# HW1. (Programming) Evaluation of postfix expression

학과: 사이버국방학과

학번: 2019330026

이름: 윤준혁

# 헤더파일

* stdio.h
* string.h
  + strtok 함수가 선언된 헤더 파일
    - 입력으로 들어온 문자열을 공백 문자를 기준으로 나누어 사용하기 위해 strtok 함수 사용
  + #define \_CRT\_SECURE\_NO\_WARNINGS
    - strtok 보안 경고로 인한 컴파일 에러 방지

# 자료구조 스택의 정의와 관련 함수 정의

* stack이라는 이름으로 1000개까지 저장할 수 있는 스택 구조 정의
* 함수
  + IsFull(stack\* ps)
    - 스택의 모든 공간이 채워져 있으면 양수, 빈 공간이 있으면 0 반환
  + IsEmpty(stack\* ps)
    - 스택의 모든 공간이 비워져 있으면 양수, 원소가 하나라도 있으면 0 반환
  + Push(stack\* ps, int in)
    - 정수 in을 매개변수로 받아 스택에 추가하는 함수
    - 만약 스택의 모든 공간이 채워져 있으면 0 반환
    - 정상 작동 시 1 반환
  + Pop(stack\* ps, int\* out)
    - 스택의 가장 상위 값을 매개변수로 들어온 주소에 저장한 후, 해당 상위 값 삭제
    - 만약 스택의 모든 공간이 비워져 있으면 0 반환
    - 정상 작동 시 1 반환

# main 함수의 변수

* Stack s
  + 정수를 저장할 스택 s
* File\* infp
  + 수식의 개수와 수식을 입력 받을 파일 스트림
* File\* outfp
  + 수식의 결과를 출력할 파일 스트림
* Int n
  + 계산할 수식의 개수
* Char expression[1001]
  + 계산할 수식, 1000글자 까지 가능
* Char\* ptr
  + 공백문자를 기준으로 나누어진 수식을 가르키는 포인터

# 프로그램 작동 과정

* postfix.in 파일에서 수식의 개수와 그 개수 만큼의 수식을 차례대로 읽어옴
  + postfix.in이라는 입력 파일이 같은 폴더에 없으면 에러 발생
* postfix.out 파일에 수식의 개수 출력
* 각각의 수식에 대해 다음의 과정 반복
  + 수식의 각 항이 숫자면 스택에 추가
  + 수식의 각 항이 +, -, \*, / 중 하나라면, 스택에서 먼저 저장된 수에 나중에 저장된 수를 입력된 연산기호에 맞게 연산하고 그 결과 값을 다시 스택에 저장
  + 각 항이 숫자나 연산자가 아니라면 무시
  + 수식이 끝나면 마지막으로 저장된 결과값을 postfix.out 파일에 출력
* 모든 수식을 계산한 후, 입력 파일과 출력 파일을 닫고 프로그램 종료