**【2024 PNU SW스터디그룹 최종(중간)보고서】**

**[프로젝트명] C/C++ 코딩테스트 준비**

상징, 로고, 등록 상표, 엠블럼이(가) 표시된 사진

자동 생성된 설명

[C언어학자와 제자들]

[정보컴퓨터공학부/202155587] [이정원]

[정보컴퓨터공학부/202211532] [임선우]

[산업공학과/202127542] [조수진]

[정보컴퓨터공학부/202455445] [이채훈]

**[프로젝트 보고서 목차]**

**1.** **서론**

* 스터디 목적 및 배경

**2. 스터디 개요**

* 기간 및 참여 인원
* 스터디 목적
* 진행 방식
* 활용 자료

**3. 스터디 진행 과정**

* 주차별 활동 내용
* 학습 방법

**4. 주요 성과**

* 개인 역량 향상
* 협업 능력 강화
* 성과 공유

**5. 어려움과 극복 과정**

* 시간 조율 문제
* 어려운 개념 학습

**6. 향후 계획 및 제언**

* 계속적인 학습 방향
* 스터디 모델 발전 방안
* 실전 대비 및 네트워크 유지

**7. 결론**

* 스터디의 의의와 성과
* 향후 활동 계획

**1. 서론**

10주간의 C언어 스터디는 프로그래밍의 기초를 다지고, 코딩 테스트 준비를 위한 실력을 키우기 위해 계획되었습니다. 본 스터디는 C언어에 대한 이론적 이해와 문제 해결 능력을 동시에 향상시키는 것을 목표로 하였으며, 참여자들의 적극적인 학습과 협력을 통해 성공적으로 마무리되었습니다. 특히, 스터디를 통해 개념의 체계적 정리가 이루어졌고, 팀원 간의 상호 작용을 통해 학습 효과를 배가시킬 수 있었습니다. 스터디는 단순히 학습에 그치지 않고, 참여자들이 주도적으로 학습 방향을 설계하고 개선하며 보다 나은 학습 환경을 만들기 위한 노력을 포함했습니다.

스터디의 과정은 단순히 주어진 자료를 암기하거나 문제를 푸는 것을 넘어, 실질적인 코드 작성과 실행을 반복하며 프로그래밍의 기본 원리를 깊이 있게 이해하고 응용하는 데 중점을 두었습니다. 이 보고서에서는 스터디의 진행 과정, 활용한 자료, 성과 및 아쉬운 점, 그리고 앞으로의 계획에 대해 상세히 정리하고자 합니다.

**2. 스터디 개요**

* **기간:** 10주 (매주 1회, 약 2시간)
* **참여 인원:** 총 4명
* **목적:**
  + C언어 기본 개념 이해
  + 코딩 테스트 대비를 위한 문제 풀이 능력 강화
  + 실습과 협업을 통한 프로그래밍 역량 향상
  + 문제 해결 과정에서의 창의적 사고력 배양
  + 프로그래밍 기초에 대한 확고한 자신감 형성
* **진행 방식:**
  + 매주 한 명의 리더가 개념 설명 및 문제 풀이를 주도
  + 이론 학습 후 관련 문제 풀이 진행
  + 주간 학습 리뷰 및 피드백 공유
  + 개인별 학습 결과 기록 및 발전 점검
  + 실습 코드와 결과물을 문서화하여 공유
* **활용 자료:**
  + 학교에서 제공한 C언어 설명 자료
  + CodeTree 문제집
  + 온라인 학습 리소스 (예: 공식 문서 및 강의 영상)
  + 추가 서적 및 참고자료 (예: "C Programming Language" by Kernighan & Ritchie)
  + GitHub를 통한 코드 관리 및 공유

**3. 스터디 진행 과정**

**3.1. 주차별 활동 내용**

* **1주차**:  
  첫 모임에서는 스터디의 전체 계획을 공유하고, 학습 목표를 설정했습니다. 이후 C언어의 기본 구조와 변수의 개념을 학습했습니다. 특히, 변수의 선언과 초기화, 데이터 타입의 활용 방법 등을 실습하며 기초를 다졌습니다.
* **2주차**:  
  조건문과 반복문을 학습했습니다. if-else, switch-case와 같은 조건문과 for, while, do-while 반복문을 활용한 간단한 프로그램을 작성했습니다. 이 과정에서 논리 연산자와 비교 연산자의 역할을 이해했습니다.
* **3주차**:  
  함수의 구조와 활용 방법을 익혔습니다. 함수 선언, 호출, 그리고 반환값을 활용하는 방법을 중점적으로 다루었으며, 재사용 가능한 코드를 작성하는 연습을 했습니다.
* **4주차**:  
  배열과 문자열을 학습했습니다. 배열의 크기 설정, 다차원 배열의 활용, 그리고 문자열 처리를 위한 라이브러리 함수(strcpy, strlen, strcmp 등)를 실습했습니다.
* **5주차**:  
  포인터와 동적 메모리 할당의 개념을 배웠습니다. 포인터 변수의 선언 및 활용 방법, 그리고 malloc, free를 이용한 메모리 관리 기법을 익혔습니다. 메모리 관련 오류를 디버깅하며 실습의 난이도가 높아졌습니다.
* **6주차**:  
  파일 입출력(File I/O)을 학습했습니다. 파일을 읽고 쓰는 방법, 텍스트 파일과 바이너리 파일의 차이점, 그리고 파일 포인터의 활용 사례를 중심으로 실습을 진행했습니다.
* **7주차**:  
  코딩 테스트에서 자주 나오는 문제 유형을 분석하고 관련 문제를 풀었습니다. 예를 들어, 정렬 알고리즘과 탐색 알고리즘을 활용한 문제를 다루며 효율적인 코드 작성법을 논의했습니다.
* **8주차**:  
  재귀 함수의 원리와 활용법을 학습했습니다. 특히, 재귀를 활용한 피보나치 수열 계산, 팩토리얼 계산 등을 실습하며 문제 해결 능력을 강화했습니다.
* **9주차**:  
  복합 데이터 구조(구조체, 연결 리스트 등)를 학습했습니다. 데이터의 효율적인 저장과 관리 방법을 배우며, 구조체와 포인터를 결합한 문제를 해결했습니다.
* **10주차**:  
  스터디 전반에 걸친 내용을 복습하고, 종합 문제 풀이를 통해 학습 결과를 점검했습니다. 구성원 각자가 난이도가 높은 문제를 제시하며 서로의 실력을 평가하고 피드백을 주고받았습니다.

**3.2. 학습 방법**

* **리더 중심 학습:** 매주 리더가 발표를 통해 학습 내용을 정리하고 참여자들에게 설명. 팀원들이 각자 돌아가며 리더 역할을 맡아 책임감을 높임. 리더는 학습 자료를 사전에 준비하고, 예상 질문에 대한 답변도 준비.
* **문제 풀이:** 학습한 개념을 적용하기 위한 CodeTree 문제 풀이. 단계별로 난이도를 높여가는 방식. 어려운 문제는 팀별로 나누어 접근하여 해결.
* **토론과 피드백:** 문제 풀이 후 각자 다른 접근법을 공유하고 개선점을 논의. 협력과 상호작용을 통해 깊이 있는 학습 도모. 이를 통해 창의적이고 효율적인 문제 해결 방안을 모색.
* **온라인 보조 학습:** 모임 외 시간에는 각자 온라인 학습 플랫폼을 활용하여 추가 학습 진행. 이를 통해 개별 약점 보완 및 추가적인 심화 학습 가능.
* **실습 기반 학습:** 코드 작성 및 실행 과정을 반복하며 실질적인 프로그래밍 능력을 향상시킴.
* **코드 리뷰:** 작성된 코드를 서로 검토하며 최적화 방안을 제시하고 실수를 발견하여 수정.

**4. 주요 성과**

**4.1. 개인 역량 향상**

* 참여자 모두 C언어의 기본 개념에 대한 이해도를 높임. 데이터 타입, 조건문, 반복문 등 기초부터 심화 개념까지 폭넓게 학습. 개념 학습을 실제 문제 해결에 연결.
* 코딩 테스트 문제 풀이 속도와 정확성 향상. 제한 시간 내에 문제를 해결하는 능력이 크게 향상됨. 실시간 피드백을 통해 코드의 정확성과 효율성을 높임.
* 포인터와 동적 메모리 할당 등 어려운 개념을 실습하며 실질적 이해도 증대. 특히, 포인터 배열과 함수 매개변수 전달 방식에서 성과를 보임. 이를 활용한 복잡한 문제 해결 능력 개발.
* 알고리즘 분석 능력 배양. 시간 복잡도와 공간 복잡도를 평가하며 최적의 해결책을 찾는 방법 학습.

**4.2. 협업 능력 강화**

* 팀원 간의 적극적인 의견 교환과 문제 해결 과정에서 협업 능력 향상. 각자 맡은 역할에 따라 책임감을 가지고 학습에 임함. 협력을 통해 개별 약점을 보완하고 팀의 강점을 극대화.
* 다른 관점에서 문제를 분석하고 접근하는 방법 학습. 다양한 해결 방식을 탐구하며 창의적 사고력 증대. 문제 해결 과정에서 도출된 해결책을 문서화하여 팀의 지식 자산으로 남김.

**4.3. 성과 공유**

* 참여자들이 공통적으로 이전보다 문제 해결 능력이 크게 향상되었다고 평가. 실력 향상을 자각하며 성취감을 얻음. 어려운 문제를 해결했을 때의 만족감이 학습 동기를 강화.
* 실질적인 코딩 테스트 모의 평가 점수 상승 확인. 정량적 데이터로 학습 성과를 확인할 수 있었음. 팀원 간의 공동 목표 달성을 위한 노력이 가시적인 결과로 나타남.
* 학습 자료와 코드 결과물을 문서화하여 이후 학습 참고 자료로 활용 가능.

**5. 어려움과 극복 과정**

**5.1. 시간 조율 문제**

* 시험 기간 중 몇몇 참여자의 일정 충돌 발생. 스터디 일정과 개인 학업 간의 균형을 맞추는 데 어려움이 있었음.
* 이를 해결하기 위해 온라인 학습 플랫폼과 그룹 채팅을 활용하여 대체 학습 진행. 녹화 자료와 공유 문서를 통해 자율적으로 학습을 보완. 팀 내 유연성을 높여 참여를 지속적으로 독려.

**5.2. 어려운 개념 학습**

* 포인터와 동적 메모리 할당은 참여자들이 특히 어려움을 느낀 부분. 초반에는 개념 이해에 시간이 걸렸으나, 반복 학습과 문제 풀이로 극복.
* 이를 극복하기 위해 추가 자료를 찾아보고, 관련 예제 문제를 반복적으로 풀며 보완. 특히, 각종 포인터 관련 실습과 메모리 디버깅 툴 사용이 효과적이었음. 협업을 통해 부족한 부분을 명확히 하며 보완.
* 심화된 예제를 통해 포인터와 동적 메모리의 실질적 활용 능력을 강화.

**6. 향후 계획 및 제언**

* **계속적인 학습:** 스터디 이후에도 개인적으로 코딩 테스트 준비를 이어갈 계획. 추가적으로 알고리즘과 자료 구조에 대한 심화 학습 필요. 예를 들어, 그래프 및 동적 프로그래밍 문제 학습.
* **고급 학습 자료 활용:** 온라인 강의와 서적을 활용하여 C언어 외에도 다양한 언어를 학습. Python과 Java를 활용한 문제 풀이 스터디 계획. 이를 통해 다양한 언어 간의 차이를 이해하고 응용 능력을 강화.
* **스터디 모델 발전:** 다른 언어(예: Java, Python)로 스터디 범위를 확장하거나, C언어를 활용한 프로젝트 진행 제안. 예를 들어, 간단한 데이터 분석 프로그램 개발 스터디.
* **실전 대비:** 향후 모의 코딩 테스트를 정기적으로 진행하며 실전 감각 유지. 대회 참여를 통해 동기를 부여받는 것도 좋은 방법. 추가적으로 코드 리뷰 문화를 도입하여 코드 품질을 높이는 방안도 고려.
* **참여자 네트워크 유지:** 스터디 종료 후에도 지속적인 학습 커뮤니케이션을 이어가며 서로의 성장을 도모.

**7. 결론**

10주간의 C언어 스터디는 참여자들의 적극적인 학습과 협력을 통해 목표를 달성한 성공적인 활동이었습니다. 기본 개념 학습부터 응용 문제 풀이까지 체계적으로 진행된 이번 스터디는 참여자들에게 큰 자신감을 심어주었으며, 앞으로의 프로그래밍 학습에도 탄탄한 기초가 될 것입니다. 또한, 이번 경험은 협업의 중요성을 재확인하는 기회였으며, 팀원들과의 관계 또한 더욱 돈독해졌습니다. 학습 과정에서 축적된 경험과 성과는 이후 프로그래밍 학습 및 프로젝트에서도 유용하게 활용될 것입니다. 이 보고서를 통해 향후 유사한 스터디를 기획하는 데 있어 유익한 참고 자료가 되기를 바랍니다.

스터디의 성과를 기반으로 참여자들은 추가적인 학습과 프로젝트 활동을 통해 실질적인 프로그래밍 능력을 더욱 심화할 계획입니다. 이를 통해 지속적으로 발전하고, 장기적인 목표를 이루기 위한 초석으로 삼고자 합니다.