[시스템프로그래밍기초 팀 프로젝트 일지\_회차]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 일시 | 2019년 11월 13일 13:00 ~ 15:00 | | | |
| 팀원 | 안세윤 | 2019064548 | 윤성빈 | 2019082206 |
| 이다은 | 2019036380 | 임단비 | 2019056462 |
| 토의내용 | **<팀 프로젝트 계획>**  - 12월 6일까지 소스코드 구현 & 활동보고서 작성 마무리하기  - 12월 13일까지 최종 보고서 완성 및 최종확인하기  - 매주 수요일 13-15시/목요일 15-24시 토의 및 코드 작성  - 구현 완성할 때 마다 일지 작성  - 팀 프로젝트 매 활동 전까지 소스코드 작성에 필요한 내용 각자 공부해오기  **<일정>**  **1주차\_과제 분석 및 계획, 역할분담, 과제요건에 필요한 내용 학습**  11/13 - 과제 요건 별 학습사항 파악, 역할분담, 길이 제한 없는 문자열, 수식 표기법 변환함수 구현방법 탐구  11/14 – 뼈대코드(구조체 관련) 작성  **2주차\_뼈대코드 완성, 후위표기법 & 스택 구현**  11/18 – 뼈대코드(연결리스트 관련) 작성  11/20 - 후위표기법 구현  11/21 - 스택 구현  **3주차\_덧셈, 뺄셈, 곱셈함수 구현**  11/25 – 덧셈함수 구현  11/27 – 뺄셈함수 구현  11/28 – 곱셈함수 구현  **4주차\_ 나눗셈 함수 완성 & 프로그램 최적화 및 코드 마무리, 최종 보고서 작성**  12/4 – 나눗셈함수 구현  12/5 - 프로그램 최적화에 대해 탐구 & 소스코드 마무리  12/6 - 최종 보고서 작성  **<역할분담>**   |  |  | | --- | --- | | 구현 설명 | 임단비 | | 일지 작성 | 안세윤/윤성빈 | | 최종 보고서 작성 | 이다은 | | 소스코드 작성 | 팀원 전체 |   **<주요내용: 입력 문자열의 길이 제한에 대한 탐구>**  무한소수계산기를 구현하기 위해 어떤 것이 필요할지 토의하였다.  -heap: 저장 메모리 공간에 한계가 있으므로 이를 보완하는데 필요할 것 같다.  -stack: 중위, 후위 표기법 등 수식 표기법 변환함수 구현에 사용하는 것이 좋을 것 같다.  -트리구조를 이용해 입력 값을 가지고 수식표기법을 구현하는 것.  -입력 문자열 길이에 제한이 없게 하려면 int, double, long 등 사용이 어렵기 때문에 해결책을 찾아야 한다. → malloc이나 calloc 사용  -malloc을 써도 배열에 제한이 있기 때문에 malloc 또한 한계가 있을 것  **<브레인스토밍>**  - 전역변수와 전적변수는 메모리를 계속 차지하고 있으니 지역변수와 매개변수를 이용하는 것이 좋다.  - stack에서 top의 초기값은 –1이다, 그런 이유는 인덱스의 0번째 자리에도 값이 들어가야 하기 때문이다.  - 구현해야 할 함수는 중위에서 후위로 바꾸는 함수, 숫자를 저장할 문자열 함수, 부호를 만나면 넣을 스택 등이 있다.  - 연산자가 들어왔는지 피연산자가 들어왔는지 구분해주는 함수도 필요하다.  - 배열을 사용하면 무한수가 받아지기 어렵다.  - 무한수이기 때문에 count 같은 걸 하면 안 된다.  - integer, string과 같은 자료형을 이용하면 overflow가 발생하기 때문에 연결리스트는 동적 할당을 해야 한다.  - 따라서 문자열을 연결리스트로 받아야 한다. | | | |
| 결론 | 문자열을 연결리스트로 받아 동적 할당을 해야 한다.  stack을 이용해서 중위표기법을 후위표기법으로 변환해야 한다.  -> 다음 모임 때까지 연결리스트와 스택, 구조체 개념을 이해해온다. 이해한 개념이 무한 수 계산기 속에 어떻게 구현해야 할지 생각해온다. | | | |