

First manual execution for the graph using approximate vertex cover algorithm:

8 8

1 4 3

2 3 7

4 3 11

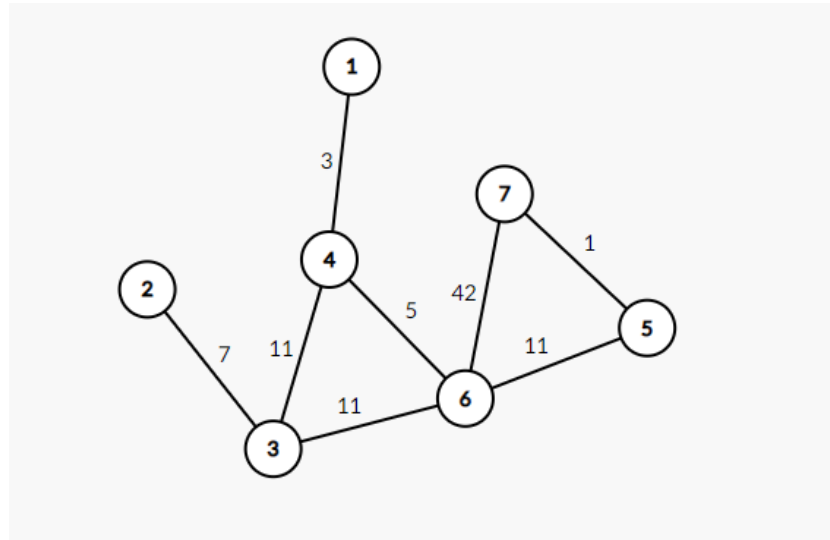
4 6 5

3 6 11

6 5 11

6 7 42

5 7 1

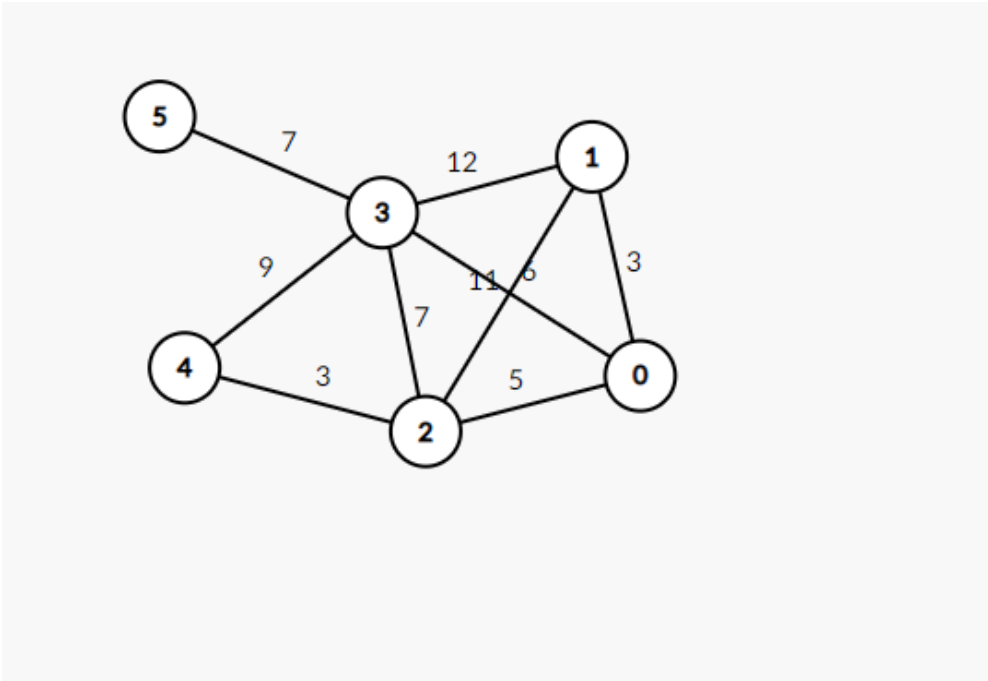


We obtained the approximate vertex cover {1, 3, 4, 5, 6, 7}

	Choice(random edge)	Result	edges
Start :			(1, 4), (4, 1), (2, 3), (3, 2), (4, 3), (3, 4), (4, 6), (6, 4), (3, 6), (6, 3), (6, 5), (5, 6), (6, 7), (7, 6), (5, 7), (7, 5)
Iteration 1	(6, 3)	{3, 6}	(1, 4), (4, 1), (5, 7), (7, 5)
Iteration 2	(1, 4)	{1, 3, 4, 6}	(5, 7), (7, 5)
Iteration 3	(5, 7)	{1, 3, 4, 5, 6, 7}	[]

Second manual execution using greedy vertex cover algorithm:

6 9
0 1 3
0 2 5
0 3 6
1 2 11
1 3 12
2 3 7
2 4 3
3 4 9
3 5 7



We obtained the minimum vertex cover {0, 2, 3}

	V (max degree vertex)	solution	dictEdges
Start:			{0: [1, 2, 3], 1: [0, 2, 3], 2: [0, 1, 3, 4], 3: [0, 1, 2, 4, 5], 4: [2, 3], 5: [3]}
Iteration 1	3	{3}	{0: [1, 2], 1: [0, 2], 2: [0, 1, 4], 4: [2], 5: []}
Iteration 2	2	{2,3}	{0: [1], 1: [0], 4: [], 5: []}
Iteration 3	0	{0,2,3}	{1: [], 4: [], 5: []}
Iteration 4	1 (degree 0)		