COMP90038
Assignment Project Exam Help
Algorit
https://eduassistpro.github.io/

Lecture 20: Greedy Algorith d Dijkstra Add WeChat edu_assist_pro (with thanks to Harald Sønde hael Kirley)

Andres Munoz-Acosta

munoz.m@unimelb.edu.au

Peter Hall Building G.83

Recap

- We have talked a lot about dynamic programming:

 - DP is bottom-up problem solving technique.
 Assignment Project Exam Help
 Similar to divide-and-conquer; however, problems are overlapping, making tabulation a requirem https://eduassistpro.github.io/
 - Solutions often involv

- We applied this idea to two graph problems:
 - Computing the transitive closure of a directed graph; and
 - Finding shortest distances in weighted directed graphs.

A practice challenge

Can you solve the problem in the figure?

Assignment Project Exam Help

```
• W = 15
```

• v = [1 2 2 10 4]

https://eduassistpro.github.io/

- Because it is a larger instance, memoing is preferable.
 - How many states do we need to evaluate?
- FYI the answer is \$15 {1,2,3,4}

The table

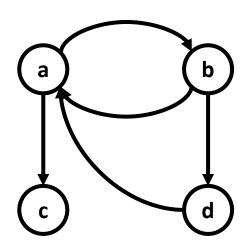
		j		0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
W	V	i				As	sigr	ımei	nt P	roje	ct E	xam	i He	lp					
			0											_					
	1	1	1		1	1	ht	tps:/	//ed	uas	sistr	oro.c	githl	ıb.io	/ 1	1	1	1	1
	1	2	2		2	-1	3	-1	-1	1	. '		3		3	-1	3	-1	3
	2	2	3		-1	-1	A	dd-N	Vec	Chat	edu	_as	sist	_pro	5	-1	-1	-1	5
	4	10	4		-1	-1	4	-1	-1	1	-1	-1	-1	-1	-1	-1	-1	-1	15
1	L2	4	5		-1	-1	-1	-1	-1	1	-1	-1	-1	-1	-1	-1	-1	-1	15

• We know that we include all the elements up to 4 because the last column (15) is the cumulative sum of the values.

- Warshall's algorithm computes the transitive closure of a directed graph.
 - An edge (a,d) is in the saignitive not represent the property and in G from a to d.

https://eduassistpro.github.io/

• **Is there a path** from node i to mode i at edu_assist_pro k] as "stepping stones"? Add we chat edu_assist_pro



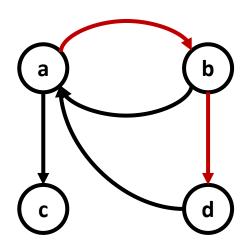
- Such path will exist if and only if we can:
 - step from *i* to *j* using only nodes [1 ... *k*-1], or
 - step from *i* to *k* using only nodes [1 ... *k*-1], and then step from *k* to *j* using only nodes [1 ... *k*-1].

```
\left[\begin{array}{cccc} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{array}\right]
```

- Warshall's algorithm computes the transitive closure of a directed graph.
 - An edge (a,d) is in the saignitive not represent the property and in G from a to d.

https://eduassistpro.github.io/

• Is there a path from node i to mode i u edu_assist_pro k] as "stepping stones"? Add Wechat edu_assist_pro



- Such path will exist if and only if we can:
 - step from *i* to *j* using only nodes [1 ... *k*-1], or
 - step from *i* to *k* using only nodes [1 ... *k*-1], and then step from *k* to *j* using only nodes [1 ... *k*-1].

$$\begin{bmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

• If G's adjacency matrix is A then we can express the recurrence relation as:

Assignment Project Exam Help

https://eduassistpro.github.io/

We examined the simpledtdversionat edu_assisithpro

```
for k \leftarrow 1 to n do

for i \leftarrow 1 to n do

if A[i,k] then

for j \leftarrow 1 to n do

if A[k,j] then

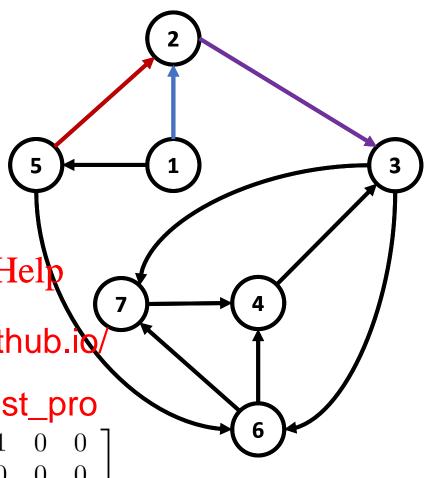
A[i,j] \leftarrow 1
```

Let's visualize the steps.

Assignment Project Exam Help

Using node 2 (k=2), we reach node 3 from no https://eduassistpro.github.io
 Add WeChat edu_assist_pro

 $\begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$

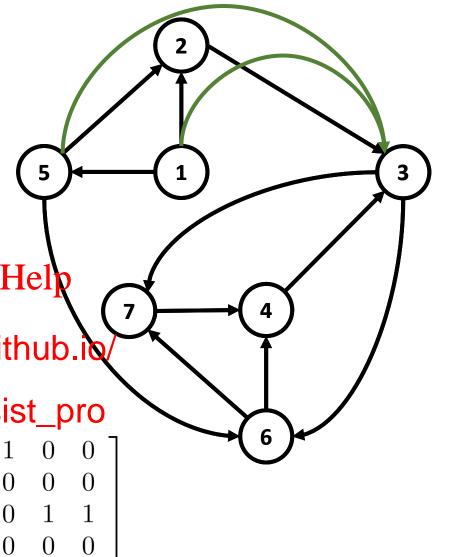


• Let's visualize the steps.

Assignment Project Exam Help

Using node 2 (k=2), we reach node 3 from no https://eduassistpro.github.io
 Add WeChat edu_assist_pro

 $\begin{bmatrix} 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$

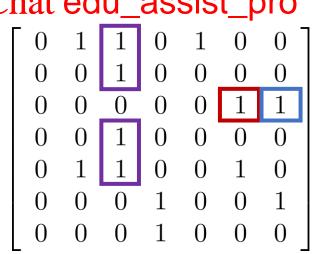


• Let's visualize the steps.

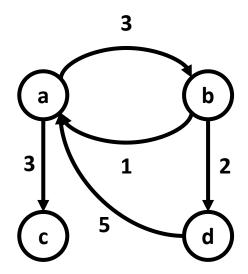
Assignment Project Exam Hel

Using node 2 (k=2), we reach node 3 from no https://eduassistpro.github.io
 Add WeChat edu_assist_pro

 Using node 3 (k=3) we can reach: Nodes [6 7] from nodes [1,2,5]



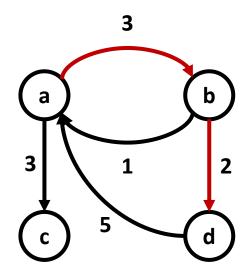
- Floyd's algorithm solves the all-pairs shortest-path problem for weighted graphs with positive weights.
 - It works for directed assign as emblifertied of Epham Help
- What is the shortest pa https://eduassistpro.github.io/ing nodes [1 ... k] as "stepping atproxication of the shortest pa https://eduassistpro.github.io/ing



- Such path will exist if and only if we can:
 - step from *i* to *j* using only nodes [1 ... *k*-1], or
 - step from *i* to *k* using only nodes [1 ... *k*-1], and then step from *k* to *j* using only nodes [1 ... *k*-1].

$$egin{bmatrix} \infty & 3 & 3 & \infty \\ 1 & \infty & \infty & 2 \\ \infty & \infty & \infty & \infty \\ 5 & \infty & \infty & \infty \end{bmatrix}$$

- Floyd's algorithm solves the all-pairs shortest-path problem for weighted graphs with positive weights.
 - It works for directed assign as emblifertied of Epham Help
- What is the shortest pa https://eduassistpro.github.io/ing nodes [1 ... k] as "stepping stowes" Chat edu_assist_pro



- Such path will exist if and only if we can:
 - step from *i* to *j* using only nodes [1 ... *k*-1], or
 - step from *i* to *k* using only nodes [1 ... *k*-1], and then step from *k* to *j* using only nodes [1 ... *k*-1].

 If G's weight matrix is W then we can express the recurrence relation as:

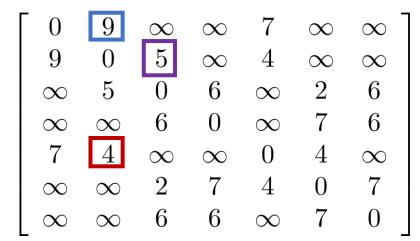
Assignment Project Exam Help

https://eduassistpro.github.io/

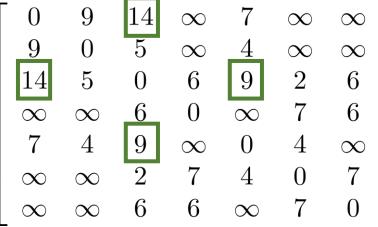
A simpler version updathold WeChat edu_assist_pro

```
\begin{aligned} & \textbf{function} \ \text{FLOYD}(W[\cdot, \cdot], n) \\ & D \leftarrow W \\ & \textbf{for} \ k \leftarrow 1 \ \text{to} \ n \ \textbf{do} \\ & \textbf{for} \ i \leftarrow 1 \ \text{to} \ n \ \textbf{do} \\ & \textbf{for} \ j \leftarrow 1 \ \text{to} \ n \ \textbf{do} \\ & D[i, j] \leftarrow \min \left(D[i, j], D[i, k] + D[k, j]\right) \\ & \textbf{return} \ D \end{aligned}
```

- For *k*=2
 - We can go $1 \rightarrow 2 \rightarrow 3$, the distance $1 \rightarrow 3$ is 9 + 5 = 14 Assignment Project Exam Help
 - We can go $5 \rightarrow 2 \rightarrow 3$, of $5 \rightarrow 3$ is 4 + 5 = 9 https://eduassistpro.githubjio



- For *k*=2
 - We can go $1 \rightarrow 2 \rightarrow 3$, the distance $1 \rightarrow 3$ is 9 + 5 = 14 Assignment Project Exam Help
 - We can go $5 \rightarrow 2 \rightarrow 3$, of $5 \rightarrow 3$ is 4 + 5 = 9 https://eduassistpro.github.id
- The distance matrix getadd WeChat edu_assist_proupdated to:



Greedy Algorithms

- A problem solving strategy is to take the locally best choice among all feasible ones.
 - Once we do this, Assignment Preject Fram Help

https://eduassistpro.github.io/

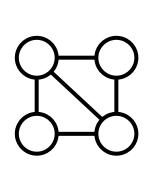
- We want to change 30 cents using t t number of coins.
 If we assume coin denominative Chateedu_assistye@ould use as many 25-
 - If we assume coin denotinations of the continuous depth of the continuous assume coin denotinations of the continuous depth of the contin
 - This **greedy** strategy would not work for denominations {25, 10, 1} (25+1+1+1+1+1 compared to 10+10+10).

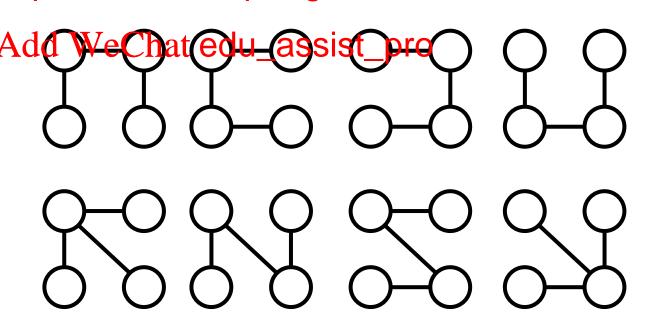
Greedy Algorithms

- In general, it is unusual that **locally best** choices yield **global best** results.
 - Assignment Project Exam Help
 However, there are problems for which a greedy algorithm is correct and fast.
 - In some other proble approximation algorithmed WeChat edu_assist_pro
- Here we shall look at:
 - Prim's algorithm for finding minimum spanning trees
 - Dijkstra's algorithm for single-source shortest paths

What is an Spanning Tree?

- Recall that a **tree** is a connected graph with no cycles.
- A spanning tree of a graph $\langle V, E \rangle$ is a tree $\langle V, E' \rangle$ where E' is a subset of Assignment Project Exam Help
- For example, the grap https://eduassistpro.gdiff@geot spanning trees:



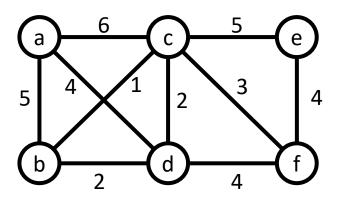


Minimum Spanning Trees of Weighted Graphs

- For a weighted graph, some spanning trees are more desirable than others.
 - For example, suppose we have a set of "stations" to connect in a network, and also some possible connections, each with its own cost.
- This is the problem of fi cost.

https://eduassistpro.github.io/ ith the smallest possible Add WeChat edu assist pro

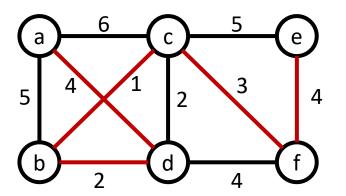
• Such tree is a minimum spanning tree for



Minimum Spanning Trees of Weighted Graphs

- For a weighted graph, some spanning trees are more desirable than others.
 - For example, suppose we have a set of "stations" to connect in a network, and also some possible connections, each with its own cost.
- This is the problem of fi cost.

https://eduassistpro.github.io/ ith the smallest possible • Such tree is a minimum spanning tree for



- Prim's algorithm is an example of a greedy algorithm.
 - It constructs a sequence of subtrees *T*, by **adding to the latest tree the closest node not classignymenit**. Project Exam Help

A simple version: https://eduassistpro.github.io/

• But how to find the **minimum-weight edge** (*v*,*u*)?

Assignment Project Exam Help

- A standard way to do t odes that are not yet included in the spanni https://eduassistpro.githebeo, organised in a min-heap by edge cost_{Add} WeChat edu_assist_pro
- The information about which nodes are connected in T can be captured by an array prev of nodes, indexed by V. Namely, when (v,u) is included, this is captured by setting prev[u] = v.

• The complete algorithm is:

Assignment Project Exam Help

https://eduassistpro.github.io/

a 6 c 5 e 5 b 2 d f

• On the first loop, we only create the table

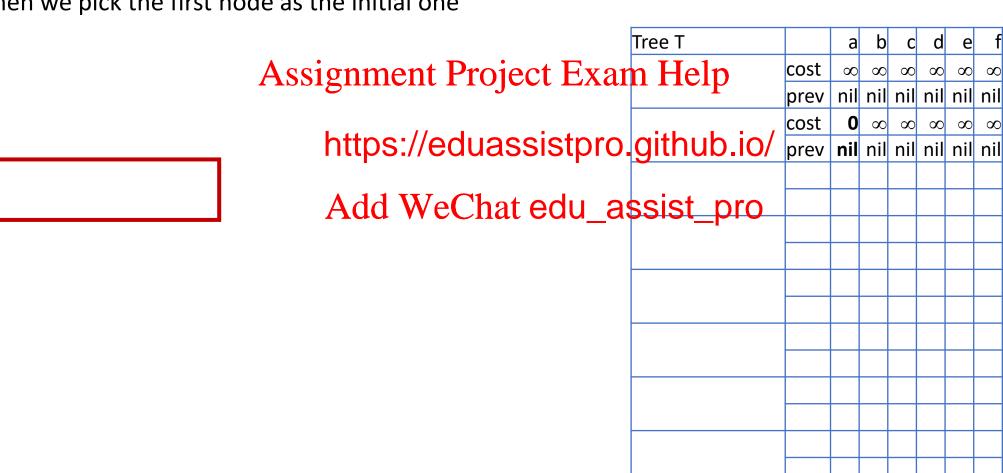


Assignment Project Exam https://eduassistpro.g

Ггее Т		a	b	С	d	e	f
n Help	cost	∞	∞	∞	∞	∞	∞
ТТОТР	prev	nil	nil	nil	nil	nil	nil
141 1 1 1							
github.io/							
ssist_pro_							
_p.o							

5 4

• Then we pick the first node as the initial one



a 6 c 5 e 5 b 2 d f

 We take the first node out of the queue and update the costs

Assignment Project Exam Help

https://eduassistpro.github.io/

Tree T		a	b	С	d	e	f
n Help	cost	∞	∞	∞	∞	∞	∞
11016	prev	nil	nil	nil	nil	nil	nil
141 1 1	cost	0	∞	∞	∞	∞	∞
.github.io/	prev	nil	nil	nil	nil	nil	nil
	cost		5	6	4	∞	∞
ssist_pro	prev		a	a	a	nil	nil
_p.o.							

a 6 c 5 e 5 b 2 d f

• We eject the node with the lowest cost and update the queue.

Assignment Project Exam Help

https://eduassistpro.github.io/

	a	b	С	d	е	f
cost	∞	∞	∞	∞	∞	∞
prev	nil	nil	nil	nil	nil	nil
cost	0	∞	∞	∞	∞	∞
prev	nil	nil	nil	nil	nil	nil
cost		5	6	4	∞	∞
prev		a	a	a	nil	nil
cost		2	2		∞	4
prev		d	d		nil	d
	prev cost prev cost prev cost	cost ∞ prev nil cost 0 prev nil cost prev cost cost	cost ∞ ∞ prev nil nil cost 0 ∞ prev nil nil cost 0 prev a cost 0	cost ∞ ∞ ∞ prev nil nil nil cost 0 ∞ ∞ prev nil nil nil cost 0 ∞ prev a a cost 0	cost ∞ ∞ ∞ ∞ prev nil nil nil nil cost 0 ∞ ∞ ∞ prev nil nil nil nil nil cost 0 0 0 0 0 0 0 0 0 0	$\begin{array}{c ccccccccccccccccccccccccccccccccccc$

a 6 c 5 e 5 b 2 d f

We eject the next node based on alphabetical order.
 Why is f not updated?

Assignment Project Exam Help

https://eduassistpro.github.io

Tree T		а	b	С	d	е	f
n Help	cost	∞	∞	∞	∞	∞	∞
ii iioip	prev	nil	nil	nil	nil	nil	nil
	cost	0	∞	∞	∞	∞	∞
github.io/	prev	nil	nil	nil	nil	nil	nil
	cost		5	6	4	∞	∞
ssist_pro	prev		a	a	a	nil	nil
a,d	cost		2	2		∞	4
a,u	prev		d	d		nil	d
a,d,b	cost			1		∞	4
a,u,u	prev			b		nil	d

a 6 c 5 e 5 b 2 d f

a nil nil

nil

nil

• We now update f

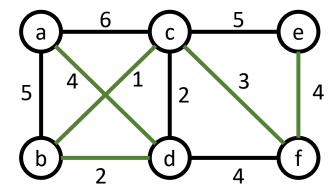
Tree T		a	b	
Assignment Project Exam Help	cost	∞	∞	_
	prev	nil	nil	L
	cost	0	∞	
https://eduassistpro.github	.IO/ prev	nil	nil	
	cost		5	
Add WeChat edu_assist_r)ro prev		a	
_	cost		2	L
a,d	prev		d	L
a d b	cost			
a,d,b	prev			
adha	cost	ost ∞ ∞ rev nil nil ost 0 ∞ rev nil nil ost 5 rev a ost 2 rev d ost rev c		
a,d,b,c	prev			

a b c 3 e 5 4 1 2 3 4 b 2 d f

• We reach the last choice

	0 2			4		U	J	
	Tree T		a	b	С	d	е	1
Assignment Project Exar	n Help	cost	∞	∞	∞	∞	∞	\propto
1 1881811110110 1 1 eje e	т	prev	nil	nil	nil	nil	nil	ni
	/	cost	0	∞	∞	∞	∞	\propto
https://eduassistpro.	github.io/	prev	nil	nil	nil	nil	nil	ni
		cost		5	6	4	∞	\propto
Add WeChat edu_as	ssist pro	prev		a	a	a	nil	ni
	- _•	cost		2	2		∞	4
	a,d	prev		d	d		nil	C
	a d b	cost			1		∞	4
	a,d,b	prev			b		nil	С
	a d b a	cost					5	3
	a,d,b,c	prev					С	(
	adbaf	cost					4	
	a,d,b,c,f	prev					f	

• The res



esulting tree is {a,d,b,c,f,e}								
	Tree T		a	b	С	d	е	f
Assignment Project Example 19 Project Example 20 Pr	m Help	cost	∞	∞	∞	∞	∞	∞
Tissisimilatio Troject Zhan		prev	nil	nil	nil	nil	nil	nil
	adda da da /	cost	0	∞	∞	∞	∞	∞
https://eduassistpro	.github.io/	prev	nil	nil	nil	nil	nil	nil
		cost		5	6	4	∞	∞
Add WeChat edu_a	ssist pro	prev		a	а	а	nil	nil
	a,d	cost		2	2		∞	4
	a,u	prev		d	d		nil	d
	a,d,b	cost			1		∞	4
	a,u,b	prev			b		nil	d
	a,d,b,c	cost					5	3
	a,u,b,c	prev					С	С
	a,d,b,c,f	cost					4	
	a,u,b,c,ı	prev					f	
	a,d,b,c,f,e	cost						
	م,۵,۵,۰,۰,۰	prev						

Analysis of Prim's Algorithm

- First, a crude analysis: For each node, we look through the edges to find those incident to the node, and pick the one with smallest cost. Thus we get $O(|V| \times |E|)$. However means are Pusing a letter that patroctures.
- Using adjacency lists for https://eduassistpro.github.io/ ap for the priority queue, we perform |V| 1 heap deletions (each at cost $O(10^{\circ})$) and |E| updates of priorities (each at cost $O(10^{\circ})$)
- Altogether $(|V|-1+|E|) O(\log |V|)$.
- Since, in a connected graph, $|V|-1 \le |E|$, this is $O(|E| \log |V|)$.

- Another classical greedy weighted-graph algorithm is Dijkstra's algorithm, whose overall structure is the same as Prim's.
 Assignment Project Exam Help
- Recall that Floyd's alg https://eduassistpro.gritegbpaths, for every pair of nodes, in a (directe ted graph. It assumed an adjacency matrix representation to but edu_assistlexity O(|V|3).
- Dijkstra's algorithm is also a shortest-path algorithm for (directed or undirected) weighted graphs. It finds all shortest paths from a fixed start node. Its complexity is the same as that of Prim's algorithm.

• The complete algorithm is:

Assignment Project Exam Help

https://eduassistpro.github.io/

• On the first loop, we only create the table



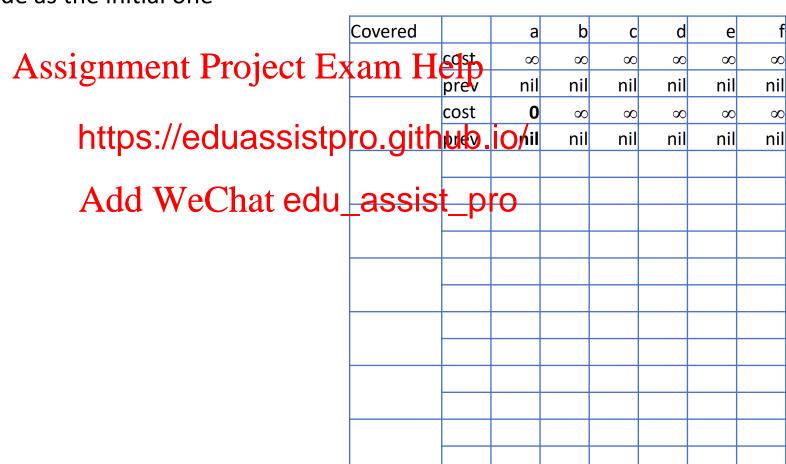
Assignment Project Exa

https://eduassistpro

Add WeChat edu

Covered		a	b	С	d	е	f
am H	cost	∞	∞	∞	8	∞	∞
am H	prev	nil	nil	nil	nil	nil	nil
ro.gith	ub.	io/					
assis	t n	rO					
_ assis	<u>-</u> Р						

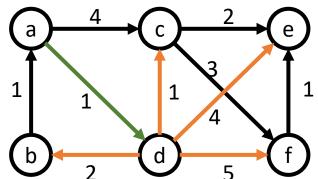
• Then we pick the first node as the initial one



Then

n we pick the first node as the initial one	C		2	U	5		り	
	Covered		а	b	С	d	е	f
Assignment Project F	vam H	sqst	∞	∞	∞	∞	∞	∞
Assignment Project E		prev	nil	nil	nil	nil	nil	nil
		cost	0	∞	∞	∞	∞	∞
https://eduassistp	ro.gith		IO/\il	nil	nil	nil	nil	nil
		cost		∞	4	1	∞	∞
Add WeChat edu	assis	prep	ro	nil	а	а	nil	nil
	T	<u> </u>						

• Then eject the node with the shortest distance from the queue. Then, we update all the paths by adding 1.



				2		5			
the shortest distance from	n the								
all the paths by adding 1.		Covered		а	b	С	d	е	f
Assignment Proj	ect Ex	am H	cast	∞	∞	∞	∞	∞	∞
	OCC LIA	carri II	prev	nil	nil	nil	nil	nil	nil
			cost	0	∞	∞	∞	∞	∞
https://edua	ssistp	ro.gith		iO/\il	nil	nil	nil	nil	nil
•	•		cost		∞	4	1	∞	∞
Add WeCha	t edu	assis	prev	ro	nil	a	a	nil	nil
		a,d	cost		3	2		5	6
		a,u	prev		d	d		d	d

 Our next node will be the one with the shortest path in overall (b)

Assignment Project Ex

https://eduassistp

Add WeChat edu_

Covered		a	b	С	d	е	f
am H	cqst	∞	∞	∞	∞	∞	∞
am H	prev	nil	nil	nil	nil	nil	nil
	cost	0	∞	∞	∞	∞	∞
ro.gith	1WO.	iO/hil	nil	nil	nil	nil	nil
			∞	4	1	∞	∞
_assis	prev	ro	nil	a	а	nil	nil
a,d	cost		3	2		5	6
a,u	prev		d	d		d	d
2 d c	cost		3			4	5
a,d,c	prev		d			C	С

• Now, we continue evaluating from (c)

				2	U	5			
ting from (c)									
		Covered		a	b	С	d	e	f
Assignment Project Ex		kam H	cost	∞	∞	∞	∞	∞	∞
			prev	nil	nil	nil	nil	nil	nil
https://eduassistp			cost	0	∞	∞	∞	∞	∞
				IO/hil	nil	nil	nil	nil	nil
			cost		∞	4	1	∞	∞
Add WeCha	t edu	assis	pre	ro	nil	a	а	nil _	nil
	_	a,d	cost		3	2		5	6
		,	prev		d	d		d	d
		a,d,c	cost		3			4	5
			prev		d			С	C
		a,d,c,b	cost					4	5
			prev					С	С
									$\overline{}$
				1 1					

• We arrive at our last decision

				2	(a)	5		リ	
ision.				_					
		Covered		a	b	С	d	е	f
Assignment Proj	ect Ex	am H	cast	∞	∞	∞	∞	∞	∞
7 issignment 1 roj		prev	nil	nil	nil	nil	nil	nil	
			cost	0	∞	∞	∞	∞	∞
Assignment Project Entry https://eduassistp. Add WeChat edu	ro.gith		iO/ /il	nil	nil	nil	nil	nil	
•		cost		∞	4	1	∞	∞	
Add WeCha	it edu	assis	prev	ro	nil	a	а	nil	nil
		a,d	cost		3	2		5	6
		a,u	prev		d	d		d	d
		a,d,c	cost		3			4	5
		a,u,c	prev		d			С	С
		a,d,c,b	cost					4	5
		u,u,c,b	prev					С	С
		a,d,c,b,e	cost						5
		u,u,c,o,c	prev						С
	•								

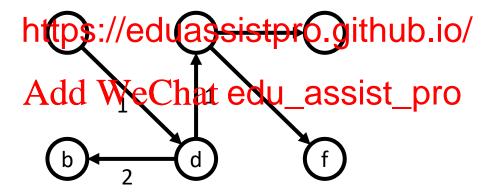
• Our complete tree is {a,d,c,b,e,f}

				2		5			
,c,b,e,f}									
		Covered		a	b	С	d	е	f
Assignment Proje	ect Ex	am H	cast	∞	∞	∞	∞	∞	∞
Assignment Project Ex			prev	nil	nil	nil	nil	nil	nil
		cost	0	∞	∞	∞	∞	∞	
https://eduas	ssistp	ro.gith	WOOD.	iO/\il	nil	nil	nil	nil	nil
•	_	cost		∞	4	1	∞	∞	
Add WeCha	t edu	assis	prev	ro	nil	a	a	nil	nil
		a,d	cost		3	2		5	6
		u,u	prev		d	d		d	d
		a,d,c	cost		3			4	5
			prev		d			С	С
		a,d,c,b	cost					4	5
	pre	prev					С	С	
		a,d,c,b,e	cost						5
			prev						С
		a,d,c,b,e,f	cost						
		۵,۵,۵,۵,۵,۱	prev						

Tracing paths

• The array prev is not really needed, unless we want to retrace the shortest paths from node a

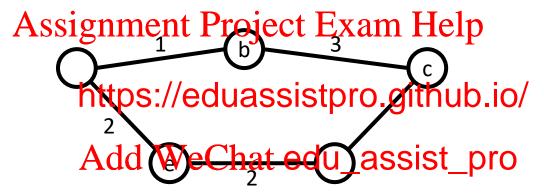
Assignment Project Exam Help



• This tree is referred as the shortest-path tree

Spanning trees and Shortest-Path trees

 The shortest-path tree that results from Dijkstra's algorithm is very similar to the minima spaning tree.



- Exercise:
 - Which edge is missing in the minimal spanning tree?
 - Which edge is missing from the shortest-path tree?
 - Assume that you always started from node a.

Next lecture

Assignment Project Exam Help

• We will have a look to https://eduassistpro.github.io/data compression Add WeChat edu_assist_pro