

HW4 REPORT

Can Duyar – 171044075

Part-1: Everything works properly as I explained.

```
?- route(edirne,X).  
X = edremit ;  
X = erzincan ;  
false.  
?- 
```

Possible flights
from edirne.

```
?- route(edirne,van).  
false.  
?- 
```

No route between
edirne and van so it
returns false.

```
?- route(istanbul,antalya).  
true ;
```

There is a route
between istanbul
and antalya so it
returns true.

Possible flights
from istanbul.

```
?- route(istanbul,X).  
X = izmir ;  
X = ısparta ;  
X = burdur ;  
X = antalya ;  
X = konya ;  
X = ankara ;  
X = van ;  
X = rize ;  
X = gaziantep ;  
X = gaziantep ;  
X = antalya ;  
X = konya ;  
X = ankara ;  
X = van ;  
X = rize ;  
X = ankara ;  
X = konya ;  
X = antalya ;  
X = gaziantep ;  
X = van ;  
X = rize ;  
X = van ;  
X = ankara ;  
X = konya ;  
X = antalya ;  
X = gaziantep ;  
X = rize ;  
X = rize ;  
X = van ;  
X = ankara ;  
X = konya ;  
X = antalya ;  
X = gaziantep ;  
false.  
?- 
```

Part – 2: Everything works properly as I explained.

```
?- sroute(edremit,erzincan,X).  
X = 736.34 ;  
false.  
?- 
```

X is 736.34 because I have a knowledge base part like following.

```
distance(edremit,erzincan,736.34).
```

```
?- sroute(van,konya,X).  
X = 1147.6499999999999 ;  
false.  
?- 
```

There is no way to go konya from van directly so I calculate the shortest route with using (van → ankara) + (ankara → konya)
X is ~ 1147.6499999999999 because I have knowledge base parts like following.

```
distance(van,ankara,920.31).
```

```
distance(ankara,konya,227.34).
```

$920.31 + 227.34 = \sim 1147.65$

Part – 3: Everything works properly as I explained.

```
?- schedule(a,P,T).  
P = z23,  
T = 10 ;  
P = z11,  
T = 12.  
  
?- usage(207,T).  
T = 16 ;  
T = 17.
```

Schedule of student-a according to our knowledge base.

Student-a uses z23 at 10 and z11 at 12.

207 is used by students at 16 and 17.

```
?- conflict(452,455).  
true.  
  
?- conflict(108,341).  
false.
```

It returns true because 452 and 455 conflicts due to classroom.

There is no confliction between 108 and 341 due to classroom or time.

```
?- meet(a,b).  
true
```

→ a and b students are in the same classroom at the same time so it returns true.

```
?- meet(b,c).  
false.
```

→ a and b students can not be in the same classroom at the same time so it returns false.

Part – 4: Everything works properly as I explained.

```
?- element(13,[2,8,13,9]).  
true.
```

→ 13 is in [2,8,13,9] so it returns true.

```
?- element(5,[2,8,0]).  
false.
```

→ 5 is not in [2,8,0] so it returns false.

```
?- union([1,2,3],[4,5,6],X).  
X = [1, 2, 3, 4, 5, 6]
```

→ Union of [1,2,3] and [4,5,6] is [1,2,3,4,5,6] so it returns it.

```
?- union([3,5],[7,8],[1,2,3,4,5]).  
false.
```

→ Union of [3,5] and [7,8] is not [1,2,3,4,5] so it returns false.

```
?- union([3,5],[7,8],[3,5,7,8]).  
true ;
```

→ Union of [3,5] and [7,8] is [3,5,7,8] so it returns true.

```
?- intersect([4,6,7,9],[0,1,2,7],X).  
X = [7].
```

→ Intersection of [4,6,7,9] and [0,1,2,7] is [7] so it returns it.

```
?- intersect([1,2,3],[7,8,9],X).  
X = [].
```

→ There is no Intersection between [1,2,3] and [7,8,9] so it returns [].

```
?- intersect([7,6,3],[6,8,1],[6]).  
true.
```

→ Intersection of [7,6,3] and [6,8,1] is [6] so it returns true.

```
?- intersect([0,4,7],[11,7,9],[4]).  
false.
```

→ Intersection of [0,4,7] and [11,7,9] is not [4] so it returns false.

```
?- equivalent([1,3,5],[7,9,8]).  
false.  
?- equivalent([1,3,5],[1,3,5]).  
true .  
?- equivalent([1,3,5],[1,5,3]).  
true
```

[1,3,5] is not equal to [7,9,8] so
it returns false.

[1,3,5] is equal to [1,3,5] so it
returns true.

[1,3,5] is also equal to [1,5,3]
so it returns true. They are
equal because I used
permutation in equivalent.

```
equivalent(S1, S2) :- permutation(S1, S2).
```

Part – 5: Everything works properly as I explained.

It finds a correct way of inserting arithmetic (operators) such that the result is a correct equation.

*Program reads from input.txt and prints to output.txt.

example(which is given on pdf):

input.txt: (don't use any spaces in input.txt/also you can write just one line)

```
File Edit Selection Find View Goto Tools Project Pr
part1.pl x input.txt — hw44444444 x ou
1 [5,3,5,7,13].|
```

output.txt:

```
File Edit Selection Find View Goto Tools Project Prefer
part1.pl x output.txt — Desktop/hw44444444 x
1 5+3*5 = 7+13
2 5*3+5 = 7+13
3 5+(3*5-7) = 13
4 5*3+(5-7) = 13
5 5+3*5-7 = 13
6 5*3+5-7 = 13
7
```

Possible equations
for given numbers.

example(extra):

input.txt: (don't use any spaces in input.txt/also you can write just one line)

```
output.txt x input.txt x p
1 [7,91,13,7,21].|
```

output.txt:

```
output.txt x input.txt x
1 7+(91/13+7) = 21
2 7+91/13+7 = 21
3 |
```

Possible equations
for given numbers.

Part – 6: Puzzle solving - Everything works properly as I explained.

there are 3 example puzzles to solve on pdf.

*Program also prints the solved puzzle to output.txt with bitmap.

*Test-2 and 3 take a lot of time to run!!!

Test Cases: Top-to-bottom or Left-to-right

1. (([3], [2,1], [3,2], [2,2], [6], [1,5], [6], [1], [2]),
[[1,2], [3,1], [1,5], [7,1], [5], [3], [4], [3]])
2. ([[3,1], [2,4,1], [1,3,3], [2,4], [3,3,1,3], [3,2,2,1,3], [2,2,2,2,2],
[2,1,1,2,1,1], [1,2,1,4], [1,1,2,2], [2,2,8], [2,2,2,4], [1,2,2,1,1,1],
[3,3,5,1], [1,1,3,1,1,2], [2,3,1,3,3], [1,3,2,8], [4,3,8], [1,4,2,5], [1,4,2,2],
[4,2,5], [5,3,5], [4,1,1], [4,2], [3,3]],
[[2,3], [3,1,3], [3,2,1,2], [2,4,4], [3,4,2,4,5], [2,5,2,4,6], [1,4,3,4,6,1],
[4,3,3,6,2], [4,2,3,6,3], [1,2,4,2,1], [2,2,6], [1,1,6], [2,1,4,2], [4,2,6],
[1,1,1,1,4], [2,4,7], [3,5,6], [3,2,4,2], [2,2,2], [6,3]])
3. ([[5], [2,3,2], [2,5,1], [2,8], [2,5,11], [1,1,2,1,6], [1,2,1,3], [2,1,1],
[2,6,2], [15,4], [10,8], [2,1,4,3,6], [17], [17], [18], [1,14], [1,1,14], [5,9],
[8], [7]], [[5], [3,2], [2,1,2], [1,1,1], [1,1,1], [1,3], [2,2], [1,3,3],
[1,3,3,1], [1,7,2], [1,9,1], [1,10], [1,10], [1,3,5], [1,8],
[2,1,6], [3,1,7], [4,1,7], [6,1,8], [6,10], [7,10], [1,4,11], [1,2,11], [2,12],
[3,13]])

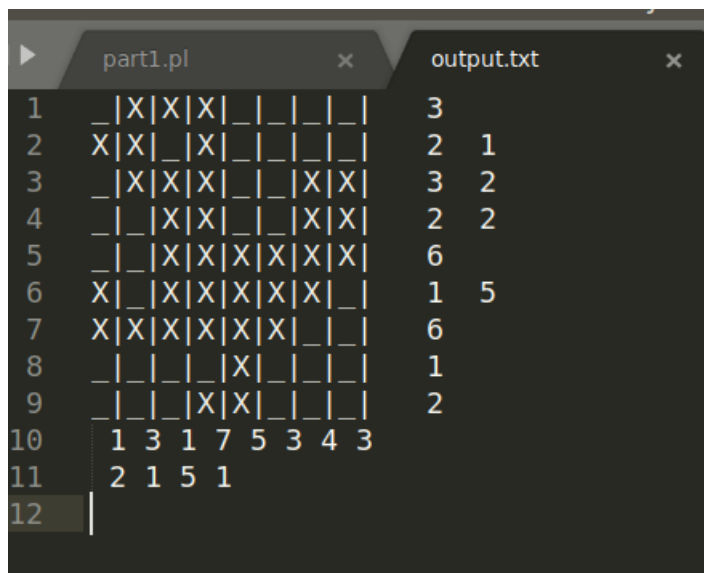
Solving of test case-1:

```
(base) can@can-ThinkPad-L13:~/Desktop/hw44444444$ swipl part6.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.3)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- test_mode('gtu1').
```

output.txt:



The screenshot shows a SWI-Prolog IDE with two tabs: 'part1.pl' and 'output.txt'. The 'part1.pl' tab contains a Prolog program with 12 lines. The 'output.txt' tab shows the results of the program's execution, which are two lists of numbers. The first list is [1, 3, 1, 7, 5, 3, 4, 3] and the second list is [2, 1, 5, 1].

```
1  _|X|X|X|_|_|_|_| 3
2  X|X|_|X|_|_|_|_| 2 1
3  _|X|X|X|_|_|X|X| 3 2
4  _|_|X|X|_|_|X|X| 2 2
5  _|_|X|X|X|X|X|X| 6
6  X|_|X|X|X|X|X|_| 1 5
7  X|X|X|X|X|X|_|_| 6
8  _|_|_|_|X|_|_|_| 1
9  _|_|_|X|X|_|_|_| 2
10 1 3 1 7 5 3 4 3
11 2 1 5 1
12
```

Solving of test case-2:

```
(base) can@can-ThinkPad-L13:~/Desktop/hw44444444$ swipl part6.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.3)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- test_mode('gtu2').
```

output.txt:

	part1.pl	output.txt	part2.pl	part3.pl	part4.pl	part5	
1	_ _ _ X X X _ X _ _ _ _ _ _ _ _ _	3	1				
2	_ _ _ X X _ X X X X _ X _ _ _ _ _	2	4	1			
3	_ _ _ X _ X X X _ X X X _ _ _ _ _	1	3	3			
4	_ _ X X _ X X X X _ _ _ _ _ _ _ _	2	4				
5	_ X X X _ X X X _ X _ _ _ _ X X X _	3	3	1	3		
6	X X X _ _ X X _ X X _ _ _ _ X _ X X X _	3	2	2	1	3	
7	X X _ _ X X _ X X _ _ _ _ X X _ X X _	2	2	2	2	2	
8	_ _ _ X X _ X _ X _ _ X X _ X _ X _	2	1	1	2	1	1
9	_ _ _ X _ X X _ X _ _ _ X X X X _ _	1	2	1	4		
10	_ _ _ X _ X _ X X _ _ _ _ _ X X _ _	1	1	2	2		
11	_ _ _ _ X X _ X X _ _ X X X X X X X X X	2	2	8			
12	_ _ _ X X _ X X _ _ _ X X _ _ X X X X	2	2	2	4		
13	_ _ _ X _ X X _ X X _ X _ _ _ X _ _ X	1	2	2	1	1	1
14	X X X _ _ X X X _ X X X X X _ _ _ X	3	3	5	1		
15	X _ X _ X X X _ X _ _ _ _ X _ _ _ X X	1	1	3	1	1	2
16	X X _ _ X X X _ X _ _ _ _ X X X _ X X X	2	3	1	3	3	
17	_ X _ X X X _ X X _ X X X X X X X X X _	1	3	2	8		
18	_ X X X X _ X X X _ X X X X X X X X X _	4	3	8			
19	_ _ X _ X X X X _ X X _ X X X X X X _	1	4	2	5		
20	_ _ X _ X X X X _ X X _ _ _ X X _ _	1	4	2	2		
21	_ _ _ _ X X X X X _ X X _ _ _ X X X X X X	4	2	5			
22	_ _ _ X X X X X _ X X X _ _ _ X X X X X X	5	3	5			
23	_ _ _ X X X X _ X _ _ _ _ _ _ _ _ X	4	1	1			
24	_ _ X X X X _ X X _ _ _ _ _ _ _ _	4	2				
25	_ _ X X X _ X X X _ _ _ _ _ _ _ _	3	3				
26	2 3 3 2 3 2 1 4 4 1 2 1 2 4 1 2 3 3 2 6						
27	3 1 2 4 4 5 4 3 2 2 2 1 1 2 1 4 5 2 2 3						
28	3 1 4 2 2 3 3 3 4 6 6 4 6 1 7 6 4 2						
29	2 4 4 4 6 6 2 2 1 2						
30	5 6 6 2 3 1 4						
31	1						

Solving of test case-3:

```
(base) can@can-ThinkPad-L13:~/Desktop/hw44444444$ swipl part6.pl
Welcome to SWI-Prolog (threaded, 64 bits, version 8.2.3)
SWI-Prolog comes with ABSOLUTELY NO WARRANTY. This is free software.
Please run ?- license. for legal details.

For online help and background, visit https://www.swi-prolog.org
For built-in help, use ?- help(Topic). or ?- apropos(Word).

?- test_mode('gtu3').
```


output.txt:

```

part1.pl x output.txt x part2.pl x part3.pl x part4.pl x part5.pl x
1 | | | | | | | | | | | | | | | | X|X|X|X|X| 5
2 | | |X|X| | | | | | | | | | |X|X|X| | |X|X| 2 3 2
3 | |X|X| | | | | | | | | | |X|X|X|X|X| | |X| 2 5 1
4 |X|X| | | | | | | | | | |X|X|X|X|X|X|X|X| | | 2 8
5 |X|X| | | | |X|X|X|X|X| |X|X|X|X|X|X|X|X|X|X|X| 2 5 11
6 |X| |X| | |X|X| | | | |X| | | |X|X|X|X|X|X|X| | 1 1 2 1 6
7 |X| | |X|X| | | | |X| | | | | |X|X|X| | | | | 1 2 1 3
8 |X|X| | | | | | |X| | | | | | | | | | | |X| 2 1 1
9 | |X|X| | | | | |X|X|X|X|X|X| | | | | | | |X|X| 2 6 2
10 | | |X|X|X|X|X|X|X|X|X|X|X|X|X| | | | |X|X|X|X| 15 4
11 | | | | |X|X|X|X|X|X|X|X|X|X|X| | |X|X|X|X|X|X|X|X| 10 8
12 | | | | |X|X| |X| |X|X|X|X|X| |X|X|X| | |X|X|X|X|X|X| 2 1 4 3 6
13 | | | | | | | |X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X| 17
14 | | | | | | | |X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X| 17
15 | | | | | | | |X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X|X| 18
16 | | | | | | | |X| | | |X|X|X|X|X|X|X|X|X|X|X|X|X|X|X| 1 14
17 | | | | | | | |X| |X| |X|X|X|X|X|X|X|X|X|X|X|X|X|X|X| 1 1 14
18 | | | | | | | |X|X|X|X|X| | | |X|X|X|X|X|X|X|X|X|X|X| 5 9
19 | | | | | | | | | | | | | | |X|X|X|X|X|X|X|X|X|X|X| 8
20 | | | | | | | | | | | | | | | |X|X|X|X|X|X|X|X|X|X| 7
21 | 5 3 2 1 1 1 2 1 1 1 1 1 1 1 2 3 4 6 6 7 1 1 2 3
22 | 2 1 1 1 3 2 3 3 7 9 10 10 3 8 1 1 1 1 10 10 4 2 12 13
23 | 2 1 1 3 3 2 1 5 6 7 7 8 11 11 1
24 | 1

```