



UIL Computer Science Competition

District 2021

JUDGES PACKET - CONFIDENTIAL

I. Instructions

1. The attached printouts of the judge test data are provided for the reference of the contest director and programming judges. Additional copies may be made if needed for this purpose.
2. This packet must remain CONFIDENTIAL. Additional copies may be made and returned to schools when other confidential contest material is returned.

II. Table of Contents

Number	Name
Problem 1	Aimi
Problem 2	Arti
Problem 3	Carol
Problem 4	Eui
Problem 5	Harish
Problem 6	Isha
Problem 7	Joyce
Problem 8	Kostya
Problem 9	Melissa
Problem 10	Pablo
Problem 11	Reka
Problem 12	Timothy

Problem #1
60 Points

1. Aimi

Program Name: Aimi.java

Input File: None

Test Output To Screen:

```
A file that big?  
It might be very useful.  
But now it is gone.  
-----  
The web site you seek  
cannot be located but  
countless more exist.  
-----  
Chaos reigns within.  
Reflect, repent, and reboot.  
Order shall return.  
-----
```

Problem #2
60 Points

2. Arti

Program Name: Arti.java

Input File: arti.dat

Test Input File:

```
10
7
8
9
51
17
153
89
15
144
3205
```

Test Output To Screen:

```
7 2.646
8 2.828
9 3.000
51 7.141
17 4.123
153 12.369
89 9.434
15 3.873
144 12.000
3205 56.613
```

Problem #3
60 Points

3. Carol

Program Name: Carol.java

Input File: carol.dat

Test Input File:

```
20
WasItACatISaw
TopSpot
1212
12332
NoLemonNo
MyGym
Tenets
1
12
123
1234
4
43
432
4321
1111111111111111112
12345678901234567890
1234512345
1212123
123123321
```

Test Output To Screen:

```
0 character(s) need to be added to convert WasItACatISaw into a palindrome.
0 character(s) need to be added to convert TopSpot into a palindrome.
1 character(s) need to be added to convert 1212 into a palindrome.
1 character(s) need to be added to convert 12332 into a palindrome.
5 character(s) need to be added to convert NoLemonNo into a palindrome.
0 character(s) need to be added to convert MyGym into a palindrome.
1 character(s) need to be added to convert Tenets into a palindrome.
0 character(s) need to be added to convert 1 into a palindrome.
1 character(s) need to be added to convert 12 into a palindrome.
2 character(s) need to be added to convert 123 into a palindrome.
3 character(s) need to be added to convert 1234 into a palindrome.
0 character(s) need to be added to convert 4 into a palindrome.
1 character(s) need to be added to convert 43 into a palindrome.
2 character(s) need to be added to convert 432 into a palindrome.
3 character(s) need to be added to convert 4321 into a palindrome.
1 character(s) need to be added to convert 1111111111111111112 into a palindrome.
19 character(s) need to be added to convert 12345678901234567890 into a palindrome.
9 character(s) need to be added to convert 1234512345 into a palindrome.
2 character(s) need to be added to convert 1212123 into a palindrome.
3 character(s) need to be added to convert 123123321 into a palindrome.
```

Problem #4

60 Points

4. Eui

Program Name: Eui.java

Input File: eui.dat

Test Input File:

16	June 14 - 15	June 1 - 28	June 1 - 28	June 1 - 28
3	April 1 - 6	June 1 - 28	June 1 - 28	June 1 - 28
January 1 - 3	March 6 - 10	June 1 - 28	June 1 - 28	June 1 - 28
January 1 - 3	May 2 - 3	June 1 - 28	June 1 - 28	June 1 - 28
January 1 - 3	February 8 - 25	June 1 - 28	June 1 - 28	June 1 - 28
5	July 9 - 25	June 1 - 28	June 1 - 28	June 1 - 28
February 5 - 10	November 1 - 28	June 1 - 28	June 1 - 28	June 1 - 28
February 3 - 4	16	June 1 - 28	June 1 - 28	June 1 - 28
March 9 - 12	January 2 - 23	June 1 - 28	June 1 - 28	June 1 - 28
March 12 - 15	January 5 - 9	June 1 - 28	June 1 - 28	June 1 - 28
March 9 - 15	March 5 - 16	June 1 - 28	June 1 - 28	June 1 - 28
1	April 3 - 7	June 1 - 28	June 1 - 28	June 1 - 28
December 12 - 12	January 19 - 26	June 1 - 28	June 1 - 28	June 1 - 28
2	May 11 - 21	June 1 - 28	June 1 - 28	June 1 - 28
January 11 - 21	June 1 - 23	June 1 - 28	June 1 - 28	June 1 - 28
February 5 - 24	April 13 - 23	June 1 - 28	June 1 - 28	June 1 - 28
4	January 4 - 8	June 1 - 28	June 1 - 28	June 1 - 28
January 4 - 22	May 1 - 26	June 1 - 28	June 1 - 28	June 1 - 28
February 10 - 19	February 7 - 13	June 1 - 28	June 1 - 28	June 1 - 28
February 5 - 22	April 15 - 19	June 1 - 28	June 1 - 28	June 1 - 28
March 7 - 23	February 15 - 25	June 1 - 28	June 1 - 28	June 1 - 28
6	September 2 - 11	June 1 - 28	June 1 - 28	June 1 - 28
January 16 - 18	April 4 - 17	June 1 - 28	June 1 - 28	June 1 - 28
January 5 - 8	May 10 - 28	June 1 - 28	June 1 - 28	June 1 - 28
March 15 - 26	18	June 1 - 28	June 1 - 28	June 1 - 28
April 9 - 18	January 20 - 24	June 1 - 28	June 1 - 28	June 1 - 28
March 6 - 9	January 6 - 19	June 1 - 28	June 1 - 28	June 1 - 28
March 2 - 10	March 12 - 28	June 1 - 28	June 1 - 28	June 1 - 28
8	March 14 - 14	June 1 - 28	June 1 - 28	June 1 - 28
January 10 - 15	May 3 - 9	June 1 - 28	June 1 - 28	June 1 - 28
February 8 - 20	January 17 - 18	June 1 - 28	June 1 - 28	June 1 - 28
January 21 - 27	April 5 - 10	June 1 - 28	June 1 - 28	June 1 - 28
March 11 - 25	June 6 - 15	June 1 - 28	June 1 - 28	June 1 - 28
April 6 - 19	March 9 - 18	June 1 - 28	June 1 - 28	June 1 - 28
January 4 - 21	September 3 - 27	June 1 - 28	June 1 - 28	June 1 - 28
June 4 - 27	May 13 - 15	June 1 - 28	June 1 - 28	June 1 - 28
June 5 - 23	April 7 - 7	June 1 - 28	June 1 - 28	June 1 - 28
10	February 19 - 25	June 1 - 28	June 1 - 28	June 1 - 28
January 6 - 28	February 4 - 6	June 1 - 28	June 1 - 28	June 1 - 28
February 2 - 17	December 5 - 12	June 1 - 28	June 1 - 28	June 1 - 28
January 9 - 11	September 5 - 26	June 1 - 28	June 1 - 28	June 1 - 28
January 3 - 11	March 16 - 21	June 1 - 28	June 1 - 28	June 1 - 28
March 15 - 25	April 16 - 22	June 1 - 28	June 1 - 28	June 1 - 28
June 6 - 15	20	June 1 - 28	June 1 - 28	June 1 - 28
July 14 - 19	January 8 - 16	June 1 - 28	June 1 - 28	June 1 - 28
February 8 - 19	January 6 - 27	June 1 - 28	June 1 - 28	June 1 - 28
March 6 - 26	January 3 - 4	June 1 - 28	June 1 - 28	June 1 - 28
August 2 - 14	January 10 - 11	June 1 - 28	June 1 - 28	June 1 - 28
12	April 1 - 24	June 1 - 28	June 1 - 28	June 1 - 28
January 22 - 26	January 3 - 21	June 1 - 28	June 1 - 28	June 1 - 28
January 21 - 25	July 6 - 25	June 1 - 28	June 1 - 28	June 1 - 28
January 10 - 17	February 7 - 18	June 1 - 28	June 1 - 28	June 1 - 28
February 3 - 9	July 16 - 28	June 1 - 28	June 1 - 28	June 1 - 28
March 8 - 23	June 1 - 28	June 1 - 28	June 1 - 28	June 1 - 28
February 6 - 24	September 8 - 20	June 1 - 28	June 1 - 28	June 1 - 28
January 9 - 21	August 1 - 20	June 1 - 28	June 1 - 28	June 1 - 28
March 8 - 22	June 1 - 17	June 1 - 28	June 1 - 28	June 1 - 28
June 12 - 27	August 15 - 20	June 1 - 28	June 1 - 28	June 1 - 28
July 6 - 21	December 5 - 28	June 1 - 28	June 1 - 28	June 1 - 28
September 7 - 16	June 17 - 27	June 1 - 28	June 1 - 28	June 1 - 28
June 7 - 23	September 7 - 26	June 1 - 28	June 1 - 28	June 1 - 28
14	February 2 - 9	June 1 - 28	June 1 - 28	June 1 - 28
January 20 - 24	January 13 - 24	June 1 - 28	June 1 - 28	June 1 - 28
February 10 - 20	October 16 - 23	June 1 - 28	June 1 - 28	June 1 - 28
January 10 - 20	200	June 1 - 28	June 1 - 28	June 1 - 28
April 21 - 24	June 1 - 28	June 1 - 28	June 1 - 28	June 1 - 28
February 13 - 24	June 1 - 28	June 1 - 28	June 1 - 28	June 1 - 28
April 12 - 22	June 1 - 28	June 1 - 28	June 1 - 28	June 1 - 28
February 23 - 26	June 1 - 28	June 1 - 28	June 1 - 28	June 1 - 28
				200
				October 2 - 10
				August 4 - 16
				August 1 - 6
				May 10 - 20
				July 2 - 25
				May 26 - 28
				December 14 - 16
				February 18 - 24
				April 1 - 20
				November 3 - 8

UIL – Computer Science Judge’s Packet – District - 2021

September 10 - 14	September 1 - 1	November 13 - 14	February 1 - 19	May 1 - 14
August 14 - 18	July 2 - 2	October 3 - 14	February 1 - 20	May 1 - 15
January 18 - 18	December 25 - 25	November 8 - 14	February 1 - 21	May 1 - 16
June 6 - 11	March 16 - 25	November 16 - 19	February 1 - 22	May 1 - 17
November 10 - 14	February 1 - 15	July 18 - 27	February 1 - 23	May 1 - 18
June 9 - 27	November 8 - 22	January 1 - 21	February 1 - 24	May 1 - 19
April 23 - 26	August 5 - 25	June 15 - 16	February 1 - 25	May 1 - 20
July 5 - 17	June 16 - 22	April 2 - 8	February 1 - 26	May 1 - 21
June 4 - 27	March 3 - 20	June 2 - 6	February 1 - 27	May 1 - 22
July 9 - 11	April 8 - 17	April 8 - 22	February 1 - 28	May 1 - 23
June 7 - 23	April 7 - 27	June 7 - 27	March 1 - 1	May 1 - 24
November 2 - 16	May 15 - 19	February 20 - 22	March 1 - 2	May 1 - 25
April 15 - 17	March 2 - 23	February 6 - 26	March 1 - 3	May 1 - 26
June 6 - 20	July 13 - 25	May 4 - 13	March 1 - 4	May 1 - 27
July 1 - 2	July 5 - 13	July 8 - 11	March 1 - 5	May 1 - 28
April 17 - 20	September 17 - 17	November 2 - 3	March 1 - 6	June 1 - 1
January 12 - 12	April 3 - 9	July 1 - 25	March 1 - 7	June 1 - 2
September 9 - 20	December 11 - 25	April 4 - 14	March 1 - 8	June 1 - 3
March 10 - 15	April 4 - 25	November 6 - 14	March 1 - 9	June 1 - 4
December 13 - 28	May 11 - 14	September 5 - 15	March 1 - 10	June 1 - 5
May 24 - 28	July 15 - 21	January 11 - 25	March 1 - 11	June 1 - 6
June 11 - 22	February 7 - 11	October 2 - 4	March 1 - 12	June 1 - 7
May 1 - 9	April 8 - 17	August 8 - 11	March 1 - 13	June 1 - 8
March 18 - 23	January 2 - 24	April 12 - 24	March 1 - 14	June 1 - 9
September 1 - 9	January 5 - 24	August 13 - 15	March 1 - 15	June 1 - 10
October 8 - 20	January 12 - 27	December 9 - 19	March 1 - 16	June 1 - 11
November 22 - 28	June 5 - 18	March 22 - 27	March 1 - 17	June 1 - 12
December 8 - 23	January 18 - 24	December 13 - 17	March 1 - 18	June 1 - 13
June 2 - 21	April 13 - 13	January 26 - 28	March 1 - 19	June 1 - 14
June 5 - 9	June 2 - 7	May 5 - 5	March 1 - 20	June 1 - 15
July 8 - 13	September 8 - 20	July 17 - 23	March 1 - 21	June 1 - 16
October 11 - 12	December 7 - 18	November 5 - 8	March 1 - 22	June 1 - 17
April 8 - 25	April 19 - 21	200	March 1 - 23	June 1 - 18
May 7 - 21	December 9 - 24	January 1 - 1	March 1 - 24	June 1 - 19
April 1 - 28	June 8 - 25	January 1 - 2	March 1 - 25	June 1 - 20
November 4 - 20	May 12 - 20	January 1 - 3	March 1 - 26	June 1 - 21
June 12 - 17	November 22 - 24	January 1 - 4	March 1 - 27	June 1 - 22
June 12 - 27	April 5 - 16	January 1 - 5	March 1 - 28	June 1 - 23
February 7 - 8	September 1 - 7	January 1 - 6	April 1 - 1	June 1 - 24
May 25 - 28	March 11 - 21	January 1 - 7	April 1 - 2	June 1 - 25
June 19 - 22	May 20 - 20	January 1 - 8	April 1 - 3	June 1 - 26
January 4 - 19	June 12 - 27	January 1 - 9	April 1 - 4	June 1 - 27
September 17 - 22	April 3 - 9	January 1 - 10	April 1 - 5	June 1 - 28
June 16 - 27	November 16 - 24	January 1 - 11	April 1 - 6	July 1 - 1
August 7 - 9	January 26 - 28	January 1 - 12	April 1 - 7	July 1 - 2
March 2 - 16	August 5 - 18	January 1 - 13	April 1 - 8	July 1 - 3
April 9 - 13	April 8 - 26	January 1 - 14	April 1 - 9	July 1 - 4
April 15 - 20	April 14 - 24	January 1 - 15	April 1 - 10	July 1 - 5
October 18 - 18	July 27 - 28	January 1 - 16	April 1 - 11	July 1 - 6
February 1 - 18	July 13 - 14	January 1 - 17	April 1 - 12	July 1 - 7
April 3 - 7	January 9 - 19	January 1 - 18	April 1 - 13	July 1 - 8
August 8 - 15	December 2 - 5	January 1 - 19	April 1 - 14	July 1 - 9
September 22 - 25	May 3 - 25	January 1 - 20	April 1 - 15	July 1 - 10
July 9 - 20	July 4 - 4	January 1 - 21	April 1 - 16	July 1 - 11
January 12 - 22	November 5 - 11	January 1 - 22	April 1 - 17	July 1 - 12
December 23 - 28	May 9 - 28	January 1 - 23	April 1 - 18	July 1 - 13
November 5 - 5	June 14 - 23	January 1 - 24	April 1 - 19	July 1 - 14
July 3 - 25	September 1 - 20	January 1 - 25	April 1 - 20	July 1 - 15
November 15 - 16	August 22 - 27	January 1 - 26	April 1 - 21	July 1 - 16
November 20 - 28	September 1 - 23	January 1 - 27	April 1 - 22	July 1 - 17
January 8 - 22	January 11 - 19	January 1 - 28	April 1 - 23	July 1 - 18
August 9 - 12	May 10 - 22	February 1 - 1	April 1 - 24	July 1 - 19
July 3 - 21	April 22 - 22	February 1 - 2	April 1 - 25	July 1 - 20
August 16 - 25	May 3 - 20	February 1 - 3	April 1 - 26	July 1 - 21
December 2 - 16	April 8 - 22	February 1 - 4	April 1 - 27	July 1 - 22
May 10 - 15	April 14 - 16	February 1 - 5	April 1 - 28	July 1 - 23
October 13 - 28	September 13 - 23	February 1 - 6	May 1 - 1	July 1 - 24
June 26 - 28	August 14 - 26	February 1 - 7	May 1 - 2	July 1 - 25
August 15 - 23	March 22 - 26	February 1 - 8	May 1 - 3	July 1 - 26
June 14 - 27	March 7 - 27	February 1 - 9	May 1 - 4	July 1 - 27
November 3 - 4	April 19 - 28	February 1 - 10	May 1 - 5	July 1 - 28
February 15 - 20	May 16 - 20	February 1 - 11	May 1 - 6	August 1 - 1
September 17 - 22	September 7 - 10	February 1 - 12	May 1 - 7	August 1 - 2
October 9 - 26	January 11 - 25	February 1 - 13	May 1 - 8	August 1 - 3
February 8 - 11	November 9 - 22	February 1 - 14	May 1 - 9	August 1 - 4
October 7 - 10	May 12 - 13	February 1 - 15	May 1 - 10	
September 5 - 19	April 26 - 28	February 1 - 16	May 1 - 11	
October 14 - 16	November 19 - 19	February 1 - 17	May 1 - 12	
June 25 - 28	February 13 - 16	February 1 - 18	May 1 - 13	

Eui continued

Test Output To Screen:

Case #1: 1 3
Case #2: 3 5
Case #3: 1 1
Case #4: 2 2
Case #5: 4 4
Case #6: 6 6
Case #7: 8 8
Case #8: 7 10
Case #9: 8 12
Case #10: 11 14
Case #11: 13 16
Case #12: 17 18
Case #13: 14 20
Case #14: 1 28
Case #15: 77 188
Case #16: 8 200

Problem #5
60 Points

5. Harish

Program Name: Harish.java

Input File: harish.dat

Test Input File:

18	aaaaa	.a.a.	-----
a...a	aaaaaa..a
a...a	-----	.a.a.	a...a
a...a	a.a.a	a.a.a	.a..a
a...a	.a..	-----a
a....a	.a..a	a...a
-----	.a..aa	-----
a...a	.a..a	.aa.a	a..a.
a...a	-----	a...a	a...a
a...a	aaaaa	.a..a	a..a.
a...a	-----	a....
a...a	a...a	a...a
-----	a....	-----
.....	aaaa.	a..a.	.aaaa
.....	-----	a...a
.....	a...a	a..a.	..a..
.....	.a.a.	-----	a.a.a
.....	..a..	a....	a....
-----	.a.a.	a.a.a	-----
a....	a...a	aa.aa
.a.a.	-----
a...a	a.a.a	aaaaa	..a..
.a.a.	.a.a.	-----	a.a.a
.a..aaa
-----	.a.a.	a.a.a	-----
aaaaa	a.a..	
aaaaa	-----	
aaaaa	a.a.a	aaaaa	

Test Output To Screen:

```

valid
invalid
invalid
invalid
invalid
invalid
valid
invalid
invalid
invalid
invalid
valid
valid
valid
valid
valid
invalid
invalid

```


Problem #6
60 Points

6. Isha

Program Name: Isha.java

Input File: isha.dat

Test Input File:

```
14
Houston 6:00 A.M. Orlando
Orlando 1:25 P.M. Charlotte
Charlotte 7:30 A.M. San Diego
San Diego 12:00 A.M. Denver
Denver 12:00 P.M. San Diego
San Diego 6:35 A.M. Houston
Houston 11:59 P.M. San Diego
San Diego 3:45 P.M. Charlotte
Denver 5:00 A.M. Houston
Houston 5:00 A.M. Denver
Houston 7:45 P.M. Charlotte
Charlotte 11:18 A.M. Houston
Charlotte 10:10 P.M. Orlando
Orlando 12:01 P.M. Houston
```

Test Output To Screen:

```
Houston 6:00 A.M. Orlando 9:01 A.M.
Orlando 1:25 P.M. Charlotte 2:54 P.M.
Charlotte 7:30 A.M. San Diego 8:53 A.M.
San Diego 12:00 A.M. Denver 2:59 A.M.
Denver 12:00 P.M. San Diego 12:59 P.M.
San Diego 6:35 A.M. Houston 11:29 A.M.
Houston 11:59 P.M. San Diego 12:53 P.M.
San Diego 3:45 P.M. Charlotte 11:08 P.M.
Denver 5:00 A.M. Houston 8:05 A.M.
Houston 5:00 A.M. Denver 6:05 A.M.
Houston 7:45 P.M. Charlotte 10:55 P.M.
Charlotte 11:18 A.M. Houston 12:28 P.M.
Charlotte 10:10 P.M. Orlando 11:39 P.M.
Orlando 12:01 P.M. Houston 1:02 P.M.
```

Problem #7
60 Points

7. Joyce

Program Name: Joyce.java

Input File: joyce.dat

Test Input File:

```
11
Two driven jocks help fax my big quiz.
Waltz, nymph, for quick jigs vex Bud.
abcdefghijklmnopqrstuvwxyz
abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
AaBbCcDdEeFfGgHhIiJjKkLlMmNnOoPpQqRrSsTtUuVvWwXxYyZz
zyxwvutsrqponmlkjihgfedcba
UIL CS is the best!
This is a random sentence in the judge's input set.
This sentence will use a bit more of the letters needed for a pangram, but will ultimately end up short.
123
```

Test Output To Screen:

```
pangram
pangram
pangram
missing z
perfect pangram
pangram
perfect pangram
missing adfgjkmnopqrvmxyz
missing bfkqlqvmxyz
missing jkqvzx
missing abcdefghijklmnopqrstuvwxyz
```

Problem #8
60 Points

8. Kostya

Program Name: Kostya.java

Input File: kostya.dat

Test Input File:

50	59 51	26 25
2 1	84 36	47 3
4 4	74 68	69 42
84 44	54 11	45 17
99 72	38 23	17 11
33 9	10 5	13 10
63 8	39 19	4 4
89 49	60 34	99 82
42 38	63 16	49 12
90 33	81 39	59 42
19 14	83 12	78 56
93 26	46 22	78 32
82 46	23 19	11 3
66 32	53 7	26 2
19 8	85 3	58 31
92 74	17 5	88 13
76 10	91 47	86 1

Test Output To Screen:

Case #1: 0.6667	Case #26: 0.8803
Case #2: 0.0000	Case #27: 1.0000
Case #3: 0.6085	Case #28: 0.8798
Case #4: 0.0000	Case #29: 0.0016
Case #5: 1.0000	Case #30: 1.0000
Case #6: 1.0000	Case #31: 1.0000
Case #7: 0.3762	Case #32: 0.9996
Case #8: 0.0000	Case #33: 0.6598
Case #9: 1.0000	Case #34: 0.0000
Case #10: 0.0456	Case #35: 1.0000
Case #11: 1.0000	Case #36: 0.1016
Case #12: 0.3143	Case #37: 0.9983
Case #13: 0.8619	Case #38: 0.2493
Case #14: 0.9597	Case #39: 0.0526
Case #15: 0.0000	Case #40: 0.0000
Case #16: 1.0000	Case #41: 0.0000
Case #17: 0.0000	Case #42: 1.0000
Case #18: 0.9909	Case #43: 0.0018
Case #19: 0.0000	Case #44: 0.0002
Case #20: 1.0000	Case #45: 0.9968
Case #21: 0.2373	Case #46: 0.9992
Case #22: 0.7879	Case #47: 1.0000
Case #23: 0.8456	Case #48: 0.5733
Case #24: 0.3443	Case #49: 1.0000
Case #25: 1.0000	Case #50: 1.0000

Problem #9
60 Points

9. Melissa

Program Name: Melissa.java

Input File: melissa.dat

Test Input File:

```
20
1
2
3
4
6
7
8
10
11
12
12345
65432
9976
6427
100000
99999
7765
4598
93367
496
```

Test Output To Screen:

```
1 -> 1
2 -> 2
3 -> 3
4 -> 4
6 -> 6
7 -> 7
8 -> 8
10 -> 1
11 -> 1
12 -> 2
12345 -> 2
65432 -> 4
9976 -> 8
6427 -> 2
100000 -> 1
99999 -> 2
7765 -> 6
4598 -> 6
93367 -> 8
496 -> 2
```

Problem #10
60 Points

10. Pablo

Program Name: Pablo.java

Input File: pablo.dat

Test Input File:

```
9
31.37646 -100.44892 31.36184 -100.43369
30.28017 -97.73871 30.27521 -97.74011
30.28642 -97.73645 30.28771 -97.73333
30.26557 -97.74465 29.76694 -95.37830
33.57851 -101.88470 33.59019 -101.87068
51.51526 -0.13781 52.36496 4.89876
-33.94493 18.41417 -33.92083 18.41494
-27.45790 153.02036 -33.85837 151.19125
61.18972 -149.81581 61.18969 -149.82203
```

Test Output To Screen:

```
The distance between (31.37646, -100.44892) and (31.36184, -100.43369) is 2176 meters.
The distance between (30.28017, -97.73871) and (30.27521, -97.74011) is 568 meters.
The distance between (30.28642, -97.73645) and (30.28771, -97.73333) is 332 meters.
The distance between (30.26557, -97.74465) and (29.76694, -95.37830) is 234481 meters.
The distance between (33.57851, -101.88470) and (33.59019, -101.87068) is 1837 meters.
The distance between (51.51526, -0.13781) and (52.36496, 4.89876) is 357867 meters.
The distance between (-33.94493, 18.41417) and (-33.92083, 18.41494) is 2681 meters.
The distance between (-27.45790, 153.02036) and (-33.85837, 151.19125) is 732845 meters.
The distance between (61.18972, -149.81581) and (61.18969, -149.82203) is 333 meters.
```

Problem #11
60 Points

11. Reka

Program Name: Reka.java

Input File: reka.dat

Test Input File:

39	46 b	180927 i	253056 j
2 1	64 q	371834 x	2179 s
2 a	69 k	259015 q	234628 s
2 2	32 k	486216 r	366213 e
1 a	81 i	148169 q	173702 r
2 b	30 d	106314 a	464004 p
4 0	10 k	208203 j	97415 n
1 0	57 c	316366 e	550539 t
2 0	36 t	352334 m	509592 y
3 0	68 a	216017 o	457368 k
2 1	52 o	171221 f	171161 a
1 a	22 d	395093 u	577176 a
4 2	70 u	259030 k	124061 q
1 z	78 l	309592 i	39582 b
4 y	8 f	50523 r	10911 b
4 2	19 o	128099 f	98464 y
2 z	63 u	64612 c	293536 n
3 y	62 s	439014 q	181928 y
5 2	29 d	105982 o	519342 y
2 a	34 n	163691 m	497327 w
4 a	89 s	248044 i	532660 y
6 2	86 h	276205 g	26300 p
3 a	84 c	455150 g	364220 f
4 b	20 i	340332 g	360126 o
6 2	41 f	223344 s	415936 a
3 a	6 f	318835 y	544961 c
4 a	50 p	180851 d	506050 h
6 2	67 n	256372 f	284868 v
2 a	51 p	395766 d	156868 u
5 b	11 z	449782 d	154111 y
5 0	38 u	272504 e	282315 e
6 0	79 d	206835 l	196815 p
7 0	91 l	397306 w	495312 p
8 0	37 q	189694 z	571089 v
9 0	1000000 68	579439 199	240851 l
10 0	86915 z	572928 y	231636 t
11 0	10377 a	330242 p	301779 c
12 0	275070 z	233988 b	233686 f
13 0	203017 j	31748 p	557273 q
14 0	511759 m	115719 x	159962 h
15 0	172305 g	221195 j	570592 y
16 0	386066 d	491540 o	146660 g
17 0	61075 o	570388 j	34532 g
18 0	185109 s	69655 m	40167 h
19 0	114325 g	252439 t	441065 o
22 2	336407 y	475162 z	8937 s
11 a	386968 o	232477 k	363762 j
12 b	240790 d	329759 o	518898 b
14 4	353693 k	370211 w	45302 w
1 z	60830 b	471080 z	130294 f
12 x	510749 q	205361 o	540921 y
13 y	563359 d	235062 n	15613 m
14 z	100771 q	366650 l	533246 n
1000000 0	181541 h	241214 d	452863 m
10000000000000 0	565798 j	272963 c	292605 e
10000000000000000 0	293542 i	71755 v	507649 m
100 44	404519 x	347212 t	428803 q
72 d	81065 t	213581 o	458500 q
31 u	568617 l	22095 c	503562 q
17 c	91051 f	510550 v	485131 i
61 d	543277 s	339031 q	58637 c
16 x	18991 q	208472 r	471823 g
49 o	248241 e	382560 i	151823 y
40 d	19889 l	290402 z	434447 k
76 w	134963 w	199780 t	377105 x
33 a	478004 n	27236 h	312596 y
88 f	484149 n	559721 k	272148 d
53 y	507320 j	537710 c	114454 g
26 l	245821 c	255601 g	509208 k

UIL – Computer Science Judge’s Packet – Invitational B - 2021

124702 n	239538 x	33724279175919658 g	378498207151230742 o
364832 i	2482 v	554486473140984366 q	223145008078535959 t
68901 o	483252 z	542258703851144240 w	162450224603820827 n
441637 u	318904 b	664863443746286640 w	873557780149935394 v
410410 m	333241 e	18863393843736626 d	824842462999382820 u
405806 k	3517 r	34176427661475895 y	513288269785769765 e
356143 g	158146 o	444789160898998330 f	18368208843096868 z
224047 c	277957 h	699170031323737661 q	288704557191952680 i
541489 q	334277 z	578591880159810117 r	459485379135097132 w
282417 h	425425 g	860821978744629837 y	359558449104418604 a
288053 h	418769 x	25991254736015437 y	973721735505720623 t
487222 x	191443 n	572743605386662993 h	597646399914737969 m
472375 b	73172 h	902330786269279317 j	283452143745464627 v
509751 z	13271 m	845800727156171349 h	813315465218662200 m
88886 b	208858 m	677788657990682709 u	614945544953095482 l
381760 e	112091 n	300933576806552663 k	759326712344467262 n
245058 x	27615 m	309110143111008864 n	688028733862841676 y
466755 a	269798 i	347021083800094820 r	304214395342472526 m
94531 l	483815 s	593890327868819556 k	35535248397489999 c
463682 b	498160 b	839581091799203945 c	491130023232562510 l
416066 r	88048 b	769146925660893294 i	692091865354009424 d
554311 e	534512 e	798766116740451440 j	836896219810658644 c
282436 l	387064 x	281006254581871731 s	92135464203330907 d
129868 r	105467 c	850898236081246323 f	292032400036282724 q
523085 x	299007 y	372290253025967233 a	748406466880940907 q
529740 b	1000000000000000000 33	178425814122372226 d	466926702916083052 d
480077 m	257839696165379585 g	319416427550230657 u	399903269933198193 n
506194 m	403381198842142089 k	307334175752284292 q	763829558685867377 l
238422 h	355954722098540810 v	800579749685950596 q	936149790211509623 h
433494 g	545608922483614858 t	92916261467513477 x	268706877006352252 k
43355 b	396773504274808462 c	302699406043783810 f	771551169771887488 s
260444 l	641849860718657296 f	857326220155890817 l	583184653035982722 d
249182 h	528934810991526801 z	123554685125031565 b	403037556337476483 o
210271 t	258606263054056720 c	105097750520617102 a	242347179618518405 r
337250 r	645317500560252180 m	955499721601047693 g	550450019361042312 i
174951 g	587188249286592793 d	562069638753383061 j	113103982065542536 c
173928 r	377451272953973149 i	751475307538253979 d	310788251986027404 q
35175 x	242024215572773539 k	3730927163146397 q	799292088946081680 z
255338 s	49443727601385268 h	856581217015763615 r	983033612498083221 i
52075 u	450213180317129211 n	612611482775156384 c	95087973792992154 x
209260 h	268164481324146238 b	286629754320711840 e	231854196759306651 p
345452 i	648660055841614657 c	782814914516544686 k	403564465875774364 x
403311 x	347592328886980294 v	863814021095834799 q	587494565466846621 r
370543 x	106555210368345542 a	516303167945214127 e	772069238246327713 k
379250 h	156881326403983689 y	766077035936184497 s	904487448255371170 z
177011 y	316038016772759764 n	757823333339938994 a	265785794268708782 h
362355 k	299906601344626904 n	826160385118218419 c	898523374381743028 u
308084 t	571822354754716888 d	450034444877414062 c	720743253698276790 c
280954 l	298964636907423845 h	4075069878820016 v	317196705851799479 d
14718 j	490388706105808101 n	874552360022567102 u	578483134247864250 o
434046 x	165608030422057579 f	787824775601601727 t	637622672762386879 b
258432 v	342981353424286827 v	447949682904903872 n	407502765972558785 d
357250 u	68038728300110960 e	694775269574137536 h	392986617706345414 m
8590 e	237279753756773746 o	734180174504309955 a	405560693295175625 u
422287 p	467570167798868727 q	917364877072556742 f	470601682988645322 z
562574 h	473176567642607865 k	490318146361512138 v	421235720590097875 u
210835 g	545822642712709498 y	930662634542840529 p	158020891728387540 d
323988 a	648226578060051579 b	581869510395610325 t	240362733507354585 o
298389 q	257643663377309822 d	881753328862033622 y	443943732312116203 q
354710 k	994116007356013390 151	768406812936437976 c	746103778049263596 d
187287 u	102429342427165700 b	986254056275883736 f	988646709854894573 p
319384 e	851031609490302469 s	314699304537009371 n	645101882065595375 o
33176 t	119434791053523464 k	579952493150844638 b	954655971450156018 d
336795 t	479324096381468175 v	67553623256435939 m	12660796253710323 k
14748 i	680617089872420886 c	602644303456246507 t	401209856495863286 l
374688 l	48991021768440343 s	127444344373039339 d	603382566001301496 a
536993 k	981530173785759262 z	611905284220492532 p	836317789751384059 h
276897 i	436731377389576735 c	81195098101239541 a	716200770793484285 d
8104 c	281078744833485348 b	86117077689073919 n	954652090449711102 h
287659 h	794494476658485797 s	645458536817959170 d	
511918 p	131579310803496489 c	41537671886106385 r	

Reka continued

Test Output To Screen:

Case #1: 1
Case #2: 0
Case #3: 676
Case #4: 26
Case #5: 26
Case #6: 676
Case #7: 1
Case #8: 0
Case #9: 0
Case #10: 676
Case #11: 0
Case #12: 676
Case #13: 0
Case #14: 17576
Case #15: 17576
Case #16: 456976
Case #17: 456976
Case #18: 11881376
Case #19: 11881376
Case #20: 308915776
Case #21: 308915776
Case #22: 031810176
Case #23: 031810176
Case #24: 827064576
Case #25: 827064576
Case #26: 503678976
Case #27: 503678976
Case #28: 095653376
Case #29: 0
Case #30: 456976
Case #31: 987109376
Case #32: 787109376
Case #33: 787109376
Case #34: 864408576
Case #35: 171557376
Case #36: 863730176
Case #37: 0
Case #38: 366834176
Case #39: 635582976

Problem #12
60 Points

12. Timothy

Program Name: Timothy.java Input File: timothy.dat

Test Input File:

```
35 5 39 46 36 27 48 19 29 26 32 38 13 43 37 46 29 8 44 28
3 36 14 16 14 42 24 47 44 3 30 49 9 44 26 41 9 38 8 11
15 41 35 29 8 48 1 16 41 11 42 44 38 45 12 1 44 2 25 42
32 38 14 33 44 3 4 6 31 42 26 24 24 49 5 23 27 48 24 45
45 45 2 23 39 9 43 30 11 38 41 38 7 32 1 15 9 10 4 27
45 36 13 3 47 36 48 45 39 4 1 29 39 24 47 0 44 47 28 14
24 2 37 15 21 4 44 28 38 31 30 20 33 41 43 41 9 41 13 5
28 24 42 41 17 2 2 2 31 41 21 25 16 48 37 48 3 5 2 37
46 8 3 37 5 0 47 10 48 8 17 37 31 27 39 12 28 39 33 21
26 38 0 30 10 12 19 16 45 13 13 2 48 45 41 15 39 21 23 11
32 31 16 15 37 46 16 11 30 46 8 29 27 33 41 15 31 38 2 4
11 27 31 20 27 39 12 1 35 35 47 7 4 31 20 41 32 49 29 46
33 41 43 30 48 42 25 27 2 13 3 36 25 36 49 36 27 14 13 25
35 5 39 30 28 42 43 28 8 4 20 35 28 24 43 37 31 2 12 41
33 43 47 28 23 15 6 14 21 16 33 31 31 4 39 43 16 46 7 17
19 2 12 34 24 25 31 13 16 16 12 19 43 42 27 16 11 23 28 29
17 3 10 31 2 13 2 11 20 45 7 46 21 35 3 20 10 26 19 29
27 29 7 41 2 0 25 21 7 16 8 38 44 7 18 47 43 36 35 11
```

Test Output To Screen:

49 3	16 11
48 8	15 6
47 7	14 5
46 7	13 8
45 8	12 6
44 8	11 8
43 9	10 4
42 7	9 4
41 14	8 7
39 9	7 6
38 9	6 1
37 7	5 5
36 7	4 7
35 7	3 8
34	2 14
33 6	1 4
32 4	0 3
31 12	
30 6	
29 8	
28 9	
27 10	
26 4	
25 6	
24 8	
23 4	
21 6	
20 5	
19 4	
18	
17 3	