

**「Korea Polytechnic University」**

**프로그래밍 과제 노트**

**2019-01학기**

<b>교과목 담당교수</b>	<b>컴퓨터공학부 박정민</b>
<b>학과</b>	<b>컴퓨터공학과</b>
<b>학번</b>	<b>2017152049</b>
<b>이름</b>	<b>정하림</b>

## 1. 온라인 강의 요약정리

### 1-1 Part1-1장. C언어의 소개와 프로그램 작성 방법

#### ■ 소프트웨어란?

- 소프트웨어
  - ▶ 명령을 받아 응답하는 컴퓨터 프로그램
- 소프트웨어 개발 프로세스
  - ▶ 소프트웨어를 만들 때 요구사항 명세화부터 분석, 설계, 구현, 테스트, 유지보수까지 소프트웨어를 만드는 절차

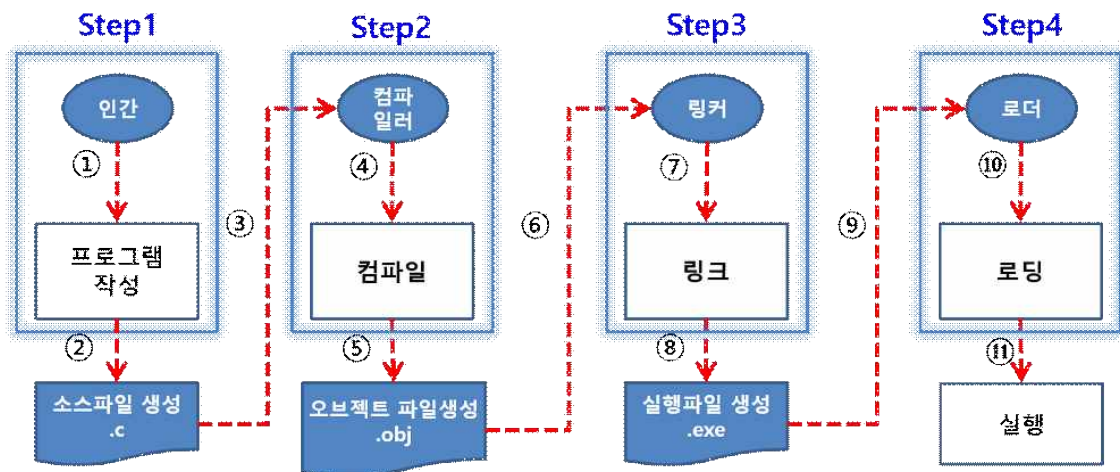
#### ■ 엔지니어와 프로그래머의 차이

- 엔지니어
  - ▶ 프로젝트 팀의 전문가, 소프트웨어 개발 프로세스를 관리
- 프로그래머
  - ▶ 작은 규모의 프로그램을 개발하는 사람

#### ■ C언어란 무엇인가?

- 인간과 컴퓨터가 의사소통을 하기 위한 프로그래밍 언어
- 벨 연구소에서 만들어 졌으며 A언어부터 B를 거쳐 배우기 쉽고 프로그램의 이식성을 높인 C언어가 됨
- 사람이 만든 소스 파일을 기계어로 번역해주는 번역기인 C컴파일러를 사용하여 의사소통을 수행

#### ■ 프로그램 작성 4단계



- ▶ 1단계 : 인간이 C언어로 프로그램을 소스파일(\*.c)로 작성
- ▶ 2단계 : 소스파일(\*.c)을 컴파일러가 컴파일을 통해 기계어파일(\*.obj)로 변환
- ▶ 3단계 : 기계어파일(\*.obj)을 링커가 라이브러리의 표준 함수와 링크하여 실행파일(\*.exe)로 변환
- ▶ 4단계 : 실행파일(\*.exe)을 로더가 로딩(메모리에 적재)하여 CPU가 프로그램을 실행함
- ▶ 2단계와 3단계의 과정을 묶어 빌드라고 함
- ▶ 1단계부터 4단계까지 한 소프트웨어 안에서 도와주는 것을 IntegratedDevelopEnviroment라고 함

## ■ C언어의 기본 구조

- 주석
  - ▶ 소스코드에 컴파일러에 의해 기계어로 변환 되지 않는 설명문
  - ▶ `//` : 표기 뒤의 한 줄을 전부 주석처리 해줌
  - ▶ `/* 내용 */` : 여러 줄 주석, 내부에 중복 사용 금지
- 전처리기와 헤더파일 `#include <stdio.h>`
  - ▶ 전처리기 - `#` : 컴파일을 수행하기 전에 먼저 처리하는 기호
  - ▶ `include` : 뒤의 헤더파일을 전처리 과정에서 포함시키라는 의미
  - ▶ 헤더파일 - `<stdio.h>` : STanDard Input Output 표준 입출력을 담당하는 헤더파일
    - 뒤에 배울 `printf()` 나 `scanf()` 등 입출력에 필수적인 함수가 미리 정의되어 있는 라이브러리
- Main함수
  - ▶ 운영체제에 의해 맨 처음 실행되고 맨 마지막에 종료되는 필수 함수

```
int main (void)
{
    함수의 기능
    return 0;
}
```

- `int`는 메인함수의 반환 형태로 `integer`(정수)형태로 함수 실행후 값을 반환하겠다는 뜻이다.
- `main`은 함수의 이름이다. 한 프로젝트 안에 하나만 존재가능.
- `(void)`는 함수 안에 값을 전달해 줄 수 있는 매개변수(입력)로 `void`는 비어있다는 뜻이다.
- `return 0;` 위에 제시한 `int` 반환형태로 정수값인 0을 반환하며 함수를 마친다.
- 세미콜론(`:`) : 세미콜론은 문장의 끝을 의미하는 마침표의 의미이다.

## ■ 모니터에 데이터 출력 - `printf()` 함수

- `PRINTFomatted` 문자열을 화면에 출력하는 함수

```
printf("내용 %d\n", 1);
```

- ▶ 위와 같은 형태로 사용하며 함수의 큰따옴표("")안에 원하는 내용을 적으면 모니터에 출력해준다
- ▶ `%d`는 정수를 입력하겠다는 뜻이다.
- ▶ `\n(Wn)`은 제어문자로 `NewLine`을 뜻하며 엔터의 기능을 한다.
- ▶ 제어문자 : 커서의 위치조정같은 특수한 역할을 하는 문자
  - `\a` : alarm 알람소리 출력
  - `\n` : newline 개행
  - `\t` : tab 탭키와 동일하게 빈칸 생성
  - `\W` : 제어문자와 동일하여 쓸 수 없었던 `\(W)` 문자를 출력해줌(작은따옴표 "", 큰따옴표 "" 동일)

- ▶ 서식문자 : 서식화된 출력문자
  - %d : 십표 뒤의 숫자를 10진수(Decimal)로 치환, 수식 또한 계산하여 치환해줌
  - %f, %lf : 숫자를 10진수 실수(Float)로 치환 %뒤에 .1을 붙이면 소수 첫째자리까지만 출력  
실수에 데이터손실이 없도록 더 큰 소수를 표현 가능한 %lf(LongFloat)를 주로 사용
  - %o : 숫자를 8진수(Octal)로 치환(양수만 표현가능)
  - %x : 숫자를 16진수(heXadeciaml)로 치환 %뒤에 #을 붙이면 생략된 데이터도 출력가능(양수만 표현가능)
  - %c : 한 문자를 'A' 형태로 입력받아 출력
  - %s : 문자열을 "string" 형태로 입력받아 출력시켜줌 문자열의 마지막은 자동으로 \0(End Of String)이 들어감
  - %u : 숫자를 10진수 양수로 치환(양수의 범위가 %d의 2배)
- ▶ 필드 폭 지정
  - %03d : 000 3칸을 확보하고 오른쪽정렬 남은 앞자리는 0으로 채움
  - %-3d : 3칸을 확보하고 왼쪽 정렬
  - %+3d : 3칸을 확보하고 오른쪽 정렬, 양수는 +, 음수는 -기호 붙여줌

## ■ 키보드로 데이터 입력 - scanf() 함수

- SCANFormatted 키보드를 통해 데이터를 입력받는 함수

```
scanf("%d", &a);
```

- ▶ 위와같은 형태로 a라는 변수의 주소값에 정수형식으로 입력받아 저장시킨다.
- ▶ printf에서 사용했던 것 같이 서식문자를 사용하여 각종 숫자나 문자를 입력할 수 있다.

### ■ 변수와 변수의 시작주소

#### • 변수

```
int a;
```

- ▶ 메모리 공간의 이름, 메모리 안에 값을 저장할 수 있는 공간
- ▶ 정수형 변수 : char, short, int, long
- ▶ 실수형 변수 : float, double, long double
- ▶ 그냥 선언만 해놓으면 쓰레기값이 들어가므로 꼭 =(대입연산자)를 통해 초기화 해주어야함

#### • 변수 선언 시 유의사항

- ▶ 변수의 선언은 제일 앞쪽에 한다.
- ▶ 변수의 이름은 의미있게 짓는다.
- ▶ 특수 기호, 공백 문자, 맨처음에 숫자를 사용하면 안되고 대소문자를 구분하여 사용한다.
- ▶ 이미 정해져있는 키워드들은 사용 불가능하다.

- 변수의 시작주소 : &라는 주소연산자를 사용하여 변수가 할당된 메모리 공간의 시작주소를 나타낼 수 있다.



## 2. 심화문제 중점 문제 분석, 디버깅

### ■ 분석내용1

- 상세내용 1
- 상세내용 2
- 상세내용 3

### ■ 분석내용2

- 상세내용 1
- 상세내용 2
- 상세내용 3

### 3. 자기성찰 및 평가

#### 3-1 1,2주차-자기성찰 및 평가

##### ■ 수업 및 실습을 통해서 배운 내용

- 소프트웨어가 어떤 것인지
- C언어의 유래와 목적
- C언어에서의 프로그램 작성 4단계
- IDE툴인 비주얼스튜디오를 활용하여 프로젝트를 생성, 컴파일, 실행, 디버깅하기
- 사용하지 않는 .c파일을 비활성화 하는 방법
- 기계어로 전환하는 단계에서 번역과 통역의 차이
- 전처리기(pre-processor) : #으로 시작하여 독립적으로 컴파일 전에 처리되는 지시자
- 유니코드 : 전세계 언어를 모두 디지털로 표현하기 위한 코드체계
- C언어의 기본 구조
- printf(), scanf()함수의 사용법과 제어, 서식문자의 종류
- 변수와 변수의 선언 시 유의사항
- 변수의 시작 주소와 &연산자
- 진수의 변환

##### ■ 느낀점 (자유롭게 기술)

- C언어를 이전에 다른 수업에서 접한 적 있었는데 그때는 모르고 있었던 새로운 사실들을 알게 되어서 좋았고 시스템함수를 사용하여 메모장을 열거나 소스파일을 프로젝트에서 제외시키는 등 흥미로운 기능들을 많이 알려주셔서 재미있었다.
- 교수님이 알려주신 2진수를 16진수, 10진수로 바꾸거나 10진수를 2진수로 바꾸는법을 어떻게 해결하면 좋을지 모르겠다. 관련 교재를 읽어보고 인터넷 검색을 활용하여 안보고도 짤 수 있도록 이해해놓아야겠다.
- C언어의 기반을 배우면서 이 부분부터 탄탄하게 익혀놓아야 앞으로 어려운 문제를 만나더라도 공부한 것들을 활용하여 문제를 해결할 수 있을 것 같다.
- 아직 이런 체계적인 과제노트를 정리하는데 미숙해서 빨리 익숙해져야겠다고 느꼈다.