

Übungsblatt 10

Abgabe bis Mittwoch, den 7. Juli um 16 Uhr

Aufgabe 1 (10 Punkte)

Eine *Anagrammgruppe* ist eine Menge von Wörtern, so dass jedes Wort in der Menge ein Anagramm von jedem anderen Wort in der Menge ist. Eine Anagrammgruppe ist maximal für ein gegebenes Wörterbuch, wenn es keine echte Obermenge gibt, die im Wörterbuch liegt und immer noch eine Anagrammgruppe ist. Die maximalen Anagrammgruppen bilden somit eine Partitionierung des Wörterbuches.

Zum Beispiel ist für das Wörterbuch vom letzten Übungsblatt, die Menge aus den Worten *Ostern*, *Sorten* und *Tensor* eine Anagrammgruppe, aber keine maximale. Die Menge aus den Worten *Ostern*, *Ortens*, *Sorten*, *Tenors*, *Tensor* und *rosten* ist eine maximale Anagrammgruppe. In dieser Terminologie war es die Aufgabe des letzten Übungsblattes für ein gegebenes Wort die maximale Anagrammgruppe zu berechnen, die dieses Wort enthält.

Ziel dieser Aufgabe ist ein Programm *AnagramGroupsMain*, das für dasselbe Wörterbuch wie vom letzten Übungsblatt und für eine gegebene positive Zahl k die k größten maximalen Anagrammgruppen ausgibt (eine Anagrammgruppe pro Zeile, mit allen Worten der Gruppe und der Größe der Gruppe).

Seien Sie so nett und committen Sie die *dictionary.txt* Datei nicht erneut in unser SVN, sondern greifen Sie auf die Datei in Ihrem Unterordner *uebungsblatt-9* vom letzten Übungsblatt zu.

Wie gehabt sollen Sie das Programm natürlich mit einer sinnvollen Klassen realisieren und in der *AnagramGroupsMain.cpp* sollten nur wenige Zeilen Code stehen, die von dieser Klasse Gebrauch machen. Das in der Vorlesung behandelte Sortieren mit beliebiger Vergleichsfunktion wird Ihnen dabei hilfreich sein. Und wie immer sollen Sie natürlich für jede Methode einen guten Test schreiben.

Committen Sie wie gehabt alles ins SVN, in ein neues Unterverzeichnis *uebungsblatt-10*, stellen Sie sicher, dass alles kompiliert, die Tests ohne Fehler laufen, Lint ohne Fehler durchläuft und auch auf Hudson alles funktioniert. Und die *erfahrungen.txt* nicht vergessen.