

컴퓨터알고리즘 9주차

2016112158 김희수



[날짜]

[회사 이름]

[회사 주소]

이번 과제에서 수행해야할 작업을 세가지로 나누면 1. Reference DNA 생성 2. ShortRead 생성 3. ShortRead로부터 trival method를 사용해 myDNA를 복원으로 나눌 수 있다.

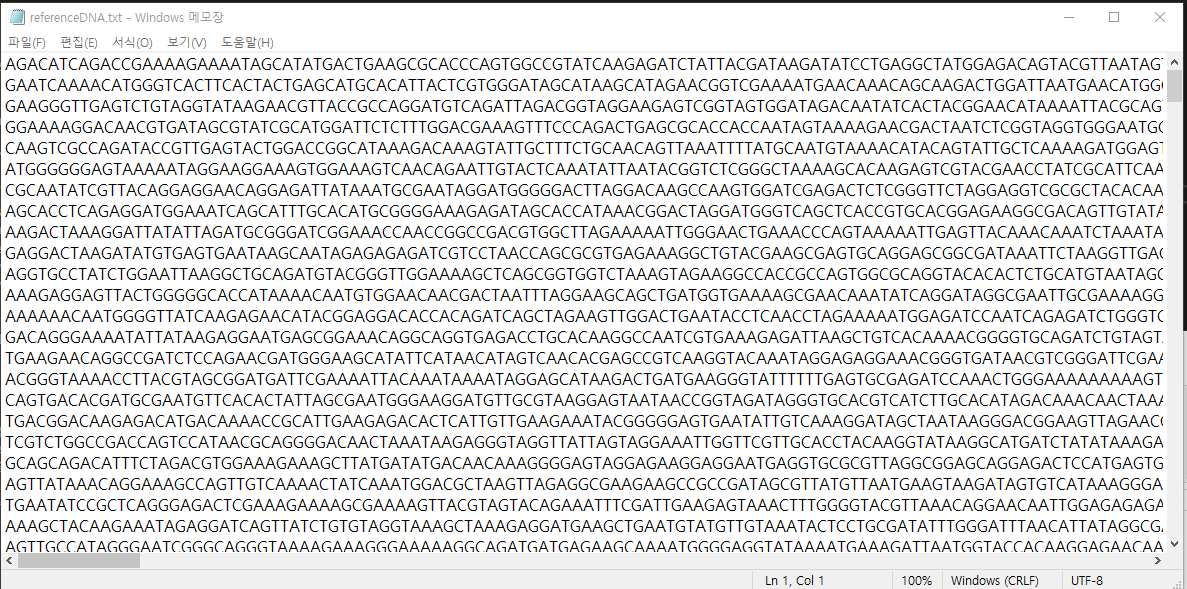
※ 처음 수행시 make\_refDNA와 make\_shortRead의 주석을 지워주세요. 또한 비주얼 스튜디오는 디버그 모드가 아닌 Release모드로 수행해주세요.

1. referenceDNA 생성

refDNA 생성시 rand() 함수를 사용하면 DNA의 반복정도가 심해진다. 따라서 선형항동방식이 적용된 rand()를 사용하지 않고 메르센 트위스터 알고리즘이 적용된 난수를 생성하겠다. 시드를 주기 위해 random\_device를 만들고, mt19937로 난수 생성엔진을 생성한 뒤, 균등분포 distribution을 정의한다. distribution(gen)은 균등분포에서 랜덤하게 뽑힌 수이다.

다만 의아한 점은 “DNA의 반복정도가 심해졌다”의 기준이 무엇인가이다. 어느정도의 반복이 발생하면 반복정도가 심해진 것인가? mt19937을 이용하지 않는 것이 치우침이 덜하다는 것만 알고 있을뿐이다.

(생성된 referenceDNA)



2. shortRead를 생성하기 위해 함수 make\_shortRead를 정의하였다. 이 함수의 조건과 그 충족은 다음과 같다

1) 길이 k와 개수 n은 동적으로 작동할 수 있어야함

파라미터로 k, n을 받아왔다.

2) refDNA의 임의의 위치에서 자른다

1에서 사용한 mt19937을 사용해서 난수엔진 gen을 만들고 균등분포 dis(0, refDNA.length-k)에서 랜덤하게 뽑아온다. dis(gen)은 refDNA안의 임의의 시작위치이다. 이 시작위치에서 길이가 k가 되도록 자른 것이 잠정 read인 temp\_read이다. 왜 잠정이냐면 3)을 수행해야하기 위한 것이기 때문

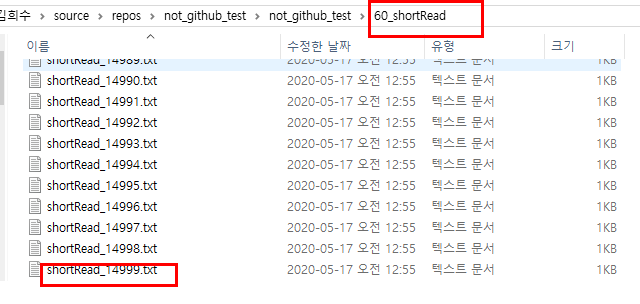
3) 자르는 과정에서 20% 확률로 2개의 문자가 기존문자와 달라야한다(돌연변이가 있어야한다)

20%확률이란건 5개 중 한번 나타난다는 것이다. 이를 위해 20%확률로 랜덤하게 추출할 균등분포 sample(0,4)를 생성하자. 그후, temp\_read를 5개씩 잘라 splited\_temp배열에 저장한다. 예를 들어 길이 10인 temp\_read(AGACTGACTA)라 하겠다. 5개씩 자르면 2개의 원소로 구성된 splited\_temp(AGACT GACTA)가 된다. 이제 돌연변이가 일어날 index를 설정하자. sample(gen)으로 index가 설정된다. 만약 splited\_temp[0]에 대해서 index = 3이면 C가 돌연변이가 된다. C의 돌연변이를 위해 shuffle함수를 정의했다. shuffle함수는 매개변수로 주어진 배열에서 임의의 원소를 랜덤하게 추출하는 함수이다. 이렇게 돌연변이가 일어난 splited\_temp의 모습은 (AGATT GTCTA)라고 하자. 이 splited\_temp의 원소들을 이어붙여주자. 그 결과가 shortRead이다.(이 과정에서 concat함수가 사용됬다. concat함수는 스트링배열의 원소들이 이어붙여진 스트링을 반환한다)

(k = 30, n = 20000의 shortRead)



(k = 60, n = 15000의 shortRead)



3. trivial method를 활용한 myDNA 복원

~~myDNA복원은 자바로 작성되었다. C++로 작성하니 k = 60, n = 15000의 경우가 너무 오래걸렸다. k = 30, n = 20000의 경우 C++로 실행하였다.~~  비주얼 스튜디오 내에서 디버그 모드가 아닌 Release모드로 수행하니 C++로도 빠르게 수행되었다.

trivial method는 두 함수로 이루어졌다. 먼저 몇번째로 만들어진 shortRead가 refDNA의 어느 위치에 존재하는 가를 위한 brute\_force\_matching함수이다. shortRead와 refDNA사이의 일치하지 않는 정도를 나타내는 mismatchex를 설정해 mismathces가 threshold보다 작으면 그 때의 인덱스 i를 리턴한다.

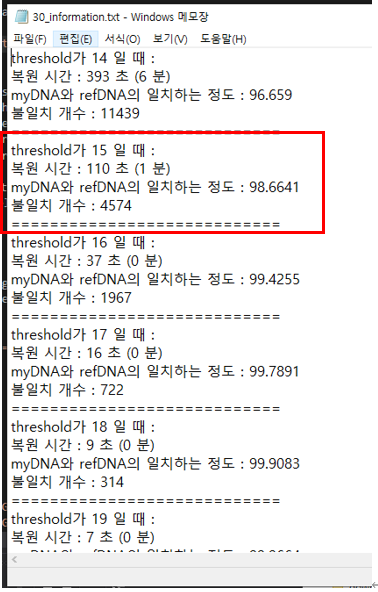
trivial\_Mapping함수는 본격적으로 myDNA의 복원이 수행되는 함수이다. 매개변수로 shortRead들의 vector인 shortReads를 받는다. 각각의 shortRead에 대해서 그 shortRead의 refDNA내의 위치를 변수 index에 저장하고 myDNA를 바꿔준다. myDNA에서 index ~ (index+ Read의 length)에 해당하는 부분을 shortRead로 바꿔주는 것이다. 이 과정을 전부 수행하면 myDNA가 복원된다.

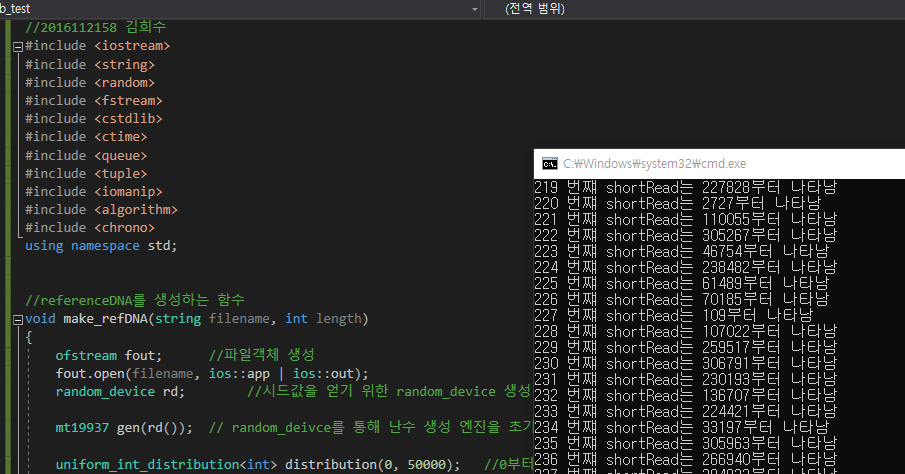
refDNA와 복원된 myDNA의 비교를 위해 일치하는 정도를 백분율로 나타내고 불일치하는 문자(A,G,C,T 뉴클레오타이드) 개수를 튜플로 반환하는 compare\_degree함수를 정의했다.

복원 과정에서 threshold를 잘 입력해줘야한다. (k=30, n=20000)의 경우를 수행할 때 threshold가 14보다 작으면 String subscript of range 에러가 발생하였다. threshold를 20이상으로 주면 불일치개수가 너무 적어졌다.

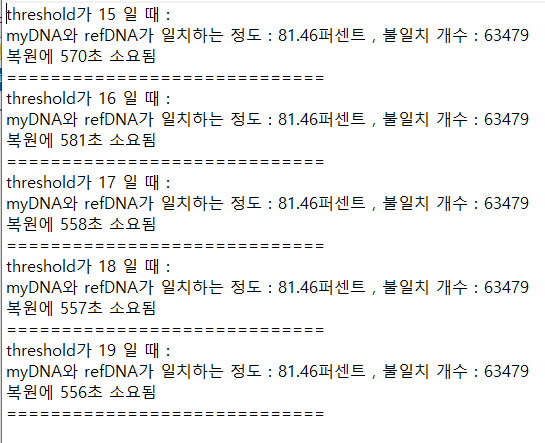
shortRead가 refDNA의 어느위치에서 나타나는 가를 콘솔창에 나타냈고, 복원시관, myDNA와 refDNA가 일치하는 정도, 불일치개수는 information.txt파일로 생성했다. 예를 들어 k=30이고 threshold가 14으로 설정했을때 생성되는건 30\_14\_myDNA.txt이다. 30\_information.txt에는 k=30일때, 각각의 threshold에 대한 일치정도, 불일치개수, 복원시간을 출력했다.

(k=30, n=20000, threshold = 15일때의 복원)

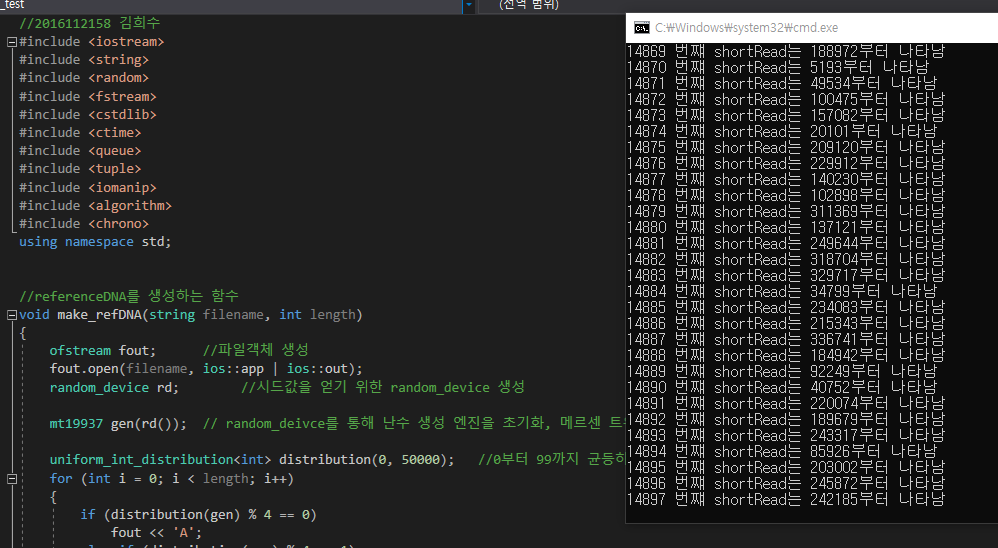


실행화면

(k = 60, n = 15000일떄 복원)



(실행화면)



프로젝트에는 k = 30일 때 만들어진 shortRead들(30\_shortRead 디렉토리)과 myDNA(30\_myDNA 디렉토리), k = 60일 때 만들어진 shortRead(60\_shortRead 디렉토리)들과 myDNA(60\_myDNA). 그리고 referenceDNA.txt가 포함되어있다. (C++)라고 적힌 디렉토리는 C++코드를 실행했을 때 만들어진 myDNA들이고 (JAVA)라고 적힌 디렉토리는 자바코드를 실행했을 때 만들어진 파일이다