13 - Nguyễn Việt Hoàng - B22DCVT214

Date No

Bài 4 đề 1

a) Kruskal(){

```
    Step 1: Initialize
    T = Ø,   // At the beginning the set of edges is empty
    d(H) = 0;  // Length equals to 0
    Step 2: Sort
    <Sort edges of the graph in the ascending order of length>;
    Step 3: Loop
    while (|T| < n-1 && E ≠ Ø){
        e = <The minimum length edges>;
        E = E \ {e};   // Remove e
        if (T ∪ {e} doesnot produce a circuit){
            T = T ∪ {e}   // Adds e to the spanning tree
            d(H) = d(H) + d(e);  // Update the length
        }
    }
    Step 4: Return result
    if (|T| < n-1) <Not connected>;
    else Return (T, d(H));
}
```

13- Nguyễn Việt Hoàng - B22DCVT214

Date No

b) # 1

	Edge	T U e
1	$E \setminus (1,2)$	$T = T \cup (1,2)$, $D(T) = 1$
2	$E = E \setminus (1,4)$	$T = T \cup (1,4)$; $D(T) = 2$
3	$E = E \setminus (1,6)$	$T = T \cup (1,6)$, $D(T) = 3$
4	$E = E \setminus (1,9)$	$T = T \cup (1,9)$; $D(T) = 4$
5	$E = E \setminus (1,10)$	$T = T \cup (1,10)$; $D(T) = 5$
6	$E = E \setminus (2,3)$	$T = T \cup (2,3)$; $D(T) = 7$
7	$E = E \setminus (2,4)$	
8	$E = E \setminus (2,6)$	
9	$E = E \setminus (2;8)$	$T = T \cup (2,8)$; $D(T) = 9$
10	$E = E \setminus (3,4)$	
11	$E = E \setminus (3,5)$	$T = T \cup (3,5)$; $D(T) = 12$
12	$E = E \setminus (3,7)$	$T = T \cup (3,7)$; $D(T) = 15$

$T = 9 = n-1$, Loop ended

$T = \{(1,2);(1,4);(1,6);(1,9);(1,10);(2;3);(2,8);(3,5);(3,7)\}$

$D(T) = 15$

Bài 4 đề 2

a) Prim (int s){

Step1: Initialize

$V_H = \{s\}$; // At the beginning V_H only contains only s

$V = V \setminus \{s\}$; // Remove s from V

$T = \emptyset$; // Spanning tree is empty

$d(H) = 0$; // length is 0

Step2: Loop

While ($V \neq \emptyset$) {

$e = (u,v)$ // Minimum length edge with $u \in V, v \in V_H$

if (e doesnot exist)

return <not connected>;

$T = T \cup \{e\}$; //Add e to the spanning tree

$d(H) = d(H) + d(e)$ //update length

$V_H = V_H \cup \{u\}$ //Add u to V_H

$V = V \setminus \{u\}$; //Remove u from V

}

Step3: Return result

return (T, d(H));

NOTEBOOK

13 - Nguyễn Việt Hoàng - B22DCVT214

Date No

b) (u, v)	$V \setminus \{u\}$	$V_H \cup \{u\}$	T , D(T)
(8,2)	1,3,4,5,6,7,9,10	2,8	$T = T \cup (8,2)$, $D(T) = 1$
(1,2)	3,4,5,6,7,9,10	1,2,8	$T = T \cup (1,2)$; $D(T) = 2$
(3,2)	4,5,6,7,9,10	1,2,3,8	$T = T \cup (3,2)$, $D(T) = 3$
(4,2)	5,6,7,9,10	1,2,3,4,8	$T = T \cup (4,2)$; $D(T) = 4$
(10,2)	5,6,7,9	1,2,3,4,8,10	$T = T \cup (10,2)$; $D(T) = 5$
(6,1)	5,7,9	1,2,3,4,6,8,10	$T = T \cup (6,1)$; $D(T) = 6$
(9,1)	5,7	1,2,3,4,6,8,9,10	$T = T \cup (9,1)$; $D(T) = 7$
(5,3)	7	1,2,3,4,5,6,8,9,10	$T = T \cup (5,3)$; $D(T) = 9$
(7,5)	∅	1,2,3,4,5,6,7,8,9,10	$T = T \cup (7,5)$; $D(T) = 13$

$V = \emptyset$; loop ended

$T = \{(8,2); (1,2); (3,2); (4,2); (10,2); (6,1); (9,1); (5,3); (7,5)\}$

$D(T) = 13$

Bài 4 đề 3:

a) Prim(int s) {

Step 1: Initialize

$V_H = \{s\}$; // At the beginning V_H contains only s

$V = V \setminus \{s\}$; // Remove s from V

$T = \emptyset$; // spanning tree is empty

$d(H) = 0$; // length is 0

Step 2: loop:

while ($V \neq \emptyset$) {

e = (u, v) // minimum length edge with $u \in V$, $v \in V_H$

if (e does not exist)

~~return~~ return <Not connected>;

$T = T \cup \{e\}$; // Add e to the spanning tree

$d(H) = d(H) + d(e)$ // update the length

~~VH~~ $V_H = V_H \cup \{u\}$ // Add u to V_H

$V = V \setminus \{u\}$; // Remove u from V

}

Step 3: Return result

return (T, d(H));

}

NOTEBOOK

13- Nguyễn Việt Hoàng - B22DCVT214

Date No

b) (u,v)	V \ {u}	V_H ∪ {u}	T, D(T)
(5,6)	1,2,3,4,7,8,9,10	5,6	T = T ∪ (5,6), D(T) = 1
(5,1)	2,3,4,7,8,9,10	1,5,6	T = T ∪ (5,1), D(T) = 2
(2,5)	3,4,7,8,9,10	1,2,5,6	T = T ∪ (2,5), D(T) = 3
(3,5)	4,7,8,9,10	1,2,3,5,6	T = T ∪ (3,5), D(T) = 4
(4,5)	7,8,9,10	1,2,3,4,5,6	T = T ∪ (4,5), D(T) = 5
(7,1)	8,9,10	1,2,3,4,5,6,7	T = T ∪ (7,1), D(T) = 12
(8,7)	9,10	1,2,3,4,5,6,7,8	T = T ∪ (8,7), D(T) = 19
(9,7)	10	1,2,3,4,5,6,7,8,9	T = T ∪ (9,7), D(T) = 26
(10,8)	∅	1,2,3,4,5,6,7,8,9,10	T = T ∪ (10,8, D(T) = 34

V = ∅, loop ended

T = {(5,6); (5,1); (2,5); (3,5), (4,5); (7,1); (8,7); (9,7); (10,8)}

D(T) = 34

Bài 4 đề 4

a) Kruskal() {
  Step1: Initialize
  T = ∅; // At the beginning the set of edge is empty
  d(H) = 0; // Length equals to 0
  Step2: Sort
  < Sort the edges of the graph in the ascending order of length
  Step3: Loop
  while (|T| < n-1 && E ≠ ∅) {
    e = < The minimum length edge >;
    E = E \ {e}; // Remove e
    if (T ∪ {e} does not produce a circuit) {
      T = T ∪ {e}; // Add e to the spanning tree
      d(H) = d(H) + d(e); // update the length
    }
  }
  Step4: Return result
  if (|T| < n-1) < Not connected >;
  else return (T, d(H));
}

NOTEBOOK

13. Nguyễn Việt Hoàng - B22DCVT214

Date

No

b)

#	Edge	T ∪ e
1	E \ (7,4)	T = T ∪ (7,4), D(T) = 1
2	E \ (7,5)	T = T ∪ (7,5), D(T) = 2
3	E \ (7,6)	T = T ∪ (7,6), D(T) = 3
4	E \ (7,8)	T = T ∪ (7,8), D(T) = 4
5	E \ (5,8)	
6	E \ (9,8)	T = T ∪ (9,8), D(T) = 7
7	E \ (9,1)	T = T ∪ (9,1), D(T) = 12
8	E \ (9,10)	T = T ∪ (9,10), D(T) = 15
9	E \ (10,1)	
10	E \ (10,2)	T = T ∪ (10,2), D(T) = 22
11	E \ (5,3)	T = T ∪ (5,3), D(T) = 33

|T| = 9 = n - 1, Loop Ended

T = { (7,4); (7,5); (7,6); (7,8); (9,8); (9,1); (9,10); (10,2); (5,3) }

D(T) = 33