**BlastX**

Eingabeparameter:

fastq, fasta, seed,size\_alp,numb\_alp

//Files reads.fq und database.fa einlesen und speichern

#include <seqan/seq\_io.h>

-int readFasta(seq, filename) seqan Funktion

-readRecord(id, seq, seqStream) seqan Funktion

//Alphabete festlegen

-getAlphabets(size\_alp,num\_alp)

->knearstneighbor

RETURN StringSet<String<int>> codierung

//Reads in Aminosäuren übersetzen mit 6 frames und verschiedenen Alphabeten

-sixFramesToAlphabet(reads, alphabet) selbst implementieren

RETURN String<CharString> newReads

//Datenbank in entsprechendes Alphabet übersetzen

-translateDatabase(database, alphabet) selbst implementieren

RETURN String<CharString> newDatabase

//Reads mit Datenbank vergleichen und matches speichern

#include <seqan/find.h>

#include <seqan/index.h>

-findMatches(reads, database,seed) selbst implementieren

->find(finder, pattern\_seeds) seqan Funktion

->Finder<TIndex, Backtracking<TDistance>>

->Pattern<TNeedle, Backtracking<TDistance>>

RETURN String<Int> matches

//Ergebnisse mit Scoringmatrix verifizieren (evtl. mit einem treshold)

#include <seqan/score.h>

-verifyMatches(matches, blossum62) selbst implementieren

RETURN String<CharString> matches, String<int> scores

//Matches und deren Scores in eine text file schreiben

-writeToFile(filename, matches, scores) selbst implementieren