updated while exploring s
 - Same for all other explored nodes



Algorithm

...

,

Patii Fillui

Vertex Se

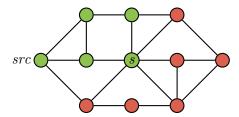
Analysis Correctness

Other

Algorithm

Appendi:

- dist[s] is the shortest distance from source to s
- After exploring s:
 - 1 still holds for s, since dist[s] is not updated while exploring s
 - Same for all other explored nodes
 - 2 still holds for all unexplored nodes *t*, since:



Algorithm

Induction step (continued):

Dath Findin

dist[s] is the shortest distance from source to s

Vertex Set

Correctness

• After exploring s:

• 1 still ho

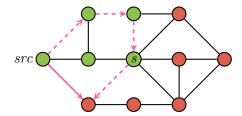
ullet a still holds for s, since dist[s] is not updated while exploring s

Same for all other explored nodes

• 2 still holds for all unexplored nodes *t*, since:

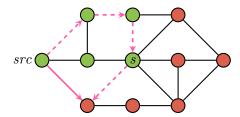
 \bullet If there is a shorter path to t via s then we would have updated $\mathsf{dist}[t]$ while exploring s





Correctness

- dist[s] is the shortest distance from source to s
- After exploring s:
 - 1 still holds for s, since dist[s] is not updated while exploring s
 - Same for all other explored nodes
 - 2 still holds for all unexplored nodes t, since:
 - If there is a shorter path to t via s then we would have updated dist[t] while exploring s
 - Otherwise, we would not have updated dist[t] and it would remain as it is



Pseudoco

Example

Vortov S

V C I C C X C I

Correctness

Time complexity

Algorithm

Appendix

Analysis:

- Each edge is considered once $\Rightarrow O(E)$
 - Undirected edges are considered once in each direction
- Outer loop has V iterations
- Every iteration, algorithm must find vertex v in vSet with minimum distance time complexity depends on vSet implementation
 - Boolean array $\Rightarrow O(V)$ per iteration \Rightarrow overall cost $= O(E + V^2) = O(V^2)$
 - Array/list of vertices $\Rightarrow O(V)$ per iteration

$$\Rightarrow$$
 overall cost = $O(E + V^2) = O(V^2)$

Priority queue $\Rightarrow O(\log V)$ per iteration \Rightarrow overall cost $= O(E + V \log V)$



Other Shortest Path Algorithms

Algorithm

Pseudoco

Example

Path Findi

Vertex Se

Analysi

Other Algorithms

Annendix

- Floyd-Warshall Algorithm
 - · All-pairs shortest path
 - Works for graphs with negative weights
- Bellman-Ford Algorithm
 - Single-source shortest path
 - Works for graphs with negative weights
 - Can detect negative cycles

Algorithm

Pseudoco

Example

...

vertex se

Anatysi

Other Algorithms

Appendix

https://forms.office.com/r/aPF09YHZ3X



COMP2521 23T3

Algorithm

Pseudocode

Example

Path Findir

vertex Se

Analysis

Other

Appendix

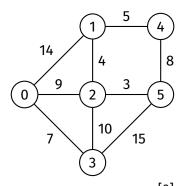
Example

Appendix

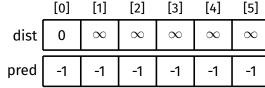
Dijkstra's Algorithm Example

tion

Initialisation



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

.

Example

vertex oc

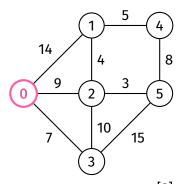
Analysis

Algorithn

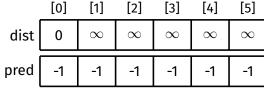
Appendix

Example

Remove 0 from vSet

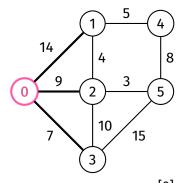


while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

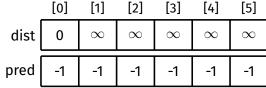


Dijkstra's Algorithm Example

Explore 0



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

Example

i dell'i i i i d

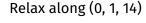
vertex Se

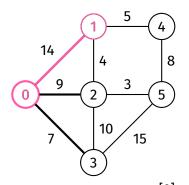
Analysis

Algorithn

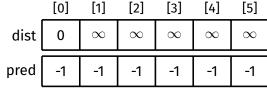
Appendix

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

711301111111

Example

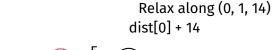
Analysis

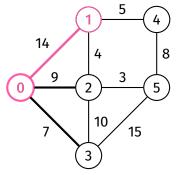
Allatysis

Algorithm

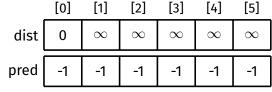
Appendix

Example



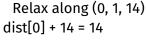


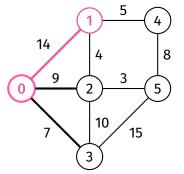
while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



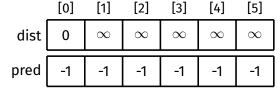
Example

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritim

Example

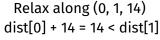
. . .

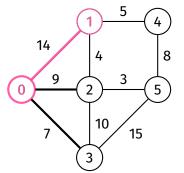
Allatysis

Algorithm

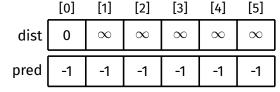
Annendix

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritim

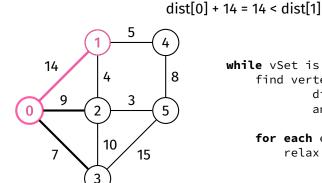
Example

Analysis

Othor

Algorithm

Append Example

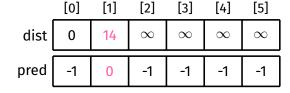


while vSet is not empty: find vertex v in vSet such that

for each edge (v, w, weight) in G: relax along (v, w, weight)

dist[v] is minimal

and remove it from vSet



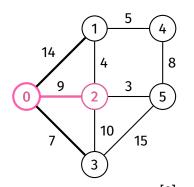
Relax along (0, 1, 14)

Example

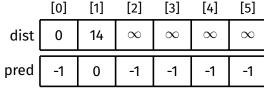
Dijkstra's Algorithm

Example

Relax along (0, 2, 9)



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

....

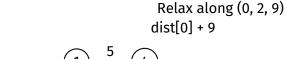
Example

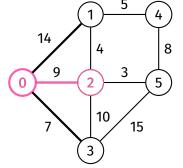
Analysis

Othor

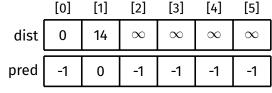
Algoritiiii

Appen Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

_

Example

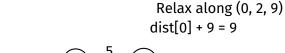
.....

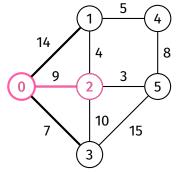
Analysis

Other

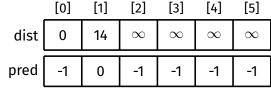
Aigoritiii

Example





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet



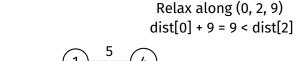
Example

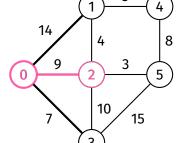
Example

Analysis

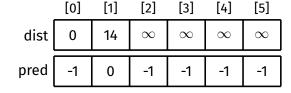
Other

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

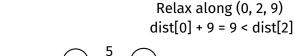
Example

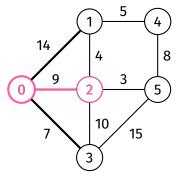
Analysis

Other

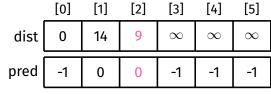
Atgoritiiii

Example



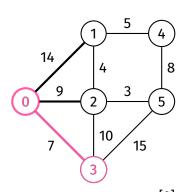


while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

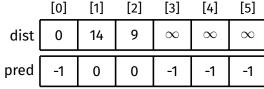


Example

Relax along (0, 3, 7)



for each edge (v, w, weight) in G: relax along (v, w, weight)



Deaudos

- .. -

Vartov Sa

Analysis

Other

Annondiv

Example

Example

5

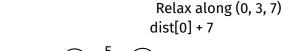
Example

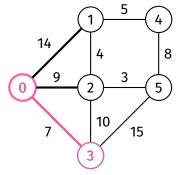
Analysis

Other

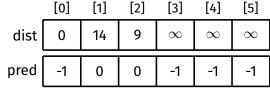
Algorithm

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

71150111111

xample

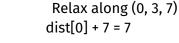
i acii i iiiai

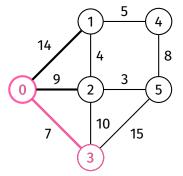
Analysis

Other

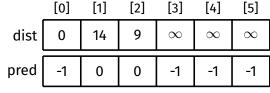
Auguritiiii

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atsoritin

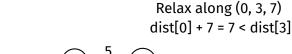
Example

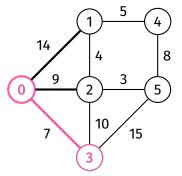
Analysis

Other

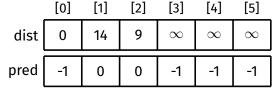
Atgoritimi

Example





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet



Example



Example

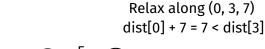
1 4411 1 11141

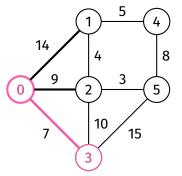
Analysis

Other

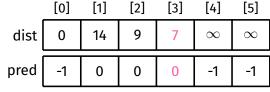
Algorithm

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

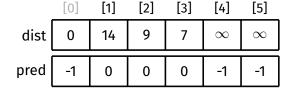


Example

Done with exploring 0

14 8 9 5 Example 0 10 15

while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

oudoc

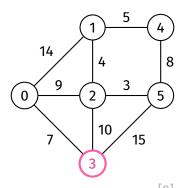
Example

Allalysis

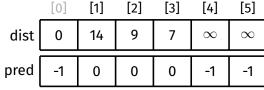
Algorithi

Example

Remove 3 from vSet



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Dijkstra's Algorithm COMP2521 23T3 Example Explore 3 14 while vSet is not empty: 8 find vertex v in vSet such that dist[v] is minimal 9 and remove it from vSet 5 Example 0 for each edge (v, w, weight) in G: 10 relax along (v, w, weight)15 [1] [2] [3] [4] [5] dist 14 9 ∞ ∞ pred 0 0 -1 0

Example

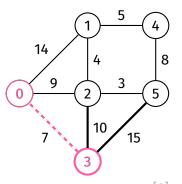
Atgoritim

Example

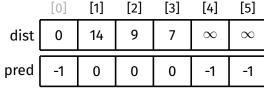
Analysis

Algorithm

Appendi Example No need to consider (3, 0, 7) (0 has already been explored)

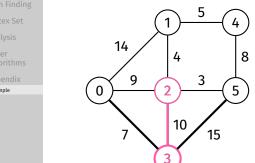


while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



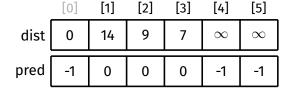
Example

Relax along (3, 2, 10)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

> for each edge (v, w, weight) in G: relax along (v, w, weight)

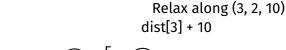


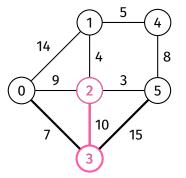
Example

Example

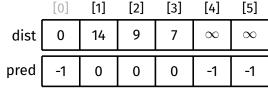


Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

.

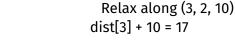
Example

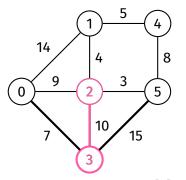
Analysis

0.1

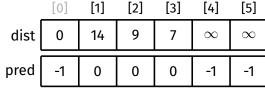
Algoritiiii

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

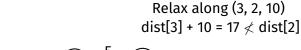
Atgoritini

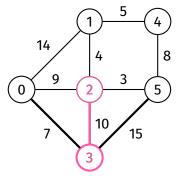
Example

Analysis

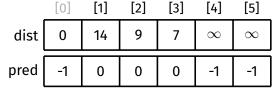
Algoritiii

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

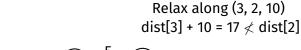
Atgoritini

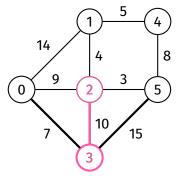
Example

Analysis

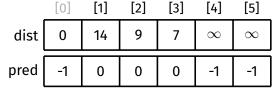
Algoritiii

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

.

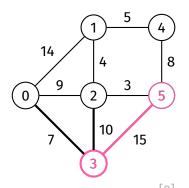
Example

vertex se

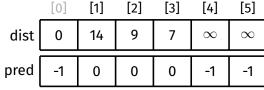
Analysis

Algorithr

Appena Example Relax along (3, 5, 15)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritim

Example

i atii i iiiai

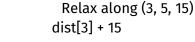
Vertex Set

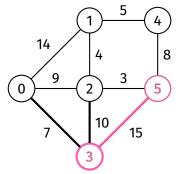
Analysis

Algorithm

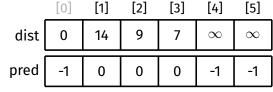
Appendi

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

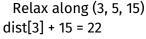
Atgoritim

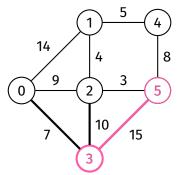
Example

Analysis

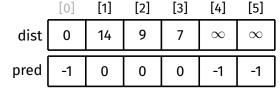
Atgoritin

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

Atgoritin

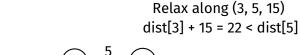
Example

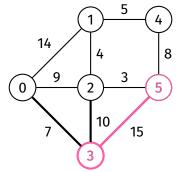
Analysis

Other

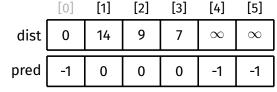
Annendi

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritim

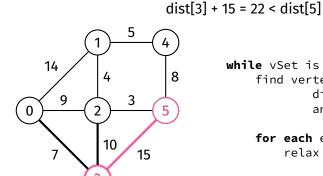
Example

i dell'i i i i di

Analysis

Algorithn

Appendi Example

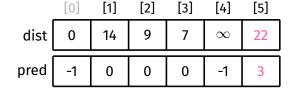


while vSet is not empty: find vertex v in vSet such that

for each edge (v, w, weight) in G:
relax along (v, w, weight)

dist[v] is minimal

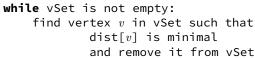
and remove it from vSet

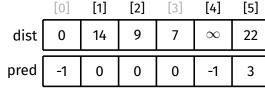


Relax along (3, 5, 15)

Example

Done with exploring 3





Example

.

Example

Vertex Se

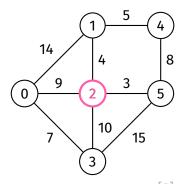
Analysis

Algorithm

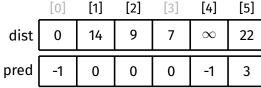
Appendi

Example

Remove 2 from vSet

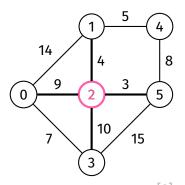


while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet

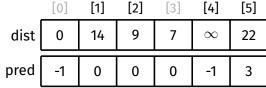


Dijkstra's Algorithm Example

Explore 2



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

.

Example

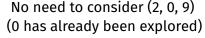
vertex se

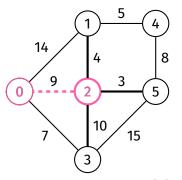
Analysis

Algorithm

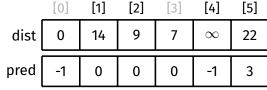
Appendix

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



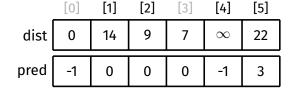
Example

Relax along (2, 1, 4)

Path Finding
Wertex Set
Analysis
Other
Algorithms
Appendix
Example

7

10
15



Example

- .

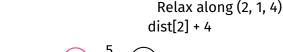
Example

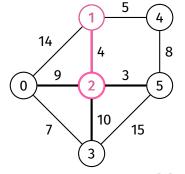
Analysis

, mary bio

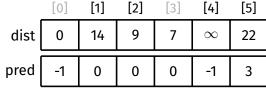
Algorium ..

Example





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet



Example

_ .

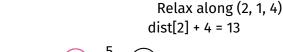
Example

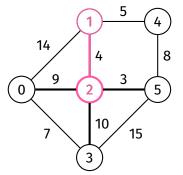
Analysis

0.1

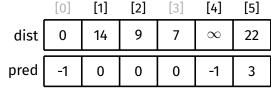
71130771111

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

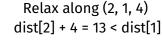
Example

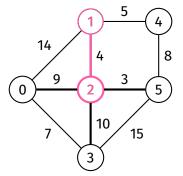
Analysis

, iii aty bio

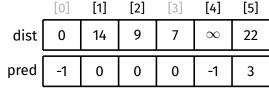
Algoritiiii

Append Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritin

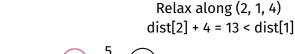
Example

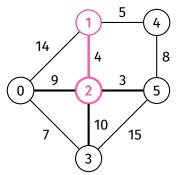
Analysis

. . .

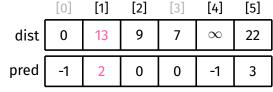
,

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

711501111111

Example

.

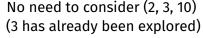
. .

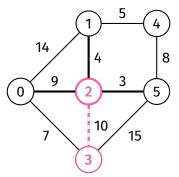
Allatysis

Algorithm

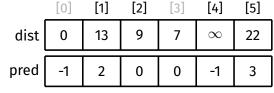
Appendix

Append Example



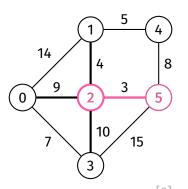


while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



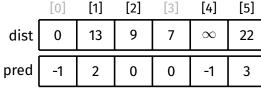
Example

Relax along (2, 5, 3)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)



Deaudoc

Example

Vortov Co

Analysis

Algoritiiii

Appenai Example

Example

Atgoritin

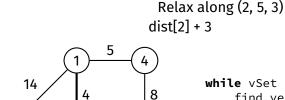
Example

Analusia

Allatysis

Algoritiii

Example



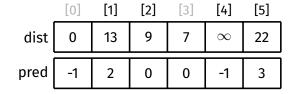
9

10

15

0

while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritin

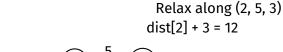
Example

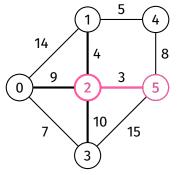
Analysis

, mary ord

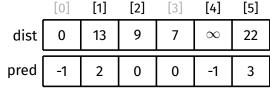
Algorium

Example





while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

Atgoritin

Example

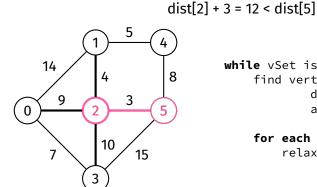
Analysis

...,

Algoritiiii

Appendi

Example

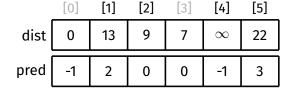


while vSet is not empty: find vertex v in vSet such that

for each edge (v, w, weight) in G: relax along (v, w, weight)

dist[v] is minimal

and remove it from vSet



Relax along (2, 5, 3)

Example

Atgoritim

Example

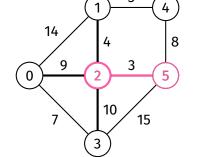
i dell'i i i i di

vertex se

Analysis

Algorithi

Append Example

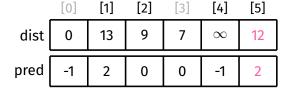


dist[2] + 3 = 12 < dist[5]

while vSet is not empty:

find vertex v in vSet such that dist $\left[v
ight]$ is minimal and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)

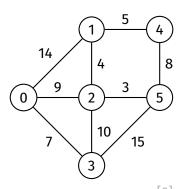


Relax along (2, 5, 3)

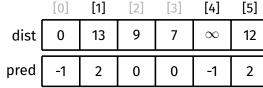
Dijkstra's Algorithm Example

LA

Done with exploring 2



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

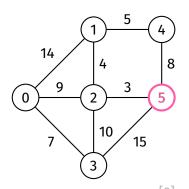


Example

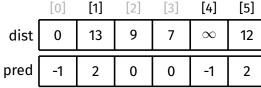
Dijkstra's Algorithm

Example

Remove 5 from vSet



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

Atgoritim

Example

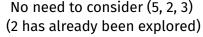
vertex 3e

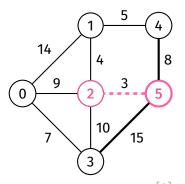
Analysis

Algorithn

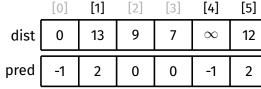
Appendi

Example





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet



Example

Algoritim

Example

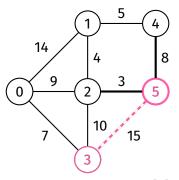
\/---h----C--4

Analysis

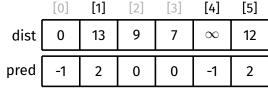
Other

Example

No need to consider (5, 3, 15) (3 has already been explored)



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

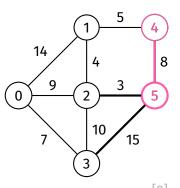


Example

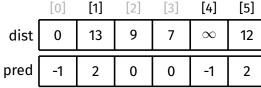
Dijkstra's Algorithm

Example

Relax along (5, 4, 8)



while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

Deaudos

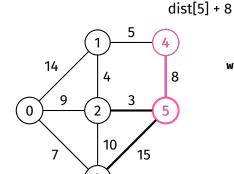
Example

Analysis

041----

Algorium ..

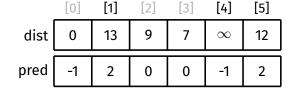
Example



8

Relax along (5, 4, 8)

while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet



Example

ample

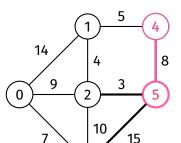
Paul Fillul

Analysis

7 11 TOLLY 5 TO

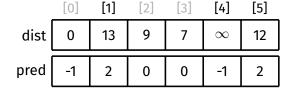
Aigoritiii

Append Example



Relax along (5, 4, 8) dist[5] + 8 = 20

while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

.

Example

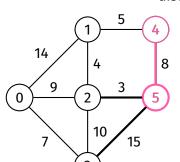
Vaukay Cal

Analysis

Other

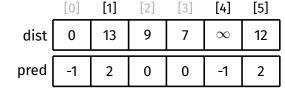
A - ... - ... - ...

Example



Relax along (5, 4, 8) dist[5] + 8 = 20 < dist[4]

while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet



Example

Atgoritin

Example

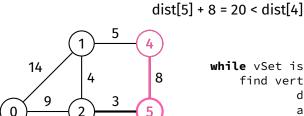
. . .

Anatysis

Algorithm

Appendix

Example

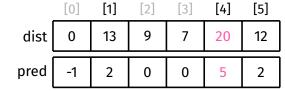


15

10

while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

for each edge (v, w, weight) in G: relax along (v, w, weight)

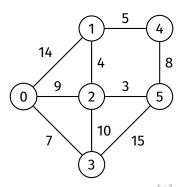


Relax along (5, 4, 8)

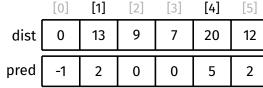
Dijkstra's Algorithm Example

Lxample

Done with exploring 5



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

oudoc

Example

Palli Filli

Vertex Se

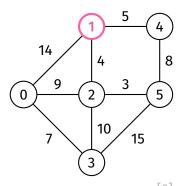
Analysis

Algorithr

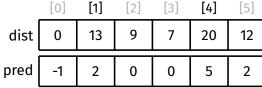
Appendix

Example

Remove 1 from vSet



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritini

Example

1 4411 1 1114

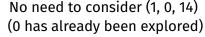
Vertex Set

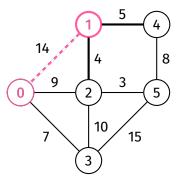
Analysis

Algorithm

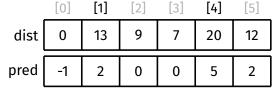
Annondiv

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

71150111111

Example

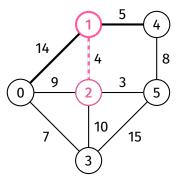
Analysis

0.1

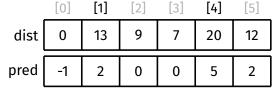
Algorithm

Appendi:

No need to consider (1, 2, 4)
(2 has already been explored)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

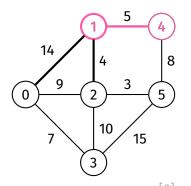


Example

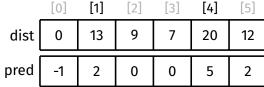
Dijkstra's Algorithm

Example

Relax along (1, 4, 5)



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example



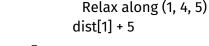
Example

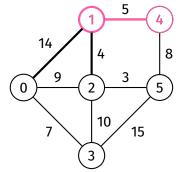
Analysis

Other

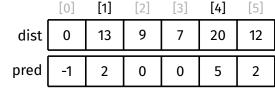
Algorithm

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Atgoritani

Example

ratii riiiui

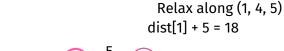
vertex sei

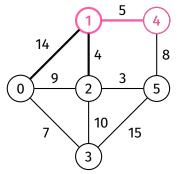
Analysis

Algorithm

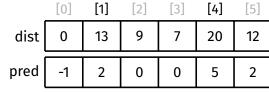
Annandiu

Example





while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Algoritani

Example

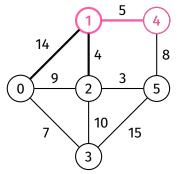
Analysis

Other

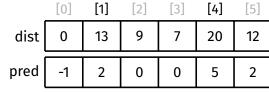
A - ... - ... - ...

Example





while vSet is not empty: find vertex v in vSet such that $\operatorname{dist}[v]$ is minimal and remove it from vSet



Example

Atgoritim

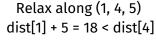
Example

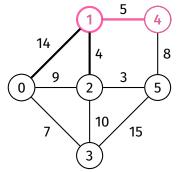
Analysis

Other

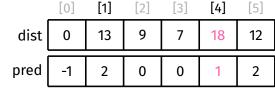
Algoritiiii

Example



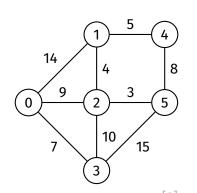


while vSet is not empty:
 find vertex v in vSet such that
 dist[v] is minimal
 and remove it from vSet

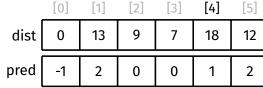


Example

Done with exploring 1

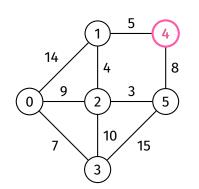


while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

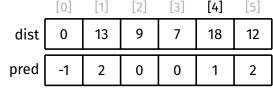


Example

Remove 4 from vSet



for each edge (v, w, weight) in G: relax along (v, w, weight)



Danielan

Example

Vartay Sa

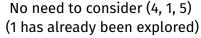
Analysis

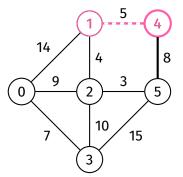
Other

Appendix Example

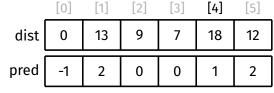
Example

Example





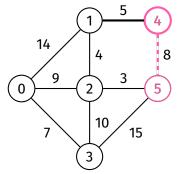
while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



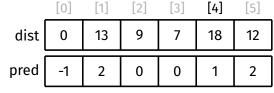
Example

14 Example 0

No need to consider (4, 5, 8) (5 has already been explored)

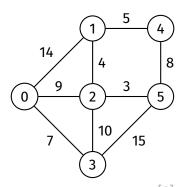


while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet



Example

Done with exploring 4



while vSet is not empty: find vertex v in vSet such that dist[v] is minimal and remove it from vSet

