1. **Phrase search**

Given is a part of a positional index in the form ‘term: docId1: posList1; docId2: posList2; …’:

- **angels**: 2: (36,174,252,651); 4: (12,22,102,432); 7: (17);
- **fools**: 2: (1,17,74,222); 4: (8,78,108,458); 7: (3,13,23,193);
- **fear**: 2: (87,704,722,901); 4: (13,43,113,433); 7: (18,328,528);
- **in**: 2: (3,37,76,444,851); 4: (10,20,110,470,500); 7: (5,15,25,195);
- **rush**: 2: (2,66,194,321,702); 4: (9,69,149,429,569); 7: (4,14,404);
- **to**: 2: (47,86,234,999); 4: (14,24,774,944); 7: (199,319,599,709);
- **tread**: 2: (57,94,333); 4: (15,35,155); 7: (20,320);
- **where**: 2: (67,124,393,1001); 4: (11,41,101,421,431); 7: (16,36,736);

1. Which documents (if any) are relevant to the following phrase queries?
   a. ‘fools rush in’
   b. ‘fools rush in’ AND ‘angels fear to tread’

2. Use the index to reconstruct (parts of) the content of document 2.

3. Create a part of the bi word index that is required to compute search results for the first query.

4. Assume the terms ‘in’ and ‘to’ are filtered out as stop words. Propose a way to combine a positional index with stop word filtering in a search system.

2. **Skip-Lists**

Given is the query ‘ice AND age’ together with the posting lists of both terms:

- **ice**: (4, 6, 10, 12, 14, 16, 18, 20, 22, 32, 47, 81, 120, 122, 157, 180)
- **age**: (47)

How many posting elements and/or pointers need to be compared in the posting list intersection for answering the query, using:

1. Traditional posting lists
2. Posting lists with skip pointers

Explain your answer!