

C in Bioinformatics

Tools and Projects

Gang Chen
chengang@genomics.cn

September 26, 2015

Outline

- 1 A project: Sequence Alignment in C
- 2 Bioinformatics Software in C

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - Implementation
 - Project Management: make and git
- 2 Bioinformatics Software in C

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - Implementation
 - Project Management: make and git
- 2 Bioinformatics Software in C
 - seqtk
 - bwa
 - libsvm

Overview

Smith-Waterman Algorithm

The algorithm is proposed by Temple F. Smith and Michael S. Waterman to perform local sequence alignment.

News

Michael Waterman has become the first honorary professor of the Chinese University of Hong Kong, Shenzhen

<http://www.cuhk.edu.cn/News/142.html>

Algorithm

Example

- Sequence 1: ACACACTA
- Sequence 2: AGCACACA
- Similarity scoring scheme:
 - $s(a, b) = 2$ if $a = b$
 - $s(a, b) = -1$ if $a \neq b$
- Gap score: -2

Algorithm

$$H = \begin{pmatrix} - & A & C & A & C & A & C & T & A \\ - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & 0 & 2 & 1 & 2 & 1 & 2 & 1 & 0 & 2 \\ G & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 \\ C & 0 & 0 & 3 & 2 & 3 & 2 & 3 & 2 & 1 \\ A & 0 & 2 & 2 & 5 & 4 & 5 & 4 & 3 & 4 \\ C & 0 & 1 & 4 & 4 & 7 & 6 & 7 & 6 & 5 \\ A & 0 & 2 & 3 & 6 & 6 & 9 & 8 & 7 & 8 \\ C & 0 & 1 & 4 & 5 & 8 & 8 & 11 & 10 & 9 \\ A & 0 & 2 & 3 & 6 & 7 & 10 & 10 & 10 & 12 \end{pmatrix}$$

Algorithm

$$T = \begin{pmatrix} - & A & C & A & C & A & C & T & A \\ - & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ A & 0 & \swarrow & \leftarrow & \swarrow & \leftarrow & \swarrow & \leftarrow & \swarrow \\ G & 0 & \uparrow & \swarrow & \uparrow & \swarrow & \uparrow & \swarrow & \uparrow \\ C & 0 & \uparrow & \swarrow & \leftarrow & \swarrow & \leftarrow & \swarrow & \leftarrow \\ A & 0 & \swarrow & \uparrow & \swarrow & \leftarrow & \swarrow & \leftarrow & \swarrow \\ C & 0 & \uparrow & \swarrow & \uparrow & \swarrow & \leftarrow & \swarrow & \leftarrow \\ A & 0 & \swarrow & \uparrow & \swarrow & \uparrow & \swarrow & \leftarrow & \swarrow \\ C & 0 & \uparrow & \swarrow & \uparrow & \swarrow & \uparrow & \swarrow & \leftarrow \\ A & 0 & \swarrow & \uparrow & \swarrow & \uparrow & \swarrow & \uparrow & \swarrow \end{pmatrix}$$

- A-CACACTA
- AGCACAC-A

Data Structure

I/O

- Input:
 - Two sequences: two string
 - Similarity scoring scheme
 - Gap score
- Output: alignment results;
- Internal: score matrix;

Files

- swa.h: constant and function declarations;
- swa.c: function definitions;
- Makefile: ?

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - **Implementation**
 - Project Management: make and git

- 2 Bioinformatics Software in C
 - seqtk
 - bwa
 - libsvm

swa.h

see swa/swa.h

swa.c

see swa/swa.c

compile and run

```
gcc swa.c -o swa ./swa
```

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - Implementation
 - Project Management: make and git
- 2 Bioinformatics Software in C
 - seqtk
 - bwa
 - libsvm

GNU Make

GNU Make

GNU Make is a tool which controls the generation of executables and other non-source files of a program from the program's source files.

Website

<http://www.gnu.org/software/make/>

Makefile for swa

```
1 swa: swa.c swa.h
2      gcc swa.c -o swa -Wall
3 clean:
4      rm swa
```


Tutorials on Makefile

- Tutorial in English: <http://makepp.sourceforge.net/>
- Official Manual: <http://www.gnu.org/software/make/manual/make.html>
- 跟我一起写Makefile : <http://blog.csdn.net/haoel/article/details/2886>
<http://coolshell.cn>

Git

- Git: <http://www.git-scm.com/>
- SourceTree: <https://www.sourcetreeapp.com/>
- GitHub Desktop: <https://desktop.github.com/>
- How to get the latest sources of GNB5010?

Project Management

- Source Version Control: git and github;
- Generation of Makefile: automake
- Documentation: Markdown

Next

- 1 A project: Sequence Alignment in C
- 2 **Bioinformatics Software in C**
 - seqtk
 - bwa
 - libsvm

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - Implementation
 - Project Management: make and git
- 2 Bioinformatics Software in C
 - seqtk
 - bwa
 - libsvm

Overview

seqtk

Toolkit for processing sequences in FASTA/Q formats.

Source Codes

<https://github.com/lh3/seqtk>

Files

- Makefile
- README.md
- khash.h
- kseq.h
- ksort.h
- kstring.h
- ksw.c
- ksq.h
- kvec.h
- seqtk.c
- trimadap.c

Compile and Run

```
make
```

```
./seqtk
```


How to read the codes?

Makefile

For most C/C++ projects, the best start point to understand the codes is the Makefile.

Why?

We have to define the relationship between all source files and executable files in the Makefile.

Makefile

```
1 CC=gcc
2 CFLAGS=-g -Wall -O2 -Wno-unused-function
3
4 all:seqtk trimadap
5
6 seqtk:seqtk.c khash.h kseq.h
7         $(CC) $(CFLAGS) seqtk.c -o $@ -lz -lm
8
9 trimadap:trimadap.c kseq.h ksw.h
10        $(CC) $(CFLAGS) ksw.c trimadap.c -o $@ -lz -lm
11
12 clean:
13        rm -fr gmon.out *.o ext/*.o a.out seqtk trimadap *~ *.a *.dSYM session*
```

seqtk.c

see [seqtk-master/seqtk.c](https://github.com/hengshen/seqtk-master/seqtk.c)

Convert FASTQ to FASTA

```
./seqtk seq seq.fq > seq.fa
```

Data

<http://trace.ncbi.nlm.nih.gov/Traces/sra/?run=SRR357733>

main and stk_seq functions

- main: see seqtk-master/seqtk.c:1371
- stk_seq: see seqtk-master/seqtk.c:1106

Summary

- Makefile can help us understand source codes;
- Typically, declarations are put in header files
- and implementation are put in source files;

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - Implementation
 - Project Management: make and git

- 2 Bioinformatics Software in C
 - seqtk
 - **bwa**
 - libsvm

Overview

BWA is a software package for mapping DNA sequences against a large reference genome, such as the human genome. It consists of three algorithms:

- BWA-backtrack: designed for Illumina sequence reads up to 100bp;
- BWA-SW: 70bp to a few megabase;
- BWA-MEM: 70bp to a few megabase, faster and more accurate;

Makefile

see bwa-master/Makefile

main function

- see bwa-master/main.c:60
- Call main_mem: bwa-master/main.c:83
- main_mem declaration: bwa-master/main.c:24

main function

- see bwa-master/main.c:60
- Call main_mem: bwa-master/main.c:83
- main_mem declaration: bwa-master/main.c:24
- Where is the function definition?

main_mem function

- Makefile
- bwamem.c, bwamem.h ...

main_mem function

- Makefile
- bwamem.c, bwamem.h ...
- fastmap.c

Next

- 1 A project: Sequence Alignment in C
 - Smith-Waterman Algorithm
 - Implementation
 - Project Management: make and git
- 2 Bioinformatics Software in C
 - seqtk
 - bwa
 - libsvm

Overview

LIBSVM

LIBSVM is an integrated software for support vector classification(C-SVC, nu-SVC), regression (epsilon-SVR, nu-SVR) and distribution estimation (one-class SVM).

Website

<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>

Overview of Codes

Source Codes

- A mixture of C and C++;
- All C codes are compiled by using C++ compiler;
- C++ is used as a C with class and template;

Makefile

see libsvm-3.18/Makefile

svm-train.c

- main function is defined in svm-train.c
- structs are defined in svm.h
- functions are declared in svm.h and implemented in svm.cpp
- classes and methods are declared and implemented in svm.cpp

svm-train.c

- main function is defined in svm-train.c
- structs are defined in svm.h
- functions are declared in svm.h and implemented in svm.cpp
- classes and methods are declared and implemented in svm.cpp
- the coding style is bad.

C and C++

- C++ is a upgrade of C?
- C++ is C with class?
- C++ is C with class, template, meta-programming and so on?
- What is the relationship between C and C++?

C and C++

- C++ is a new programming languages;
- Currently, C++ is compatible with C;
- The latest version of C is C99, C11 is in progress;
- The latest version of C++ is C++14, C++17 is in progress.

Thanks!