The purpose of this problem set is to gain familiarity with heaps and priority queues.

The Internet Movie Database (http://www.imdb.com) is an online database of information related to movies, tv programs, and videos. You can download plain text data files; directions are at http://www.imdb.com/interfaces.

In this exercise we will work with movie ratings. I have used the files title.basics.tsv and title.ratings.tsv to produce the file ratings.tsv, which contains one line of tab-separated values for each movie that has been rated. The first five lines (of 260,191) of the file are:

8 4.5 Bohemios
623 6.1 The Story of the Kelly Gang
14 4.5 Robbery Under Arms
11 3.2 Hamlet
9 4.9 Don Quijote

The first value on each line is the number of ratings that the movie received, and the second value is the average score. Scores range from 1 (worst) to 10 best. After the second tab comes to the title of the movie.

Write a program called MovieRanker that reads in the data in ratings.tsv and responds to queries of the following form. The user inputs two integers, say k and m, with $0 \le k \le 2 \times 106$ and $0 \le m \le 105$. Your program should then list, in decreasing order of rating, the m highest rated movies among those that had at least k reviews. Your program should loop, accepting input and producing lists until at least one of the input numbers is 0. After each list, it should report the time in milliseconds that it took to produce the list. A screenshot appears on the next page.

Several files are on Vocareum, in addition to ratings.tsv. The file MaxHeap.java contains an implementation of a priority queue based on a max heap. MovieRating.java is a class for storing a record of movie information, and MovieRanker.java contains starter code with the required input and output statements.