iHS

자료구조 (Data Structure)

Programming Assignment 2

▪ 문제 1:

강의 자료의 pmatch() 함수는 KMP algorithm을 사용하여 주어진 string내에 pattern이 존재하는지를 찾아주는 함수이다.

만약 string에서 pattern이 발견되면, pmatch()는 가장 처음 pattern과 string이 matching되는 부분의 starting index를 return한다.

pmatch()를 수정하여, string과 pattern이 matching되는 모든 부분의 starting indices를 출력하는 함수 pmatch_all()을 작성하라.

▶ 문제 1:

예제	
입력	출력
bbbbbabbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb	0 1 2 6 7 8
bbbbbabbbbbbbbbbbbbbbbbbbbbbbbbbbbbbbb	

- 문제 1:

- 입력:
 - 첫 번째 줄-string
 두 번째 줄-pattern
 - · 각 입력의 길이는 최소 1글자, 최대 30글자 (그 외의 경우는 들어오지 않음)
- 출력:
 - matching되는 모든 부분의 starting index를 한 줄에 하나씩 출력
 - matching되는 부분이 없을 경우 아무 것도 출력하지 않음

문제 1:

- 제약 조건:
 - 모든 입출력은 stdin, stdout 사용
 - 전역변수 사용 불가
 - 정적할당 가능

문제 2:

Array을 parameter로 받은 후, 연속된 숫자만이 저장되어 있는지 확인하는 함수 check_array()를 만들고자 한다. 예를 들어, {4, 1, 5, 2, 3}와 같은 array의 경우에는 1부터 5까지의 연속적인 숫자들이 저장되어 있으므로, check_array()가 1을 return하도록 한다. 그러나 {10, 14, 12, 15, 11, 9}와 같은 array의 경우에는 중간에 13이 빠져 있으므로 check_array()가 0을 return하도록 한다.

check_array()를 구현하는 데에는 여러 가지가 있지만, 가장 쉬운 방법은 배열을 먼저 sorting한 후 중간에 빠진 숫자가 있는지 찾는 것이다. 이 경우, array의 크기를 n이라 할 때, 함수의 시간 복잡도는 O(nlogn)이 된다.

물론 이보다 더 나은 방법도 존재한다. O(n)의 시간 복잡도를 가지도록 check_array()를 작성해보자.

▶ 문제 2:

예제	
입력	출력
5 4 1 5 2 3	1
6 10 14 12 15 11 9	0
1	1
4 2 2 3 1	0

문제 2:

- 입력:
 - · 첫 번째 줄-원소의 개수 n 두 번째 줄-n개의 원소
 - · n과 각각의 원소의 크기는 1 이상 100 이하

- 출력:
 - · 배열이 연속된 숫자로만 이루어져 있을 경우 1, 아니면 0

문제 2:

- 제약 조건:
 - 모든 입출력은 stdin, stdout 사용
 - 전역변수 사용 불가
 - 정적할당 사용 불가
 - 완성된 함수의 시간 복잡도는 O(n)이어야 함.

• 소스코드

- 이름: HW2_학번_문제번호.c
 - ex) HW2_20180000_1.c
- cpp, txt 파일 등은 <u>절대</u> 받지 않음
 - 반드시 .c 파일로 제출할 것
- 컴파일 에러가 발생할 경우 0점 처리
- 무한 루프/세그멘테이션 오류는 해당 testcase 0점 처리
- 입출력 양식이 틀릴 경우 감점



- 보고서
 - 이름: Document.pdf
 - 반드시 pdf 파일로 변환 후 제출
 - ▶ 보고서의 hard copy를 제출할 것
 - 4/8 09:00부터 4/11 18:00까지 AS1010 앞에 제출함이 있을 예정임
 - Hard copy를 제출하지 않을 경우 보고서 0점 처리



- 압축 파일
 - 이름: HW2_학번.zip
 - ex) HW2_20180000.zip
 - 압축을 풀면 아래의 파일들이 있어야 함:
 - HW2_학번_1.c
 - HW2_학번_2.c
 - Document.pdf
 - 제출 형식이 틀릴 경우 과제 점수의 30% 감점



- 4/11 18:00까지 <u>2019ds1@gmail.com</u>으로
 - 압축파일(HW2_학번.zip) 제출
 - 메일 제목: HW2_학번_이름
- 4/11 18:00까지 AS1010 앞 제출함에 보고서 hard copy 제출

- 과제 채점은 cspro의 gcc compiler 기준
- Late 받지 않음
- Copy 검사



제출시 유의 사항

- 과제는 반드시 C언어로만 작성되어야 함.
- 과제 마감시까지 최대 3회 재제출 가능.
 - 가장 최근에 도착한 제출본으로 성적 처리함.
 - 신중하게 재제출 할 것.
- 과제를 교우들과 논의 할 수는 있으나 반드시 독자적으로 하여 제출할 것.
 - 카피를 허용한 자와 카피를 한 자 모두 1회 적발시 과제 0점 처리, 2회 적발시 F학점
- 프로그램이 일정시간 안에 답을 출력 안하는 경우 틀린 출력이라고 간주함.

