

## 후위식을 계산하는 프로그램

5671144-강범창

### 1. 후위식을 입력 받는다.

```
char inputTerm[100];

printf("후위식을 입력해주세요. 숫자 나 연산자를 입력해주세요. 단 괄호와 문자는 불가능합니다. Wn");
printf("Current Case => [0 1 +] [0 1 + ] [ 0 1 + ] [01+] Wn");
printf("Error Case => [ddf +] [ (1+2) ] Wn");
printf("Postfix expression: ");
scanf("%[^Wn]s", inputTerm);    // ( 공백 포함 문자열 입력 )
```

후위식을 입력 받는 코드입니다. 프로그램의 사양( 문자, 괄호 사용 불가.)의 내용과 예시를 프린트 해 놓았습니다. 그리고, 입력을 받을 때 공백 문자를 포함하기위해 scanf 에서 서식 지정자를 작성해주었습니다.

### 2. 입력 받은 후위식을 계산하고 출력한다.

```
for (int i = 0; i < strlen(inputTerm); i++) {
    if (inputTerm[i] != ' ') {
        ch = inputTerm[i];

        if (ch != '+' && ch != '-' && ch != '*' && ch != '/')
        {
            value = ch - '0';
            push(stack, value);
        }
        else
        {
            op2 = pop(stack);
            op1 = pop(stack);
            switch (ch)
            {
                case '+':
                    push(stack, op1 + op2);
                    break;
```

```

        case '-':
            push(stack, op1 - op2);
            break;
        case '*':
            push(stack, op1 * op2);
            break;
        case '/':
            push(stack, op1 / op2);
            break;
    }
}
}
}

```

문자열에서 공백이 아닐 경우 연산자인지, 아닌지를 보고 각각 숫자라면 동적 배열에 넣어둔 다음, 연산자가 왔을 때 pop을 하여, 연산자를 계산한후 다시 배열에 넣는 과정을 반복하여 후위식을 계산합니다.

### 3. 오류가 있으면 이를 처리한다.

```

for (int i = 0; i < strlen(inputTerm); i++) {
    if (inputTerm[i] != ' ') {
        ch = inputTerm[i];
        if (isCurrentInput(ch) != 1) {
            fprintf(stderr, "Error: Invalid character\n");
            exit(1);
        }
    }
}
}

```

유저에게 인풋을 받은 후 동작을 하며, 위와 같이 공백을 없애 주는 조건을 걸어준 다음, input의 값이 유효하지 않을 경우 0을 리턴 하는 isCurrentInput 함수를 통해 유효하지 않은 경우 프로그램을 중지 합니다.

isCurrentInput 함수입니다.

```

int isCurrentInput(char string)
{
    if (string == '+' || string == '-' || string == '*' || string == '/')
    {
        return 1;
    }
}

```

```
}  
  
if (isdigit(string))  
{  
    return 1;  
};  
  
return 0; // 위의 조건에 포함되지 않으면 error처리  
}
```

올바르지 않은 값을 체크하는 함수입니다. 받아온 값이 숫자가 아니거나  
[+ , - , \* , /] 오퍼레이터 가 아니라면 올바르지 않은 값이라 정하고  
0( error ) 를 리턴 합니다.