

Übungen zum Bioinformatik-Tutorium

Blatt 8

Termin: Dienstag, 11.12.2018, 11 Uhr

1. Proteom-Debugging

Importiere die Datei `/home/proj/tutorium_bioinformatik/DebugProteome.java` in dein Eclipse-Projekt. Korrigiere zuerst alle rot unterringelten Compiler-Fehler (gegebenenfalls mit Hilfe der Quick-Fix Funktion). Verwende anschließend den Debugger, um alle semantischen Fehler zu verbessern.

Rufe `DebugProteome` mit der Datei `/home/proj/tutorium_bioinformatik/C_elegans.pep.all.fa` und leite seine Ausgaben in eine Datei `test.fasta` um. Verwende anschließend das Shell-Kommando `diff`, um `test.fasta` mit `/home/proj/tutorium_bioinformatik/C_elegans.pep.all_check.fa` zu vergleichen (die beiden Dateien sollten gleich sein).

```

import java.io.*;
import java.util.ArrayList;

public class DebugProteome {
    private ArrayList<Protein> proteins = new
        ArrayList<Protein>();

    public DebugProteome(File fastaFile){
        try{
            BufferedReader br = new BufferedReader(new
                FileReader(fastaFile));

            String line;
            StringBuilder sb = new StringBuilder();
            String id = "";
            while ((line=br.readLine())!=null) {
                if (line.startsWith(">")) {
                    if (sb.length()>0) {
                        proteins.add(new Protein(id,sb.toString()));
                        sb = new StringBuilder();
                    }
                    id = line.substring(1);
                }
                else {
                    sb.append(line);
                }
            }
            proteins.add(new Protein(id,sb.toString()));
            br.close();
        }catch(IOException e) {
            e.printStackTrace();
        }
    }

    public int getProteinCount() {
        return proteins.size();
    }

    public Protein getProtein(int i) {
        return proteins.get(i);
    }
}

```

```

public static void main(String [] args) {
    if (args.length==0) {
        System.err.println("Usage:Proteome <fasta-file >");
        System.exit(1);
    }
    File fastaFile = new File(args[0]);
    if (!fastaFile.exists()) {
        System.err.println("File not found!");
        System.exit(1);
    }
    DebugProteome p = new DebugProteome(fastaFile);
    for (int i=0; i<p.getProteinCount(); i++) {
        Protein protein = p.getProtein(i);
        System.out.println(">" + protein.getId() + "\n" +
                           protein.getSequence());
    }
}
}

```

2. ScopeTest2-Debugging

Überlege dir zuerst, was wohl am Ende auf der Konsole ausgegeben wird.

Verwende dann den Debugger um `/home/proj/tutorium_bioinformatik/ScopeTest2.java` nachzuvollziehen und mit der eigenen Lösung zu vergleichen.

```

7
4
7
3
7
1

```