

2018년 2학기 알고리즘 과제 #3

■ 과제내용

▲ 문제 : Shortest Common Superstring (NP-complete Problem)

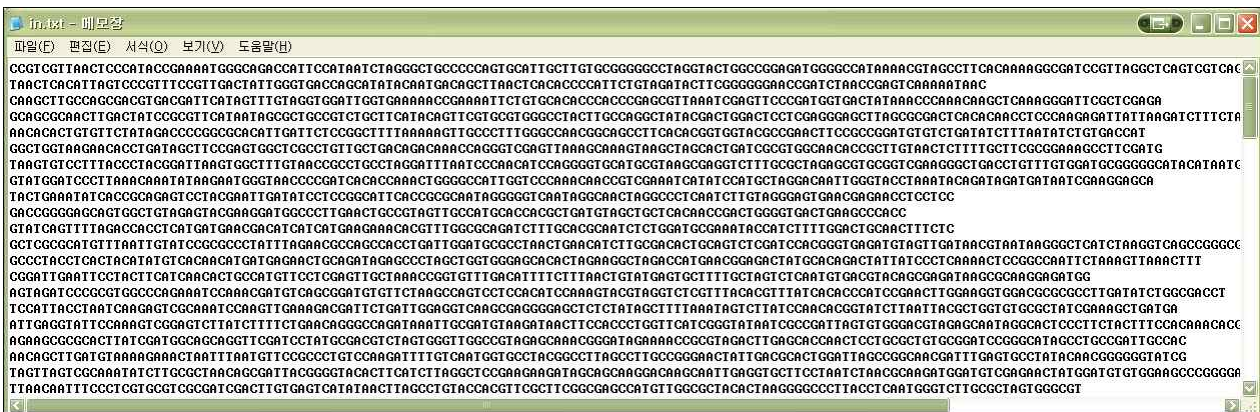
INSTANCE : Finite alphabet Σ , finite set R of strings from Σ^* , and a positive integer K .

QUESTION : Is there a string $w \in \Sigma^*$ with $|w| \leq K$ such that each string $x \in R$ is a substring of w , i.e., $w = w_0 x w_1$ where each $w_i \in \Sigma^*$?

- 요약 : 임의의 string 집합 R 내의 모든 string들을 substring으로 가지는 string (= "superstring") 중에서 되도록 짧은 길이의 것을 찾는 문제 (수업시간에 공지)
- 사용가능 기법 : **branch-and-bound, reducing another NP-complete problem, dynamic programming**
- 사용가능 언어 : C, C++를 위주로 함.
(Java 언어까지는 허용하나, 시간적으로 불리할 수 있음)

▲ 명세

- 입력(파일)
 - 파일제공 : "input.txt"
 - **DNA string 집합 R 이 기록된 파일**
 - $\Sigma = \{A, C, G, T\}$ ($\therefore |\Sigma| = 4$)
 - string의 개수($|R|$) : 20개
 - string x ($x \in R$)의 최대 길이 : 200
 - 예제 : "input.txt"



※ 한 line이 하나의 string을 나타냅니다. 따라서 "input.txt"는 $|R| (= 20)$ line입니다.

- 출력(파일)
 - 파일명 : "output.txt"
 - **입력(DNA string 집합 R)에 대해 자신이 얻은 가장 짧은 superstring w 가 기록된 파일**

- ※ 자신이 얻은 가장 짧은 superstring w 하나만 한 line으로 들어가 있어야 함
- 명세 이외에 답을 출력하면 오답처리입니다.

- 평가기준 : 임의로 생성한 "input.txt"에 대한 성능 평가
- 등수를 매겨 1등부터 높은 점수를 부여할 예정

평가항목	평가내용	설명
1-1	reasonable한 시간 내에 답을 얻을 수 있는지 검증	프로그램은 반드시 30분 내에 종료되어야 합니다. 30분 내에 결과를 출력하지 못할 경우 평가항목 2, 3의 대상에서 제외됩니다.
1-2	superstring의 correctness 검증	"input.txt" 내의 모든 string을 substring으로 가지는지 여부를 먼저 검증합니다. 제대로 된 superstring이 아닐 경우 평가항목 2, 3의 대상에서 제외됩니다.
2	superstring w의 길이 측정	등수를 매기는 기본적인 기준입니다. 즉 짧을수록 높은 등수를 얻을 수 있습니다.
3	수행시간 측정	항목 3에서 길이가 동일한 그룹이 발생할 경우에는 그 수행시간이 짧을수록 동일한 그룹 내에서 높은 등수를 얻을 수 있습니다.

- 기본점수 처리기준

- 입·출력 형식이 과제의 명세와 달라 코드의 수정이 필요할 경우
- 컴파일 에러나 실행 중 에러가 발생하는 경우
- 평가항목 1-1 또는 1-2를 만족시키지 못하는 경우

- 미제출 처리기준

- 카피로 판단될 경우
- 제출된 과제의 성격에 따라 'F'학점이상의 징벌도 있을 수 있음

■ 과제 제출

▲ 제출기한 : 2018년 12월 25일 (화요일)

※ 2018년 12월 25일 (화요일) 자정까지 upload된 과제만 인정합니다.

▲ 제출해야 하는 결과물 : source, document, output.txt

1. source : 상세한 comment를 한글로 달 것
2. document
 - 구현환경 및 인터페이스 언급
 - **아이디어 설명**
 - 직접 **branch-and-bound** 기법을 사용하였을 경우 : 문제 접근 방법, {upper, lower} bound, constraint 등 branch-and-bound 기법을 위한 자신의 아이디어를 구체적으로 기술할 것
 - 다른 **NP-complete** 문제로 **reducing**하여 해결하였을 경우 : 어떤 과정을 통해서 어떤 문제로 reducing하였는지, reducing 후 어떤 방법으로 해결하였는지, 해결 후 다시 본 문제에 적용하는 방법 등을 구체적으로 기술할 것
 - **dynamic programming**으로 해결하였을 경우 : 문제를 풀기위한 풀이 설계과정, 점화식의 정의, correctness를 증명을 상세히 기재
 - 기타 : 참조한 source가 있을 경우 언급할 것
3. output.txt : 자신이 생성한 superstring이 기록된 실행 결과파일

▲ 제출방법

1. 자신의 학번으로 새 폴더를 생성
2. source와 document를 폴더에 넣은 후, 폴더 자체를 압축 (virus check 권장)
3. 과제 제출 메일(algo.hyu@gmail.com)로 제출

■ 주의사항

▲ 「명세」의 입·출력 형식을 정확하게 맞추어 주시기 바랍니다.

▲ 자신의 프로그램이 제한시간(30분) 내에 종료하지 못할 경우를 대비해, 별도의 시간 체크 루틴을 내부적으로 작성하여 제한시간(30분)에 임박했을 때에는 중간 결과를 출력하고 프로그램을 곧장 종료하도록 구현하시는 방법을 권장합니다.

▲ 해당 source를 구현한 환경을 명시해주시기 바랍니다, windows visual studio 2015 이상 추천합니다.

담당조교 : FTC 1107호 임베디드보안연구실 심호재
(algo.hyu@gmail.com)

[input 예시]

GTTGAATCCATGTATGTGAGATAAGGCCCAAGGAGAGGTAATAAGCTGGGAATGCCATCAGCTCATTTTTCTTCAGGCCTATATTTGTCATTGTCACTTGTAGA
AGCAGGACAGCCCTGGCGTTGGGGTTGGTAGAAACAGAGAG
AAACAGAGAGTATCAAAGGCAAAACTGAACTCCCTAATTTTTGGAAAACAGCAGATTGGAAACA
GAAACAGATGGGCTCCAGCGTTTTCCATGTGTGAGGTCATTGTCCCAGGTAGCCTTGCTCAGGA
AGAAACAGAGA
GATTGGAAACAGATGGGCTCCAGCGTTTTCCATGTGTGAGGTCATTGTCCCAGGTAGCCTTGCTCAGGACATTTCTTGTGAGCAAAACAGAAGTCAAATGATATTT
CTACCTTCCAAGATAATAG
GATAATAGAACATAATGTGAGATTTTCTCATGGATTCCCACAAGTTCAAGAAAAGTTTCATGGCCTTATTTAA
AATGTCAGATTTTCTCATGGATTCCCACAAGTTCAAGAAAAGTTTCATGGCCTTATTTAACTGCTTAAGCATTTCAACTGAAAGAGTATTTGCTTTCAAATACAC
AGGAATCTATTTGGAAAAATTTAACCAGAAAAATTTGGAATTTCTAAAGTAAAAAATGCATAAAGCAATCAAATTT
CAATCAAATTTTTTACAGCTTTTAATTTAGATGTGATGGGGAAAAAAATTTCTGAAATTTGGCTTTT
AACTGCTTAAGCATTTCAACTGAAAGAGTATTTGCTTTCAAATACACAGGAATCTATTTGGAAAAATTTAACCAGAAA
CTCTGAAATTTGGCTTTTATACCATTAAGACTTATTTTTATTACCAGCAATACAGGGCAACTCATT
TTCCATGTGTGAGGTCATTGTC
ACCAGCAATACAGGGCAACTCATTGAGTTGAATCTTGAAGGTAAACTTTAACTTAATTTAAGTTTTGGCT
AATTTAAGTTTTGGCTAATTTTAAGCATTTATCAGTCACCTACCATGATTTTCATCTCAGAAACCAAATCTCAATTTTATTAGACCTTTGAA
CTTTGAAATATTAACAGAAAGTTAAATGCTTCAAAAAATATTTCATGTAGAGGCTTATATATGTGG
CATGTAGAGGCTTATATATGTGGACCAGGAATCTCCATGTATTACAAAGTTTATGAGAACAATAACAAATGTTGATACACACATTTAATTCTAAAAATAAAA
AAAATAAAACTTACAACAAATAAACTGTAACAAATCAAGAAAAATTTGTAGGTTTCACATTTTATGTCTAAAAATATAGGT
TAGGTATTAACACTCAAGGATGGCTAAAAGAAATCACACAGAAAAATAGAAAAATATCTAGAGACAAATTAATAATATGAAATACCAAAGA
CTACCATGATTTTATCTCAG
AAATATGAAATACCAAAGAGATATCAAAAACAGTACCATAAGGGAAAAATTTATAGCTATAAATGATAAA
ATAAATGATAAAAAATAAGATACCGAATCAACAACCTTACTCCTAAGGAACTAAAAACAGAGGGAAAAAGGGACTACTAAAAGAGGG

[output 예시]

```
AGTTGAATCCATGTATGTGAGATAAGGCCCCCAAGGAGAGGTAATAAGCTGGGAATGCCATCAGCTCATTTTTCTTCAGGCCTATATTTGTCATTGTCACTTGTAGAAG
CAGGACAGCCCTGGCGTTGGGTTGGTAGAAACAGAGAGTATCAAAGGCAAACTGAACTTCCCTAATTTTGGAAAACAGCAGATTGGAAACAGATGGGCTCCAGCGT
TTTCCATGTGTGAGGTCATTGTCCCAGGTAGCCTTGCTCAGGACATTTCTTGTGTCAGCAAAACAGAAGTCAAATGATATTTCTACCTTCCAAGATAATAGAACATAATGT
CAGATTTTCTCATGGATTCCCACAAGTTCAAGAAAGTTTCATGGCCTTATTTAACTGCTTAAGCATTTCAACTGAAAGAGTATTTGTCTTTCAAATACACAGGAATCTA
TTTGGAAAATTTTAAACGAGAAAATTTGGAAATCTAAAGTAAAAATGCATAAAGCAATCAAAATTTTTTACAGCTTTTAATTTAGATGTCATGGGGAAAAAAATTT
CTCTGAAAATTTGGCTTTTATACCATTAAAGACTTATTTTTTATTACCAGCAATACAGGGCAACTCATTGAGGTTGAATCTTGAAGGTAAACTTTAACTTAATTTAAG
TTTTGGCTAATTTTTAAGCATTATCAGTCACCTACCATGATTTTCATCTCAGAAACAAAATCTCAATTTTATTAGACCTTTGAAATATTTAAACAGAAGGTTAAATG
CTTCAAAAAATATTTCATGTAGAGGCTTATATATGTGGACCAGGAATCTCCATGTATTACAAAGTTTATGAGAACATAACAAATGTTGATACACACATTTAATTTCTAAA
ATAAAAACTTACAACAAATAAACTGTAACAAATCAAGAAAATTTGTAGGTTTCACATTTTATGTCTAAAAATATAGGTATTAACACTCAAGGATGGCTAAAAGAAA
TCACAACAGAAAATAGAAAATCTAGAGACAAATTAATAATGAAATACCAAAGAGATATATCAAAAAACGTACCATAAGGGAAAAATTTATAGCTATAAATGATA
AAAAATAAGATACCGAATCAACAACTTTACTCCTAAGGAACTAAAAACAGAGGGAAAAAGGGACTACTAAAAGAGGG
```