# Comparison of Brute Force VS FPT implementation of Vertex Cover problem for varied input graphs

- For the comparison, the input instances' number of vertices generated are upper-bounded by a small integer (here 70) because of computability issues. However, the graph generating python program is generalised and can be used to generate graphs of any size by changing a single parameter to the random function.
- The input specification is like this: The first line is the parameter 'k'. Next following lines are the edges in the graph. The single number lines are the isolated vertices in the graph.

Comparison of running times:

*For comparison purposes, the 'k' value is taken as 5 uniformly across all the cases.

| Number of vertices | parameter k | Brute force algorithm running time (seconds) | FPT algorithm running time (seconds) |
|---|---|---|---|
| **27** | 13 | 753.95 | 1.3 |
| **32** | 15 | 28.99 | 0.14 |
| **19** | 8 | 1.6 | 0.003099 |
| **30** | 20 | 1236 | 335.10 |
| **40** | 15 | 26.33 | 2.2881 |
| **15** | 5 | 0.06 | 0.001811 |
| **10** | 5 | 0.018 | 8.702 |
| **57** | 30 | 38 | 2.408 |
| **54** | 20 | 32 | 0.15 |
| **70** | 30 | 32.84 | 5.602 |
| **37** | 13 | 47 | 0.178 |
| **13** | 6 | 0.08 | 0.488 |
| **16** | 7 | 0.265 | 0.0012 |
| **58** | 27 | 43.28 | 2.002 |
| **26** | 14 | 392.485 | 1.340 |
| **39** | 19 | 28.51 | 1.898 |
| **11** | 5 | 0.046 | 0.00391 |
| **37** | 17 | 62.13 | 2.09 |
| **33** | 9 | 605.763 | 1.50 |
| **44** | 15 | 93.33 | 0.588 |

| Number of vertices | parameter k | Brute force algorithm running time (seconds) | FPT algorithm running time (seconds) |
|---|---|---|---|
| **36** | 17 | 34.23 | 2.59 |
| **39** | 21 | 128.02 | 57.22 |
| **33** | 20 | 32 | 1.14 |
| **18** | 8 | 54 | 35.89 |
| **17** | 9 | 0.899 | 0.021 |