

자람 신입생 c 스터디 가이드라인

intro

이 스터디는 프로그래밍 학습 경험이 없는 신입생을 대상으로 하며, 대표적 프로그래밍 언어인 c의 사용법과 자료구조, 알고리즘의 기초를 공부하며 앞으로의 학습에 토대가 될 기본적인 프로그래밍 논리를 정립하는 것을 목표로 한다.

다만 이 스터디가 단순히 프로그래밍 언어의 사용법을 답습하는 것에 그치지 않고, 실제 프로그램의 실행 과정이나 하드웨어 구조 등 프로그래밍 전반에 대한 이해가 이루어지도록 개별적 개념에 대한 상세한 설명을 수반할 것을 권장한다.

아래 기재된 목차는 전체 스터디 기간 동안 스터디원이 습득하기를 권장하는 개념의 목록이며, 그 순서는 스터디원의 이해도나 진행할 과제에 필요한 선행지식에 따라 임의로 소폭 변경할 수 있다. 또한, 스터디장의 판단에 따라 권장 사항 바깥의 추가적 학습을 이어가기 바란다.

syllabus

1. 환경 구축, 컴퓨터 프로그래밍과 프로그래밍 언어의 이해

본격적인 학습에 앞서서 c 언어 개발환경을 구축하고 Hello World를 출력해본 후 프로그래밍과 그 언어, 그리고 c 프로그램 실행 과정 등의 배경 지식을 습득한다.

2. 자료형, 변수의 종류와 메모리 구조

c의 자료형과 형변환, 변수의 종류를 공부하고 변수가 어떻게 저장되는지 이해한다.

3. 조건문, 반복문

조건문과 반복문의 사용 방법을 숙지하고 연산자의 종류, 우선순위 등을 공부한다. 학습 내용을 바탕으로 별찍기, 구구단 출력 등의 과제를 해결하며 개념을 숙달한다.

4. 함수의 정의와 호출 과정

함수의 개념과 함수를 선언하는 방법을 학습한다. 이후 재귀함수나 반복문을 사용하여 피보나치 수열 출력 함수 설계 등의 과제를 해결해보며 각 방식의 장단점에 대해 생각해본다. (추가 도전과제) 앞서 제작한 함수를 동적계획법을 적용해 설계해보며 효과적인 함수 설계 방식에 대해 생각해본다.

5. 포인터와 배열의 이해

메모리의 주소를 가리키는 자료형인 포인터와, 동일한 타입의 데이터들의 그룹인 배열형을 학습한다.

6. 문자열의 이해, 문자열 함수 사용과 직접 구현

c언어에서 문자를 표현할 때 사용하는 문자열 형에 대해 이해하고 문자열을 다룰 때 사용하는 함수를 사용해보며, 이를 직접 구현해본다.

7. 구조체와 동적 메모리 할당

여러 데이터를 하나로 묶는 사용자 데이터형인 구조체에 대해 학습한다. 이후 다양한 데이터를 담은 구조체를 선언하고 인스턴스를 생성하여 구조체 내 데이터에 접근하는 실습을 진행한다. 또 동적 메모리 할당의 개념과 그 과정을 살펴보고 예제 코드를 작성하며 학습내용을 습득한다.

8. (추가 도전과제)링크드 리스트의 구현을 통한 자료구조의 이해

대표적인 선형 자료구조인 링크드 리스트(linked list)를 통해 자료구조에 대해 이해하고, 구조체를 이용해 이를 구현해보며 그간의 학습내용을 점검한다.