



# 03

강

## 프로그래밍 기법과 알고리즘

### 학습목표

- ▶ 다양한 프로그래밍 기법의 종류와 특징을 이해할 수 있다.
- ▶ 알고리즘을 표현하는 방법과 의미를 이해할 수 있다.

## 1 프로그래밍 기법

### (1) 프로그래밍 기법의 개념

- ① 소프트웨어 개발비와 시간을 최소화시키는 방법의 하나로 프로그래밍 작성 방법의 표준화를 들 수 있다.
- ② 표준화된 프로그램은 코드화, 디버그, 관리 및 수정하기가 쉽다.

## 2 구조화 프로그래밍

### (1) 구조화 프로그래밍의 개념

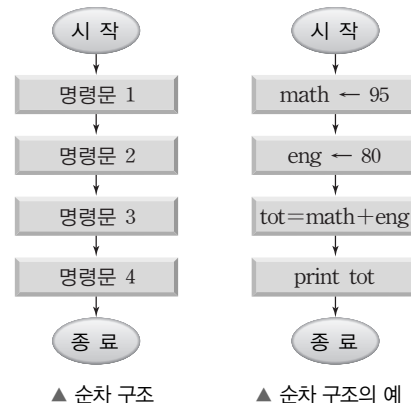
- ① 1960년대 말 E.W.다이크스트라가 주창한 프로그램 작성 기법이다. 프로그램이 프로그래머의 개성에 따라 복잡 다단하게 만들어지기 때문에 개발하는 데도 문제가 많지만, 후에 다른 사람이 유지·보수를 할 때는 더욱 어려웠던 문제를 해결하기 위해 개발된 방식이다.
- ② 프로그램 개발 시 이해와 수정이 쉽고, 정확성을 검증하기 용이하여 최종적으로 제어 구조가 명확한 프로그램을 만들 수 있다. 코드를 몇 개의 단위로 구분(책의 내용을 단원과 페이지별로 구분하듯)하여 이해하기 쉽고, 수정하기 쉽도록 블록 형태로 프로그램을 작성하는 기법이다.

### (2) 구조화 프로그래밍의 구조

- ① 컴퓨터 프로그램의 구조를 여러 갈래로 분기하여 복잡하게 하지 않고, 프로그램을 구성하는 모든 요소 상호 간의 관계를 가능한 한 순차, 선택, 반복의 세 가지 기본 구조를 사용하여 작성한다.
- ② 프로그램을 실행할 때에 제어가 위에서 아래로 가는 성질을 가지게 되어, 역방향으로 진행됨이 없이 기술된 순서에 따라 순차적으로 실행하도록 한다.(가급적 GO TO 문을 사용하지 않는다)
- ③ 주어진 조건에 따라 둘 중 어느 하나를 선택하여 실행하도록 한다. 특정 블록을 반복 실행하도록 하여 프로그램의 처리 절차를 간략하게 한다.

### (3) 구조화 프로그래밍의 기본 구조

- ① 순차 구조 : 일을 단계적으로 차례로 처리하는 구조로서 하나의 처리를 마친 후에 바로 다음의 처리 과정이 시작된다. 즉 일이 순차적으로 처리된다.



EBS tip

- 표준화된 프로그래밍 기법에는 구조화 프로그래밍 기법과 객체 지향 프로그래밍 기법이 있다.

### ● 구조화 프로그래밍의 특징

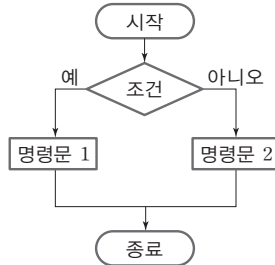
- 구조가 단순하다.
- 프로그램을 이해하기 쉽다
- 오류가 나타나거나 기능을 변경할 경우 쉽게 처리할 수 있다.
- 선택 또는 반복문을 발췌하여 재사용하기도 쉬워 프로그램의 생산성도 높일 수 있다.

### ● 구조화 프로그래밍 기법의 좁은 관점

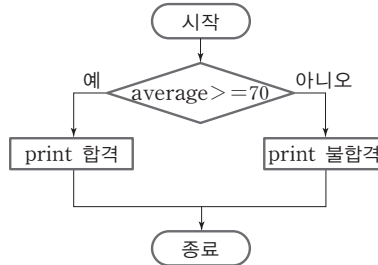
- goto 문이 없는 프로그램
- 순차, 선택, 반복 구조만 사용

수학 성적 95점, 영어성적 80점을 입력하여 총점을 구하여 출력하는 프로그램의 구조를 나타낸 것이다.

- ② 선택 구조 : 조건에 따라 서로 다른 처리 과정을 진행하도록 한 것으로, 어떤 조건을 비교한 후 조건을 만족하거나 만족하지 않는 경우에 서로 다른 처리를 실행하는 구조이다. 프로그램의 제어 순서가 바뀌는 구조이다.



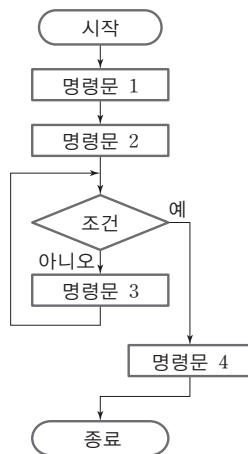
▲ 선택 구조



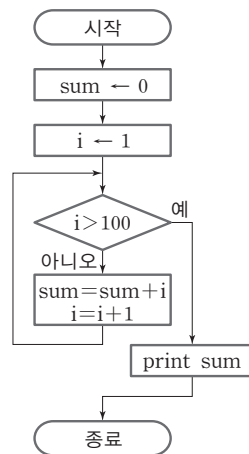
▲ 선택 구조의 예

산정된 평균값을 비교하여 평균이 70점 이상이면 “합격”을 출력하고 평균이 70점 미만이면 “불합격”을 출력하는 프로그램의 구조를 나타낸 것이다.

- ③ 반복 구조 : 어떤 조건을 만족하는 동안 또는 만족할 때까지 같은 처리를 반복하여 실행하는 구조이다.



▲ 반복 구조



▲ 반복 구조의 예

1부터 100까지의 합을 구하여 그 결과를 출력하는 프로그램의 구조를 나타낸 것이다.

## (2) 구조화 프로그래밍의 필요성 및 효과

- ① 프로그램의 흐름을 쉽게 이해할 수 있다.
- ② 프로그램의 구조가 간결하다.
- ③ 프로그램의 신뢰성을 높인다.
- ④ 프로그램의 테스트를 용이하게 한다.
- ⑤ 프로그램의 오류 수정 및 삽입, 삭제가 용이하다.
- ⑥ 프로그래밍에 소요되는 경비를 감소시킨다.



### ● 모듈화 프로그래밍 기법

전체 프로그램을 기능별로 작은 단위인 모듈(module)로 분리하여 부분적으로 코딩한 후 전체 모듈을 결합하여 프로그램을 구성

### ● 하향식 프로그래밍 기법

프로그램을 상위의 모듈에서 하위의 모듈 순으로 작성해 나가는 기법



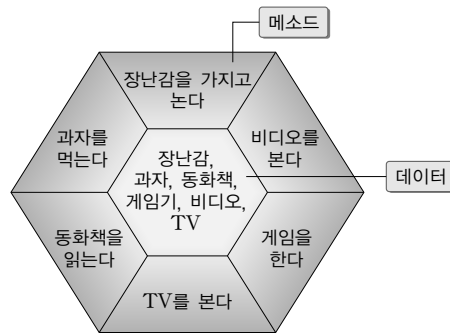
### 3 객체 지향 프로그래밍

#### (1) 객체 지향 프로그래밍의 개념

- ① 객체 지향 프로그래밍 기법은 실제 세계에서 나타나는 다양한 현상을 객체(object)라는 단위로 실현함으로써 다양한 문제를 해결하기 위한 프로그래밍 기법으로서 절차보다는 실체(데이터=자료)를 중요시 한다.
- ② 하향식 방식보다는 상향식 방식으로 프로그래밍한다. 실체에 대한 요소들을 먼저 구성한 후 해당 요소를 결합하여 상위 요소를 만들어 내고, 상위 요소들을 결합한 후 더 상위 요소를 만들어 내는 방식을 사용한다.
- ③ 객체는 우리가 다루는 모든 사물을 일컫는 말로 상태(state)와 행위(behavior)의 요소를 가진다.
- ④ 프로그램에서 객체는 고유의 데이터와 데이터를 처리하는 메소드(method)로 구성한다.
- ⑤ 절차 지향 프로그래밍에서는 데이터의 구조와 그 데이터를 처리하는 일련의 과정을 통하여 결과를 얻는 데 비해, 객체 지향 프로그래밍은 객체들이 메시지를 통하여 통신함으로써 원하는 결과를 얻는다.
- ⑥ 클래스는 객체를 생성하는 틀이다.

#### (2) 객체 지향 프로그래밍의 용어

- ① 객체 : 실체(데이터)와 실체에 관련된 동작(절차, 방법, 기능 등)을 모두 포함하는 것을 의미한다.
- ② 클래스 : 동일한 속성, 관계 등을 가진 객체들의 집합을 클래스라고 한다.
- ③ 인스턴스 : 클래스에 속하는 각 객체를 의미한다.
- ④ 속성 : 객체가 갖는 성질이나 특징을 의미한다.



▲ 객체의 예

#### (3) 객체 지향 프로그래밍의 특성

- ① 상속성(inheritance) : 상위 클래스와 하위 클래스 사이에 메소드와 데이터를 자동으로 공유하는 특성이 있다. 상속에는 하나의 클래스로부터 속성과 메소드를 상속받는 단일 상속과 하나 이상의 클래스로부터 속성과 메소드를 상속받는 복수 상속이 있다.
- ② 캡슐화(encapsulation) : 객체 내에 메소드와 속성을 넣어 두고, 오직 해당 객체의 고유의 메소드를 통해서만 데이터에 접근할 수 있다. 이는 작업 수행에 필요한 것은 객체 안에 숨기고 사용자에게는 인터페이스만 노출시킨다.

● 객체 지향 프로그래밍 언어  
C++, 자바(JAVA)

#### ● 객체 지향 프로그래밍의 용어

- 객체 : 실체(데이터)와 실체에 관련된 동작(절차, 방법, 기능 등)을 모두 포함하는 것을 의미한다.
- 클래스 : 동일한 속성, 관계 등을 가진 객체들의 집합을 클래스라고 한다.
- 인스턴스 : 클래스에 속하는 각 객체를 의미한다.
- 속성 : 개체가 갖는 성질이나 특징을 의미한다.

#### ● 객체의 구성

- 속성의 값을 나타내는 변수(데이터)
- 데이터를 변경하거나 조작할 수 있는 메소드로 구성

#### ● 속성(attribute)

클래스나 객체의 상태를 나타내는 것

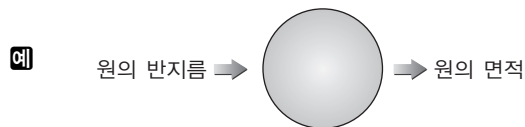
#### ● 메소드(method)

클래스나 객체의 값을 변경하는 기능을 하는 것

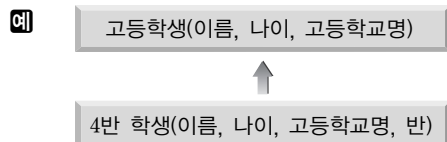
- ③ 추상화(abstraction) : 클래스나 객체로부터 공통적인 질의나 특성을 추출하여 상위 클래스를 만들 수 있다.
- ④ 다형성(polymorphism) : 동일한 메시지에 대하여 수신 측에 따라 다른 결과를 얻을 수 있는 특성이다. 이 특성 때문에 상속성을 보다 강화할 수 있다.
- ⑤ 지속성(persistence) : 객체의 인스턴스는 대부분 프로그램이 실행될 때 만들어지고 실행이 끝나면 없어진다.
- ⑥ 메시지(message) : 객체와 객체 사이에서 주고받는 지시를 말하는 것으로, 객체에서 연산을 수행할 수 있도록 하는 유일한 인터페이스 규약이다. 이 메시지는 다른 언어에서 함수와 비슷한 것으로 메소드로 구현된다.

#### 4 객체 지향 프로그래밍 방법

- ① 객체를 프로그램 구성 단위로 한다.
- ② 사건(Event)의 발생에 따라 그에 대한 기능을 작성한다.
- ③ 사건은 외부에서 어떤 일이 발생했을 때 일어난다.
  - 마우스로 특정 객체를 클릭하거나 키보드의 특정 키를 누르는 것
- ④ 메시지를 사용하여 발생한 사건을 특정 객체에 전달한다.
- ⑤ 캡슐화, 상속성, 다형성의 특징을 지원한다.
  - 캡슐화 : 일정 기능을 캡슐에 담아 캡슐 단위로 사용하는 것



- 상속성 : 미리 만들어진 객체의 일부를 재정의하여 사용하는 것



- 다형성 : 같은 기능이라도 객체에 따라 다르게 동작하는 것

예 '4' 기호 : 산술 연산에서는 두 수를 더하는 작업  
문자 연산에서는 주어진 문자를 결합하는 작업

#### 5 알고리즘(Algorithm)

- ① 알고리즘은 어떤 문제를 해결하기 위한 절차를 말한다.
- ② 알고리즘을 표현하는 방법으로는 다음과 같은 것들이 있다.
  - 일반적인 언어를 이용한 표현
  - 의사 코드(pseudo code)를 이용한 표현
  - 순서도
  - 나시-슈나이더만(NS) 차트
- ③ 알고리즘의 조건
  - 명확성 : 각 명령들은 명확한 의미가 있어야 한다.
  - 유효성 : 일정한 시간 내에 실행 가능해야 한다.
  - 입력 : 입력 자료는 유한하여야 한다.
  - 출력 : 적어도 하나 이상의 결과가 나와야 한다.



##### ● 객체 지향 프로그래밍 특징

- 객체 지향 모델은 실세계를 잘 반영한다.
- 개발 생산성 증대 및 유지 보수를 용이하게 한다.
- 복잡하고 다양한 정보의 표현이 가능하다.

- 객체 지향 프로그래밍의 시작은 클래스의 생성이다.
- 클래스로부터 생성된 객체를 인스턴스(instance)라 한다.
- 객체를 생성할 때 숨겨야 하는 정보와 공개해야 하는 정보를 구분함
- 상속성은 상위 클래스의 속성과 메소드를 물려받는다.
- 객체 지향 프로그래밍 언어로는 C++, 자바 등이 대표적이다.

##### ● 알고리즘(algorithm)

문제 해결을 위한 일련의 논리적인 절차

##### ● 의사 코드(pseudo code)

알고리즘의 명령을 나타내기 위해 도표가 아닌 자유로운 문장 형태로 논리 체계를 작성한 코드

• 유한성 : 여러 단계를 거친 후에는 끝나야 한다.

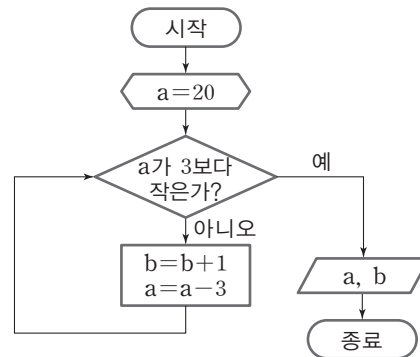
④ 좋은 알고리즘은 절차가 간단하고 빠르며, 메모리를 많이 사용하지 않는다.

## 6 알고리즘의 실제

예 일반적인 언어를 사용하여 나타낸 알고리즘

ㄱ. 게임 시작  
 ㄴ. 비행기의 연료량을 저장하는 변수 fuel을 500으로 초기화  
 ㄷ.  $\text{fuel} \geq 10$ 인지 비교  
 ㄹ. 참이면  $\text{fuel} = \text{fuel} - 5$  식을 계산한 후 'ㄷ' 으로 이동  
 ㄴ. 거짓이면 '비상 착륙 유도' 메시지 출력  
 ㄷ. 게임 종료

예 순서도를 사용하여 나타낸 알고리즘



▲ 몫과 나머지를 구하는 알고리즘



**EBS tip**

### ● 나시 슈나이더만 도표

프로그램의 절차 중에서 복합적 조건 처리를 명확하게 구분하여 구조적 프로그래밍 방법에 사용하는 논리 표현 기법의 도표

조건이 참	조건	조건이 거짓
참일 때 수행할 내용 기재		거짓일 때 수행할 내용 기재



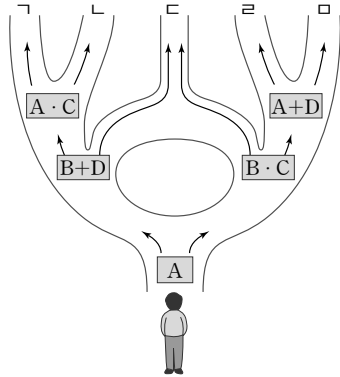
# 기출 모의고사

정답 및 해설 p. 3

1

2009학년도 대수능

다음의 [게임 요령]에 따라 [퀴즈]를 풀고 논리식의 결과에 따라 진행할 때 철수가 나갈 출구로 옳은 것은?



[게임 요령]

- 각 [퀴즈]를 풀고 답이 맞으면 '참', 틀리면 '거짓'을 변수(A~D)에 저장한다.
- 논리식의 '+'는 OR 연산자이고 '·'는 AND 연산자이다.
- 갈림길을 만나면 상자(□) 안의 각 논리식을 계산한 뒤, 결과가 참이면 왼쪽(←), 거짓이면 오른쪽(→)길로 간다.

퀴즈	변수	논리값
알고리즘은 문제 해결 과정을 기술한 것이다.	A	참/거짓
순서도와 의사 코드는 알고리즘을 표현하는 방법의 일종이다.	B	참/거짓
어떤 문제이든 결과를 구하기 위한 알고리즘은 한 가지이다.	C	참/거짓
프로그램은 문제 해결을 위한 명령문들을 순차적을 나열한 것이다.	D	참/거짓

- ① ㄱ    ② ㄴ    ③ ㄷ    ④ ㄹ    ⑤ ㅁ

2

입력된 점수를 이용하여 총점과 평균을 구하는 알고리즘을 구현하려고 한다. <보기>의 순서를 올바르게 나열한 것은?

**보기**

- 평균(총점/4)을 구한다.
- 이름, 점수(국어, 영어, 수학, 프로그래밍)를 읽는다
- 총점(국어+영어+수학+프로그래밍)을 구한다.
- 이름이 공백(null)이면 프로그램을 끝낸다.

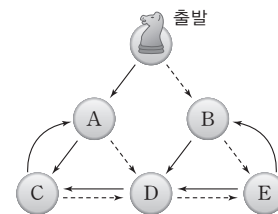
- ① ㄱ-ㄴ-ㄷ-ㄹ    ② ㄴ-ㄷ-ㄱ-ㄹ  
 ③ ㄴ-ㄹ-ㄷ-ㄱ    ④ ㄷ-ㄴ-ㄱ-ㄹ  
 ⑤ ㄹ-ㄴ-ㄷ-ㄱ

3

2011학년도 대수능

다음은 [규칙]과 [명제]에 따라 말을 움직이는 놀이이다. 놀이를 종료했을 때 말의 최종 위치로 옳은 것은?

- [명제]는 1번부터 차례로 모두 본다.
- 말은 [명제]가 참이면 실선(→), 거짓이면 점선(⋯→)을 따라 한 칸만 이동한다.

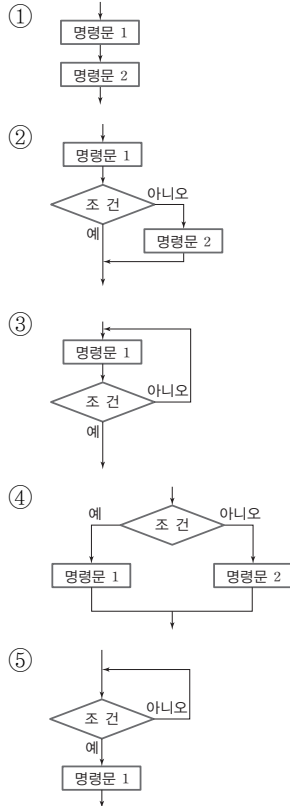


번호	명제
1	알고리즘은 문제 해결을 위한 방법과 절차이다.
2	알고리즘은 순서도나 의사 코드로 표현할 수 있다.
3	순서도는 약속된 기호를 사용한다.

- ① A    ② B    ③ C  
 ④ D    ⑤ E

#### 4 다음 작업을 수행하는 순서도로 올바른 것은?

x를 2로 나누어서 나머지가 0이면 “짝수”를 0이 아니면 “홀수”를 출력한다.



#### 5 객체 지향 프로그래밍의 특징을 설명한 내용이다. 옳은 것을 <보기>에서 있는 대로 고른 것은?

##### 보기

- ㄱ. 객체보다는 작업의 진행 절차를 중요시한다.
- ㄴ. 캡슐화, 상속성, 다형성 등의 특징을 가지고 있다.
- ㄷ. 각각의 사건(Event)의 발생에 반응하는 모듈을 작성한다.
- ㄹ. 이 기법을 사용하는 프로그래밍 언어로는 C++, 자바 등이 있다.

- ① ㄱ, ㄴ      ② ㄷ, ㄹ      ③ ㄱ, ㄴ, ㄷ  
④ ㄱ, ㄷ, ㄹ      ⑤ ㄴ, ㄷ, ㄹ

#### 6 다음은 프로그래밍 관련 용어에 대한 설명이다. (가)와 (나)에 들어갈 내용으로 옳은 것은?

(가)

- 컴퓨터로 어떤 문제를 해결하기 위하여 필요한 자료들과 해결 절차를 알기 쉽게 기술한 것을 말한다.
- 표현 방법으로는 순서도와 의사 코드 등이 있다.

(나)

- 객체 내에 메소드와 속성을 넣어 두고, 오직 해당 개체 고유의 메소드를 통해서만 데이터에 접근할 수 있다.
- 작업 수행에 필요한 것은 객체 안에 숨기고 사용자에게는 인터페이스만 노출시킨다

(가)

(나)

- ① 알고리즘      다형성  
② 알고리즘      캡슐화  
③ 컴파일      캡슐화  
④ 컴파일      메소드  
⑤ 디버깅      캡슐화

#### 7 다음에 설명하는 용어의 조건으로 올바른 것을 <보기>에서 있는 대로 고른 것은?

##### <용어 설명>

- 어떤 특정한 작업을 처리하기 위하여 어떠한 순서와 방식에 의하는가를 나타내는 과정
- 문제 해결을 위한 일련의 논리적인 절차

##### 보기

- ㄱ. 입력 자료는 무한하여야 한다.
- ㄴ. 각 명령은 명확한 의미가 있어야 한다.
- ㄷ. 컴퓨터 기종에 따라 달라야 한다.
- ㄹ. 기계어와 쉽게 호환이 되어야 한다.
- ㅁ. 여러 단계를 거친 후에는 끝나야 한다.

- ① ㄱ, ㄷ      ② ㄴ, ㅁ      ③ ㄱ, ㄴ, ㄷ  
④ ㄱ, ㄷ, ㅁ      ⑤ ㄴ, ㄷ, ㅁ