목차

1. 개요
   1. 프로젝트 주요 기능 및 특징
   2. 프로젝트 추진 내역
      1. 조원 구성 및 역할 분담
      2. 일정
2. 시스템 요구사항 및 개발 환경
3. 시스템 개요
   1. 시스템의 주요 기능
   2. 시스템 구조
4. 함수 설명
5. 개요
   1. 프로젝트 목적 및 기능

목적 : 공학용 계산기와 스케줄 관리 프로그램 개발을 통해 함수, 배열 사용 능력과 다양한 라이브러리 함수의 활용 능력을 배양한다.

기능 : 1. 공학용 계산기

* 사용자에게 수식을 입력 받아 계산하여 결과를 보여주는 프로그램이다.
* 입력된 수식을 Stack을 이용하여 후위 표기법(Postfix Notation)으로 변경한다.
* Stack은 배열을 이용하여 구현한다.
* C언어에서 제공하는 수학함수(log, pow, sqrt, sin, cos, tan)를 구현한다.

log : base가 10이며 수식에서는 log 30으로 표기

pow : 수식에서는 5^3으로 표기

sqrt : root 10으로 표기

sin : sin 45로 표기

cos : cos 45로 표기

tan : tan 45로 표기

* 사용자가 입력한 수식에 수학함수가 존재 할 경우 후위 표기법 전에 해당 수학함수를 실행한다.

23 \* 4 + log 10 \* 3^4를 입력할 경우 23 \* 4 + 1 \*81로 변경 후 후위 표기법을 진행한다.

* 대문자 알파벳(A~Z)을 변수로 취급하여 계산할 수 있다.
* 계산 과정은 일정한 시간(대략 3초)을 두고 순차적으로 출력하고 출력이 끝나면 엔터키를 받을 때까지 기다린다. 엔터를 누르면 화면을 clear하고 다시 입력을 받는다.
* !@#$을 입력하면 화면을 clear하고 Main menu로 넘어간다.

- 2. 스케줄 관리

* 스케줄을 입력하면 보여주는 프로그램이다
* 스케줄 관리에 관한 Sub menu가 있어야 한다.
* 스케줄은 0000(년) 00(월) 00(일) xxxxxx(내용) 처럼 입력하고, 추가 후 화면을 clear한다.
* 기존에 같은 날에 스케줄이 있을 경우 추가할 것인지 덮어쓸 것인지 사용자에게 확인받는다.
* 스케줄 삭제를 누를 경우 해당 날짜에 일정이 있을 경우 삭제를 하고, 없을 경우 일정이 없다고 알려준다.
* 스케줄 보기를 하면 가장 빠른 스케줄부터 월단위로 출력하고 엔터키를 입력하면 다음 스케줄을 보여준다.
* 4. Main menu를 입력하면 화면을 clear하고 Main menu로 이동한다.
  1. 프로젝트 추진 내역
     1. 조원 구성 및 역할 분담

조장 : 컴퓨터학부 1학년 가반 20162448 김병준 – 공학용 계산기, 스케줄 관리 제작 (75%)

조원 : 컴퓨터학부 1학년 가반 20162528 함정완 – 프로그램 디버거 (25%)

* + 1. 일정

2016. 5. 12(목) – 프로그래밍1 및 실습 프로젝트 공지

2016. 5. 16(월) – 프로젝트 역할 지정

2016. 5; 17(화) – 메인 프로그램 프레임워크 구축

2016. 5. 20(금) - 스케줄 관리 프로그램 프로토타입 완성

2016. 5. 23(월) – 공학용 계산기 변수 지정 완성

2016. 5. 26(목) – 공학용 계산기 Transformation 완성

2016. 5. 30(목) – 공학용 계산기 프로토타입 완성

2016. 5. 31(금) – 스케줄 관리 프로그램 리팩토링 및 완성

2016. 6. 06(월) – 스케줄 관리 프로그램 일정 정렬 알고리즘 트러블슈팅

1. 시스템 요구사항 및 개발 환경

운영체제 : Windows CE를 제외한 모든 Windows 버전, Mac OS X 10.6, Ubuntu 10.04LTS 이상

최소사양 : Pentium 3 2.4GHz, 256MB RAM, 10GB Storage

개발환경 : Xcode 7.3.1, Cygwin 5.4(x86, 64)

개발언어 : C Language

1. 시스템 개요
   1. 시스템의 주요기능

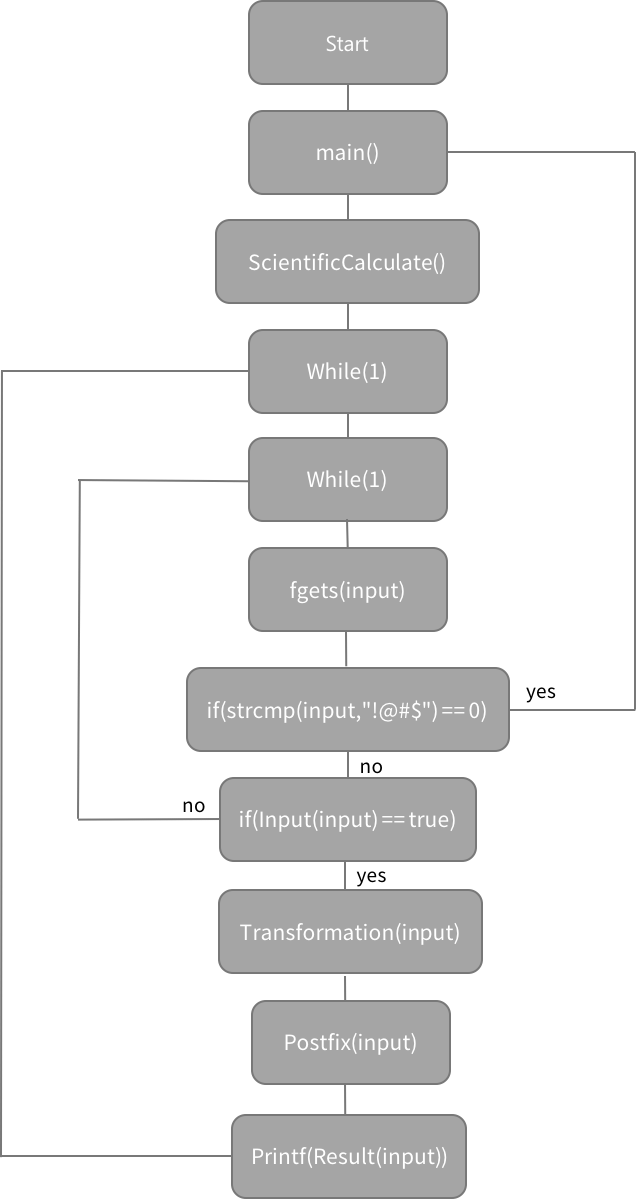
이 프로그램은 공학용 계산기와 스케줄 관리 프로그램으로 구성되어 있으며 각 프로그램들은 Main menu에서 선택하여 구동시킬 수 있다.

공학용 계산기는 알파벳 A~Z까지 변수를 지정할 수 있으며, 이를 Stack을 활용하여 후위연산법으로 계산할 수 있다. 또한 루트를 비롯한 삼각함수와 로그를 사용할 수 있다.

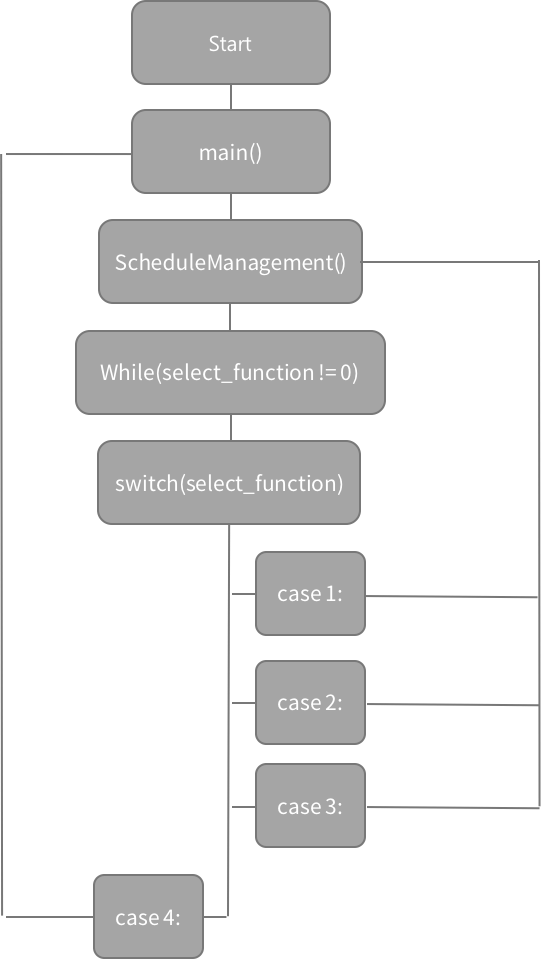
스케줄 관리 프로그램은 1년 1월 1일부터 추가한 날짜까지의 모든 날짜를 계산한다. 또한 일정을 각 날짜 마다 추가할 수 있으며, 일정을 추가 및 덮어쓰기가 가능하다. 일정이 이미 존재할 때 추가를 할 경우 추가된 일정 순으로 구분자로 .(점)이 존재하며 이를 통해 여러가지 일정 내용을 추가할 수 있다. 또한 일정을 Bubble Sort로 날짜순으로 저장하며 이를 일정 보기 기능을 통해 볼 수 있다.

* 1. 시스템 알고리즘

공학용 계산기 :



스케줄 관리 :

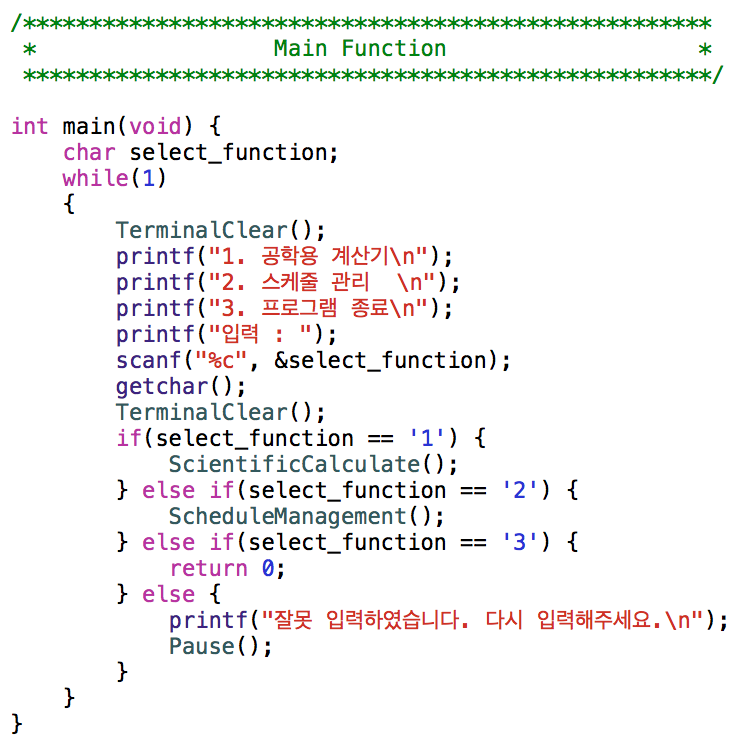


1. 함수설명

함수 정의 :



함수설명 :



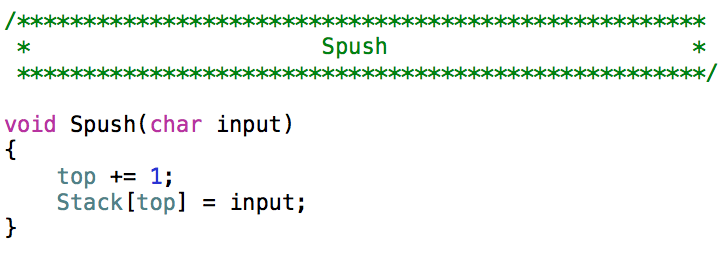
메인 함수는 select\_function이라는 char형 변수를 입력 받아 메뉴에서 기능을 선택한다.

1을 입력 할 경우 ScientificCalculate함수를 실행한다.

2를 입력 할 경우 ScheduleManagement함수를 실행한다.

3을 입력 할 경우 0을 반환하며 프로그램을 종료한다.

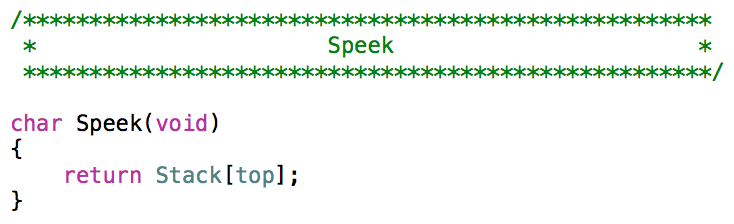
이 외의 값을 입력 할 경우 오류구문을 출력하며 다시 입력을 받는다.



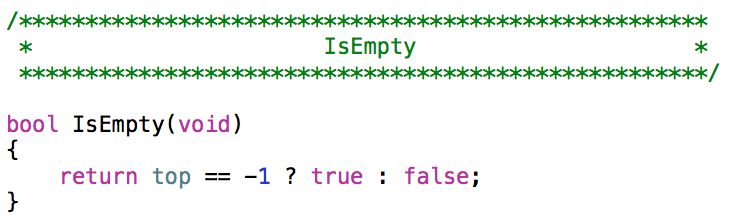
Spush함수는 Stack에 값을 넣음과 동시에 Stack의 크기를 1칸 늘린다.



Spop함수는 Stack의 값을 뽑아내는 역할을 함과 동시에 Stack의 top을 -1한다.



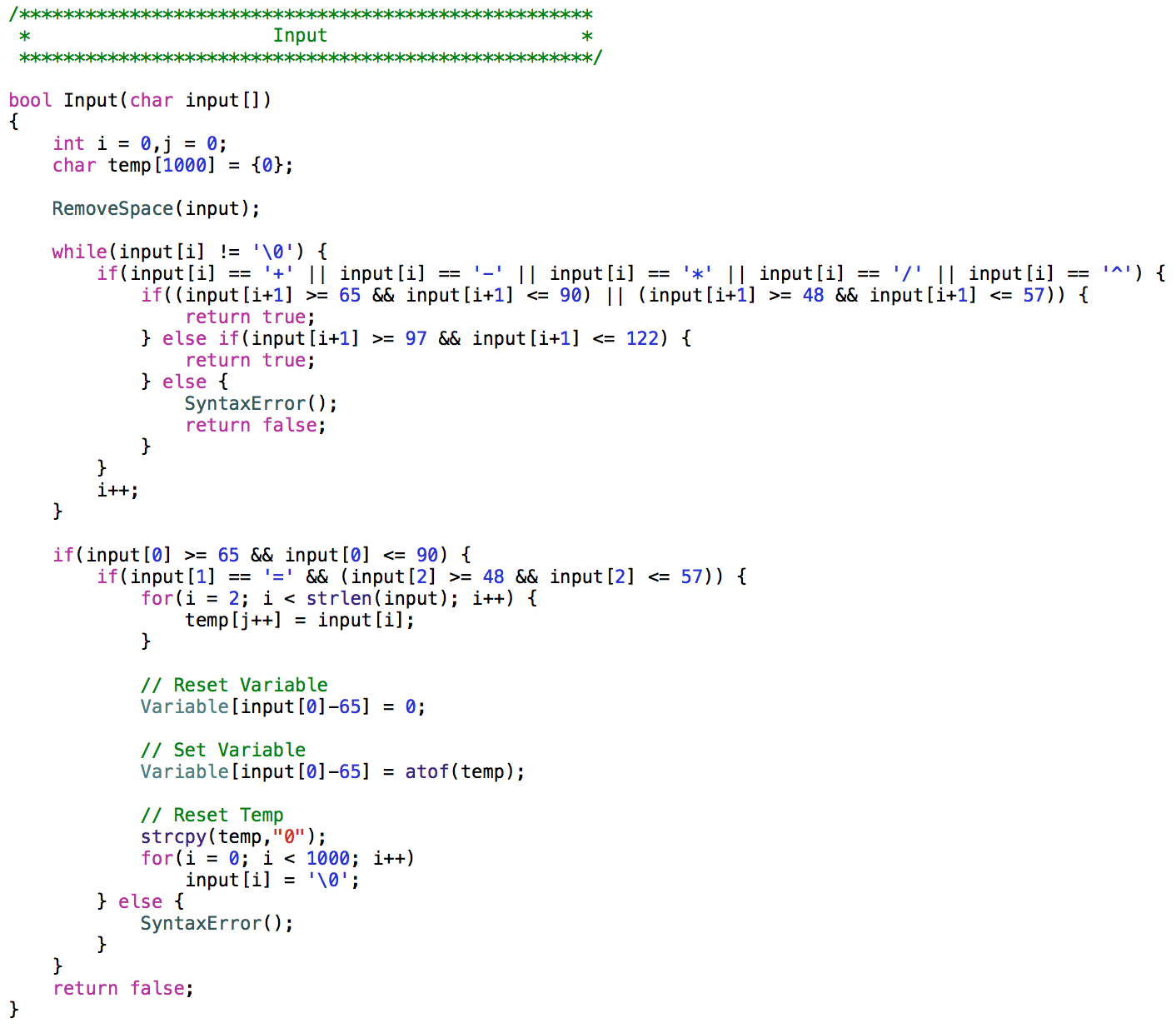
Speek함수는 Stack의 맨 꼭대기 즉 Stack의 제일 마지막 값을 반환한다.



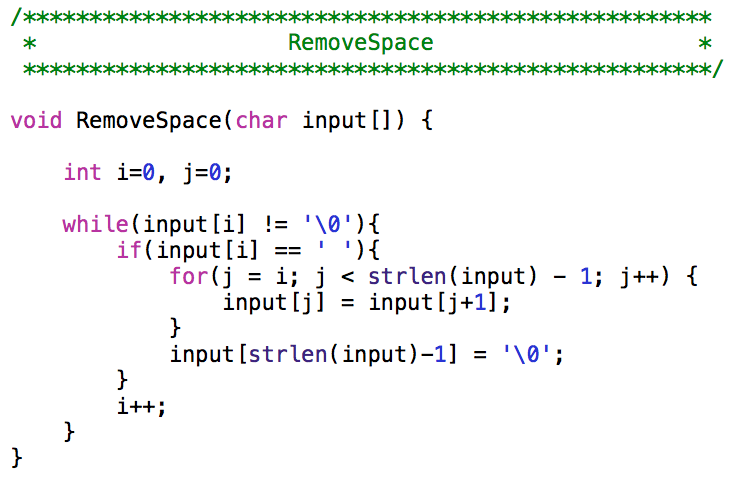
IsEmpty함수는 Stack의 top을 검사하며 만약 top이 -1이라면 true(1)을 반환하고, 아니라면 false(0)을 반환한다.



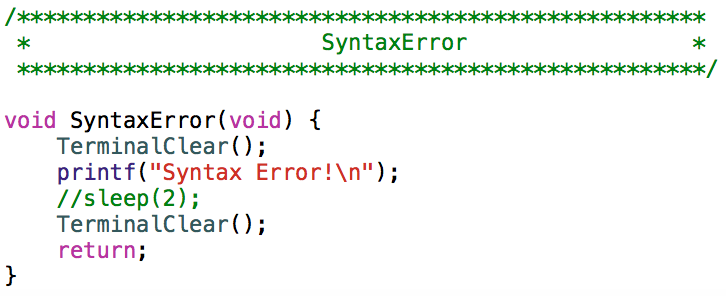
ScientificCalculate는 공학용 계산기 기능의 메인함수로써 작동한다. Input이라는 char형 배열에 수식을 입력 받으며 input배열의 맨 끝의 Enter Buffer을 제거한다. 그리고 만약 input배열에 !@#$이 입력되면 함수를 반환하며 Main menu로 돌아간다. !@#$이 입력되지 않았을 경우 Input함수에 input을 넘겨 수식을 검토 및 변수 지정을 하며 반환 값이 true(1)이면 반복문을 탈출하고, 아니라면 다시 input을 입력 받는다. 반복문을 탈출한 뒤 Transformation, Postfix에 input을 넘겨 변환, 후위 연산 후 결과값을 출력한다. 출력한 뒤 CalStack을 Null문자(\0)로 초기화하고 Enter을 받을 때까지 대기한다. 입력 받은 뒤 위의 과정을 다시 반복한다.



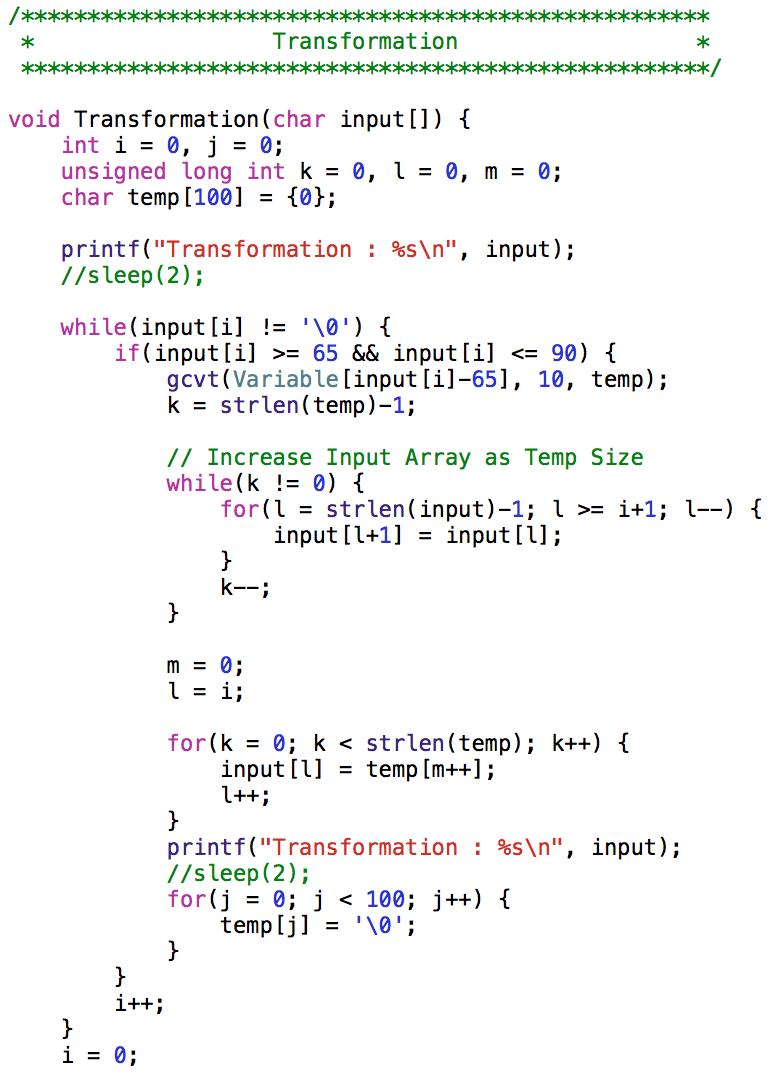
Input함수는 넘어온 input배열의 수식을 RemoveSpace로 넘겨서 수식의 공백을 제거한 뒤, 만약 알파벳 A~Z와 ‘=’문자가 들어가 있을 경우 Variable이라는 double형 배열에 변수를 저장 하고 false(0)을 반환한다. 또한 만약 연산자가 들어가 있을 경우 true(1)을 반환한다.



RemoveSpace함수는 넘어온 input배열의 공백을 제거하고 배열을 축소하는 함수이다. 배열에서 Null문자(\0)이 나오기 전까지 공백( )이 존재할 경우 그 위치의 배열부터 마지막 배열까지 한 칸씩 축소한다.



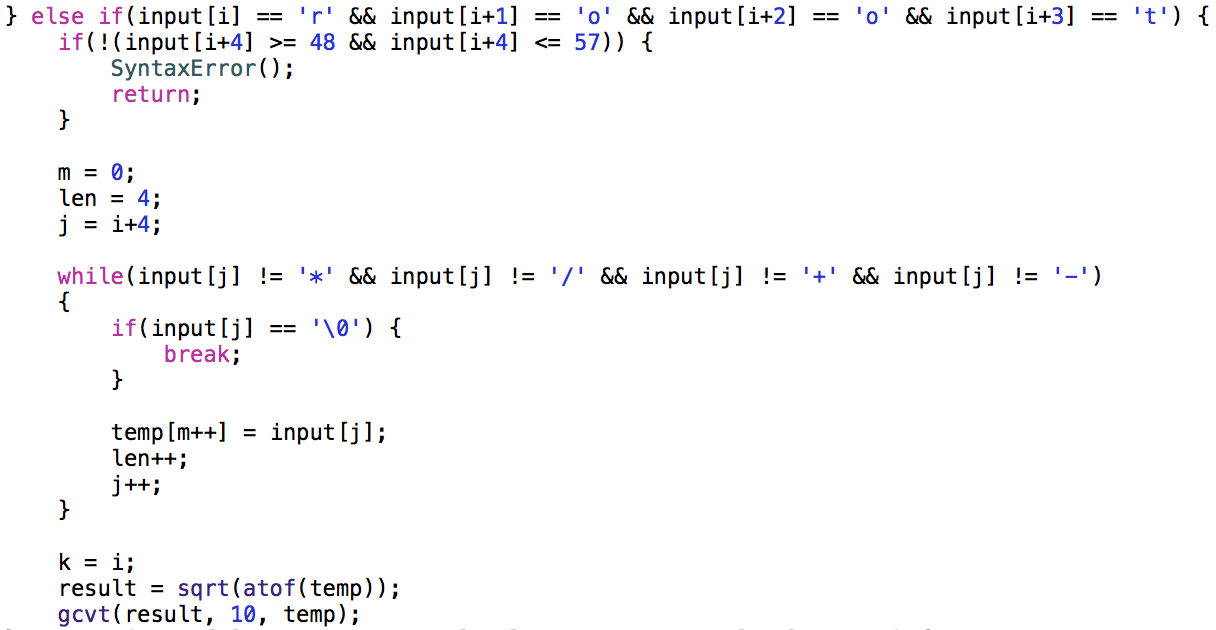
SyntaxError함수는 수식의 구문 오류 처리 기능을 하는 함수이다. 만약 구문 오류가 발생 할 시 해당 함수를 선언하면 구문 오류 처리 후 함수를 반환한다.



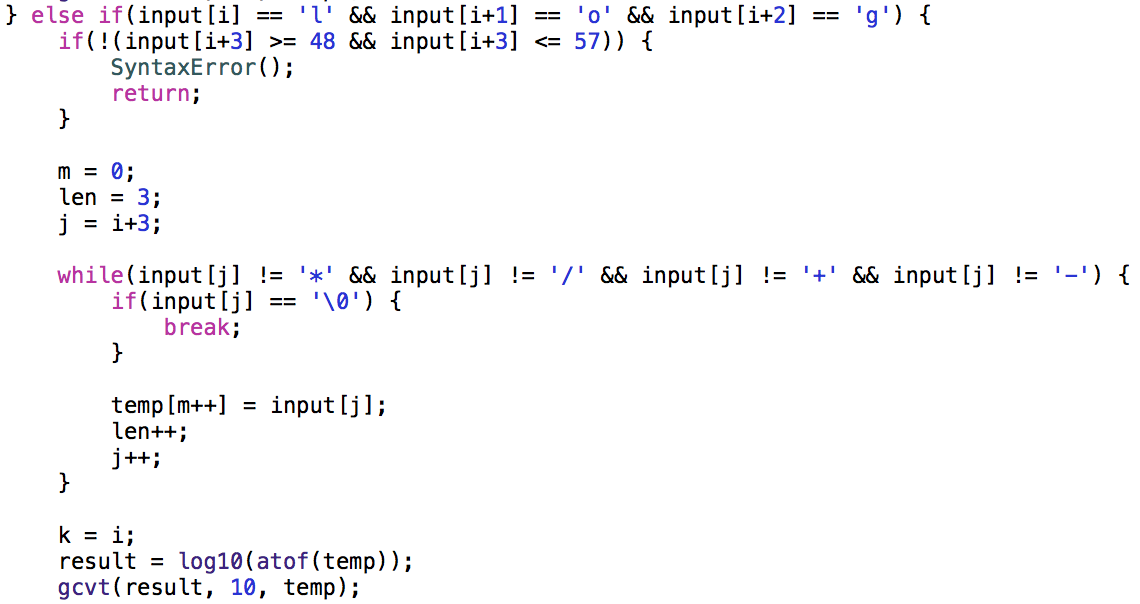
Transformation함수는 input배열에 존재하는 변수 혹은 수학 함수를 지정된 값으로 변환시킨다. Input배열을 탐색하면서 Null문자(\0)가 나오기 전까지 A~Z가 있을 경우 그 값을 Variable에서 빼냄과 동시에 gcvt로 문자열로 변환시켜 char형의 temp배열에 저장한다. 또한 k라는 변수에 temp의 길이를 지정하면서 변수가 있던 위치의 배열의 크기를 늘려준다. 또한 이 과정이 끝나면 변환된 수식을 출력하면서 모든 변수 혹은 수학 함수가 변환이 되었을 경우 이 함수는 종료된다.



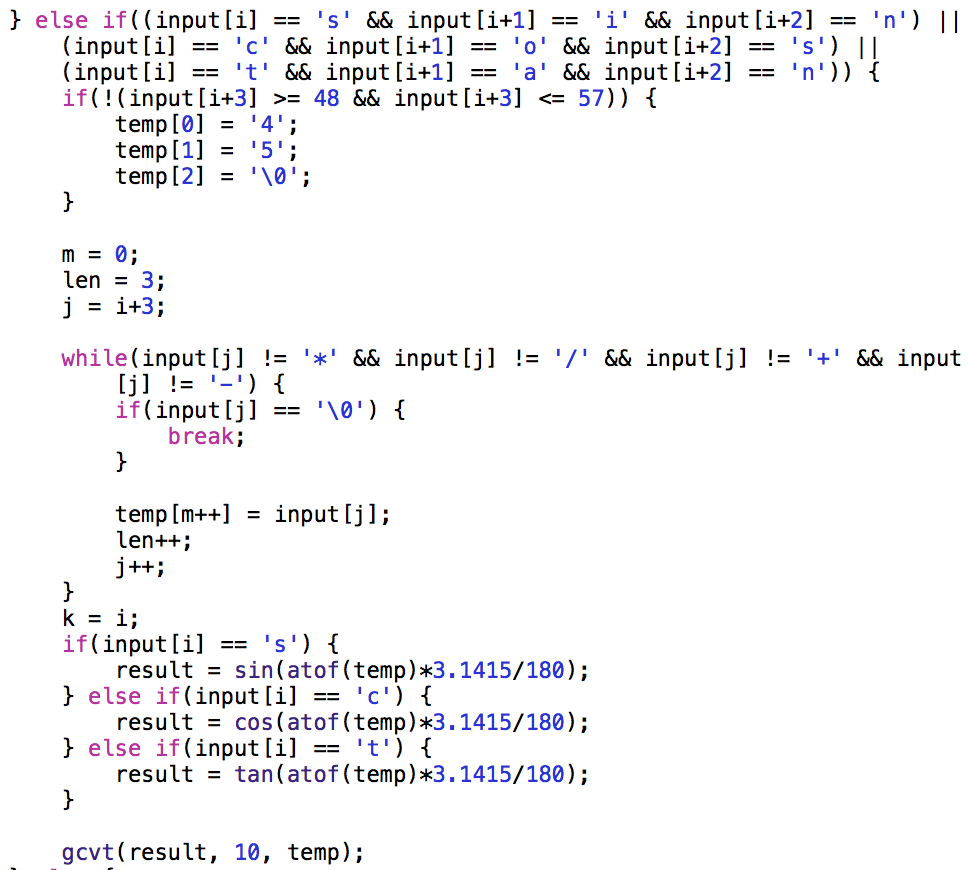
위 소스코드는 Transformation함수에서 지수(^)의 경우이다.



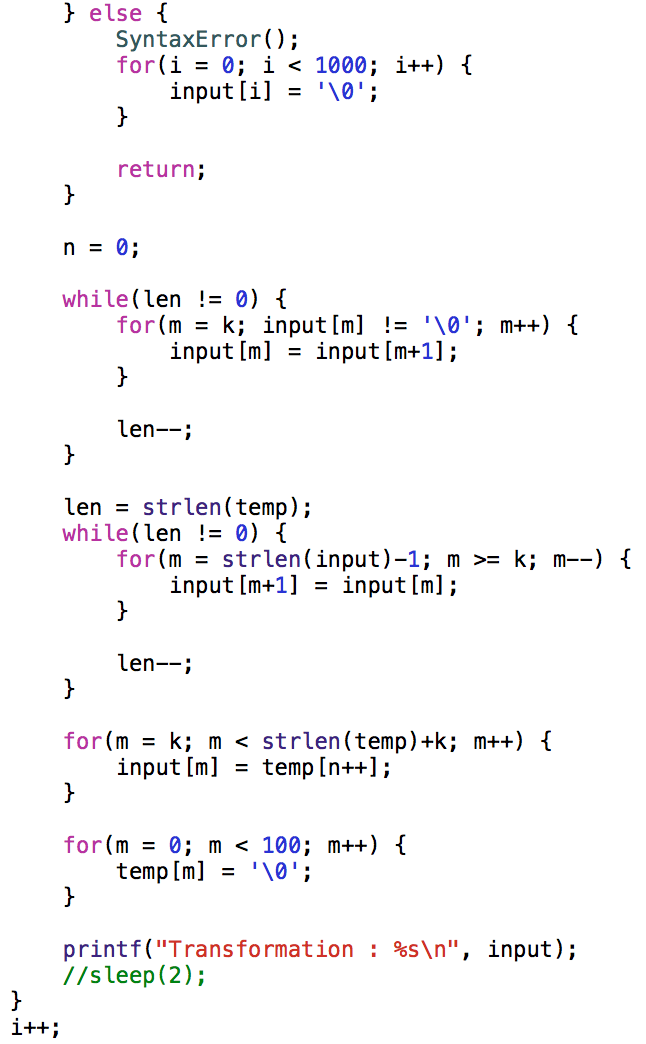
위 소스코드는 Transformation함수에서 루트(root)의 경우이다.



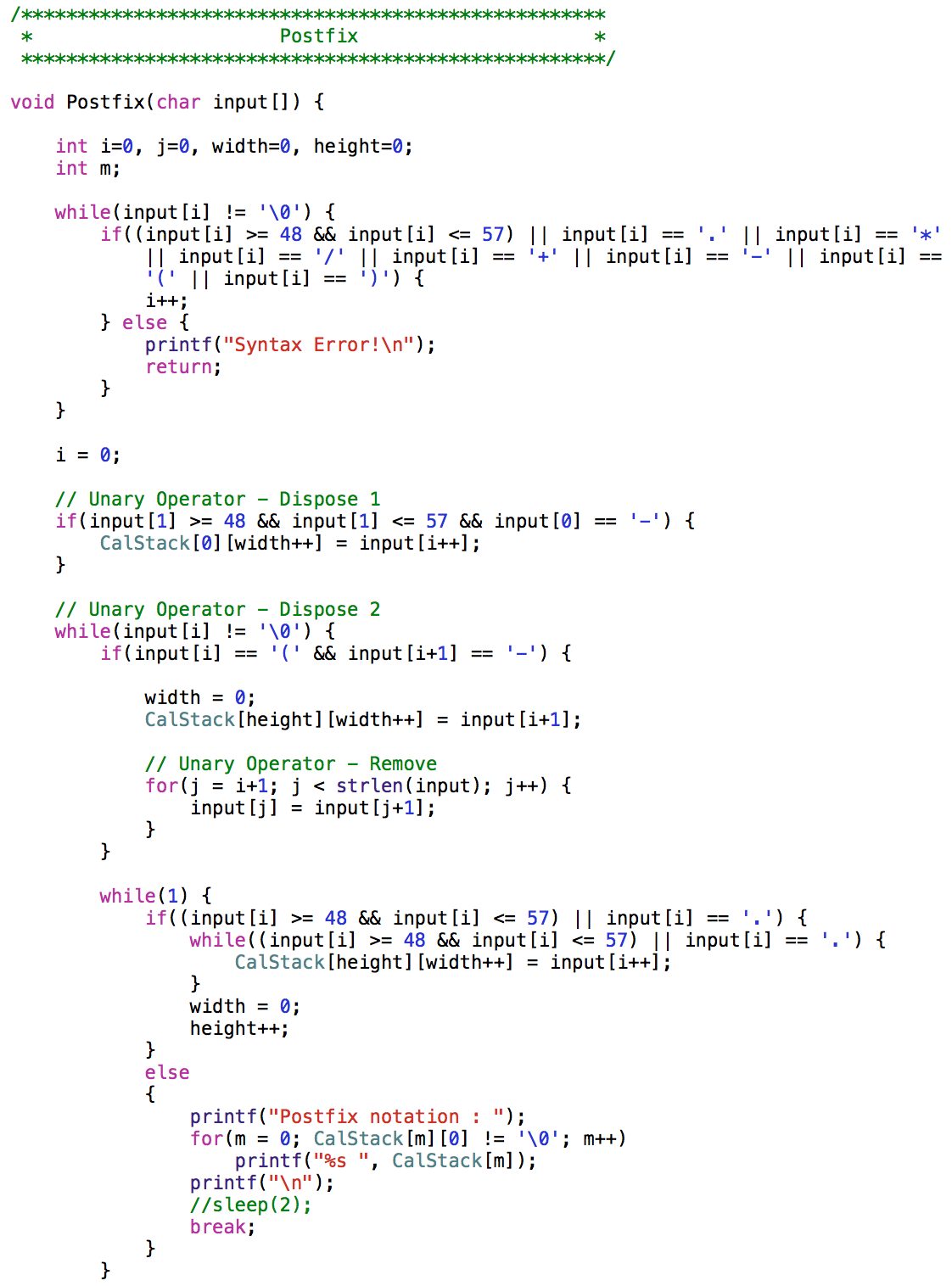
위 소스코드는 Transformation함수에서 로그(log)의 경우이다.



위 소스코드는 Transformation함수에서 삼각함수(sin, cos, tan)의 경우이다.



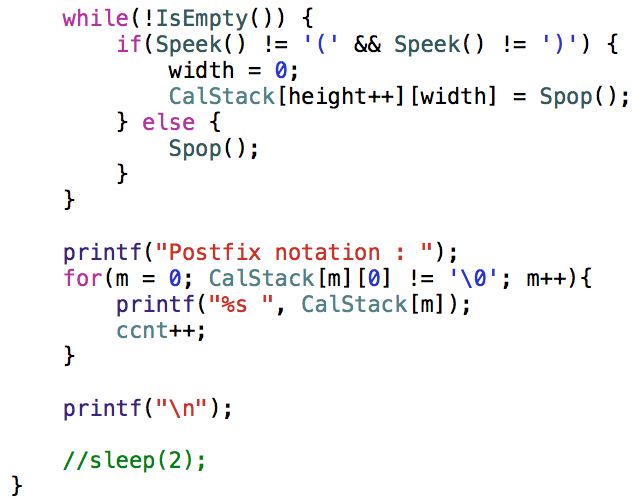
위의 소스코드는 Transformation함수에서 수학 함수 외의 기호 혹은 문자열이 입력되었을 경우 구문 오류 처리를 하는 경우이다. 또한 모든 단계가 마무리 되면 temp배열을 초기화한다.



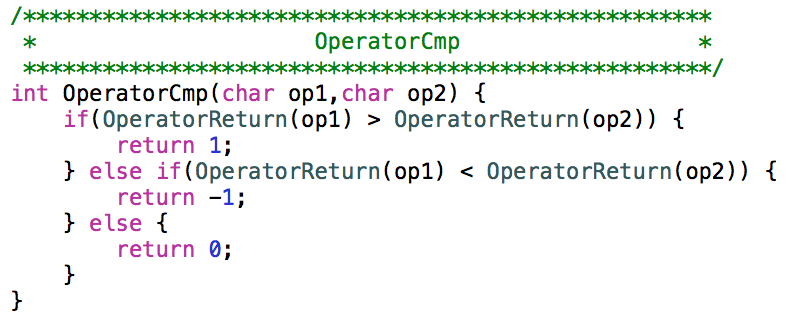
Postfix함수는 Transformation에서 변환된 input배열의 수식을 후위 연산식으로 변환해주는 함수이다. 먼저 수식을 검토한 다음 연산자를 추출하여 CalStack의 배열에 담는다. 그 다음 추출된 CalStack의 후위연산식으로 변환된 식을 출력한다.



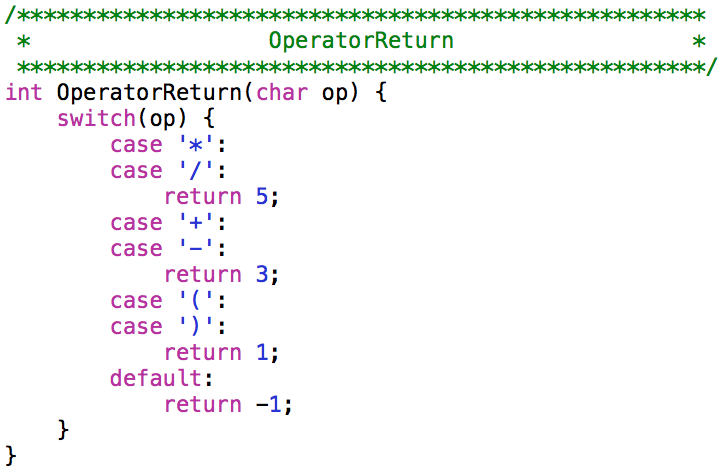
위의 소스코드는 Postfix함수에서 연산자 우선순위에 따라 Stack에 연산자와 피연산자를 추가 및 수식을 후위연산으로 변환하는 과정이다.



위의 소스코드는 Postfix함수에서 연산자 우선순위에 따라 Stack에 연산자와 피연산자를 추출 및 변환된 수식을 출력을 하는 과정이다.



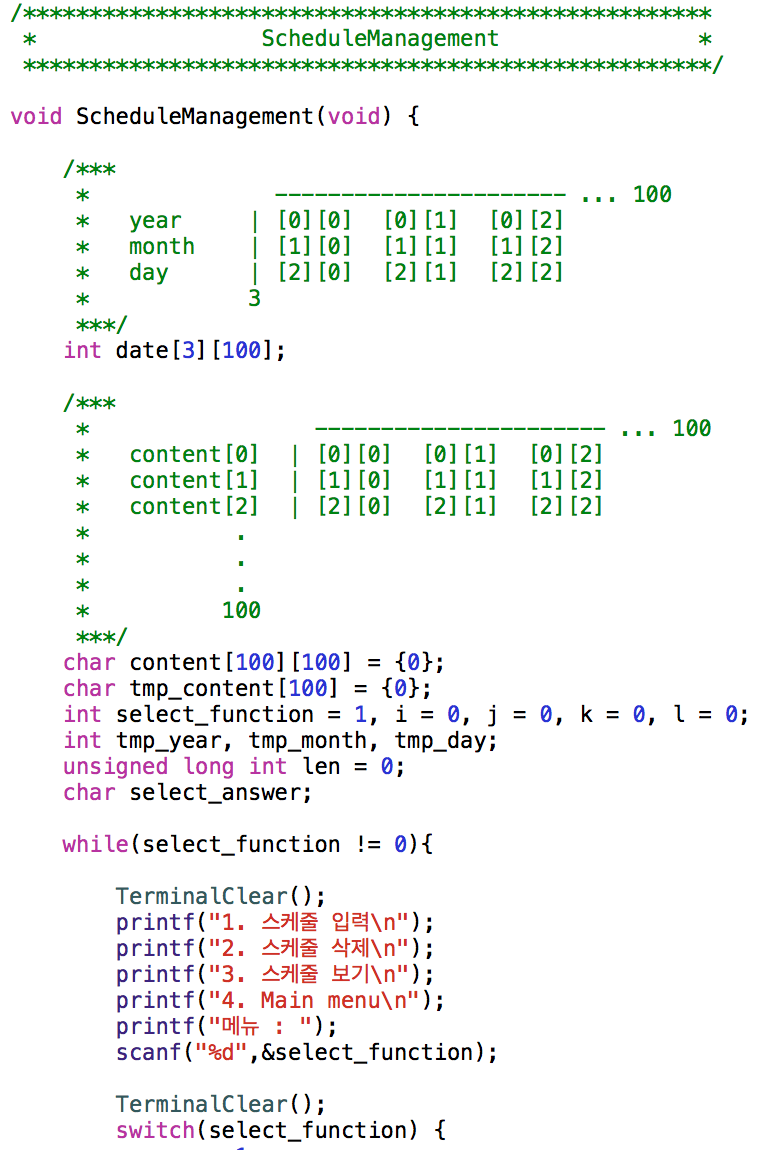
OperatorCmp는 연산자의 우선순위를 판단한다. 만약 op1이 op2보다 크면 1을 반환한다. 반대로 op1이 op2보다 작으면 -1을 반환한다. 이 외의 경우는 함수를 종료한다.



OperatorReturn은 연산자의 경우에 따라 반환값을 지정한다. 곱하기(\*)와 나누기(/)의 경우 5를 반환하며 더하기(+)와 빼기(-)는 3을 반환, 괄호(())는 1을 반환하며 이외의 것들은 -1을 반환하며 함수를 종료한다.



Result함수는 Transformation과 Postfix가 모두 완료된 시점에서 결과값을 추출하는 함수이다. Null문자(\0)가 나오기 전까지 CalStack을 검토하고, 문제가 없으면 조건문을 통해 연산 후 결과값을 반환한다.



ScheduleManagement함수는 스케줄 관리 기능의 메인함수로써 작동한다. Select\_function이라는 int형 변수로 기능을 선택할 수 있다.

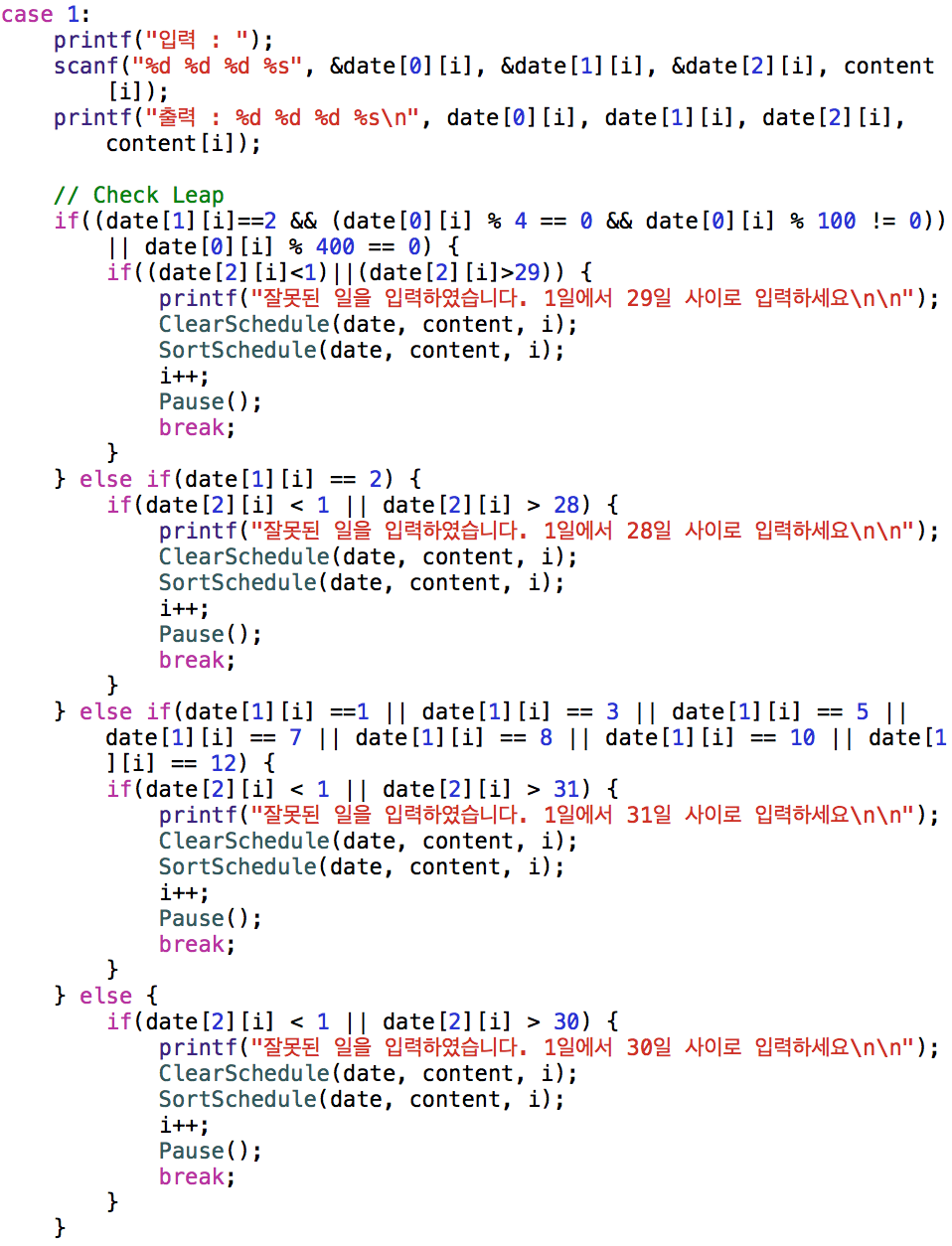
1을 입력 할 경우 스케줄 입력 기능을 실행한다.

2를 입력 할 경우 스케줄 삭제 기능을 실행한다.

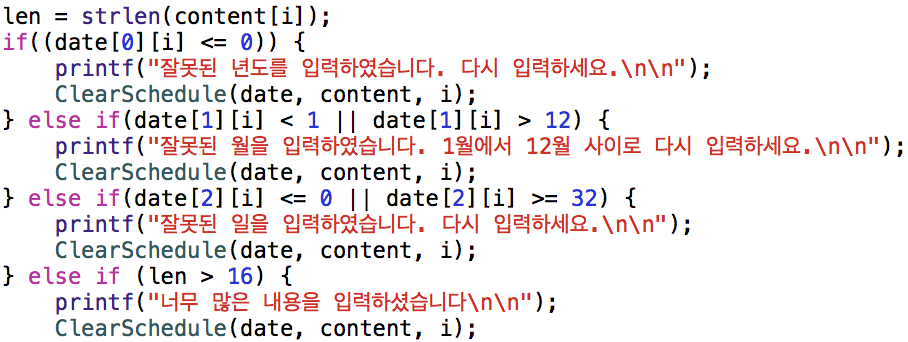
3을 입력 할 경우 ViewSchedule함수를 실행한다.

4를 입력 할 경우 함수를 반환하며 Main menu로 돌아간다.

이 외의 값을 입력 할 경우 오류구문을 출력하며 다시 입력을 받는다.



위의 소스코드는 ScheduleManagement함수에서 일정을 입력 하였을 때 윤년의 규칙에 어긋났을 때 구문 오류 처리를 하는 경우이다. 모든 경우는 오류 문장을 출력한 뒤 입력했던 임시 일정을 초기화하고 일정 배열을 정렬한 뒤 일정을 다시 입력 받는다.



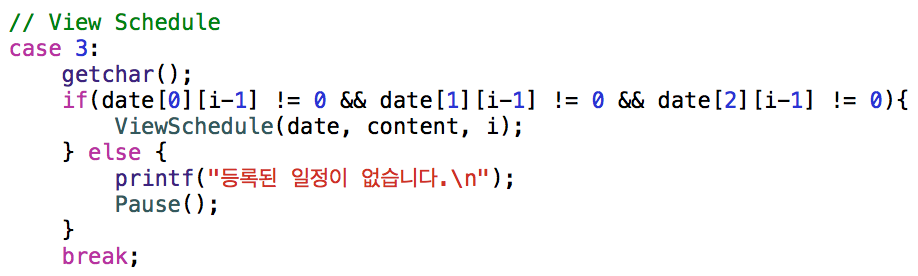
위의 소스코드는 ScheduleManagement함수에서 일정을 입력 하였을 때 날짜의 규칙에 어긋났을 때 구문 오류 처리를 하는 경우이다. 이때의 경우는 윤년과 달리 임시 일정만 초기화 하고 일정을 다시 입력받는다.



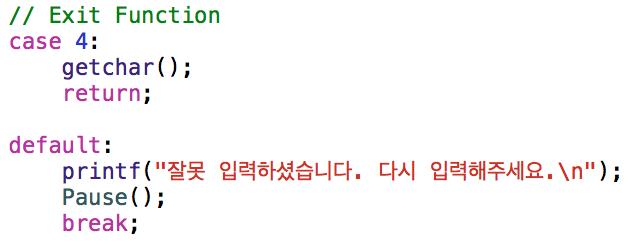
위의 소스코드는 ScheduleManagement함수에서 일정을 입력 하였을 때 일정이 중복되었을 경우이다. 첫번째 일정을 입력 받았을 때의 경우만 일정 추가 구문을 출력하고, 그 이후부터는 for문이 모두 돌았을 때 중복된 일정이 없을 경우 일정 추가 구문을 출력한다. . 만약 일정을 추가하면 구분자로 점(.)을 사용하여 기존의 내용에 “.내용”을 추가하고 덮어쓸 경우 모든 내용을 삭제하고 다시 입력한다.



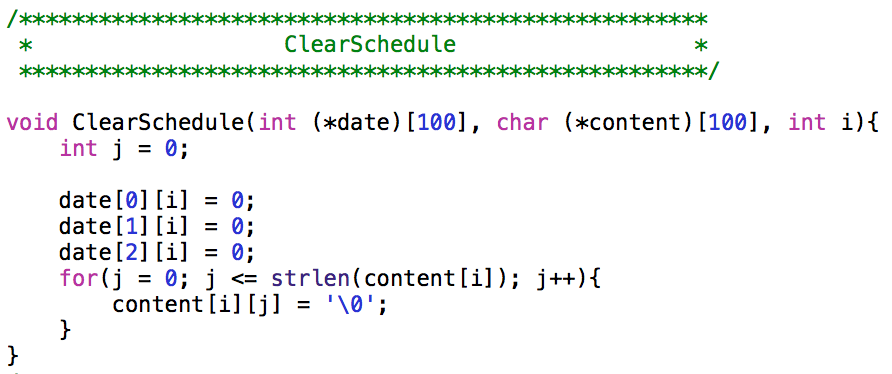
위의 소스코드는 ScheduleManagement함수에서 일정을 삭제하는 경우이다. 먼저 삭제할 일정을 입력 받고 equality\_date와 equality\_content를 통해서 일정을 검토하고 만약 true라면 일정 삭제 구문을 출력하고 일정을 초기화, false라면 일정 없음 구문을 출력한다.



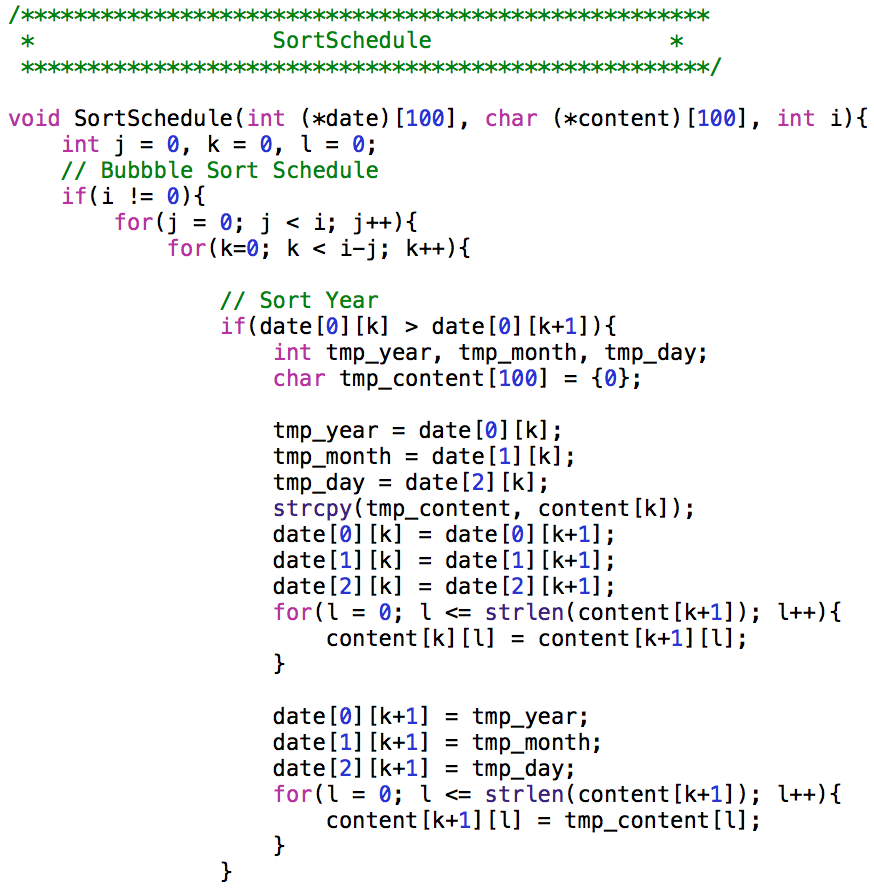
위의 소스코드는 ScheduleManagement함수에서 일정을 보는 경우이다. 먼저 일정의 존재 유무를 검토한 뒤 일정이 존재하면 ViewSchedule함수로 넘어가고, 만약 없다면 일정 없음 구문을 출력하고 SchduleManagement의 Main menu로 넘어간다.



위의 소스코드는 ScheduleManagement함수에서 Main menu로 나가는 경우와 구문 오류처리이다. 4를 입력하면 함수를 반환하고 Main menu로 돌아간다. 또한.메뉴에 나온 선택 예시 외에 다른 구문을 입력하면 구문 오류 처리를 하고 다시 입력 받는다.

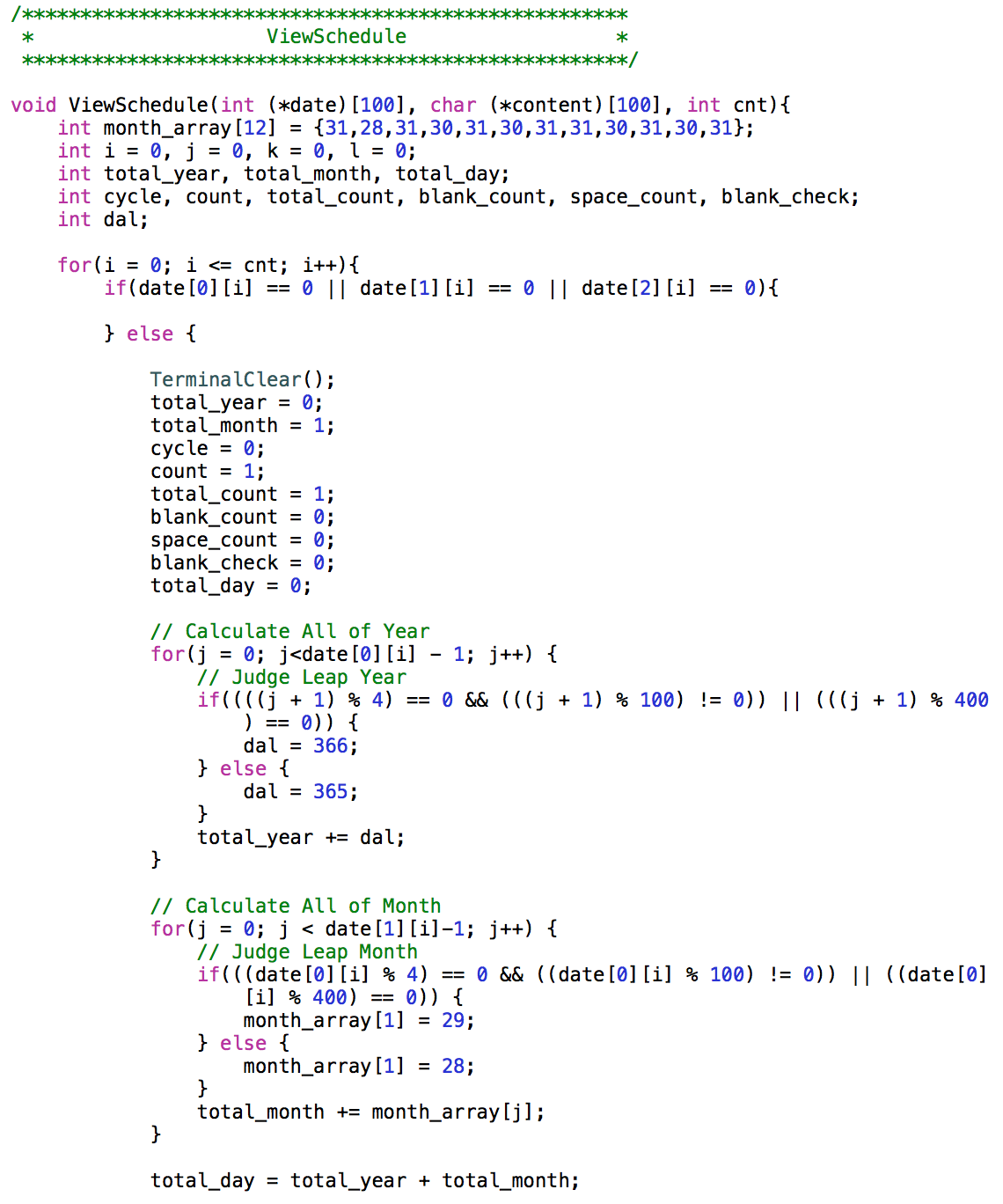


ClearSchedule함수는 인자 값을 통해 해당 위치의 일정을 모두 초기화한다. 날짜의 경우 모두 0, 내용의 경우 Null문자(\0)로 초기화한다.

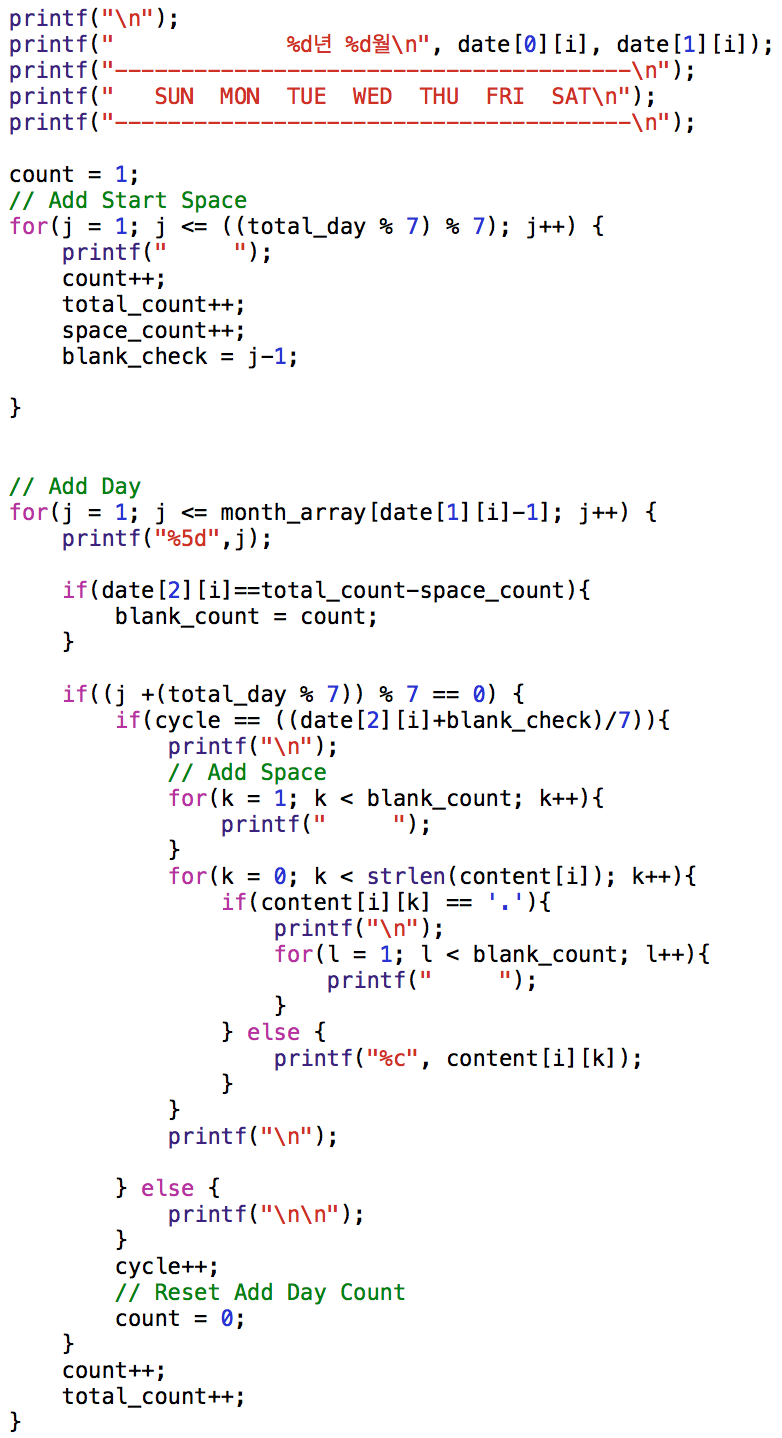


SortSchedule함수는 일정배열을 오름차순으로 정렬한다. 모든 정렬은 Bubble Sort로 이루어진다. 위의 경우는 년도를 정렬하는 과정이다.



위의 소스코드는 SortSchedule함수에서 월과 일을 정렬하는 과정이다. 모든 과정은 년도의 정렬 과정과 동일하며 정렬의 오차를 줄이기 위해 년-월-일-월-년 순으로 정렬한다.

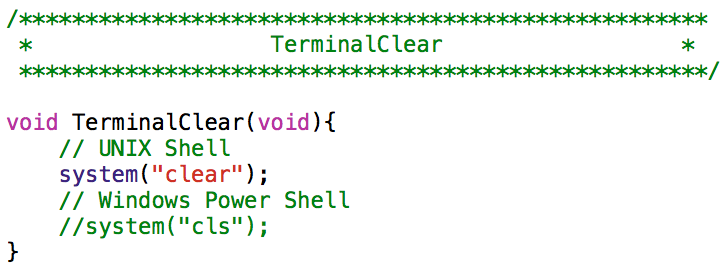
ViewSchedule함수는 일정을 보여주는 함수이다. date배열과 content배열에 저장된 일정들을 바탕으로 달력을 출력한다. 만약 일정이 0년 혹은 0월 0일로 되어있을 경우 일정 보여주기를 생략하며 그외의 경우 일정을 1년 1월 1일부터 연산하여 각 날짜를 출력한다. 또한 윤년을 계산하여 윤년일 경우 2월 29일로 처리한다. 마지막으로 모든 일정 바로 밑에 해당 일정의 내용을 출력한다.



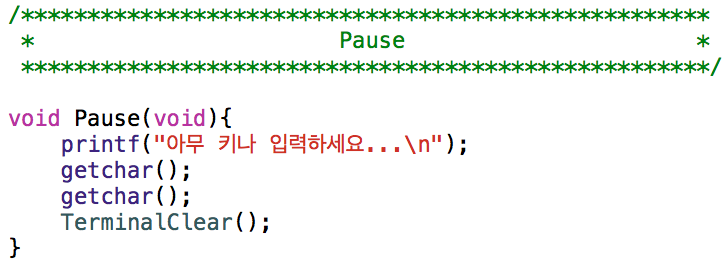
위의 소스코드는 ViewSchedule함수에서 달력을 출력하는 소스코드이다. 7일을 주기로 회전 수를 계산하며 일정 내용의 출력 위치는 회전수의 위치에 출력한다. 또한 출력할 때에는 content배열에서 해당 날짜의 일정 중 구분자 점(.)을 읽어서 만약 점(.)이 있을 경우엔 개행 후 추가로 일정을 출력한다. 만약 없을 경우엔 개행 과 동시에 날짜를 이어서 출력한다.



위의 소스코드는 ViewSchedule함수에서 일정 출력 과정 중 마지막 회전 수의 예외처리를 보강하는 과정과, 마무리하는 과정이다. 회전수가 최대일 경우 회전 수를 미처 따라가지 못하고 출력 과정을 마무리하는 경우가 존재하여 해당 문제를 해결하기 위한 Trouble Shooting이다. 이 과정은 앞서 말한 일정 출력 과정과 동일하며 단 마지막 회전수의 문제를 보강하기 위한 소스코드이다. 이후 모든 출력이 마무리 되면 키를 입력 받아 다음 일정을 계속해서 출력한다. 만약 일정이 더이상 없을 경우 일정 없음 구문을 출력 후 함수를 종료한다.



TerminalClear함수는 터미널을 초기화하는 기능을 하는 함수이다. Linux나 OS X, Cygwin 환경에서는 clear을, Windows에서는 cls를 사용한다.



Pause함수는 프로그램을 일시정지하는 기능을 하는 함수이다.