

# Contents

<b>1</b>	<b>Rubric</b>	<b>8</b>
<b>2</b>	<b>Metadata</b>	<b>9</b>
2.1	Submitted Files	9
2.2	webgrader Runs	10
2.3	diffs	11
<b>3</b>	<b>Written Exercises</b>	<b>12</b>
<b>4</b>	<b>Programming Exercises</b>	<b>18</b>
4.1	csce310h0mework03part01	18
4.1.1	Test 01	18
	diff	18
	Input	18
	Submission Output	18
	Solution Output	18
	stderr	18
4.1.2	Test 02	18
	diff	18
	Input	18
	Submission Output	19
	Solution Output	19
	stderr	19
4.1.3	Test 03	19
	diff	19
	Input	19
	Submission Output	19
	Solution Output	19
	stderr	19
4.1.4	Test 04	19
	diff	19
	Input	19
	Submission Output	20
	Solution Output	20
	stderr	20
4.1.5	Test 05	20
	diff	20
	Input	20
	Submission Output	20
	Solution Output	20
	stderr	20
4.1.6	Test 06	20
	diff	20
	Input	20
	Submission Output	21
	Solution Output	21
	stderr	21
4.1.7	Test 07	21
	diff	21

	Input	21
	Submission Output	21
	Solution Output	21
	<b>stderr</b>	21
4.1.8	Test 08	21
	<b>diff</b>	21
	Input	21
	Submission Output	22
	Solution Output	22
	<b>stderr</b>	22
4.1.9	Test 09	22
	<b>diff</b>	22
	Input	22
	Submission Output	22
	Solution Output	22
	<b>stderr</b>	22
4.1.10	Test 10	22
	<b>diff</b>	22
	Input	22
	Submission Output	23
	Solution Output	23
	<b>stderr</b>	23
4.1.11	Test 11	23
	<b>diff</b>	23
	Input	23
	Submission Output	23
	Solution Output	23
	<b>stderr</b>	23
4.1.12	Test 12	23
	<b>diff</b>	23
	Input	23
	Submission Output	24
	Solution Output	24
	<b>stderr</b>	24
4.1.13	Test 13	24
	<b>diff</b>	24
	Input	24
	Submission Output	24
	Solution Output	24
	<b>stderr</b>	24
4.1.14	Test 14	24
	<b>diff</b>	24
	Input	25
	Submission Output	25
	Solution Output	25
	<b>stderr</b>	25
4.1.15	Test 15	25
	<b>diff</b>	25
	Input	25
	Submission Output	25
	Solution Output	25
	<b>stderr</b>	25
4.1.16	Test 16	26
	<b>diff</b>	26
	Input	26
	Submission Output	26
	Solution Output	26
	<b>stderr</b>	26
4.1.17	Test 17	26
	<b>diff</b>	26

	Input	26
	Submission Output	26
	Solution Output	26
	<b>stderr</b>	27
4.1.18	Test 18	27
	<b>diff</b>	27
	Input	27
	Submission Output	27
	Solution Output	27
	<b>stderr</b>	27
4.1.19	Test 19	27
	<b>diff</b>	27
	Input	27
	Submission Output	27
	Solution Output	28
	<b>stderr</b>	28
4.1.20	Test 20	28
	<b>diff</b>	28
	Input	28
	Submission Output	28
	Solution Output	28
	<b>stderr</b>	28
4.1.21	Test 21	28
	<b>diff</b>	28
	Input	28
	Submission Output	28
	Solution Output	28
	<b>stderr</b>	29
4.1.22	Test 22	29
	<b>diff</b>	29
	Input	29
	Submission Output	29
	Solution Output	29
	<b>stderr</b>	29
4.1.23	Test 23	29
	<b>diff</b>	29
	Input	29
	Submission Output	29
	Solution Output	29
	<b>stderr</b>	30
4.1.24	Test 24	30
	<b>diff</b>	30
	Input	30
	Submission Output	30
	Solution Output	30
	<b>stderr</b>	30
4.1.25	Test 25	30
	<b>diff</b>	30
	Input	30
	Submission Output	30
	Solution Output	30
	<b>stderr</b>	31
4.1.26	Source Code	31
4.2	<b>csce310h0mework03part02</b>	32
4.2.1	Test 01	32
	<b>diff</b>	32
	Input	32
	Submission Output	32
	Solution Output	32
	<b>stderr</b>	32

4.2.2	Test 02	32
	diff	32
	Input	32
	Submission Output	33
	Solution Output	33
	stderr	33
4.2.3	Test 03	33
	diff	33
	Input	33
	Submission Output	33
	Solution Output	34
	stderr	34
4.2.4	Test 04	34
	diff	34
	Input	34
	Submission Output	34
	Solution Output	34
	stderr	34
4.2.5	Test 05	34
	diff	34
	Input	34
	Submission Output	35
	Solution Output	35
	stderr	35
4.2.6	Test 06	35
	diff	35
	Input	35
	Submission Output	36
	Solution Output	36
	stderr	36
4.2.7	Test 07	36
	diff	36
	Input	36
	Submission Output	36
	Solution Output	36
	stderr	37
4.2.8	Test 08	37
	diff	37
	Input	37
	Submission Output	37
	Solution Output	37
	stderr	37
4.2.9	Test 09	37
	diff	37
	Input	37
	Submission Output	38
	Solution Output	38
	stderr	38
4.2.10	Test 10	38
	diff	38
	Input	38
	Submission Output	38
	Solution Output	38
	stderr	39
4.2.11	Test 11	39
	diff	39
	Input	39
	Submission Output	39
	Solution Output	39
	stderr	39

4.2.12	Test 12	39
	diff	39
	Input	40
	Submission Output	40
	Solution Output	40
	stderr	40
4.2.13	Test 13	40
	diff	40
	Input	40
	Submission Output	41
	Solution Output	41
	stderr	41
4.2.14	Test 14	41
	diff	41
	Input	41
	Submission Output	41
	Solution Output	41
	stderr	41
4.2.15	Test 15	42
	diff	42
	Input	42
	Submission Output	42
	Solution Output	42
	stderr	42
4.2.16	Test 16	42
	diff	42
	Input	43
	Submission Output	43
	Solution Output	43
	stderr	43
4.2.17	Test 17	43
	diff	43
	Input	43
	Submission Output	43
	Solution Output	44
	stderr	44
4.2.18	Test 18	44
	diff	44
	Input	44
	Submission Output	44
	Solution Output	44
	stderr	45
4.2.19	Test 19	45
	diff	45
	Input	45
	Submission Output	45
	Solution Output	45
	stderr	45
4.2.20	Test 20	45
	diff	45
	Input	45
	Submission Output	46
	Solution Output	46
	stderr	46
4.2.21	Test 21	46
	diff	46
	Input	46
	Submission Output	46
	Solution Output	46
	stderr	47

4.2.22	Test 22	47
	diff	47
	Input	47
	Submission Output	47
	Solution Output	47
	stderr	47
4.2.23	Test 23	47
	diff	47
	Input	47
	Submission Output	48
	Solution Output	48
	stderr	48
4.2.24	Test 24	48
	diff	48
	Input	48
	Submission Output	48
	Solution Output	48
	stderr	49
4.2.25	Test 25	49
	diff	49
	Input	49
	Submission Output	49
	Solution Output	49
	stderr	49
4.2.26	Source Code	49
4.3	csce310h0mework03part03	50
4.3.1	Test 01	50
	diff	50
	Input	50
	Submission Output	51
	Solution Output	51
	stderr	51
4.3.2	Test 02	51
	diff	51
	Input	51
	Submission Output	51
	Solution Output	51
	stderr	51
4.3.3	Test 03	51
	diff	51
	Input	51
	Submission Output	52
	Solution Output	52
	stderr	52
4.3.4	Test 04	52
	diff	52
	Input	52
	Submission Output	52
	Solution Output	52
	stderr	52
4.3.5	Test 05	52
	diff	52
	Input	52
	Submission Output	53
	Solution Output	53
	stderr	53
4.3.6	Test 06	53
	diff	53
	Input	53
	Submission Output	53

	Solution Output	53
	<b>stderr</b>	53
4.3.7	Test 07	53
	<b>diff</b>	53
	Input	53
	Submission Output	54
	Solution Output	54
	<b>stderr</b>	54
4.3.8	Test 08	54
	<b>diff</b>	54
	Input	54
	Submission Output	54
	Solution Output	54
	<b>stderr</b>	54
4.3.9	Test 09	54
	<b>diff</b>	54
	Input	54
	Submission Output	55
	Solution Output	55
	<b>stderr</b>	55
4.3.10	Test 10	55
	<b>diff</b>	55
	Input	55
	Submission Output	55
	Solution Output	55
	<b>stderr</b>	55
4.3.11	Source Code	55

# Chapter 1

## Rubric

Question	Points
Question 1	10
Question 2	10
Question 3	10
Question 4	10
<b>maximumST</b>	
Test Cases	$1 \times 25$
Compilation	5
<b>maximumST Total</b>	30
<b>allPairsSP</b>	
Test Cases	$1 \times 25$
Compilation	5
<b>allPairsSP Total</b>	30
Total	100



# Chapter 2

## Metadata

### 2.1 Submitted Files

handin.time					
1	10/26/2019	20:25:47	fsandhu:	csce310h0mework03part01.cpp	- OK
2	10/26/2019	20:25:49	fsandhu:	csce310h0mework03part01.h	- OK
3	10/26/2019	20:25:51	fsandhu:	csce310h0mework03part02.cpp	- OK
4	10/26/2019	20:25:54	fsandhu:	csce310h0mework03part02.h	- OK
5	10/26/2019	20:25:57	fsandhu:	csce310h0mework03part03.h	- OK
6	10/26/2019	20:25:59	fsandhu:	csce310h0mework03part03.cpp	- OK
7	10/26/2019	20:33:03	fsandhu:	csce310h0mework03part02.cpp	- OK
8	10/26/2019	20:35:01	fsandhu:	csce310h0mework03part02.cpp	- OK
9	10/26/2019	20:36:24	fsandhu:	csce310h0mework03part02.cpp	- OK
10	10/26/2019	20:38:09	fsandhu:	csce310h0mework03part02.cpp	- OK
11	10/26/2019	20:38:43	fsandhu:	csce310h0mework03part02.cpp	- OK
12	10/26/2019	20:39:52	fsandhu:	csce310h0mework03part02.cpp	- OK
13	10/26/2019	20:41:12	fsandhu:	csce310h0mework03part02.cpp	- OK
14	10/26/2019	20:42:27	fsandhu:	csce310h0mework03part02.cpp	- OK
15	10/26/2019	20:46:32	fsandhu:	csce310h0mework03part02.cpp	- OK
16	10/26/2019	20:47:52	fsandhu:	csce310h0mework03part02.cpp	- OK
17	10/26/2019	20:51:29	fsandhu:	csce310h0mework03part02.cpp	- OK
18	10/26/2019	21:30:09	fsandhu:	csce310h0mework03part01.cpp	- OK
19	10/26/2019	21:30:12	fsandhu:	csce310h0mework03part01.h	- OK
20	10/26/2019	21:32:08	fsandhu:	csce310h0mework03part01.cpp	- OK
21	10/26/2019	21:33:11	fsandhu:	csce310h0mework03part01.cpp	- OK
22	10/26/2019	21:33:42	fsandhu:	csce310h0mework03part01.cpp	- OK
23	10/26/2019	21:39:23	fsandhu:	csce310h0mework03part01.cpp	- OK
24	10/28/2019	14:06:25	fsandhu:	csce310h0mework03part01.cpp	- OK
25	10/29/2019	13:37:12	fsandhu:	csce310h0mework03part01.cpp	- OK
26	10/29/2019	14:05:45	fsandhu:	csce310h0mework03part01.cpp	- OK
27	10/29/2019	14:08:50	fsandhu:	csce310h0mework03part01.cpp	- OK
28	10/29/2019	14:09:57	fsandhu:	csce310h0mework03part01.cpp	- OK
29	10/29/2019	14:13:49	fsandhu:	csce310h0mework03part01.cpp	- OK
30	10/29/2019	14:30:55	fsandhu:	csce310h0mework03part01.cpp	- OK
31	10/29/2019	15:02:19	fsandhu:	csce310h0mework03part01.cpp	- OK
32	10/29/2019	15:02:50	fsandhu:	csce310h0mework03part01.cpp	- OK
33	10/29/2019	15:04:00	fsandhu:	csce310h0mework03part01.cpp	- OK
34	10/29/2019	15:15:12	fsandhu:	csce310h0mework03part01.cpp	- OK
35	10/29/2019	15:18:24	fsandhu:	csce310h0mework03part01.cpp	- OK
36	10/29/2019	15:21:06	fsandhu:	csce310h0mework03part01.cpp	- OK
37	10/29/2019	15:22:48	fsandhu:	csce310h0mework03part01.cpp	- OK
38	10/29/2019	15:31:27	fsandhu:	csce310h0mework03part01.cpp	- OK
39	10/29/2019	15:34:14	fsandhu:	csce310h0mework03part02.cpp	- OK
40	10/29/2019	15:34:17	fsandhu:	csce310h0mework03part03.cpp	- OK
41	10/30/2019	13:44:35	fsandhu:	csce310h0mework03part01.cpp	- OK

42	10/30/2019	13:45:52	fsandhu:	csce310h0mework03part01.cpp	-	OK
43	10/30/2019	13:48:11	fsandhu:	csce310h0mework03part01.cpp	-	OK
44	10/30/2019	13:52:50	fsandhu:	csce310h0mework03part01.cpp	-	OK
45	10/30/2019	13:53:51	fsandhu:	csce310h0mework03part01.cpp	-	OK
46	10/30/2019	13:57:09	fsandhu:	csce310h0mework03part01.cpp	-	OK
47	10/30/2019	13:58:08	fsandhu:	csce310h0mework03part01.cpp	-	OK
48	10/30/2019	14:01:00	fsandhu:	csce310h0mework03part01.cpp	-	OK
49	10/30/2019	14:02:29	fsandhu:	csce310h0mework03part01.cpp	-	OK
50	10/30/2019	14:26:39	fsandhu:	csce310h0mework03part01.cpp	-	OK
51	10/30/2019	14:28:15	fsandhu:	csce310h0mework03part01.cpp	-	OK
52	10/30/2019	14:38:58	fsandhu:	csce310h0mework03part01.cpp	-	OK
53	10/30/2019	14:40:14	fsandhu:	csce310h0mework03part01.cpp	-	OK
54	10/30/2019	14:41:10	fsandhu:	csce310h0mework03part01.cpp	-	OK
55	10/30/2019	14:43:24	fsandhu:	csce310h0mework03part01.cpp	-	OK
56	10/30/2019	14:47:31	fsandhu:	csce310h0mework03part01.cpp	-	OK
57	10/30/2019	14:55:44	fsandhu:	csce310h0mework03part01.cpp	-	OK
58	10/30/2019	15:02:03	fsandhu:	csce310h0mework03part01.cpp	-	OK
59	10/30/2019	15:04:12	fsandhu:	csce310h0mework03part01.cpp	-	OK
60	10/30/2019	15:13:04	fsandhu:	csce310h0mework03part01.cpp	-	OK
61	10/30/2019	15:22:52	fsandhu:	csce310h0mework03part01.cpp	-	OK
62	10/30/2019	15:23:57	fsandhu:	csce310h0mework03part01.cpp	-	OK
63	10/30/2019	15:24:48	fsandhu:	csce310h0mework03part01.cpp	-	OK
64	10/30/2019	15:24:51	fsandhu:	csce310h0mework03part01.h	-	OK
65	10/30/2019	15:29:51	fsandhu:	csce310h0mework03part01.cpp	-	OK
66	10/30/2019	15:29:53	fsandhu:	csce310h0mework03part01.h	-	OK
67	10/30/2019	15:30:05	fsandhu:	csce310h0mework03part02.cpp	-	OK
68	10/30/2019	15:30:10	fsandhu:	csce310h0mework03part02.h	-	OK
69	10/30/2019	15:30:16	fsandhu:	csce310h0mework03part03.cpp	-	OK
70	10/30/2019	15:30:21	fsandhu:	csce310h0mework03part03.h	-	OK
71	10/30/2019	21:27:51	fsandhu:	hw3.pdf	-	OK

## 2.2 webgrader Runs

				webgrader.time
1	2019-10-26T20:26:12-0500	97.98.163.171	fsandhu	003
2	2019-10-26T20:33:05-0500	97.98.163.171	fsandhu	003
3	2019-10-26T20:35:03-0500	97.98.163.171	fsandhu	003
4	2019-10-26T20:36:27-0500	97.98.163.171	fsandhu	003
5	2019-10-26T20:38:11-0500	97.98.163.171	fsandhu	003
6	2019-10-26T20:38:46-0500	97.98.163.171	fsandhu	003
7	2019-10-26T20:39:09-0500	97.98.163.171	fsandhu	003
8	2019-10-26T20:39:47-0500	97.98.163.171	fsandhu	003
9	2019-10-26T20:39:56-0500	97.98.163.171	fsandhu	003
10	2019-10-26T20:41:14-0500	97.98.163.171	fsandhu	003
11	2019-10-26T20:42:29-0500	97.98.163.171	fsandhu	003
12	2019-10-26T20:43:37-0500	97.98.163.171	fsandhu	003
13	2019-10-26T20:46:34-0500	97.98.163.171	fsandhu	003
14	2019-10-26T20:47:55-0500	97.98.163.171	fsandhu	003
15	2019-10-26T20:51:31-0500	97.98.163.171	fsandhu	003
16	2019-10-26T21:30:15-0500	97.98.163.171	fsandhu	003
17	2019-10-26T21:30:28-0500	97.98.163.171	fsandhu	003
18	2019-10-26T21:32:10-0500	97.98.163.171	fsandhu	003
19	2019-10-26T21:33:13-0500	97.98.163.171	fsandhu	003
20	2019-10-26T21:33:44-0500	97.98.163.171	fsandhu	003
21	2019-10-26T21:39:25-0500	97.98.163.171	fsandhu	003
22	2019-10-28T14:06:38-0500	10.43.86.250	fsandhu	003
23	2019-10-29T13:37:17-0500	10.43.43.198	fsandhu	003
24	2019-10-29T14:05:47-0500	10.43.43.198	fsandhu	003
25	2019-10-29T14:08:52-0500	10.43.43.198	fsandhu	003

26	2019-10-29T14:09:19-0500	10.43.43.198	fsandhu	003
27	2019-10-29T14:10:06-0500	10.43.43.198	fsandhu	003
28	2019-10-29T14:13:57-0500	10.43.43.198	fsandhu	003
29	2019-10-29T14:30:58-0500	10.43.43.198	fsandhu	003
30	2019-10-29T15:02:20-0500	10.43.43.198	fsandhu	003
31	2019-10-29T15:02:59-0500	10.43.43.198	fsandhu	003
32	2019-10-29T15:04:02-0500	10.43.43.198	fsandhu	003
33	2019-10-29T15:15:15-0500	10.43.43.198	fsandhu	003
34	2019-10-29T15:16:10-0500	10.43.43.198	fsandhu	003
35	2019-10-29T15:18:33-0500	10.43.43.198	fsandhu	003
36	2019-10-29T15:21:25-0500	10.43.43.198	fsandhu	003
37	2019-10-29T15:22:49-0500	10.43.43.198	fsandhu	003
38	2019-10-29T15:31:28-0500	10.43.43.198	fsandhu	003
39	2019-10-30T13:29:50-0500	10.43.107.14	fsandhu	003
40	2019-10-30T13:44:54-0500	10.43.107.14	fsandhu	003
41	2019-10-30T13:45:54-0500	10.43.107.14	fsandhu	003
42	2019-10-30T13:48:13-0500	10.43.107.14	fsandhu	003
43	2019-10-30T13:48:51-0500	10.43.107.14	fsandhu	003
44	2019-10-30T13:52:51-0500	10.43.107.14	fsandhu	003
45	2019-10-30T13:53:53-0500	10.43.107.14	fsandhu	003
46	2019-10-30T13:57:10-0500	10.43.107.14	fsandhu	003
47	2019-10-30T13:58:09-0500	10.43.107.14	fsandhu	003
48	2019-10-30T14:01:02-0500	10.43.107.14	fsandhu	003
49	2019-10-30T14:02:31-0500	10.43.107.14	fsandhu	003
50	2019-10-30T14:02:41-0500	10.43.107.14	fsandhu	003
51	2019-10-30T14:26:41-0500	10.43.107.14	fsandhu	003
52	2019-10-30T14:28:17-0500	10.43.107.14	fsandhu	003
53	2019-10-30T14:39:01-0500	10.43.107.14	fsandhu	003
54	2019-10-30T14:40:10-0500	10.43.107.14	fsandhu	003
55	2019-10-30T14:40:31-0500	10.43.107.14	fsandhu	003
56	2019-10-30T14:41:12-0500	10.43.107.14	fsandhu	003
57	2019-10-30T14:43:26-0500	10.43.107.14	fsandhu	003
58	2019-10-30T14:47:32-0500	10.43.107.14	fsandhu	003
59	2019-10-30T14:55:46-0500	10.43.107.14	fsandhu	003
60	2019-10-30T15:02:05-0500	10.43.107.14	fsandhu	003
61	2019-10-30T15:04:13-0500	10.43.107.14	fsandhu	003
62	2019-10-30T15:13:06-0500	10.43.107.14	fsandhu	003
63	2019-10-30T15:22:55-0500	10.43.107.14	fsandhu	003
64	2019-10-30T15:23:59-0500	10.43.107.14	fsandhu	003
65	2019-10-30T15:24:35-0500	10.43.107.14	fsandhu	003
66	2019-10-30T15:24:52-0500	10.43.107.14	fsandhu	003
67	2019-10-30T15:26:45-0500	10.43.107.14	fsandhu	003
68	2019-10-30T15:30:46-0500	10.43.107.14	fsandhu	003
69	2019-10-30T18:57:24-0500	76.84.50.181	fsandhu	003
70	2019-10-30T21:28:00-0500	76.84.50.181	fsandhu	003
71	2019-12-15T19:52:44-0600	76.84.219.87	fsandhu	003

## 2.3 diffs

submission.diffs

## Chapter 3

### Written Exercises

Fateh Karan Singh Sandhu

NUID: 17286643

Assignment 3 → written part.

Q1).

a). Input: AVL tree of real numbers

Output: range.

```
presentNode ← root
while (presentNode.leftChild != NULL)
{
    presentNode ← presentNode.leftChild
    smallest ← presentNode.value
}
```

```
presentNode ← root
while (presentNode.rightChild != NULL)
{
    presentNode ← presentNode.rightChild
    largest ← presentNode.value
}
```

range ← largest - smallest

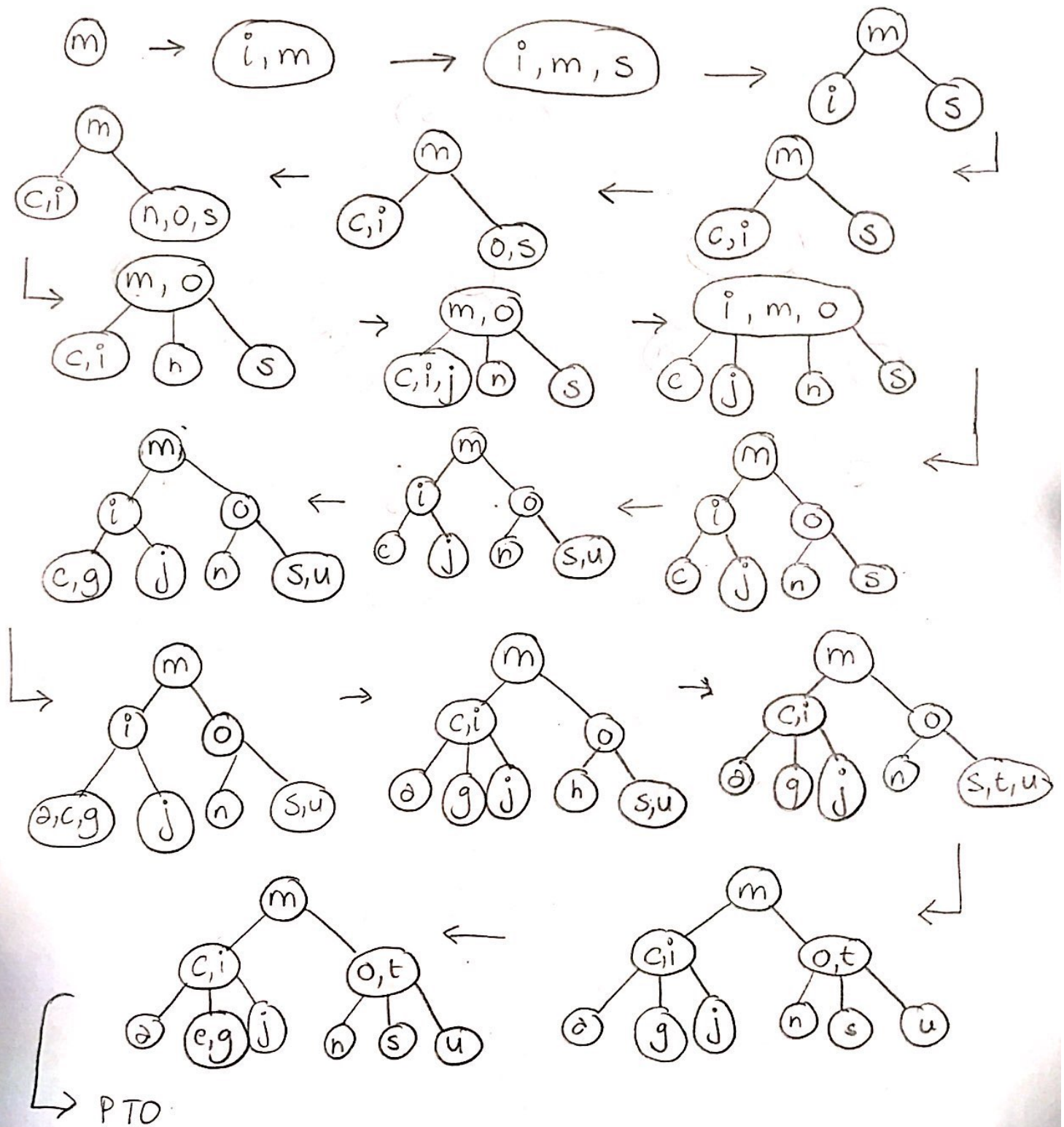
output range

★ worst case efficiency  
 $O(\log(n))$

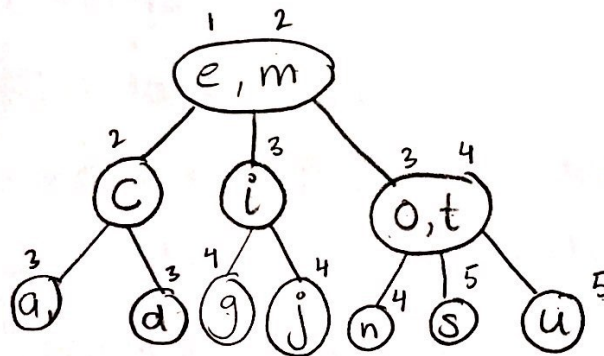
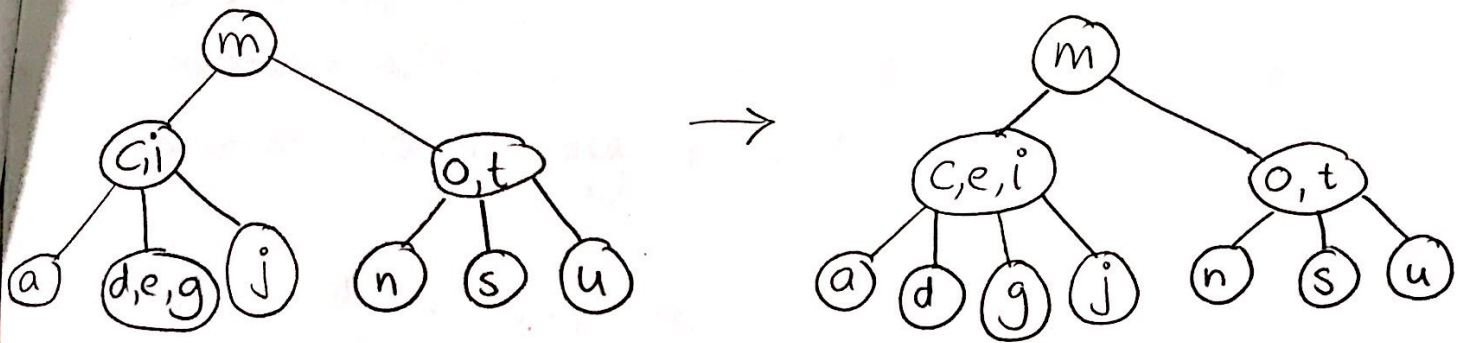
b). False, smallest value in AVL will always be the leftmost  
largest will always be the rightmost

Q2)

The 2-3 tree will look like







b) The largest number of comparisons will be 5

average no. of comparison:

$$1 + 2 + 2 + 3 + 3 + 3 + 4 + 4 + 3 + 4 + 4 + 5 + 5$$

13

$$\Rightarrow 3.3076 \text{ comparisons}$$

Q3). In 2-3 tree, the leftmost node's left element has the smallest value and the rightmost node's right element has the largest value.

Input: A 2-3 tree

Output: Range

presentNode  $\leftarrow$  root

while (presentNode.leftChild  $\neq$  NULL)

presentNode  $\leftarrow$  presentNode.leftChild

if (presentNode is a 3 node)

smallest  $\leftarrow$  presentNode.leftElementValue  
else

smallest  $\leftarrow$  presentNode.value

presentNode  $\leftarrow$  root

while (presentNode.rightChild  $\neq$  NULL)

presentNode  $\leftarrow$  presentNode.rightChild

if (presentNode is a 3 node)

largest  $\leftarrow$  presentNode.rightElementValue  
else

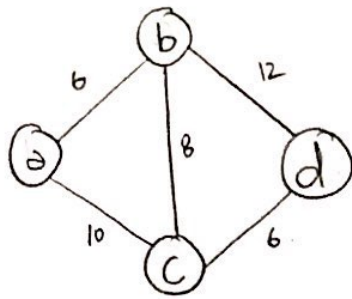
largest  $\leftarrow$  presentNode.value

range  $\leftarrow$  largest - smallest  $\star$  worst case

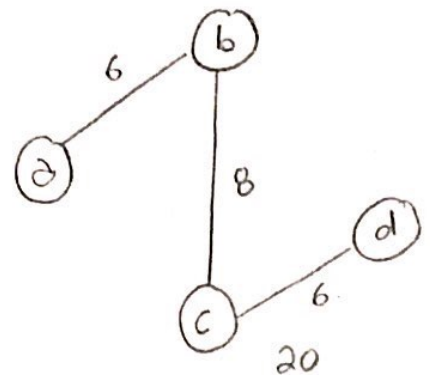
output range  $O(\log(n))$



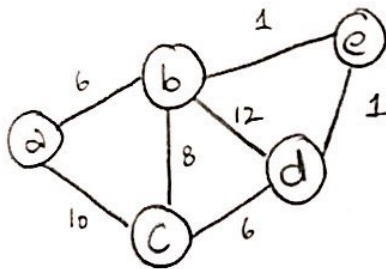
Q4) No, we can give a counter example



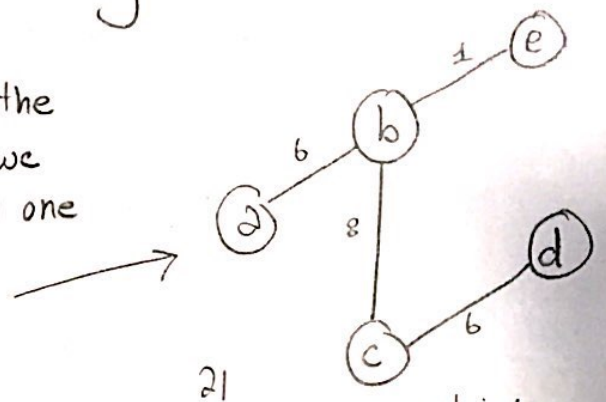
Minimum Spanning Tree



If we add a new vertex and new edges

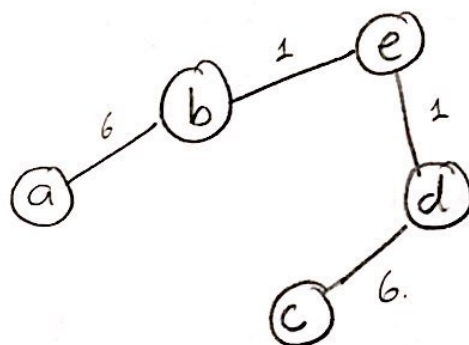


according to the question, we can add any one edge



X which is not the correct MST

actual minimum spanning tree will now be



14

Algorithm is not valid

The added vertex may have edges to other nodes which may change the steps to proceed in Prim's algorithm.

## Chapter 4

# Programming Exercises

### 4.1 csce310h0mework03part01

#### 4.1.1 Test 01

diff

part01test01.diff

**Input**

part01test01.matrix.input

```
0 7 0 7 7 2 2 4
7 0 9 3 11 12 12 1
0 9 0 5 5 7 7 3
7 3 5 0 1 5 8 0
7 11 5 1 0 9 15 6
2 12 7 5 9 0 3 4
2 12 7 8 15 3 0 1
4 1 3 0 6 4 1 0
```

**Submission Output**

part01test01.output

Maximum spanning tree value: 69

**Solution Output**

part01test01.solution

Maximum spanning tree value: 69

stderr

part01test01.err

#### 4.1.2 Test 02

diff

part01test02.diff

**Input**

part01test02.matrix.input

```
0 1 6 9
1 0 7 1
6 7 0 8
9 1 8 0
```

## Submission Output

part01test02.output

Maximum spanning tree value: 24

## Solution Output

part01test02.solution

Maximum spanning tree value: 24

stderr

part01test02.err

## 4.1.3 Test 03

diff

part01test03.diff

## Input

part01test03.matrix.input

```
0 4 6 2 0 1 4 5 5
4 0 10 6 4 17 10 7 3
6 10 0 10 2 7 17 8 1
2 6 10 0 6 10 11 10 3
0 4 2 6 0 3 3 1 4
1 17 7 10 3 0 7 11 2
4 10 17 11 3 7 0 4 3
5 7 8 10 1 11 4 0 3
5 3 1 3 4 2 3 3 0
```

## Submission Output

part01test03.output

Maximum spanning tree value: 83

## Solution Output

part01test03.solution

Maximum spanning tree value: 83

stderr

part01test03.err

## 4.1.4 Test 04

diff

part01test04.diff

## Input

part01test04.matrix.input

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

```
7 2 9 7 0 9 5 13 6 3
2 1 9 5 9 0 3 12 9 8
8 2 1 1 5 3 0 3 0 0
6 4 2 10 13 12 3 0 2 5
3 3 1 3 6 9 0 2 0 0
3 6 9 8 3 8 0 5 0 0
```

#### Submission Output

part01test04.output

Maximum spanning tree value: 83

#### Solution Output

part01test04.solution

Maximum spanning tree value: 83

stderr

part01test04.err

#### 4.1.5 Test 05

diff

part01test05.diff

#### Input

part01test05.matrix.input

```
0 6 1 4 8 6 3 5 9
6 0 9 17 9 8 3 8 8
1 9 0 13 11 5 4 5 6
4 17 13 0 2 11 11 8 9
8 9 11 2 0 8 10 2 4
6 8 5 11 8 0 14 8 4
3 3 4 11 10 14 0 1 8
5 8 5 8 2 8 1 0 0
9 8 6 9 4 4 8 0 0
```

#### Submission Output

part01test05.output

Maximum spanning tree value: 92

#### Solution Output

part01test05.solution

Maximum spanning tree value: 92

stderr

part01test05.err

#### 4.1.6 Test 06

diff

part01test06.diff

#### Input

part01test06.matrix.input

```
0 0 4 2 1
0 0 5 4 4
4 5 0 1 0
2 4 1 0 8
1 4 0 8 0
```

#### Submission Output

part01test06.output

```
Maximum spanning tree value: 21
```

#### Solution Output

part01test06.solution

```
Maximum spanning tree value: 21
stderr
```

part01test06.err

#### 4.1.7 Test 07

diff

part01test07.diff

#### Input

part01test07.matrix.input

```
0 8 6
8 0 1
6 1 0
```

#### Submission Output

part01test07.output

```
Maximum spanning tree value: 14
```

#### Solution Output

part01test07.solution

```
Maximum spanning tree value: 14
stderr
```

part01test07.err

#### 4.1.8 Test 08

diff

part01test08.diff

#### Input

part01test08.matrix.input

```
0 9 8 4 2 7 1
9 0 4 12 12 8 7
8 4 0 7 9 14 8
4 12 7 0 13 9 4
2 12 9 13 0 9 8
7 8 14 9 9 0 2
1 7 8 4 8 2 0
```

## Submission Output

part01test08.output

Maximum spanning tree value: 65

## Solution Output

part01test08.solution

Maximum spanning tree value: 65

stderr

part01test08.err

## 4.1.9 Test 09

diff

part01test09.diff

## Input

part01test09.matrix.input

```
0 9 5 3 5 7 7 8 7
9 0 6 4 10 5 9 5 3
5 6 0 6 15 3 11 9 4
3 4 6 0 14 8 12 6 4
5 10 15 14 0 3 10 18 8
7 5 3 8 3 0 6 4 0
7 9 11 12 10 6 0 8 4
8 5 9 6 18 4 8 0 1
7 3 4 4 8 0 4 1 0
```

## Submission Output

part01test09.output

Maximum spanning tree value: 94

## Solution Output

part01test09.solution

Maximum spanning tree value: 94

stderr

part01test09.err

## 4.1.10 Test 10

diff

part01test10.diff

## Input

part01test10.matrix.input

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

```
7 4 10 9 0 4 10 13 6
3 4 8 2 4 0 9 2 0
9 3 10 9 10 9 0 9 5
6 9 4 8 13 2 9 0 6
8 7 4 1 6 0 5 6 0
```

#### Submission Output

part01test10.output

```
Maximum spanning tree value: 77
```

#### Solution Output

part01test10.solution

```
Maximum spanning tree value: 77
```

```
stderr
```

part01test10.err

#### 4.1.11 Test 11

```
diff
```

part01test11.diff

#### Input

part01test11.matrix.input

```
0 7 7 4 4 1 8
7 0 9 12 7 13 1
7 9 0 13 6 9 7
4 12 13 0 2 12 7
4 7 6 2 0 3 0
1 13 9 12 3 0 8
8 1 7 7 0 8 0
```

#### Submission Output

part01test11.output

```
Maximum spanning tree value: 61
```

#### Solution Output

part01test11.solution

```
Maximum spanning tree value: 61
```

```
stderr
```

part01test11.err

#### 4.1.12 Test 12

```
diff
```

part01test12.diff

#### Input

part01test12.matrix.input

```
0 0 6 6 6 7 6 2
0 0 8 4 4 3 4 7
6 8 0 14 8 8 6 4
6 4 14 0 11 7 4 1
6 4 8 11 0 3 5 4
7 3 8 7 3 0 0 0
6 4 6 4 5 0 0 0
2 7 4 1 4 0 0 0
```

#### Submission Output

part01test12.output

Maximum spanning tree value: 61

#### Solution Output

part01test12.solution

Maximum spanning tree value: 61

stderr

part01test12.err

### 4.1.13 Test 13

diff

part01test13.diff

#### Input

part01test13.matrix.input

```
0 1 9 6 5 4 3 4 6
1 0 6 11 11 15 4 8 1
9 6 0 12 9 5 13 15 4
6 11 12 0 9 10 15 12 8
5 11 9 9 0 6 12 13 1
4 15 5 10 6 0 5 8 8
3 4 13 15 12 5 0 13 1
4 8 15 12 13 8 13 0 2
6 1 4 8 1 8 1 2 0
```

#### Submission Output

part01test13.output

Maximum spanning tree value: 99

#### Solution Output

part01test13.solution

Maximum spanning tree value: 99

stderr

part01test13.err

### 4.1.14 Test 14

diff



part01test14.diff

### Input

part01test14.matrix.input

```
0 0 4 8 1 4 9
0 0 8 2 8 9 8
4 8 0 11 9 11 1
8 2 11 0 15 12 5
1 8 9 15 0 17 7
4 9 11 12 17 0 1
9 8 1 5 7 1 0
```

### Submission Output

part01test14.output

Maximum spanning tree value: 69

### Solution Output

part01test14.solution

Maximum spanning tree value: 69

stderr

part01test14.err

## 4.1.15 Test 15

diff

part01test15.diff

### Input

part01test15.matrix.input

```
0 0 5 9 2 5 9 4 5 5
0 0 7 7 2 1 9 8 1 6
5 7 0 13 10 15 9 10 12 2
9 7 13 0 9 7 8 3 11 5
2 2 10 9 0 2 1 12 10 1
5 1 15 7 2 0 3 12 4 5
9 9 9 8 1 3 0 8 8 0
4 8 10 3 12 12 8 0 12 7
5 1 12 11 10 4 8 12 0 8
5 6 2 5 1 5 0 7 8 0
```

### Submission Output

part01test15.output

Maximum spanning tree value: 99

### Solution Output

part01test15.solution

Maximum spanning tree value: 99

stderr

part01test15.err

#### 4.1.16 Test 16

diff

part01test16.diff

##### Input

part01test16.matrix.input

```
0 0 7 2 2 4 2 6
0 0 8 3 4 7 1 4
7 8 0 12 13 4 8 6
2 3 12 0 10 9 8 4
2 4 13 10 0 6 4 1
4 7 4 9 6 0 0 0
2 1 8 8 4 0 0 0
6 4 6 4 1 0 0 0
```

##### Submission Output

part01test16.output

Maximum spanning tree value: 63

##### Solution Output

part01test16.solution

Maximum spanning tree value: 63

stderr

part01test16.err

#### 4.1.17 Test 17

diff

part01test17.diff

##### Input

part01test17.matrix.input

```
0 0 5 4 0 3 1 4 8
0 0 2 9 0 8 9 6 4
5 2 0 9 3 6 8 8 8
4 9 9 0 4 13 8 1 1
0 0 3 4 0 3 4 9 2
3 8 6 13 3 0 2 5 4
1 9 8 8 4 2 0 0 0
4 6 8 1 9 5 0 0 0
8 4 8 1 2 4 0 0 0
```

##### Submission Output

part01test17.output

Maximum spanning tree value: 73

##### Solution Output

part01test17.solution

Maximum spanning tree value: 73

stderr

part01test17.err

#### 4.1.18 Test 18

diff

part01test18.diff

##### Input

part01test18.matrix.input

```
0 0 9 0 2 3 8 3 8
0 0 1 0 6 5 3 4 1
9 1 0 5 9 6 12 8 4
0 0 5 0 4 4 3 2 3
2 6 9 4 0 5 10 5 2
3 5 6 4 5 0 9 4 6
8 3 12 3 10 9 0 1 8
3 4 8 2 5 4 1 0 0
8 1 4 3 2 6 8 0 0
```

##### Submission Output

part01test18.output

Maximum spanning tree value: 67

##### Solution Output

part01test18.solution

Maximum spanning tree value: 67

stderr

part01test18.err

#### 4.1.19 Test 19

diff

part01test19.diff

##### Input

part01test19.matrix.input

```
0 9 0 1 0 9 8 3 3 9
9 0 3 9 9 5 6 2 4 9
0 3 0 4 0 9 8 9 9 8
1 9 4 0 8 12 7 7 3 8
0 9 0 8 0 8 7 6 7 1
9 5 9 12 8 0 5 6 3 9
8 6 8 7 7 5 0 2 5 2
3 2 9 7 6 6 2 0 0 0
3 4 9 3 7 3 5 0 0 0
9 9 8 8 1 9 2 0 0 0
```

##### Submission Output

part01test19.output

Maximum spanning tree value: 83

## Solution Output

part01test19.solution

Maximum spanning tree value: 83  
stderr

part01test19.err

## 4.1.20 Test 20

diff

part01test20.diff

## Input

part01test20.matrix.input

0 4 0 2 4  
4 0 5 6 1  
0 5 0 9 4  
2 6 9 0 0  
4 1 4 0 0

## Submission Output

part01test20.output

Maximum spanning tree value: 23

## Solution Output

part01test20.solution

Maximum spanning tree value: 23  
stderr

part01test20.err

## 4.1.21 Test 21

diff

part01test21.diff

## Input

part01test21.matrix.input

0 0 4 2 8 9  
0 0 9 6 3 2  
4 9 0 12 3 7  
2 6 12 0 1 7  
8 3 3 1 0 0  
9 2 7 7 0 0

## Submission Output

part01test21.output

Maximum spanning tree value: 45

## Solution Output

part01test21.solution

Maximum spanning tree value: 45

stderr

part01test21.err

#### 4.1.22 Test 22

diff

part01test22.diff

##### Input

part01test22.matrix.input

```
0 1 8 8 0 3
1 0 11 18 3 3
8 11 0 7 9 5
8 18 7 0 4 7
0 3 9 4 0 2
3 3 5 7 2 0
```

##### Submission Output

part01test22.output

Maximum spanning tree value: 53

##### Solution Output

part01test22.solution

Maximum spanning tree value: 53

stderr

part01test22.err

#### 4.1.23 Test 23

diff

part01test23.diff

##### Input

part01test23.matrix.input

```
0 0 2 8 4 8 2 4
0 0 3 6 6 9 7 5
2 3 0 7 13 5 10 2
8 6 7 0 8 4 7 0
4 6 13 8 0 12 11 8
8 9 5 4 12 0 16 7
2 7 10 7 11 16 0 1
4 5 2 0 8 7 1 0
```

##### Submission Output

part01test23.output

Maximum spanning tree value: 74

##### Solution Output

part01test23.solution

Maximum spanning tree value: 74

stderr

part01test23.err

#### 4.1.24 Test 24

diff

part01test24.diff

##### Input

part01test24.matrix.input

```
0 0 3 9 4 7 7
0 0 8 6 6 9 8
3 8 0 8 9 3 5
9 6 8 0 12 15 4
4 6 9 12 0 10 2
7 9 3 15 10 0 6
7 8 5 4 2 6 0
```

##### Submission Output

part01test24.output

Maximum spanning tree value: 62

##### Solution Output

part01test24.solution

Maximum spanning tree value: 62

stderr

part01test24.err

#### 4.1.25 Test 25

diff

part01test25.diff

##### Input

part01test25.matrix.input

```
0 0 8 6
0 0 9 6
8 9 0 3
6 6 3 0
```

##### Submission Output

part01test25.output

Maximum spanning tree value: 23

##### Solution Output

part01test25.solution

Maximum spanning tree value: 23

stderr

part01test25.err

#### 4.1.26 Source Code

csce310h0mework03part01.h

```
1 #ifndef CSCE310HOMEWORK03PART01_H
2 #define CSCE310HOMEWORK03PART01_H
3 #include <vector>
4 using namespace std;
5
6 double maximumST( vector< vector<double> > );
7
8 #endif
```

csce310h0mework03part01.cpp

```
1 /**
2  * Author: Fateh Karan Singh Sandhu
3  *
4  * This program uses Prim's algorithm to produce a maximum spanning tree
5  * from a given adjacency matrix
6  */
7
8 #include <vector>
9 #include "csce310h0mework03part01.h"
10 #include <cmath>
11 #include <iostream>
12 #include<algorithm>
13
14 using namespace std;
15
16 double maximumST( vector< vector<double> > adjacencyMatrix ){
17
18     double maximumSpanningTree = 0;
19     vector<int> nodesVisited; //initialize vector
20     nodesVisited.push_back(0); //start at the first node
21     int maxElement = 0;
22     int row = 0;
23     int column = 0;
24
25     for (int l = 0 ; l < adjacencyMatrix.size() ; l++) {
26         for (int i = 0 ; i < nodesVisited.size() ; i++) {
27             for (int j = 0 ; j < adjacencyMatrix.size() ; j++) {
28                 if (adjacencyMatrix[nodesVisited[i]][j] > maxElement) {
29                     maxElement = adjacencyMatrix[nodesVisited[i]][j]; //get max
30                     row = i;
31                     column = j;
32                 }
33             }
34         }
35         maximumSpanningTree += maxElement; //add maxEdge to Tree
36         for (int k = 0 ; k < adjacencyMatrix.size() ; k++) {
37             if (l == 0) {
38                 adjacencyMatrix[k][row] = 0; //take column 0 out of consideration for first node
39             }
40             adjacencyMatrix[k][column] = 0; //take columns out of consideration
41         }
42         nodesVisited.push_back(column); //push back node visited
```

```

43     row = 0;
44     column = 0;
45     maxElement = 0;
46 }
47
48 return maximumSpanningTree;
49 }

```

## 4.2 csce310h0mework03part02

### 4.2.1 Test 01

diff

part02test01.diff

#### Input

part02test01.matrix.input

```

0 -1 -1 9 3 9
-1 0 -1 -1 -1 -1
-1 -1 0 3 2 3
9 -1 3 0 6 8
3 -1 2 6 0 -1
9 -1 3 8 -1 0

```

part02test01.i.input

5

#### Submission Output

part02test01.output

```

0 -1 5 8 3 8
-1 0 -1 -1 -1 -1
5 -1 0 3 2 3
8 -1 3 0 5 6
3 -1 2 5 0 5
8 -1 3 6 5 0

```

#### Solution Output

part02test01.solution

```

0 -1 5 8 3 8
-1 0 -1 -1 -1 -1
5 -1 0 3 2 3
8 -1 3 0 5 6
3 -1 2 5 0 5
8 -1 3 6 5 0

```

stderr

part02test01.err

### 4.2.2 Test 02

diff

part02test02.diff

#### Input



part02test02.matrix.input

```
0 -1 4 -1 1
-1 0 4 -1 9
4 4 0 -1 1
-1 -1 -1 0 -1
1 9 1 -1 0
```

part02test02.i.input

5

**Submission Output**

part02test02.output

```
0 6 2 -1 1
6 0 4 -1 5
2 4 0 -1 1
-1 -1 -1 0 -1
1 5 1 -1 0
```

**Solution Output**

part02test02.solution

```
0 6 2 -1 1
6 0 4 -1 5
2 4 0 -1 1
-1 -1 -1 0 -1
1 5 1 -1 0
```

stderr

part02test02.err

### 4.2.3 Test 03

diff

part02test03.diff

**Input**

part02test03.matrix.input

```
0 -1 3 -1 9 -1 9
-1 0 7 -1 2 -1 5
3 7 0 -1 -1 6 -1
-1 -1 -1 0 -1 -1 -1
9 2 -1 -1 0 7 -1
-1 -1 6 -1 7 0 5
9 5 -1 -1 -1 5 0
```

part02test03.i.input

2

**Submission Output**

part02test03.output

```
0 -1 3 -1 9 -1 9
-1 0 7 -1 2 -1 5
3 7 0 -1 9 6 12
-1 -1 -1 0 -1 -1 -1
9 2 9 -1 0 7 7
-1 -1 6 -1 7 0 5
9 5 12 -1 7 5 0
```

## Solution Output

```
0 -1 3 -1 9 -1 9
-1 0 7 -1 2 -1 5
3 7 0 -1 9 6 12
-1 -1 -1 0 -1 -1 -1
9 2 9 -1 0 7 7
-1 -1 6 -1 7 0 5
9 5 12 -1 7 5 0
stderr
```

part02test03.solution

part02test03.err

## 4.2.4 Test 04

diff

part02test04.diff

## Input

```
0 9 -1 1
9 0 -1 8
-1 -1 0 -1
1 8 -1 0
```

part02test04.matrix.input

part02test04.i.input

4

## Submission Output

```
0 9 -1 1
9 0 -1 8
-1 -1 0 -1
1 8 -1 0
```

part02test04.output

## Solution Output

```
0 9 -1 1
9 0 -1 8
-1 -1 0 -1
1 8 -1 0
stderr
```

part02test04.solution

part02test04.err

## 4.2.5 Test 05

diff

part02test05.diff

## Input

part02test05.matrix.input

```
0 7 -1 5 2 7 4 6 6 1
7 0 5 5 17 5 16 4 6 7
-1 5 0 1 1 2 2 4 1 5
5 5 1 0 14 10 6 9 6 5
2 17 1 14 0 13 9 3 7 3
7 5 2 10 13 0 8 8 13 1
4 16 2 6 9 8 0 6 11 5
6 4 4 9 3 8 6 0 3 -1
6 6 1 6 7 13 11 3 0 7
1 7 5 5 3 1 5 -1 7 0
```

part02test05.i.input

9

### Submission Output

part02test05.output

```
0 7 3 4 2 5 4 5 4 1
7 0 5 5 6 5 7 4 6 6
3 5 0 1 1 2 2 4 1 3
4 5 1 0 2 3 3 5 2 4
2 6 1 2 0 3 3 3 2 3
5 5 2 3 3 0 4 6 3 1
4 7 2 3 3 4 0 6 3 5
5 4 4 5 3 6 6 0 3 6
4 6 1 2 2 3 3 3 0 4
1 6 3 4 3 1 5 6 4 0
```

### Solution Output

part02test05.solution

```
0 7 3 4 2 5 4 5 4 1
7 0 5 5 6 5 7 4 6 6
3 5 0 1 1 2 2 4 1 3
4 5 1 0 2 3 3 5 2 4
2 6 1 2 0 3 3 3 2 3
5 5 2 3 3 0 4 6 3 1
4 7 2 3 3 4 0 6 3 5
5 4 4 5 3 6 6 0 3 6
4 6 1 2 2 3 3 3 0 4
1 6 3 4 3 1 5 6 4 0
```

stderr

part02test05.err

## 4.2.6 Test 06

diff

part02test06.diff

### Input

part02test06.matrix.input

```
0 9 2 4 2 4
9 0 14 4 7 9
2 14 0 3 5 8
4 4 3 0 -1 -1
2 7 5 -1 0 -1
4 9 8 -1 -1 0
```

part02test06.i.input

2

### Submission Output

```
0 9 2 4 2 4
9 0 11 4 7 9
2 11 0 3 4 6
4 4 3 0 6 8
2 7 4 6 0 6
4 9 6 8 6 0
```

### Solution Output

```
0 9 2 4 2 4
9 0 11 4 7 9
2 11 0 3 4 6
4 4 3 0 6 8
2 7 4 6 0 6
4 9 6 8 6 0
```

stderr

part02test06.output

part02test06.solution

part02test06.err

### 4.2.7 Test 07

diff

part02test07.diff

### Input

```
0 -1 -1 7 7
-1 0 -1 9 3
-1 -1 0 1 4
7 9 1 0 -1
7 3 4 -1 0
```

part02test07.matrix.input

part02test07.i.input

2

### Submission Output

```
0 -1 -1 7 7
-1 0 -1 9 3
-1 -1 0 1 4
7 9 1 0 12
7 3 4 12 0
```

### Solution Output

```
0 -1 -1 7 7
-1 0 -1 9 3
-1 -1 0 1 4
7 9 1 0 12
7 3 4 12 0
```

part02test07.output

part02test07.solution

stderr

part02test07.err

#### 4.2.8 Test 08

diff

part02test08.diff

##### Input

part02test08.matrix.input

```
0 4 2 2 2 6
4 0 9 9 1 2
2 9 0 7 6 3
2 9 7 0 -1 -1
2 1 6 -1 0 -1
6 2 3 -1 -1 0
```

part02test08.i.input

5

##### Submission Output

part02test08.output

```
0 3 2 2 2 5
3 0 5 5 1 2
2 5 0 4 4 3
2 5 4 0 4 7
2 1 4 4 0 3
5 2 3 7 3 0
```

##### Solution Output

part02test08.solution

```
0 3 2 2 2 5
3 0 5 5 1 2
2 5 0 4 4 3
2 5 4 0 4 7
2 1 4 4 0 3
5 2 3 7 3 0
```

stderr

part02test08.err

#### 4.2.9 Test 09

diff

part02test09.diff

##### Input

part02test09.matrix.input

```
0 7 7 5 1
7 0 3 -1 -1
7 3 0 7 3
5 -1 7 0 -1
1 -1 3 -1 0
```

part02test09.i.input

2

### Submission Output

part02test09.output

```
0 7 7 5 1
7 0 3 12 8
7 3 0 7 3
5 12 7 0 6
1 8 3 6 0
```

### Solution Output

part02test09.solution

```
0 7 7 5 1
7 0 3 12 8
7 3 0 7 3
5 12 7 0 6
1 8 3 6 0
```

stderr

part02test09.err

## 4.2.10 Test 10

diff

part02test10.diff

### Input

part02test10.matrix.input

```
0 -1 7 4 1 8 -1 6
-1 0 -1 -1 -1 -1 -1 -1
7 -1 0 3 11 6 -1 3
4 -1 3 0 14 12 -1 8
1 -1 11 14 0 13 -1 1
8 -1 6 12 13 0 -1 8
-1 -1 -1 -1 -1 -1 0 -1
6 -1 3 8 1 8 -1 0
```

part02test10.i.input

6

### Submission Output

part02test10.output

```
0 -1 7 4 1 8 -1 2
-1 0 -1 -1 -1 -1 -1 -1
7 -1 0 3 8 6 -1 3
4 -1 3 0 5 9 -1 6
1 -1 8 5 0 9 -1 1
8 -1 6 9 9 0 -1 8
-1 -1 -1 -1 -1 -1 0 -1
2 -1 3 6 1 8 -1 0
```

### Solution Output

```

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

```

stderr

part02test10.solution

part02test10.err

#### 4.2.11 Test 11

diff

part02test11.diff

##### Input

```

0 -1 -1 5 3
-1 0 -1 3 3
-1 -1 0 -1 -1
5 3 -1 0 -1
3 3 -1 -1 0

```

part02test11.matrix.input

2

part02test11.i.input

##### Submission Output

```

0 -1 -1 5 3
-1 0 -1 3 3
-1 -1 0 -1 -1
5 3 -1 0 6
3 3 -1 6 0

```

part02test11.output

##### Solution Output

```

0 -1 -1 5 3
-1 0 -1 3 3
-1 -1 0 -1 -1
5 3 -1 0 6
3 3 -1 6 0

```

part02test11.solution

stderr

part02test11.err

#### 4.2.12 Test 12

diff

part02test12.diff

## Input

```
0 -1 -1 -1 7 -1 4 4
-1 0 -1 -1 3 -1 3 6
-1 -1 0 -1 4 -1 3 8
-1 -1 -1 0 -1 -1 -1 -1
7 3 4 -1 0 -1 -1 -1
-1 -1 -1 -1 -1 0 -1 -1
4 3 3 -1 -1 -1 0 -1
4 6 8 -1 -1 -1 -1 0
```

part02test12.matrix.input

part02test12.i.input

7

## Submission Output

```
0 7 7 -1 7 -1 4 4
7 0 6 -1 3 -1 3 6
7 6 0 -1 4 -1 3 8
-1 -1 -1 0 -1 -1 -1 -1
7 3 4 -1 0 -1 6 9
-1 -1 -1 -1 -1 0 -1 -1
4 3 3 -1 6 -1 0 8
4 6 8 -1 9 -1 8 0
```

part02test12.output

## Solution Output

```
0 7 7 -1 7 -1 4 4
7 0 6 -1 3 -1 3 6
7 6 0 -1 4 -1 3 8
-1 -1 -1 0 -1 -1 -1 -1
7 3 4 -1 0 -1 6 9
-1 -1 -1 -1 -1 0 -1 -1
4 3 3 -1 6 -1 0 8
4 6 8 -1 9 -1 8 0
```

part02test12.solution

stderr

part02test12.err

## 4.2.13 Test 13

diff

part02test13.diff

## Input

```
0 -1 -1 8 1
-1 0 -1 6 4
-1 -1 0 6 3
8 6 6 0 -1
1 4 3 -1 0
```

part02test13.matrix.input

part02test13.i.input

3



## Submission Output

```
0 -1 -1 8 1
-1 0 -1 6 4
-1 -1 0 6 3
8 6 6 0 9
1 4 3 9 0
```

part02test13.output

## Solution Output

```
0 -1 -1 8 1
-1 0 -1 6 4
-1 -1 0 6 3
8 6 6 0 9
1 4 3 9 0
```

part02test13.solution

stderr

part02test13.err

## 4.2.14 Test 14

diff

part02test14.diff

## Input

```
0 -1 1 1
-1 0 8 7
1 8 0 -1
1 7 -1 0
```

part02test14.matrix.input

part02test14.i.input

2

## Submission Output

```
0 -1 1 1
-1 0 8 7
1 8 0 2
1 7 2 0
```

part02test14.output

## Solution Output

```
0 -1 1 1
-1 0 8 7
1 8 0 2
1 7 2 0
```

part02test14.solution

stderr

part02test14.err

#### 4.2.15 Test 15

diff

part02test15.diff

**Input**

part02test15.matrix.input

```
0 -1 -1 9 6 8 3 2 5 2
-1 0 -1 3 7 3 8 8 7 4
-1 -1 0 4 8 3 8 1 2 1
9 3 4 0 5 16 8 16 5 2
6 7 8 5 0 5 2 7 -1 -1
8 3 3 16 5 0 4 6 5 3
3 8 8 8 2 4 0 6 4 1
2 8 1 16 7 6 6 0 8 2
5 7 2 5 -1 5 4 8 0 -1
2 4 1 2 -1 3 1 2 -1 0
```

part02test15.i.input

5

**Submission Output**

part02test15.output

```
0 12 13 9 6 8 3 2 5 2
12 0 7 3 7 3 8 8 7 4
13 7 0 4 8 3 8 1 2 1
9 3 4 0 5 6 7 5 5 2
6 7 8 5 0 5 2 7 10 7
8 3 3 6 5 0 4 4 5 3
3 8 8 7 2 4 0 5 4 1
2 8 1 5 7 4 5 0 3 2
5 7 2 5 10 5 4 3 0 3
2 4 1 2 7 3 1 2 3 0
```

**Solution Output**

part02test15.solution

```
0 12 13 9 6 8 3 2 5 2
12 0 7 3 7 3 8 8 7 4
13 7 0 4 8 3 8 1 2 1
9 3 4 0 5 6 7 5 5 2
6 7 8 5 0 5 2 7 10 7
8 3 3 6 5 0 4 4 5 3
3 8 8 7 2 4 0 5 4 1
2 8 1 5 7 4 5 0 3 2
5 7 2 5 10 5 4 3 0 3
2 4 1 2 7 3 1 2 3 0
```

stderr

part02test15.err

#### 4.2.16 Test 16

diff

part02test16.diff

## Input

```
0 -1 2 -1 4
-1 0 3 -1 4
2 3 0 -1 -1
-1 -1 -1 0 -1
4 4 -1 -1 0
```

part02test16.matrix.input

part02test16.i.input

5

## Submission Output

```
0 5 2 -1 4
5 0 3 -1 4
2 3 0 -1 6
-1 -1 -1 0 -1
4 4 6 -1 0
```

part02test16.output

## Solution Output

```
0 5 2 -1 4
5 0 3 -1 4
2 3 0 -1 6
-1 -1 -1 0 -1
4 4 6 -1 0
```

part02test16.solution

stderr

part02test16.err

## 4.2.17 Test 17

diff

part02test17.diff

## Input

```
0 -1 -1 9 9 4 8
-1 0 -1 5 7 3 8
-1 -1 0 -1 -1 -1 -1
9 5 -1 0 3 -1 -1
9 7 -1 3 0 6 9
4 3 -1 -1 6 0 -1
8 8 -1 -1 9 -1 0
```

part02test17.matrix.input

part02test17.i.input

4

## Submission Output

part02test17.output

```
0 14 -1 9 9 4 8
14 0 -1 5 7 3 8
-1 -1 0 -1 -1 -1 -1
9 5 -1 0 3 8 13
9 7 -1 3 0 6 9
4 3 -1 8 6 0 11
8 8 -1 13 9 11 0
```

#### Solution Output

part02test17.solution

```
0 14 -1 9 9 4 8
14 0 -1 5 7 3 8
-1 -1 0 -1 -1 -1 -1
9 5 -1 0 3 8 13
9 7 -1 3 0 6 9
4 3 -1 8 6 0 11
8 8 -1 13 9 11 0
```

stderr

part02test17.err

### 4.2.18 Test 18

diff

part02test18.diff

#### Input

part02test18.matrix.input

```
0 -1 5 8 -1 1 7 1 1
-1 0 2 7 -1 2 7 2 1
5 2 0 7 -1 4 5 2 9
8 7 7 0 -1 -1 -1 -1 -1
-1 -1 -1 -1 0 -1 -1 -1 -1
1 2 4 -1 -1 0 -1 -1 -1
7 7 5 -1 -1 -1 0 -1 -1
1 2 2 -1 -1 -1 -1 0 -1
1 1 9 -1 -1 -1 -1 -1 0
```

part02test18.i.input

5

#### Submission Output

part02test18.output

```
0 7 5 8 -1 1 7 1 1
7 0 2 7 -1 2 7 2 1
5 2 0 7 -1 4 5 2 3
8 7 7 0 -1 9 12 9 8
-1 -1 -1 -1 0 -1 -1 -1 -1
1 2 4 9 -1 0 8 2 2
7 7 5 12 -1 8 0 7 8
1 2 2 9 -1 2 7 0 2
1 1 3 8 -1 2 8 2 0
```

#### Solution Output

```

0 7 5 8 -1 1 7 1 1
7 0 2 7 -1 2 7 2 1
5 2 0 7 -1 4 5 2 3
8 7 7 0 -1 9 12 9 8
-1 -1 -1 -1 0 -1 -1 -1 -1
1 2 4 9 -1 0 8 2 2
7 7 5 12 -1 8 0 7 8
1 2 2 9 -1 2 7 0 2
1 1 3 8 -1 2 8 2 0

```

stderr

part02test18.solution

part02test18.err

#### 4.2.19 Test 19

diff

part02test19.diff

**Input**

```

0 -1 -1
-1 0 3
-1 3 0

```

part02test19.matrix.input

part02test19.i.input

1

**Submission Output**

```

0 -1 -1
-1 0 3
-1 3 0

```

part02test19.output

**Solution Output**

```

0 -1 -1
-1 0 3
-1 3 0

```

part02test19.solution

stderr

part02test19.err

#### 4.2.20 Test 20

diff

part02test20.diff

**Input**

part02test20.matrix.input

```
0 3 -1 9 4
3 0 -1 5 5
-1 -1 0 -1 -1
9 5 -1 0 -1
4 5 -1 -1 0
```

part02test20.i.input

3

**Submission Output**

part02test20.output

```
0 3 -1 8 4
3 0 -1 5 5
-1 -1 0 -1 -1
8 5 -1 0 10
4 5 -1 10 0
```

**Solution Output**

part02test20.solution

```
0 3 -1 8 4
3 0 -1 5 5
-1 -1 0 -1 -1
8 5 -1 0 10
4 5 -1 10 0
```

stderr

part02test20.err

## 4.2.21 Test 21

diff

part02test21.diff

**Input**

part02test21.matrix.input

```
0 -1 4 1
-1 0 4 3
4 4 0 -1
1 3 -1 0
```

part02test21.i.input

2

**Submission Output**

part02test21.output

```
0 -1 4 1
-1 0 4 3
4 4 0 5
1 3 5 0
```

**Solution Output**

part02test21.solution

```
0 -1 4 1
-1 0 4 3
4 4 0 5
1 3 5 0
```

stderr

part02test21.err

#### 4.2.22 Test 22

diff

part02test22.diff

**Input**

part02test22.matrix.input

```
0 -1 3 6
-1 0 7 2
3 7 0 -1
6 2 -1 0
```

part02test22.i.input

4

**Submission Output**

part02test22.output

```
0 8 3 6
8 0 7 2
3 7 0 9
6 2 9 0
```

**Solution Output**

part02test22.solution

```
0 8 3 6
8 0 7 2
3 7 0 9
6 2 9 0
```

stderr

part02test22.err

#### 4.2.23 Test 23

diff

part02test23.diff

**Input**

part02test23.matrix.input

```
0 8 9 1 5
8 0 4 1 3
9 4 0 -1 -1
1 1 -1 0 -1
5 3 -1 -1 0
```

part02test23.i.input

4

### Submission Output

part02test23.output

```
0 2 6 1 5
2 0 4 1 3
6 4 0 5 7
1 1 5 0 4
5 3 7 4 0
```

### Solution Output

part02test23.solution

```
0 2 6 1 5
2 0 4 1 3
6 4 0 5 7
1 1 5 0 4
5 3 7 4 0
```

stderr

part02test23.err

## 4.2.24 Test 24

diff

part02test24.diff

### Input

part02test24.matrix.input

```
0 -1 -1 -1 5 5 3
-1 0 -1 -1 -1 -1 -1
-1 -1 0 -1 -1 -1 -1
-1 -1 -1 0 2 3 7
5 -1 -1 2 0 5 -1
5 -1 -1 3 5 0 3
3 -1 -1 7 -1 3 0
```

part02test24.i.input

4

### Submission Output

part02test24.output

```
0 -1 -1 -1 5 5 3
-1 0 -1 -1 -1 -1 -1
-1 -1 0 -1 -1 -1 -1
-1 -1 -1 0 2 3 7
5 -1 -1 2 0 5 8
5 -1 -1 3 5 0 3
3 -1 -1 7 8 3 0
```

### Solution Output



part02test24.solution

```
0 -1 -1 -1 5 5 3
-1 0 -1 -1 -1 -1 -1
-1 -1 0 -1 -1 -1 -1
-1 -1 -1 0 2 3 7
5 -1 -1 2 0 5 8
5 -1 -1 3 5 0 3
3 -1 -1 7 8 3 0
```

stderr

part02test24.err

#### 4.2.25 Test 25

diff

part02test25.diff

##### Input

part02test25.matrix.input

```
0 -1 -1 -1
-1 0 -1 -1
-1 -1 0 7
-1 -1 7 0
```

part02test25.i.input

1

##### Submission Output

part02test25.output

```
0 -1 -1 -1
-1 0 -1 -1
-1 -1 0 7
-1 -1 7 0
```

##### Solution Output

part02test25.solution

```
0 -1 -1 -1
-1 0 -1 -1
-1 -1 0 7
-1 -1 7 0
```

stderr

part02test25.err

#### 4.2.26 Source Code

csce310h0mework03part02.h

```
1 #ifndef CSCE310HOMEWORK03PART02_H
2 #define CSCE310HOMEWORK03PART02_H
3 #include <vector>
4 using namespace std;
5
6 vector< vector<double> > allPairsSP( vector< vector<double> > , int );
7
8 #endif
```

```

1  /**
2   * Author: Fateh Karan Singh Sandhu
3   *
4   * This program uses Floyd's algorithm to produce a all pairs shortest path tree
5   * from a given adjacency matrix, output is computed upto the i'th stage.
6   */
7
8  #include <vector>
9  #include "csce310h0mework03part02.h"
10 #include <cmath>
11 #include <iostream>
12
13 using namespace std;
14
15 vector< vector<double> > allPairsSP( vector< vector<double> > adjacencyMatrix , int i
    ){
16
17     for (int k = 0 ; k < i ; k++) {
18         for (int j = 0 ; j < adjacencyMatrix.size() ; j++) {
19             for (int l = 0 ; l < adjacencyMatrix.size() ; l++) {
20                 if (adjacencyMatrix[j][l] == -1) {
21                     adjacencyMatrix[j][l] = 100000; //set -1 to a large value
22                 }
23                 adjacencyMatrix[j][l] = min(adjacencyMatrix[j][l], adjacencyMatrix[j][k]+
                    adjacencyMatrix[k][l]);
24             }
25         }
26     }
27
28     for (int a = 0 ; a < adjacencyMatrix.size() ; a++) {
29         for (int b = 0 ; b < adjacencyMatrix.size() ; b++) {
30             if (adjacencyMatrix[a][b] == 100000) {
31                 adjacencyMatrix[a][b] = -1; //change large value back to -1
32             }
33         }
34     }
35
36     return adjacencyMatrix;
37 }

```

## 4.3 csce310h0mework03part03

### 4.3.1 Test 01

diff

part03test01.diff

Input

part03test01.coeffs.input

```

-3
-2
-3
3
2
-3

```

part03test01.x.input

0

### Submission Output

part03test01.output

The value of  $-3x^5-2x^4-3x^3+3x^2+2x^1-3$  at 0 is -3

### Solution Output

part03test01.solution

The value of  $-3x^5-2x^4-3x^3+3x^2+2x^1-3$  at 0 is -3

stderr

part03test01.err

## 4.3.2 Test 02

diff

part03test02.diff

### Input

part03test02.coefs.input

-2  
2  
-3  
2

part03test02.x.input

-2

### Submission Output

part03test02.output

The value of  $-2x^3+2x^2-3x^1+2$  at -2 is 32

### Solution Output

part03test02.solution

The value of  $-2x^3+2x^2-3x^1+2$  at -2 is 32

stderr

part03test02.err

## 4.3.3 Test 03

diff

part03test03.diff

### Input

part03test03.coefs.input

2  
-1  
-1

part03test03.x.input

-2

#### Submission Output

part03test03.output

The value of  $2x^2-1x^1-1$  at -2 is 9

#### Solution Output

part03test03.solution

The value of  $2x^2-1x^1-1$  at -2 is 9

stderr

part03test03.err

### 4.3.4 Test 04

diff

part03test04.diff

#### Input

part03test04.coeffs.input

-1

-3

-1

2

3

part03test04.x.input

2

#### Submission Output

part03test04.output

The value of  $-1x^4-3x^3-1x^2+2x^1+3$  at 2 is -37

#### Solution Output

part03test04.solution

The value of  $-1x^4-3x^3-1x^2+2x^1+3$  at 2 is -37

stderr

part03test04.err

### 4.3.5 Test 05

diff

part03test05.diff

#### Input

part03test05.coeffs.input

-2

2

-1

3

part03test05.x.input

3

#### Submission Output

part03test05.output

The value of  $-2x^3+2x^2-1x^1+3$  at 3 is -36

#### Solution Output

part03test05.solution

The value of  $-2x^3+2x^2-1x^1+3$  at 3 is -36

stderr

part03test05.err

### 4.3.6 Test 06

diff

part03test06.diff

#### Input

part03test06.coeffs.input

-3  
-1  
3  
-2  
2  
3

part03test06.x.input

-1

#### Submission Output

part03test06.output

The value of  $-3x^5-1x^4+3x^3-2x^2+2x^1+3$  at -1 is -2

#### Solution Output

part03test06.solution

The value of  $-3x^5-1x^4+3x^3-2x^2+2x^1+3$  at -1 is -2

stderr

part03test06.err

### 4.3.7 Test 07

diff

part03test07.diff

#### Input

part03test07.coeffs.input

3  
3

part03test07.x.input

2

#### Submission Output

part03test07.output

The value of  $3x^1+3$  at 2 is 9

#### Solution Output

part03test07.solution

The value of  $3x^1+3$  at 2 is 9

stderr

part03test07.err

### 4.3.8 Test 08

diff

part03test08.diff

#### Input

part03test08.coeffs.input

1  
2  
-2  
3

part03test08.x.input

-1

#### Submission Output

part03test08.output

The value of  $1x^3+2x^2-2x^1+3$  at -1 is 6

#### Solution Output

part03test08.solution

The value of  $1x^3+2x^2-2x^1+3$  at -1 is 6

stderr

part03test08.err

### 4.3.9 Test 09

diff

part03test09.diff

#### Input

part03test09.coeffs.input

1  
3  
-3  
-1  
-1  
-3

part03test09.x.input

-2

#### Submission Output

part03test09.output

The value of  $1x^5+3x^4-3x^3-1x^2-1x^1-3$  at -2 is 35

#### Solution Output

part03test09.solution

The value of  $1x^5+3x^4-3x^3-1x^2-1x^1-3$  at -2 is 35

stderr

part03test09.err

### 4.3.10 Test 10

diff

part03test10.diff

#### Input

part03test10.coeffs.input

1  
3  
1

part03test10.x.input

0

#### Submission Output

part03test10.output

The value of  $1x^2+3x^1+1$  at 0 is 1

#### Solution Output

part03test10.solution

The value of  $1x^2+3x^1+1$  at 0 is 1

stderr

part03test10.err

### 4.3.11 Source Code

csce310h0mework03part03.h

```
1 #ifndef CSCE310HOMEWORK03PART03_H
2 #define CSCE310HOMEWORK03PART03_H
3 #include <vector>
4 using namespace std;
5
6 int hornersRule( vector<int> , int );
7
8 #endif
```

```
1  /**
2   * Author: Fateh Karan Singh Sandhu
3   *
4   * This program computes a polynomial at a given value using Horners's rule
5   */
6
7  #include <vector>
8  #include "csce310h0mework03part03.h"
9  #include <cmath>
10 #include <iostream>
11 #include <vector>
12
13 using namespace std;
14
15 int hornersRule( vector<int> coeffs , int value ){
16
17     double result = coeffs[0]; //set the first coeff
18     for (int i = 1 ; i < coeffs.size() ; i++) {
19         result = (result * value) + coeffs[i];
20     }
21     return result;
22 }
```