

## Char1. 프로그래밍 언어 활용

### 데이터 타입 종류

- 정수 타입(integer type)
- 부동 소수점 타입(floating point type)
- 문자 타입(character type) : 작은 따옴표 안에 표시
- 문자열 타입(charater string type) : 큰 따옴표 안에 표시
- 불린 타입(Booleean type) : 기본값은 거짓 false
- 배열 타입(array type) : 중괄호{}안에 콤마,로 구분

### 변수 variable

컴퓨터가 명령을 처리하는 도중 발생하는 값을 저장하기 위한 공간, 변할 수 있는 값

예약어 : 정해진 기능을 수행하도록 이미 용도가 정해져 있는 단어를 가르치는 용어, 다른 용도로 활용 불가

### 기억 클래스 storage class

변수 선언 시 메모리 내에 변수의 값을 저장하기 위한 기억영역을 할당되는데 할당되는 기억 영역에 따라 사용 범위에 제한이 있다. 이러한 기억영역을 결정하는 작업을 의미

#### 1. 자동변수 automatic variable

함수나 코드의 범위를 한정하는 블록 내에서 선언되는 변수

#### 2. 외부 변수 external variable

현재 파일이나 다른 파일에서 선언된 변수나 함수를 참조하기 위한 변수

#### 3. 정적 변수 static variable

◆ 내부 정적 변수 : 함수나 블록 내에서 선언하는 변수

◆ 외부 정적 변수 : 함수 외부에서 선언하는 외부 정적 변수

#### 4. 레지스터 변수 register variable

메모리가 아닌 CPU 내부의 레지스터에 기억영역을 할당받는 변수

자주 사용되는 변수를 레지스터에 저장하여 처리속도를 높이기 위해 사용

| 종류       | 기억영역     | 예약어      | 생존기간 | 사용 범위 |
|----------|----------|----------|------|-------|
| 자동 변수    | 메모리(스택)  | Auto     | 일시적  | 지역적   |
| 레지스터 변수  | 레지스터     | Register |      |       |
| 내부 정적 변수 | 메모리(데이터) | Static   | 영구적  | 전역적   |
| 외부 정적 변수 |          |          |      |       |
| 외부 변수    |          | extern   |      |       |

Char aa='A' 문자형 변수 aa에 문자'A'를 저장한다.

## 표준 입출력 함수 input-output functions

키보드로 입력 받아 화면으로 출력할 때 사용하는 함수

<C언어>

|    |                   |   |
|----|-------------------|---|
| 입력 | Scanf(서식문자열, 주소)  | 키보드로 입력받아 변수에 저장하는 함수<br>한번에 여러 데이터를 입력 받을 수 있다.<br>서식문자열과 변수의 자료형은 일치해야 한다.<br>입력받을 데이터의 자료형, 자릿수 등을 지정할 수 있다. |
|    | Getchar()         | 키보드로 한 문자를 입력받아 변수에 저장하는 함수   |
|    | Gets()            | 키보드로 문자열을 입력받아 변수에 저장하는 함수<br>Enter을 누르기 전까지 한 문자열로 인식하여 저장   |
| 출력 | Printf(서식문자열, 변수) | 인수로 주어진 값을 화면에 출력하는 함수  |
|    | Putschar()        | 인수로 주어진 한 문자를 화면에 출력하는 함수   |
|    | Puts()            | 인수로 주어진 문자열을 화면에 출력한 후<br>커서를 자동으로 다음 줄 앞으로 이동하는 함수   |

서식 문자열

|    |               |
|----|---------------|
| %d | 정수형 10진수      |
| %o | 정수형 8진수       |
| %x | 정수형 16진수      |
| %u | 부호없는 정수형 10진수 |
| %c | 문자            |
| %s | 문자열           |
| %p | 주소            |

주요제어문자

| 문자   | 의미              | 기능                |
|------|-----------------|-------------------|
| \b   | Backspace       | 커서를 왼쪽으로 한 칸 이동   |
| \r   | Carriage return | 커서를 현재 줄의 처음으로 이동 |
| \0   | Null            | 널 문자 출력           |
| '\'' | Single quote    | 작은 따옴표 출력         |
| '\"' | Double quote    | 큰 따옴표 출력          |
| \a   | Alert           | 스피커로 벨 소리 출력      |
| \\   | backslash       | 역 슬래시를 출력         |
| \f   | Form feed       | 한 페이지를 넘긴다        |

| 코드   | 입력 데이터          | 풀이 과정  | 결과                         |
|--|-----------------|--|----------------------------|
| Scanf("%2d\$%4f",&i, &j);                    | 123\$45.678     | \$\$을 기준으로 앞뒤에 두 개의 서식문자열이 쓰여졌으므로 입력데이터 123\$45.678을 \$\$을 기준으로 나눈다.<br>서식 문자열이 %2d이므로 앞에 2자리까지만 저장. 따라서 12.<br>서식 문자열이 %4f이므로 소수점을 포함하여 앞에 4자리까지 저장. 따라서 45.6 | i:12,<br>j:45.6            |
| Char b[8];<br>Scanf("%2d %4f %s",&i, &j, b); | 123 45.678 love | 문자나 문자열을 입출력 받기 위해서는 우선 배열을 만들어 줘야 하므로 배열b를 만들어 준다.<br>입력 데이터가 빈칸으로 구분되어 나뉜다.<br>%2d이므로 12, %4f이므로 45.6, %s이므로 love  | i:12,<br>j:45.6,<br>b:love |
| Char b[6], c[8];<br>Scanf("%c %5c",b,c);     | Love me         | %c이므로 L, %5c이므로 나머지 ove me중 5문자 ove m  | b:L,<br>c:ove m            |
| Char b[6], c[8];<br>Scanf("%s %2s",b,c);     | Love you        | %s이므로 Love, %2s이므로 빈칸 이후 두 자리 yo   | b:Love,<br>c:yo            |

| 코드   | 입력 데이터       | 풀이 과정   |    |   |   |    |   |   |  |   |   |  |   |   |   |   |   |   |   |   |  |  |  |   |   |   |   |   |
|--|--------------|---|----|---|---|----|---|---|--|---|---|--|---|---|---|---|---|---|---|---|--|--|--|---|---|---|---|---|
| Printf("W'AW'는 문자로 %c, 아<br>스키 코드로 %d\n", 'A', 'A');   |              | 변수'A'를 받았는데 서식 문자열이 %d라면 문자 'A'에 해당하<br>는 아스키 코드 값을 정수형으로 출력<br>'A'는 문자로 A, 아스키 코드로 65  |    |   |   |    |   |   |  |   |   |  |   |   |   |   |   |   |   |   |  |  |  |   |   |   |   |   |
| Scanf("%d %1d %f", &i, &a<br>&j);<br>Printf("%2d", i);<br>Printf("[%2d], [%-2d], [%02d]",<br>a,a,a);<br>Printf("%8.2f",j); | 300 12.34567 | i는 300, a는 3, j는 12.34567<br>출력 데이터에서 전체 2자리 확보 후 오른쪽부터 출력하는<br>데 출력할 값이 지정 자릿수 보다 큰 경우 자리수를 무시하<br>고 모두 출력. 300<br><table border="1"><tr><td>[</td><td></td><td></td><td>3]</td><td>,</td><td></td><td></td><td>[</td><td>3</td><td></td><td>]</td><td>,</td><td></td><td></td><td>[</td><td>0</td><td>3</td><td>]</td></tr></table><br>출력 데이터에서 8자리 확보 후 소수점 이하 3자리에서 반<br>올림하여 2자리까지만 출력. 남은 6자리에 정수 부분 출력<br><table border="1"><tr><td></td><td></td><td></td><td>1</td><td>2</td><td>.</td><td>3</td><td>5</td></tr></table> | [  |   |   | 3] | , |   |  | [ | 3 |  | ] | , |   |   | [ | 0 | 3 | ] |  |  |  | 1 | 2 | . | 3 | 5 |
| [  |              |   | 3] | , |   |    | [ | 3 |  | ] | , |  |   | [ | 0 | 3 | ] |   |   |   |  |  |  |   |   |   |   |   |
|  |              |   | 1  | 2 | . | 3  | 5 |   |  |   |   |  |   |   |   |   |   |   |   |   |  |  |  |   |   |   |   |   |

<java>

표준 입력

Scanner scan01 = new Scanner(System.in);

InNum = scan01.nextInt();

- Scanner: 객체 변수 생성시 사용하는 클래스 명
- Scan01: 객체 변수 명
- New: 객체 생성 예약어
- System.in: 표준입력장치. 즉, 키보드를 의미
- nextInt(): 입력값을 정수형으로 변환

표준 출력

System.out.printf(서식 문자열, 변수)

System.out.printf("%-8.2f", 200.2);

|   |   |   |   |   |   |  |  |
|---|---|---|---|---|---|--|--|
| 2 | 0 | 0 | . | 2 | 0 |  |  |
|---|---|---|---|---|---|--|--|

<python>

표준 입력

Input(출력문자)

Input(출력문자).split(분리문자)

표준 출력

Print(출력값1, ..., sep=분리문자, end=종료문자)

Print(서식 문자열 %(출력값1,...) )

- capitalize() : 문자열 첫글자만 대문자 나머지는 소문자로 변경 ('ABC deg'.capitalize() = 'Abc def')
- title() : 각 단어의 첫글자만 대문자 나머지는 소문자로 변경 ('ABC deg'.capitalize.title() = 'Abc Def')

## 제어문

프로그램의 순서를 변경할 때 사용하는 명령문

종류 : if, swich, goto, 반복문 등등

- if문 : 조건에 따라서 실행할 문장을 달리하는 제어문
- swich문 : 조건에 따라 분기할 곳이 여러 곳인 경우 간단하게 처리할 수 있는 제어문
- goto문 : 프로그램 실행 중 현재 위치에서 원하는 다른 문장으로 건너뛰어 수행을 계속하기 위해 사용하는 제어문
- 반복문 : 일정한 횟수를 반복하는 명령문

## 반복문

- For문 : 초기값, 최종값, 증가값을 지정하는 수식을 이용해 정해진 횟수를 반복하는 제어문
- While문 : 조건이 참인 동안 실행할 문장을 반복 수행하는 제어문
- Do~while문 : 조건이 참인 동안만 정해진 문장을 반복 수행하고 조건이 거짓이면 반복문을 벗어나지만, 실행할 문장을 무조건 한 번은 실행하는 제어문
- Break: switch문이나 반복문을 제어하는 예약어로 switch문이나 반복문에서 나오면 블록에서 벗어난다.
- Continue: 반복문(only!!)을 제어하는 예약어로 이후의 문장을 실행하지 않고 반복문의 처음으로 옮긴다.

## 배열

동일한 데이터 유형을 여러 개 사용해야 할 경우 이를 손쉽게 처리하기 위해 여러 개의 변수들을 조합해서 하나의 이름으로 정의해 사용하는 것을 의미

## 포인터

변수의 주소. C언어에서는 주소를 제어할 수 있는 기능을 제공

### 포인터변수

변수의 주소를 저장할 때 사용하는 변수

자료형 \*변수; (int \*a;)

포인터 변수에 주소저장 (a = &b;)

배열a가 존재할 때 **a[i]**는 **\*(a+i)**와 같다.

### 사용자 정의 함수

사용자가 필요한 기능을 취향대로 만들어 사용할 수 있는 함수

## 라이브러리

프로그램을 효율적으로 개발할 수 있도록 자주 사용하는 함수나 데이터들을 미리 만들어 모아 놓은 집합체 종류로는 표준라이브러리, 외부 라이브러리가 있다.

### <C>

헤더파일을 사용하려면 '#include <stdio.h>'를 선언 후 사용

| 헤더 파일           | 기능   |
|-----------------|--|
| <b>Stdio.h</b>  | <b>데이터의 입출력에 사용되는 기능을 제공</b><br>Printf, scanf, fprintf, fscanf, fclose, fopen, ... |
| <b>Math.h</b>   | 수학 함수들을 제공<br>Sqrt, pow, abs, ...  |
| <b>String.h</b> | 문자열 처리에 사용되는 기능들을 제공<br>Strlen, strcpy, strcmp, ...                                |
| <b>Stdlib.h</b> | 자료형 변환, 난수 발생, 메모리 할당에 사용되는 기능들을 제공<br>atoi, atof, srand, rand, malloc, free, ...  |
| <b>Time.h</b>   | 시간 처리에 사용되는 기능들을 제공<br>time, clock, ...  |

### <java>

패키지 사용 시 'import java.util'을 선언 후 사용

클래스의 메소드 사용 시 '클래스.메소드()'와 같이 사용 (Math.abs())

| 패키지              | 기능   |
|------------------|--|
| <b>java.lang</b> | 기본적으로 필요한 인터페이스, 자료형, 예외 처리 등에 관련된 기능을 제공<br>String, System, Process, Runtime, Math, Error |
| <b>java.util</b> | 날짜 처리, 난수 발생, 복잡한 문자열 처리 등에 관련된 기능을 제공<br>Date, Calender, Random, Rtring Tokenizer         |
| <b>java.io</b>   | 파일 입출력과 관련된 기능 및 프로토콜을 제공<br>InputStream, OutputStrime, Reader, Writer                     |
| <b>java.net</b>  | 네트워크와 관련된 기능을 제공<br>Socket, URL, InetAddress   |
| <b>java.awt</b>  | 사용자 인터페이스UI와 관련된 기능을 제공<br>Frame, Panel, Dialog, Button, Checkbox                          |

### <python>

클래스 statistics는 통계값 산출을 위한 기능을 제공하며 주요 메소드에는 mean(), median(), variance()등이 있다.

## 절차형 프로그래밍 언어

일련의 처리 절차를 정해진 문법에 따라 순서대로 기술해 나가는 언어

### 장단점

- 컴퓨터의 처리 구조와 유사하여 실행 속도가 빠르다.
- 같은 코드를 복사하지 않고 다른 위치에서 호출하여 사용할 수 있다.
- 모듈 구성이 용이하며 구조적 프로그래밍이 가능하다.
- 프로그램을 분석하기 어렵다.
- 유지 보수나 코드의 수정이 어렵다.

### 종류

- C : 컴파일러 방식의 언어, UNIX의 일부가 C언어로 구성, 시스템 소프트웨어를 개발하기 편리
- ALGOL : 수치계산이나 논리 연산을 위한 과학기술용 언어로, PASCAL과 C언어의 모체
- COBOL : 사무처리용 언어, 영어 문장형식으로 구성, 4개의 DIVISION으로 구성
- FROTRAIN : 과학기술계산용언어로 수학과 공학 분야 공식이나 수식과 같은 프로그래밍 가능

## 객체지향 프로그래밍 언어

현실 세계의 개체Entity를 기계의 부품처럼 하나의 객체로 만들어 객체를 조립하여 프로그램을 작성가능하도록 한 프로그래밍 기법

### 장단점

- 상속을 통한 재사용과 시스템의 확장이 용이
- 코드의 재활용성이 높다.
- 자연적인 모델링에 의해 분석과 설계를 쉽고 효율적으로 할 수 있다.
- 사용자와 개발자 사이의 이해를 쉽게 해준다.
- 대형 프로그램의 작성이 용이하다.
- 소프트웨어 개발 및 유지보수가 용이하다.
- 프로그래밍 구현을 지원해 주는 정형화된 분석 및 설계 방법이 없다.
- 구현 시 처리 시간이 지연된다.

### 종류

- Java : 분산 네트워크환경 적용 가능, 멀티스레드기능 제공
- C++: C언어에 객체지향 개념 적용
- Smalltalk: 1세대 객체지향 프로그래밍 언어, 최초로 GUI 제공한 언어

## 구성요소

|                  |   |
|------------------|---|
| 객체<br>(Object)   | <ul style="list-style-type: none"> <li>• 데이터(속성)와 이를 처리하기 위한 연산(메소드)을 결합시킨 실체</li> <li>• 데이터 구조와 그 위에서 수행되는 연산들을 가지고 있는 소프트웨어 모듈임</li> <li>• 속성(Attribute) : 한 클래스 내에 속한 객체들이 가지고 있는 데이터 값들을 단위별로 정의하는 것으로서 성질, 분류, 식별, 수량 또는 현재 상태 등을 표현함</li> <li>• 메소드(Method) : 객체가 메시지를 받아 실행해야 할 때 구체적인 연산을 정의하는 것으로, 객체의 상태를 참조하거나 변경하는 수단이 됨</li> </ul> |
| 클래스<br>(Class)   | <ul style="list-style-type: none"> <li>• 두 개 이상의 유사한 객체들을 묶어서 하나의 공통된 특성을 표현하는 요소임. 즉 공통된 특성과 행위를 갖는 객체의 집합이라고 할 수 있음</li> <li>• 객체의 유형 또는 타입(Object Type)을 의미함</li> <li>• 데이터를 추상화하는 단위임</li> </ul>  |
| 메시지<br>(Message) | <ul style="list-style-type: none"> <li>• 객체들 간에 상호작용을 하는데 사용되는 수단으로 객체의 메소드(동작, 연산)를 일으키는 외부의 요구 사항</li> <li>• 메시지를 받은 객체는 대응하는 연산을 수행하여 예상된 결과를 반환하게 됨</li> </ul>  |

## 특징

|                               |   |
|-------------------------------|---|
| 캡슐화<br>(Encapsulation)        | <ul style="list-style-type: none"> <li>• 데이터(속성)와 데이터를 처리하는 함수를 하나로 묶는 것</li> <li>• 캡슐화된 객체의 세부 내용이 외부에 은폐(정보 은닉)되어, 변경이 발생할 때 오류의 파급 효과가 적음</li> <li>• 캡슐화된 객체들은 재사용이 용이함</li> </ul>         |
| 정보 은닉<br>(Information Hiding) | 캡슐화에서 가장 중요한 개념으로, 다른 객체에게 자신의 정보를 숨기고 자신의 연산만을 통하여 접근을 허용하는 것  |
| 추상화<br>(Abstraction)          | <ul style="list-style-type: none"> <li>• 불필요한 부분을 생략하고 객체의 속성 중 가장 중요한 것에만 중점을 두어 개략화하는 것, 즉 모델화하는 것</li> <li>• 데이터의 공통된 성질을 추출하여 슈퍼 클래스를 선정하는 개념임</li> </ul>                                 |
| 상속성<br>(Inheritance)          | <ul style="list-style-type: none"> <li>• 이미 정의된 상위 클래스(부모 클래스)의 모든 속성과 연산을 하위 클래스가 물려받는 것</li> <li>• 상속성을 이용하면 하위 클래스는 상위 클래스의 모든 속성과 연산을 자신의 클래스 내에서 다시 정의하지 않고서도 즉시 사용할 수 있음</li> </ul>     |
| 다형성<br>(Polymorphism)         | <ul style="list-style-type: none"> <li>• 메시지에 의해 객체(클래스)가 연산을 수행하게 될 때 하나의 메시지에 대해 각 객체(클래스)가 가지고 있는 고유한 방법(특성)으로 응답할 수 있는 능력을 의미</li> <li>• 객체(클래스)들은 동일한 메소드명을 사용하며 같은 의미의 응답을 함</li> </ul> |

## 스크립트 언어 Script Language

HTML 문서 안에 직접 프로그래밍 언어를 삽입하여 사용하는 것으로 기계어로 컴파일 되지 않고 별도의 번역기가 소스를 분석하여 동작하게 하는 언어

클라이언트의 웹 브라우저에서 해석되어 실행되는 **클라이언트용 스크립트 언어**에는 ASP, JHP, PHP, 파이썬이 있고 서버에서 해석되어 실행 된 후 결과만 클라이언트로 보내는 **서버용 스크립트 언어**에는 자바 스크립트가 있다.

|                          |   |  |  |
|--------------------------|---|--|--|
| 자바 스크립트<br>(Java Script) | <ul style="list-style-type: none"> <li>• 넷스케이프(Netscape)의 브렌던 아이크(Brendan Eich)가 개발한 언어</li> <li>• 웹 페이지의 동작을 제어하며, 변수 선언이 필요 없음</li> </ul> | PHP(Professional Hypertext Preprocessor) | <ul style="list-style-type: none"> <li>• 서버용 스크립트 언어로, Linux, Unix, Windows 운영체제에서 사용이 가능함</li> <li>• C, Java 등과 문법이 유사하여 배우기 쉽고 웹 페이지 제작에 많이 사용됨</li> </ul> |
| ASP(Active Server Page)  | <ul style="list-style-type: none"> <li>• 서버 측에서 동적으로 수행되는 페이지를 만들기 위한 언어</li> <li>• 마이크로 소프트사에서 제작하였으며, Windows 계열에서만 수행 가능함</li> </ul>     | 파이썬<br>(Python)                          | <ul style="list-style-type: none"> <li>• 객체지향 기능을 지원하는 대화형 인터프리터 언어</li> <li>• 플랫폼에 독립적이고 문법이 간단하여 배우기 쉬움</li> </ul>   |
| JSP(Java Server Page)    | <ul style="list-style-type: none"> <li>• Java로 만들어진 서버용 스크립트 언어</li> <li>• 다양한 운영체제에서 사용이 가능함</li> </ul>                                    |  |  |

**명령형 언어**는 문제를 해결하기 위한 방법을 기술한다면 **선언형 언어**는 프로그램이 수행해야 할 문제를 기술하는 언어이다.

명령형 언어는 절차적 언어와 객체지향 언어가 있다.

선언형 언어는 목표를 명시하고 알고리즘은 명시하지 않는다. 종류로 함수형 언어, 논리형 언어가 있다.

선언형 언어의 종류

|         |  |
|---------|--|
| HTML    | <ul style="list-style-type: none"><li>• 인터넷의 표준 문서인 하이퍼텍스트 문서를 만들기 위해 사용하는 언어</li><li>• 특별한 데이터 타입이 없는 단순한 텍스트이므로 호환성이 좋고 사용이 편리함</li></ul>              |
| LISP    | 인공지능 분야에 사용되는 언어로, 기본 자료 구조가 연결 리스트 구조이며 재귀(Recursion) 호출을 많이 사용함  |
| PROLOG  | 논리학을 기초로 한 고급 언어로, 인공 지능 분야에서의 논리적인 추론이나 리스트 처리 등에 주로 사용됨  |
| XML     | <ul style="list-style-type: none"><li>• 기존 HTML의 단점을 보완하여 웹에서 구조화된 폭넓고 다양한 문서들을 상호 교환할 수 있도록 설계된 언어</li><li>• 사용자가 HTML에 새로운 태그(Tag)를 정의할 수 있음</li></ul> |
| Haskell | 함수형 프로그래밍 언어로, 부작용(Side Effect)이 없고 코드가 간결하여 에러 발생 가능성이 낮음   |