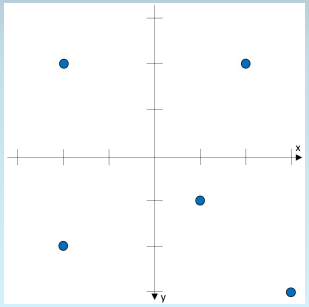
Use a Small Graph as an Example

Here's a small graph with **5 vertices** and **no edges**.

The pyramid tree has **7 nodes**.

L = $\left\lfloor \frac{7+2}{4} \right\rfloor$ = 2.

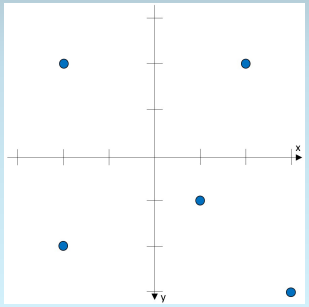


Look at the Output for a Small Graph

The ASCII file for the pyramid tree is:

```
7
1 1 0
-1 -2 1
2
0
3
1
4
```

total: 26 bytes



Look at the Output for a Small Graph

The ASCII file for the pyramid tree is:

```
7
1 1 0
-1 -2 1
2
0
3
1
4
```

total: 26 bytes

The binary file for the pyramid tree is:

- **4B** for `n_nodes`
- **16B** per node
- **7** nodes

total: 116 bytes

ASCII is Smaller Because We Left Out Unnecessary Bits

The results are similar for the graph of the streets in the Champaign-Urbana area:

- **570,555B for ASCII**, and
- **942,164B for binary**.

Why is the binary file larger?

We saved a lot of space by not writing everything.

- If we had written all four fields
- for all nodes in **ASCII**,
- the result is over **1.5MB**.

And most numbers are small.

Could have done the same with the binary file.