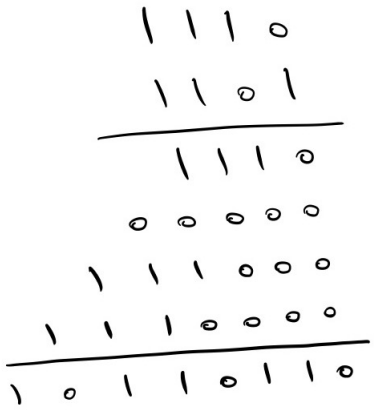
quiz 2 cmput 304 fall 2022

Each question has a multiple variants, here is a solution to one of them.

Solution for 1:

Part (a):



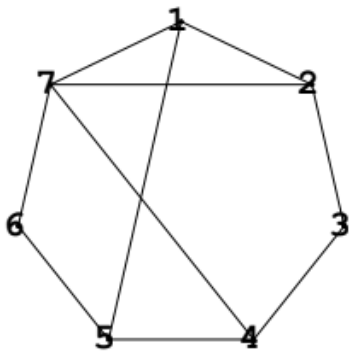
part (b): Note that we multiply every digits of the second number to every digits of the first number and since both numbers have n bits, we are doing n^2 such operations so the running time is $\Omega(n^2)$. Also note that, we are adding two numbers $n-1$ times. Each number involve in the addition operation has up to $2n$ bits. Since each addition operation takes $\Omega(n)$ time, in total for adding all these numbers, we spend $\Omega(n^2)$ time. So in total our algorithm runs in $\Omega(n^2)$.

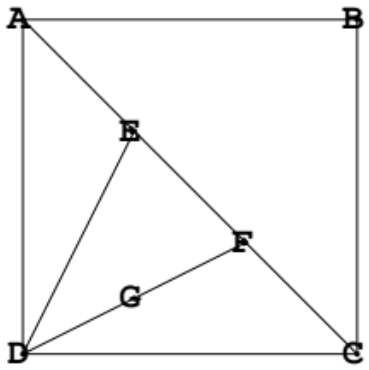
It is easy to see that the running time of this multiplication algorithm is in fact $\theta(n^2)$.

2. If the two graphs are isomorphic, complete the isomorphism below by filling in the blanks. If the graphs are not isomorphic, sketch a proof of why (you do not have to give all details).

Solution: All the 3 problem sets in the quiz had both the graphs isomorphic. One of the graphs that came in the quiz is as follows.

```
isomorphism: left graph node     1  2  3  4  5  6  7
             right graph node    _  _  _  _  _  _  _
```





The isomorphism is pretty straightforward by noticing the degree of each vertex in the left and right graph. 7 clearly gets mapped to D, 6 gets mapped to G, 5 gets mapped to F, 4 gets mapped to C, 3 gets mapped to B, 2 gets mapped to A and lastly 1 gets mapped to E.

Hence the isomorphism labels are given by D, G, F, C, B, A, E. Similarly for the other graphs, we can do the same way.

3.
Version 1

```
Capacity: 6

item    1   2   3   4

value   3   1   2   2

weight  2   1   1   3

--------------------

K[0]    0   0   0   0

K[1]    0   1   2   2

K[2]    3   3   3   3

K[3]    3   4   5   5

K[4]    3   4   6   6

K[5]    3   4   6   6

K[6]    3   4   6   7
```

Version 2

```
Capacity: 6

item    1   2   3   4

value   4   1   2   2

weight  3   1   1   3

---------------------

K[0]    0   0   0   0

K[1]    0   1   2   2

K[2]    0   1   3   3

K[3]    4   4   4   4

K[4]    4   5   6   6

K[5]    4   5   7   7

K[6]    4   5   7   7
```

Version 3

```
Capacity: 6

item    1   2   3   4

value   4   1   2   2

weight  3   1   3   1

---------------------

K[0]    0   0   0   0

K[1]    0   1   1   2

K[2]    0   1   1   3

K[3]    4   4   4   4

K[4]    4   5   5   6

K[5]    4   5   5   7

K[6]    4   5   6   7
```