1a. 4

1b. 2/3

1c. 2/3

2a. 99

2b. You will want to use triangular matrix, because tabular approach beats out triangular when 33% of all pairs have a nonzero count, in this scenario it is 10%.

3a.

|  |  |  |
| --- | --- | --- |
|  | Absolute | Relative |
| {1} | 4 | 1/3 |
| {1,2} | 2 | 1/6 |
| {1,3} | 3 | 1/4 |
| {1,4} | 2 | 1/6 |
| {1,5} | 1 | 1/12 |
| {1,6} | 0 | 0 |
| {2} | 6 | 5/12 |
| {2,3} | 3 | 1/4 |
| {2,4} | 4 | 1/4 |
| {2,5} | 2 | 1/6 |
| {2,6} | 1 | 1/12 |
| {3} | 8 | 7/12 |
| {3,4} | 4 | 1/3 |
| {3,5} | 4 | 1/3 |
| {3,6} | 2 | 1/6 |
| {4} | 8 | 7/12 |
| {4,5} | 3 | 1/4 |
| {4,6} | 3 | 1/4 |
| {5} | 6 | 1/2 |
| {5,6} | 2 | 1/6 |
| {6} | 4 | 1/3 |

3b. [0, 5, 5, 3, 6, 1, 3, 2, 6, 3, 2]

3c. 1, 2, 4, 8 (0 based array above)

3d. {3 ,4}, {2,4}, {1,2}, {4,5}, {4,6},{1,4}, {3,5}, {1,4}, {3,5}, {2,4}, {5,6},{2,6}

4. The main point of the article is talking about documenting fingerprinting in the digital world. This is when you partially copy someone else’s work and use it in your own. One major point discussed are the three properties a copy-detection algorithm should contain. It should have whitespace insensitivity, noise suppression against grammar articles, and position independence which is adding or removing to the text will not change the fact it was copied. Other major points discussed were the matching algorithms discussed. They discussed Karp-Robin string matching that finds substring matches and discussed the weakness. The article also discussed All-to-all matching which using the ASCII Table to find matches. The main point and algorithm discussed is the winnowing algorithm. When given a set of documents it will look to satisfy two properties. One being if there is a substring match at least as long as the guaranteed threshold then there is a match. The other property being there are no matches shorter than the noise threshold. Within the winnowing algorithm they made a few points, one being the expected density. The density is the expected fraction of fingerprints from the hash values and discussed good hashes (0 mod p), and bad hashes. The next main point discusses queries and how to choose hashes well. It discusses how the application will build a database, query the fingerprints from each document against the database, and then how winnowing allows them to treat the two fingerprinting times differently and be flexible to use it as needed. The last main topic discussed before going into the experiments were local algorithms. They use local algorithms to see if they perform better than winnowing. They do this by proving the lower bound for the density of the local algorithm can be improved. It is interesting how there is always new algorithms being though of, produced, and tested for not only plagiarism but for all forms of software that requires efficient code. It leaves me wondering how relevant these algorithms will be in a few years. Will winnowing even be considered efficient in 5 years?