Grundlagen der Bioinformatik

Exercises – Assignment 3

Oğuz Şerbetci

Competition

Competition	Assignment 1			Assignment 2			
		sequence (58M)	chr1 (240M)			pair_long (38K)	chr1 (240M)
Group 1	rust	0m0.226s	0m0.613s				
Group 2				cpp	CORRECT	0m24.483s	–
Group 4	cpp	0m0.105s	0m0.250s	cpp	CORRECT	0m0.073s	0m30.925s
Group 11				rust	CORRECT	0m1.922s	–
Group 12	rust	0m0.516s	0m2.025s	rust	CORRECT	0m0.438s	0m19.041s
Group 14	java	0m2.764s	0m10.850s	c	CORRECT	0m5.281s	–
Group 16	python	2m6.656s	8m39.437s				

Overview

- Compute optimal global alignments between protein sequences using a BLOSUM scoring matrix
- Implement hierarchical clustering to compute phylogenetic trees
- Perform some BLAST searches

Task 1.1

- In moodle, you find two files
  - blosum62.txt (BLOSUM62 scoring matrix)
  - sequence_pair.fasta (two protein sequences)
- Write a program with two parameters
  - Path of a FASTA file with two protein sequences
  - Path of a file containing a scoring matrix (same format as blosum62 from above)
- Output the global alignment score between the two sequences using the scoring matrix
  - No need to output alignments
  - Scoring matrix must be loaded and must not be hardcoded
  - For the provided files, result should be 2216.
- Submission: program called "**alignwithmatrix**"

Task 1.2

- In moodle, you find another file: sequences.fasta
- Write a program that takes two parameters
  - Path of a FASTA file with many protein sequences
  - Path of a file containing a scoring matrix
- Program should output
  - The similarity matrix M of all pairs of sequences using global alignment with the given scoring matrix
    - Use metadata in FASTA file as row / column header
  - The guide tree as computed by hierarchical clustering of M
    - Number sequences in rising order (1, 2, 3 …) as in file
    - If node i is merged with node j – output (i, j) and name new node "i+j"
      - Output every merge in a new line, labels in alphabetical order (not (3,2))
      - Inner nodes closer to the root get longer sum terms, ala "i+j+k+l"
- Submission: program called "**guidetree**"

Example

- Input

```
> HS
AGGTAGAC
> MM
AGGTGACT
> RN
TGGAGACT
```

- Scoring matrix M (numbers are arbitrary)

```
    HS     MM     RN
HS         32     27
MM                36
RN
```

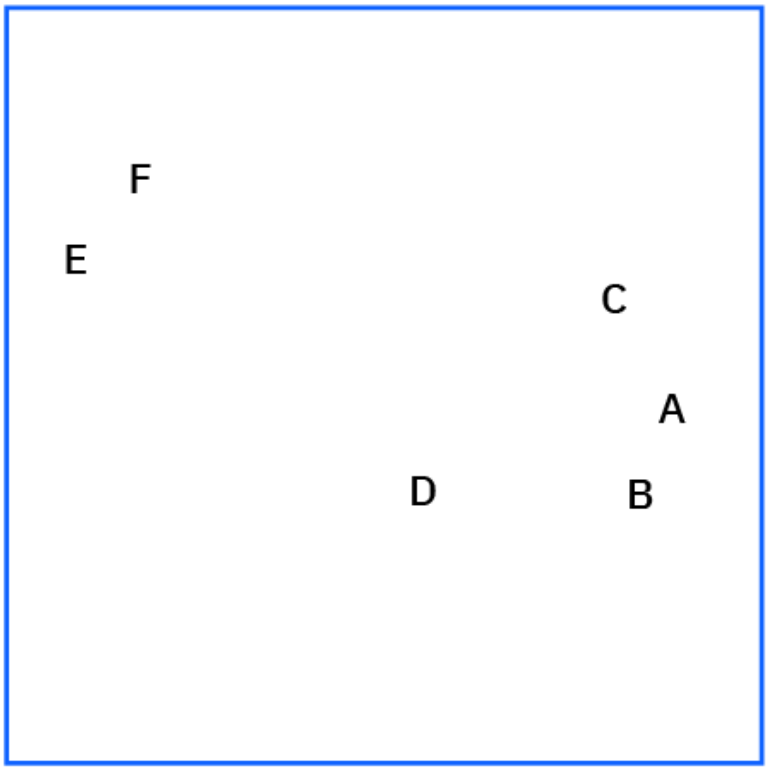
Example continued

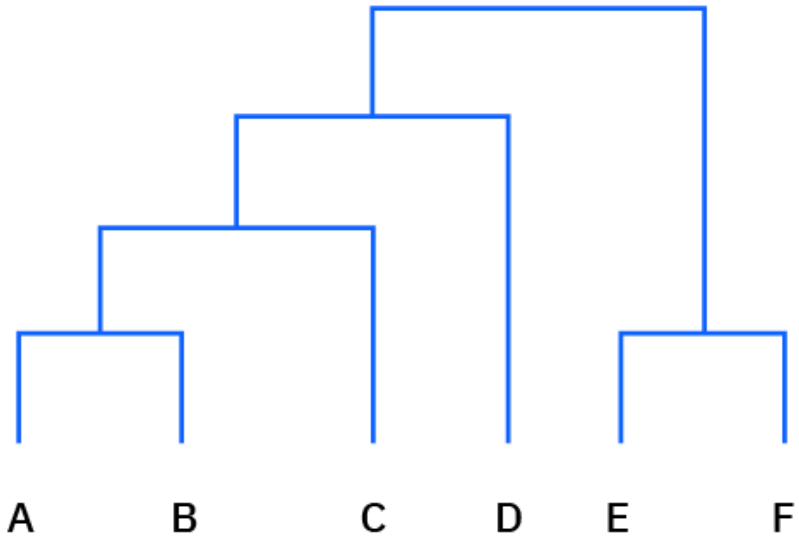
- Scoring matrix M (numbers are arbitrary)

	HS	MM	RN
HS		32	27
MM			36
RN			

- Guide Tree (HS=1, MM=2, RN=3)
  - (1, 3)
    (1+3, 2)

Example





Competition

- Compute the guide tree as fast as possible
  - Use whatever tricks you find
  - Implementation may be different from solution to Task 1.2
  - Output: Only the guide tree (no scoring matrix etc.)
  - Submission (voluntarily): program called "**guidetree_competition**" with two parameters
- We will measure wall clock time (unix time) for a few new file of protein sequences

Things you may consider for speeding-up alignment

- Recall: Complexity is $O(k^2*n^2+k^2*\log(k))$

- Compute global alignments efficiently ($O(k^2*n^2)$)
- Compute guide tree in $O(k^2*\log(k))$
  - Efficient management of matrix – you need to delete rows / columns and to add rows / columns
  - Finding the currently largest value in the matrix in $O(\log(k))$
- We will measure with many rather short sequences

Task 2.1: Run BLAST

- OMIM is a database of hereditary diseases in humans
- Search the disease PHENYLKETONURIA
  - Give a 10 line description of the disease: diagnosis, symptoms, therapy, prognosis
- Extract the sequence of the disease-causing protein from UniProt (use links on page)
  - Give the sequence in FASTA format
- Run BLAST (at ncbi) in default settings to find best matching other proteins
  - Extract and give most similar sequences in chimpanzee, gorilla, tufted capuchin (Kapuziner Affe), arabian camel, alpaca, horse, Canada lynx (Luchs), and domestic cat
  - Copy into FASTA file in this order; put human sequence first

Task 2.2. Compute Guide Tree

- Compute the guide tree using your program from task 1.2
  - Output the tree in pre-defined format
- Provide all outputs in one PDF
  - The 10-line description of the disease
  - The FASTA sequences of all species in correct order
  - The guide tree

General requirements

- Remember to name all programs as requested
- All programs must run without further installations on GRUENAU2
  - *ssh username@gruenau2.informatik.hu-berlin.de*
- For all programs, source code must be submitted as well
  - Document your code
  - For Java/C etc.: Submit the source code and the compiled binary
- All responses must be submitted as PDF, where the task /assignment of every answer is clearly recognizable
- Zip everything into one file per task and upload via Moodle
  - **AssignmentX_groupY_taskZ.zip**
- Deadline for submissions: Tuesday 24.06.2025, 23:59

Questions?