Let a **dab** be a set of disjoint segments on the real axis.

(a) Write a **Dab** class in Java that supports equality testing between two dabs in linear time. Provide only the data members without any constructors or methods, and highlight any noteworthy features and invariants.

*Note:* You may not use any pre-made lists, resizable arrays or other library collections of any kind; use only integers, doubles, pointers, arrays and classes. You are allowed to define additional classes if necessary. [4 marks]

(b) Draw a records-and-pointers diagram representation of the following dab

\[ \{(-2.3, -1), (10, 24.53), (2, 6)\} \]

using your class from part (a). [2 marks]

(c) Given a set \( S \) of dabs, find an algorithm that returns a dab of maximum cardinality containing only segments from the dabs in \( S \). The algorithm should run in \( O(n^2) \) time, where \( n \) is the total number of segments of all dabs contained in \( S \).

(i) Clearly describe and explain your algorithm. [4 marks]

(ii) Describe your algorithm in a few lines of pseudocode. [4 marks]

(iii) Prove that your algorithm is correct. [4 marks]

(iv) Derive the asymptotic running time of your algorithm. [2 marks]