Graph problems!

Time-outs:

- Python: 5 seconds on Q1 through Q4, 10 seconds on Q5
- C++ (w/ g++ -O3 flag): 2 seconds on Q1 through Q4, 4 seconds on Q5

You may solve the programming problems in either Python or C++. To use Python, name the file containing your code `main.py`. To use C++, name the file `main.cpp`. You are encouraged to test your code locally.

At any time, you can choose to run the tests that will be used to grade your solution. To do so, navigate to the Gradescope assignment for that particular homework question and upload **only** `main.py` or `main.cpp`. Gradescope will tell you how many test cases passed, and what error occurred on the ones that failed.

When you run the tests on Gradescope, debugging output will be printed to tell you which test cases you passed, which ones you failed, and what your total score currently is for that problem. If a test case failed, the output will specify a failure mode (e.g. timeout, compiler error, wrong answer, etc.).

The first few test cases for each problem are visible test cases. This means that if you fail them, Gradescope will tell you the the expected output and the output your code actually gave. Note that these outputs may not be listed in order. The rest of the test cases for each problem are hidden test cases, which means you will not be shown the input or expected output.

All assignments MUST be manually submitted to Gradescope in order to complete submission! There is no auto-submit feature enabled for any Gradescope coursework. There will be no exemptions granted if you forget to manually submit your assignment. If time permits, you can contact the course staff and they can check on the Gradescope platform and confirm your assignment submission status.

1. Establishing Communications

(100 pts)

Description

As each team of explorers from Oganesson Dynamics arrives at their destination, the first thing they want to do is setup a communications base. But where should they put it?

To decide, they build a graph of the area. Each node represents a work site, and each edge indicates two sites that are close enough to communicate if one of them has the communications base. In fact, pairs of sites can relay through the base to communicate, as long as they are both connected to it. It can be slow, but it works well.

You must determine where to put the base such that the MOST pairs of sites can communicate.

Input Format

- The first line contains *N,* the number of sites being analyzed, and *M*, the number of pairs (edges) that could potentially communicate.
- The next *M* lines each have two values identifying all pairs of sites that could communicate with a base station at one of them.

Constraints

- $1 \leq N \leq 200$
- $1 \leq M \leq 10{,}000$
- $0 \leq$ site IDs $< N$

Output Format

Output the count of how many PAIRS of sites would be able to communicate if the communications base were placed in the optimal position.

Example 0

Sample Input

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

Sample Output

```
 10
```

Explanation

If a communications node is placed at size 1, it will connect to 0, 2, 3, and 4.

This means that there are 10 pairs that can communicate. Node 1 can directly communicate with each of the four others PLUS it can relay communications between (0,2), (0,3), (0,4), (2,3), (2,4), and (3,4).

Example 1

Sample Input

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

Sample Output

```
 28
```

Example 2

Sample Input

```
10 20
0 3
0 5
0 6
0 9
1 4
1 6
1 8
1 9
2 3
2 4
2 5
2 7
2 8
3 6
3 9
4 6
5 6
5 8
6 9
7 8
```

Sample Output

```
 21
```

2. Expanding Communications

(100 pts)

Description

Some teams from Oganesson dynamics were sent with two communications bases. These bases must be connected to each other by an edge, but have the ability to relay messages through one another, connecting any of their neighbors and broadly expanding the range covered.

You must determine where to put the two bases such that the most pairs of sites can communicate.

Input Format

- The first line contains *N,* the number of sites being analyzed and *M*, the number of pairs of sites that could potentially communicate, including through related messages.
- The next *M* lines each have two values identifying all pairs of sites that could directly communicate if a base station is at one of them.

Constraints

- $1 \leq N \leq 200$
- $1 \leq M \leq 10{,}000$
- $0 \leq$ site IDs $< N$

Output Format

Output the count of how many PAIRS of sites would be able to communicate if the communications bases were placed in the optimal position.

Example 0

Sample Input

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

Sample Output

```
  10
```

Explanation

If communications nodes are placed at sites 0 and 2, they can facilitate communication among all 5 sites, this is allowed because sites 0 and 2 are connected to each other. On top of that, site 0 is connected to 1 and 4, while site 2 is connected to 3 and 4.

With fives sites, there are (5 x 4)/2 = 10 pairs.

Example 1

Sample Input

```
7 15
0 3
0 4
0 5
0 6
1 2
1 3
1 5
2 3
2 4
2 6
3 4
3 5
3 6
4 6
5 6
```

Sample Output

```
  21
```

Example 2

Sample Input

```
10 20
0 3
0 5
0 7
0 8
0 9
1 4
1 6
1 7
2 3
3 4
3 5
3 6
3 9
4 6
4 8
4 9
5 7
6 7
7 9
8 9
```

Sample Output

```
 36
```

Example 3

Sample Input

```
25 30
0 20
1 3
1 21
1 24
2 9
2 10
2 11
2 17
3 5
3 9
4 6
4 16
4 17
4 19
6 21
7 13
8 17
9 19
10 21
12 13
12 19
12 20
13 17
13 18
14 20
15 20
15 22
19 20
21 22
23 24
```

Sample Output

```
 28
```

3. Analyzing Communication Clusters

(200 pts)

Description

Over time, exploration teams will build more communications bases and set them up at each site. However, there's still a problem! Some sites simply cannot be connected at all, no matter how many bases are used.

You must analyze a connectivity network and identify how many distinct (unconnected) components a graph is divided into.

Input Format

- The first line contains N, the number of sites being analyzed, and M, the number of connections that will exist when all nodes are in place.
- The next M lines each have two values identifying a pair of connected sites.

Constraints

- $1 \leq N \leq 200$
- $1 \leq M \leq 10{,}000$
- $0 \leq$ site IDs $< N$

Output Format

Print the number of independent site clusters in the input graph.

Example 0

Sample Input

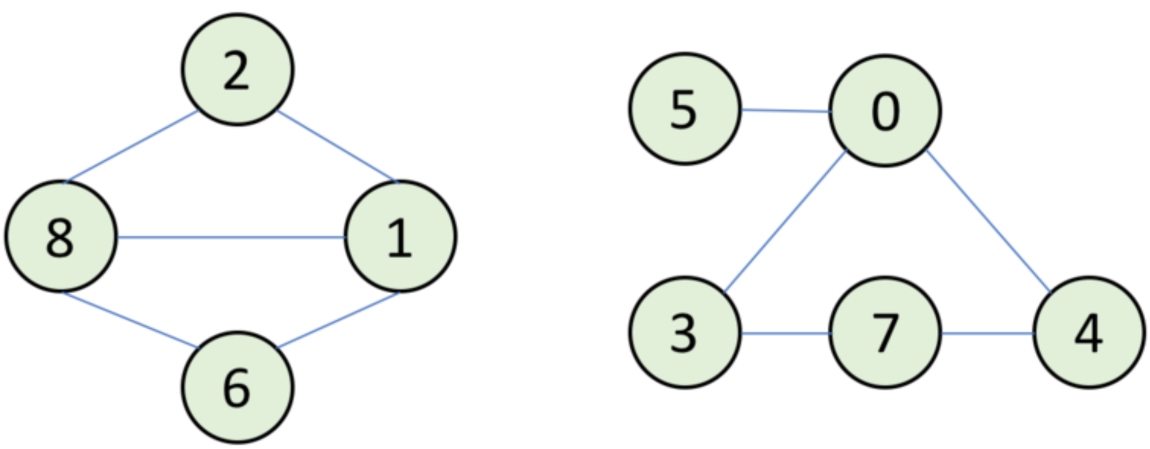
```
9 10
0 3
0 4
0 5
1 2
1 6
1 8
2 8
3 7
4 7
6 8
```

Sample Output

```
 2
```

Explanation

This graph looks like this:



Example 1

Sample Input

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

Sample Output

```
 4
```

Example 2

Sample Input

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

Sample Output

```
 1
```

Example 3

Sample Input

```
224 318
0 75
0 113
0 215
1 65
1 95
1 138
2 15
2 168
3 106
3 118
3 212
4 5
4 101
4 169
4 198
5 101
5 113
5 169
5 198
6 16
6 18
6 72
6 167
7 23
7 80
8 43
8 145
9 53
10 38
10 82
10 158
10 183
11 50
12 166
13 15
```

13 66
13 160
13 191
14 48
14 88
14 142
16 105
16 127
16 164
17 128
17 182
18 72
18 129
18 164
19 54
19 122
19 159
19 196
19 199
19 207
20 42
20 138
20 218
21 133
21 161
21 189
22 84
22 140
23 200
24 45
24 195
24 214
25 28
25 41
25 130
25 173
26 58
26 79
27 63
27 165

```
27 216
28 96
28 112
28 219
28 222
29 163
30 32
31 47
31 137
32 160
33 53
33 135
33 182
34 43
34 55
35 67
35 180
35 204
35 221
36 80
37 56
38 71
38 183
39 218
40 48
40 88
40 115
41 152
42 218
43 194
44 63
44 103
44 157
45 163
46 118
46 212
47 210
48 147
48 220
49 93
```

```
49 144
49 212
50 210
51 71
51 94
51 121
52 122
54 102
55 117
56 154
57 170
57 182
58 86
58 98
58 110
58 176
59 221
60 179
60 181
60 203
61 97
61 136
61 163
62 86
62 111
63 103
63 162
63 165
64 153
65 95
65 141
65 218
67 68
67 116
67 171
67 204
67 221
69 74
69 100
69 114
```

69	133
69	134
69	201
70	170
71	158
71	217
72	127
72	167
73	203
74	133
74	134
75	101
75	198
76	116
76	171
76	180
76	204
76	221
77	96
77	112
77	219
78	83
78	89
78	181
79	86
80	153
80	200
80	206
81	195
82	94
83	125
83	181
84	184
84	193
85	131
85	135
85	170
85	182
86	98
86	110

86	111
87	214
88	151
89	125
89	203
90	117
90	194
90	197
91	163
92	205
93	106
93	185
93	212
95	141
96	173
99	183
99	223
100	114
100	133
101	113
101	198
102	122
102	156
103	165
103	216
104	166
104	190
105	127
105	129
105	167
106	172
106	185
107	126
107	213
108	162
109	182
110	143
111	176
113	198
113	215

114	189
115	147
115	177
118	175
119	181
120	171
121	158
121	183
122	207
123	186
123	187
123	193
123	211
124	157
124	216
125	179
125	181
126	192
127	167
128	170
129	167
129	178
130	150
132	190
133	161
133	189
133	201
136	163
136	195
137	154
137	192
137	210
139	194
139	197
140	184
140	186
141	202
142	146
142	151
142	220

```
144 174
144 212
145 197
146 147
147 151
148 173
148 219
149 191
153 166
153 190
155 198
156 159
156 199
157 162
157 216
158 217
158 223
160 168
161 189
162 216
163 214
164 167
165 216
166 200
168 191
168 205
171 180
172 174
174 175
175 185
177 220
184 186
184 188
186 211
187 188
187 193
193 211
194 197
199 209
202 218
```

```
204 221
208 214
210 213
```

Sample Output

```
 20
```

4. Pricing Instantaneous Communications

(200 pts)

Description

Great news! Oganesson Dynamics has worked out the technology to provide instantaneous communications between two sites, MUCH faster than the old system. The problem is that it is quite expensive to setup, especially since a sending unit and receiving unit must be built for each site and the price varies depending on how far apart they are and if there are any obstacles in between.

Given a graph where nodes represent sites and edges provide the cost of connecting two sites, determine the cheapest it would be to make sure that ALL of the sites can communicate with one another. (Again, relaying of information is allowed, and much faster this time!)

Input Format

- The first line contains N, the number of sites, and M, the number of possible pairs of sites that could communicate.
- The next M lines each have three integer values. The first two identify the two sites involved, while the third provides the cost to setup a communications channel.
- All graphs are guaranteed to be connected.

Constraints

- $1 \leq N \leq 200$
- $1 \leq M \leq 10{,}000$
- $0 \leq$ cost to setup a communication channel $\leq 1{,}000{,}000$
- $0 \leq$ site IDs $< N$

Output Format

A single value indicating the minimum total cost to setup communications.

Example 0

Sample Input

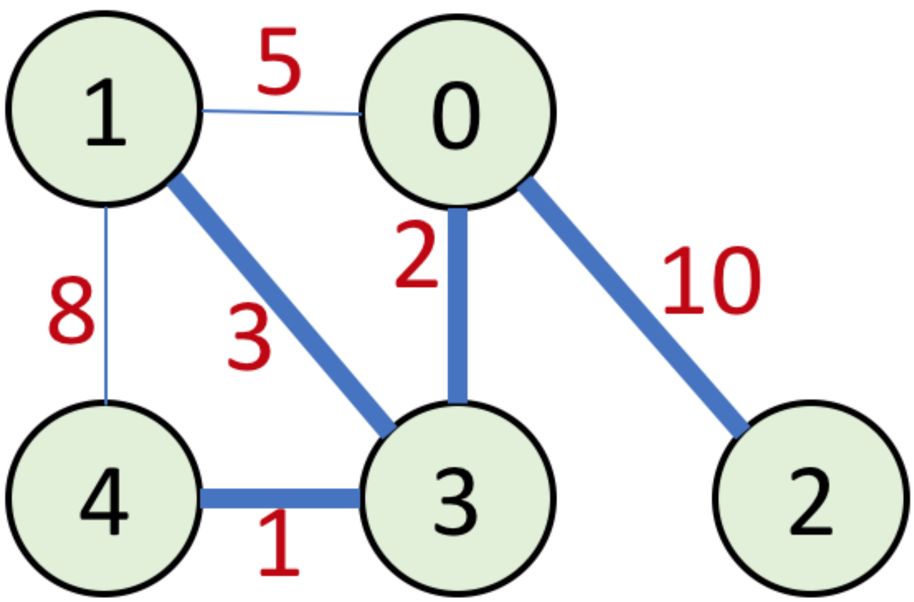
```
5 6
0 1 5
0 2 10
0 3 2
1 3 3
1 4 8
3 4 1
```

Sample Output

```
 16
```

Explanation

The graph is as follows, with the minimum-cost communication channels highlighted:



As such, the total cost is 1 + 2 + 3 + 10 = 16.

Example 1

Sample Input

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

Sample Output

```
 11
```

Example 2

Sample Input

```
6 15
0 1 17
0 2 13
0 3 10
0 4 11
0 5 10
1 2 11
1 3 15
1 4 15
1 5 14
2 3 12
2 4 16
2 5 12
3 4 14
3 5 10
4 5 12
```

Sample Output

```
 54
```

5. Securing Communications

(595 pts)

Description

Oganesson Dynamics wants to ensure that all of its communications within a network remain private, but this requires extra equipment for at least one end of each communicating pair of sites.

Given a graph where vertices represent work sites and edges represent pairs of sites that must be able to communicate securely, what are the fewest work sites that can have security-monitoring nodes to guarantee that each pair has at least one?

Hint: this is a brute-force optimization problem in which you will want to use many different optimizations (e.g., backtracking, branch and bound, etc.). As in homework 4's optimization problem, 400 points is considered to be full credit on the problem. However, 595 points are possible.

IMPORTANT NOTE 1: For this problem, your program must be guaranteed to output the correct answer (or timeout in the process of searching). If your program's output is incorrect for a single test case, it will receive 0 credit on this problem (regardless of how many other test cases it passes). Timing out is allowed, however. Test cases on which you time out will result in no points from that test case, but will not zero out points earned on test cases where your program returned the correct output.

IMPORTANT NOTE 2: Only the "Graded testcases" problem (HW5 Q5a on Gradescope) is graded. There is only a single testcase which actually consists of 100 different sub-testcases; each sub-testcase is worth either 2.5 points or 14 points, and the total amount of points possible across all sub-testcases is 592 points.

All other testcases in the “Ungraded testcases" problem (HW5 Q5b on Gradescope) are worth 0 points and are for testing purposes only. The first few test cases for the “Ungraded testcases” problem are visible test cases. This means that if you fail them, Gradescope will tell you the the expected output and the output your code actually gave. Note that these outputs may not be listed in order. The rest of the test cases for the “Ungraded testcases” problem are hidden test cases, which means you will not be shown the input or expected output.

IMPORTANT NOTE 3: Full credit for the problem will be 400 points. Thus, 195 bonus points are available.

Input Format

- The first line contains two values: the number of work sites (N) and the number of pairs of sites that must be able to communicate securely (M).
- The next M lines each describe a pair of communicating sites, indicating the unique id of each.

Constraints

- $1 \le N \le 200$
  $1 \le M \le 10{,}000$
  $0 \le$ Work Site IDs $< N$

Output Format

A single number indicating the minimum number of work sites that must have security monitoring nodes to ensure that at least one exists for every communicating pair.

Example 0

Sample Input

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

Sample Output

```
 3
```

Explanation

Security nodes at sites 0, 1, and 2 will ensure that all communications in the network are safe.