

Example Test BFS-DFS-MST

Due	No due date	Points	60	Questions	6	Time Limit	None
Allowed Attempts	Unlimited						

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	less than 1 minute	0 out of 60
LATEST	Attempt 2	less than 1 minute	0 out of 60
	Attempt 1	29 minutes	0 out of 60

Submitted Oct 18 at 9:48am

Question 1

0 / 10 pts

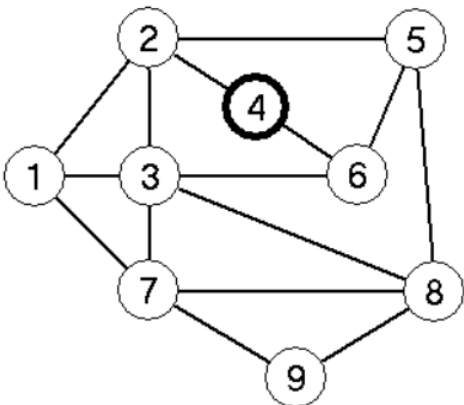
BFS (breadth-first search)

Attention! If an answer consists of a sequence of numbers, the numbers must be separated with commas, with no blank. No sign is to be put at the ends of the sequence.

For example: 1,3,4,6

Exercise 1:

Given the graph below.



Run BFS (breadth-first search) on this graph. In all cases, the neighbours of a node must be processed in the increasing order of their indices. Source vertex: **4**

Answer the following questions.

1.a: Consider the first five iterations of the main loop. In each iteration, present the content of the queue at the end of the body of the main loop. The vertices must be separated with commas. Do not use blanks nor other extra characters.

At the end of iteration 1:

At the end of iteration 2:

At the end of iteration 3:

At the end of iteration 4:

At the end of iteration 5:

Consider now the result of BFS.

1.b: What is the maximal d value:

1.c: Present the d and π values of the following vertices. (Write first the d value, and then the π value separated by a comma. Use no other character, no blank.)

Vertex 1:

Vertex 7:

Vertex 8:

Vertex 9:

1.d: Let us suppose that we delete edge (3,8) from the graph, and we run the algorithm again. Does the d value of any vertex change? (Answer yes or no.)

Exercise 2:

Given a digraph, we have run BFS. The parent (i.e. π) values of the vertices:

vertices:

π values:

1	2	3	4	5	6	7	8	9	10
3	8	6	3	2	0	6	7	2	8

Draw the corresponding breadth-first tree and answer the following questions.

2.a: Present the path leading from the source vertex to vertex 5.
(Separate the vertices with "," signs. Use no blank.)

The path:

2.b: Enter the vertices which are at distance 2 from the source. (The vertices must be entered in increasing order according to their indices. They must be separated by commas. Use no blank.)

The vertices at distance 2:

2.c The distance of the vertex farthest from the source:

Answer 1:

You Answered

(You left this blank)

Correct Answer

2,6

Answer 2:

You Answered

(You left this blank)

Correct Answer

6,1,3,5

Answer 3:

You Answered

(You left this blank)

Correct Answer

1,3,5

Answer 4:

You Answered

(You left this blank)

Correct Answer

3,5,7

Answer 5:

You Answered

(You left this blank)

Correct Answer

5,7,8

Answer 6:

You Answered

(You left this blank)

Correct Answer

4

Answer 7:

You Answered

(You left this blank)

Correct Answer

2,2

Answer 8:

You Answered

(You left this blank)

Correct Answer

3,1

Answer 9:

You Answered

(You left this blank)

Correct Answer

3,3

Answer 10:

You Answered

(You left this blank)

Correct Answer

4,7

Answer 11:

You Answered

(You left this blank)

Correct Answer

no

Correct Answer

No

Answer 12:

You Answered

(You left this blank)

Correct Answer

6,7,8,2,5

Answer 13:

You Answered

(You left this blank)

Correct Answer

1,4,8

Answer 14:

You Answered

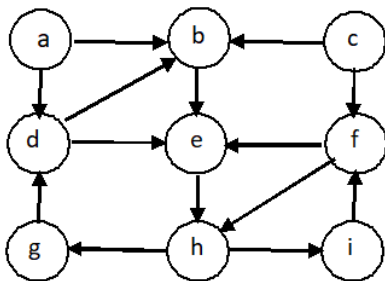
(You left this blank)

Correct Answer

4

Question 2**0 / 10 pts****DFS (depth-first search)**

Given the following graph:



Run DFS on this graph. Use the alphabetical convention. Classify the edges of the graph.

Answer the following questions.

a) **Enter the d and f values of the following vertices.**

(Write first the d value, and then the f value separated by a comma. Use no other character, no blank.)

Vertex c:

Vertex h:

b) Provide the classification of the following edges entering the appropriate letter.

(T -- Tree edge, F -- Forward edge, C -- Cross edge, B -- Back edge)

Edge (a,d) :

Edge (b,e):

Edge (c,b):

c) The number of back edges =

d) The number of trees of the depth-first forest =

e) Is the original graph a DAG? (Y - yes, N - no)

f) Consider the subgraph containing vertices b, c, e, f and the edges between them. Enter the topological order of the vertices computed by DFS topological sort on this subgraph. Use the alphabetical convention in DFS.

, , ,

Answer 1:

You Answered

(You left this blank)

Correct Answer

17,18

Answer 2:

You Answered

(You left this blank)

Correct Answer

4,13

Answer 3:

You Answered

(You left this blank)

Correct Answer

F

Correct Answer

f

Answer 4:

You Answered

(You left this blank)

Correct Answer

T

Correct Answer

t

Answer 5:

You Answered

(You left this blank)

Correct Answer

C

orrect Answer

c

Answer 6:

ou Answered

(You left this blank)

orrect Answer

4

Answer 7:

ou Answered

(You left this blank)

orrect Answer

2

Answer 8:

ou Answered

(You left this blank)

orrect Answer

N

orrect Answer

n

orrect Answer

no

orrect Answer

No

Answer 9:

ou Answered

(You left this blank)

orrect Answer

c

Answer 10:

ou Answered

(You left this blank)

orrect Answer

f

Answer 11:

ou Answered

(You left this blank)

orrect Answer

b

Answer 12:

ou Answered

(You left this blank)

orrect Answer

e

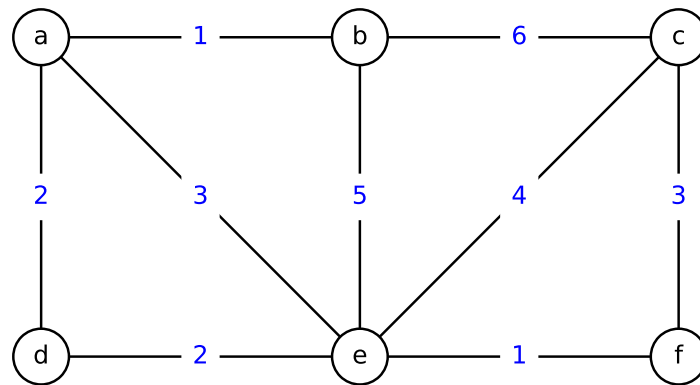
Question 3

0 / 10 pts

Algorithm of Prim:

Attention! If an answer consists of a sequence of numbers or letters, the elements of the sequence must be separated with commas, with no blank. No sign is to be put at the ends of the sequence.

Illustrate the algorithm of Prim on the graph below as you have seen it in the classroom where the *root* is vertex *c*. Then answer the following questions.



1. Give the final c values of the vertices separated by commas, and then the p values of the vertices similarly (pointer null must be written as 0).
2. Give the sequence numbers of the iterations of the main loop of the algorithm where the c value of vertex b is changed.
3. Give the different values of $p(b)$ according to the chronological order where each $p(b)$ value must be given only once.
4. Give the preorder traversal of the MST calculated. The vertices must be separated with commas. (The root is vertex c . The children of a parent are processed according to the increasing weights of the edges leading to them. In case of equal weights use the alphabetical order.)

Answer 1:

You Answered

(You left this blank)

Correct Answer

2,1,0,2,1,3

Answer 2:**You Answered**

(You left this blank)

Correct Answer

d,a,0,e,f,c

Correct Answer

D,A,0,E,F,C

Answer 3:**You Answered**

(You left this blank)

Correct Answer

1,3,5

Answer 4:**You Answered**

(You left this blank)

Correct Answer

0,c,e,a

Correct Answer

0,C,E,A

Answer 5:**You Answered**

(You left this blank)

Correct Answer

c,f,e,d,a,b

Correct Answer

C,F,E,D,A,B

Question 4**0 / 10 pts****Algorithm of Kruskal:**

Attention! If an answer consists of a sequence of numbers or letters, the elements of the sequence must be separated with commas, with no blank. No sign is to be put at the ends of the sequence.

For example: **a,b,d,f**

Illustrate the algorithm of Kruskal on the graph below as you have seen it in the classroom. Then answer the following questions. (Edges with equal weights are sorted according to their endpoints. For example:

$$(b, e) = (e, b) < (b, f) = (f, b) < (c, e) = (e, c))$$

a — $b, 4; d, 2$
 b — $c, 1; d, 4; e, 1$
 c — $e, 1; f, 3$
 d — $e, 5$
 e — $f, 2$
 f .

- Let us say that an edge of the graph is outsider if it is not part of the MST calculated. How many outsider edges had been processed before the MST was received?
- Give the preorder traversal of the MST calculated. The vertices must be separated with commas. (The root is vertex f . The children of a parent are processed according to the increasing weights of the edges leading to them. In case of equal weights use the alphabetical order.)
- Give the sizes of the sets in decreasing order after processing edge (a, d) .
- Give the elements of the largest set of vertices in alphabetical order when edge (a, d) has just been processed. The vertices must be separated with commas. Use no extra character, no blank, no sign at the ends of the sequence.
- How many edges are processed (in the main loop) until vertices d and e get into the same set?

Answer 1:

You Answered

(You left this blank)

Correct Answer

2

Answer 2:

You Answered

(You left this blank)

Correct Answer

f,e,b,c,a,d

Correct Answer

F,E,B,C,A,D

Answer 3:

You Answered

(You left this blank)

Correct Answer

3,2,1

Answer 4:**You Answered**

(You left this blank)

Correct Answer

b,c,e

Correct Answer

B,C,E

Answer 5:**You Answered**

(You left this blank)

Correct Answer

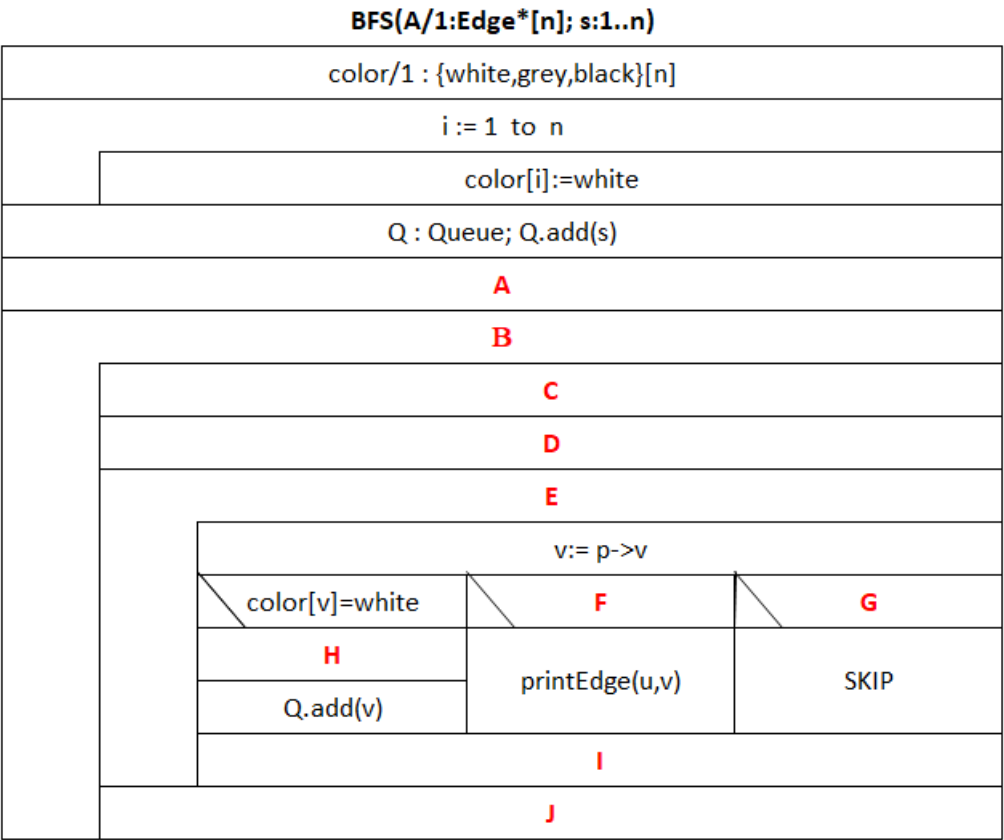
7

Question 5**0 / 10 pts**

A network of streets consists of only two-way streets. Given a graph describing this network. The vertices represent the crossroads, and each edge represents a road section between two crossroads. In turn, the graph is given in adjacency list representation using array $A[1]$ of n elements. (The elements of the array are pointers identifying S1Ls, i.e. simple one-way lists.) We know that the network is connected. The city government is preparing for road renovation. Certain road sections will be closed for renovation.

Write an algorithm based on BFS (breadth-first search) which prints all the lockable road sections so that the rest of the network still remains connected. It is enough to print one possible solution. Operational complexity: $O(n + m)$ where m is the number of edges of the graph.

Select the appropriate code segments from the drop-down lists.



- A: [Select]
- B: [Select]
- C: [Select]
- D: [Select]
- E: [Select]
- F: [Select]
- G: [Select]
- H: [Select]
- I: [Select]
- J: [Select]

Answer 1:

You Answered

(You left this blank)

Correct Answer

color[s] := grey

Answer 2:

You Answered

(You left this blank)

Correct Answer $\neg Q.isEmpty()$ **Answer 3:****You Answered**

(You left this blank)

Correct Answer $u:=Q.rem()$ **Answer 4:****You Answered**

(You left this blank)

Correct Answer $p:=A[u]$ **Answer 5:****You Answered**

(You left this blank)

Correct Answer $p \neq 0$ **Answer 6:****You Answered**

(You left this blank)

Correct Answer $color[v]=grey$ **Answer 7:****You Answered**

(You left this blank)

Correct Answer $color[v]=black$ **Answer 8:****You Answered**

(You left this blank)

Correct Answer $color[v]:=grey$ **Answer 9:****You Answered**

(You left this blank)

Correct Answer $p:=p->next$ **Answer 10:****You Answered**

(You left this blank)

Correct Answer $color[u]:=black$

Question 6

0 / 10 pts

Give the algorithm of topological sort for adjacency list representation of the graph.

In the structogram below we numbered the vertices of the digraph from 1 to n where n is given. The elements of array A identify S1Ls, i.e. simple one-way lists. The colours: 0 is used instead of *white*, 1 is used instead of *grey*, and 2 is used instead of *black*. The topological order is produced in array T provided that we do not find directed loop. In this case the procedure returns with $k=0$. Otherwise k is the index of the vertex where we find the loop. In this later case the value of k and array π together identify the directed loop found (backwards).

Select the appropriate code segments from the drop-down lists.

topologicalSort(A/1 : Edge*[n]; T/1,π/1 : N[n]; &k : N)

color/1 : N[n]		
i := 1 to n		
color[i] := 0		
m := n		
k := 0		
u := 1 to n		
A		
π[u] := 0	SKIP	
B		

DFvisit(A/1 : Edge*[n]; u,&m : N; T/1,π/1 : N[n]; &k : N)

C		
D		
p ≠ ∅ ∧ k = 0		
v := p → v; p:=p→next		
E		
π[v] := u	F	
DFvisit(A, v, m, T, π, k)	π[v] := u	H
	G	
I		
J		
m := m-1		

Válassza ki a fenti struktogramok hiányzó utasításait a lenyíló listákból!

A: [Select]

B: [Select]

C: [Select]

D: [Select]

E: [Select]

F: [Select]

G: [Select]

H: [Select]

I: [Select]

J: [Select]

Answer 1:

You Answered

(You left this blank)

Correct Answer

color[u] = 0

Answer 2:

You Answered

(You left this blank)

Correct Answer

DFvisit(A,u,m,T, π ,k)

Answer 3:

You Answered

(You left this blank)

Correct Answer

color[u]:=1

Answer 4:

You Answered

(You left this blank)

Correct Answer

p:= A[u]

Answer 5:

You Answered

(You left this blank)

Correct Answer

color[v] = 0

Answer 6:**You Answered**

(You left this blank)

Correct Answer`color[v] = 1`**Answer 7:****You Answered**

(You left this blank)

Correct Answer`k := v`**Answer 8:****You Answered**

(You left this blank)

Correct Answer

SKIP

Answer 9:**You Answered**

(You left this blank)

Correct Answer`color[u] := 2`**Answer 10:****You Answered**

(You left this blank)

Correct Answer`T[m] := u`