



04

강

순서도 작성

학습목표

▶ 순서도의 개념을 이해할 수 있으며 쓰임에 대해서도 설명할 수 있다.



EBS tip

● 순서도 작성 원칙

순서도는 프로그래머 자신은 물론, 그 프로그램의 개발과 운영 및 유지·보수에 관계되는 모든 사람과 연관성을 가지게 된다. 그러므로 누구나 이해할 수 있는 약속된 기호를 사용하여야 한다.

1 순서도

- ① 주어진 문제를 해결하는 방법과 절차, 그리고 내용에 대하여 기호를 사용하여 순서대로 나타낸 그림이다.
- ② 순서도는 문제를 분석하여 처리하는 순서를 논리적인 흐름에 따라 단계화시키고, 약속된 기호로 일관성 있게 나타낸다.

2 순서도 작성 방법

- ① 국제 표준화 기구(ISO)에서 정한 표준 기호를 사용하여 작성한다.
- ② 시작, 끝 처리 과정의 각 기호를 흐름선으로 연결한다.
- ③ 처리 순서는 위에서 아래로, 왼쪽에서 오른쪽으로 서로 교차하지 않도록 하고, 화살표로 표시한다.
- ④ 간단명료하게 작성하고, 필요하면 주석을 함께 기술한다.
- ⑤ 논리적인 흐름이 복잡하고 어려울 때에는 여러 단계로 구분하여 작성한다.
- ⑥ 순서도 기호 내부에 처리 순서를 기술하고, 필요한 경우 기호 외부에 주석을 단다.
- ⑦ 순서도가 길거나 복잡한 경우 기능별로 분할한 후 연결 기호를 사용하여 연결한다.
- ⑧ 순서도를 작성할 때 논리적인 오류가 있으면 코딩된 프로그램에도 오류가 발생할 수 있으므로 논리에 맞도록 정확하게 작성한다.

3 순서도의 역할

- ① 프로그램 작성의 직접적인 자료가 된다.
- ② 업무의 흐름을 그림으로 직접 볼 수 있기 때문에 프로그램의 논리적인 절차 및 처리 내용을 쉽게 파악할 수 있고, 다른 사람에게 전달이 쉽다.
- ③ 일정 기간이 지난 후에도 프로그램을 이해하는데 도움을 주는 문서화의 역할을 한다.
- ④ 프로그램의 정확성 여부를 판단하는 자료가 되며, 오류가 발생하였을 때 그 원인을 찾아 수정하기가 쉽다.
- ⑤ 프로그램의 논리적인 체계 및 처리 내용을 쉽게 파악할 수 있다.
- ⑥ 프로그래밍 언어에 관계없이 공통으로 사용할 수 있다.
- ⑦ 프로그램의 인수, 인계를 용이하게 한다.
- ⑧ 프로그램의 갱신 및 유지 관리가 용이하다.

4 순서도 기호

명칭	기호	의미
처리		산술 연산, 데이터의 이동, 편집 등의 처리
입출력		데이터의 입출력 기능을 표시



명칭	기호	의미
준비		처리전 준비 및 선언
비교 · 판단		주어진 조건을 비교, 판단하여 흐름을 결정
단말기		처리의 시작과 종료를 표시
자기 테이프		자기 테이프의 입출력
온라인 기억 장치		대용량 기억 장치를 이용한 온라인 처리
연결자		다음에 처리할 순서가 있는 곳으로 연결
흐름선		제어의 흐름과 실행 순서를 표시
서류 문서		서류를 매체로 하는 입출력 기능을 표시
수동 입력		자판을 통한 수작업 입력
통신 연결		통신 회선에 의한 연결

5 순서도의 실제

순서도는 크게 직선형, 분기형, 반복형으로 나눌 수 있다.

(1) 직선형 순서도

- ① 순서가 위에서 아래로 일직선의 형태로 이루어진 구조이다.
- ② 반복이나 분기가 되지 않고 한번 처리하면 끝난다.
- ③ 간편하게 작성할 수 있으며, 오류 수정이 쉽다.

예 두 수를 입력하여 합과 평균을 구하여 출력

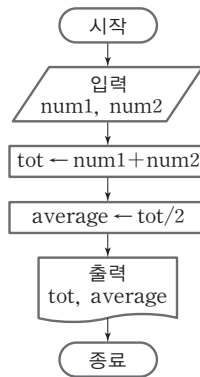
알고리즘(자연 언어로 작성)	알고리즘(의사 코드로 작성)
1. 시작 2. 두 수 num1, num2 입력. 3. 합을 계산 ($\text{tot} = \text{num1} + \text{num2}$) 4. 평균을 계산 ($\text{average} = \text{tot} / 2$) 5. 합(tot)과 평균(average)을 출력 6. 프로그램을 종료	Begin input num1, num2 $\text{tot} = \text{num1} + \text{num2}$ $\text{average} = \text{tot} / 2$ print tot, average End

- 순서도는 크게 직선형, 분기형, 반복형으로 나눌 수 있다.

● 직선형 순서도

조건에 의하여 분기되거나 반복 처리되지 않는다.

• 순서도



(2) 분기형 순서도

처리 중 주어진 조건을 비교, 판단하여 그 결과에 따라 서로 다른 처리를 한다.

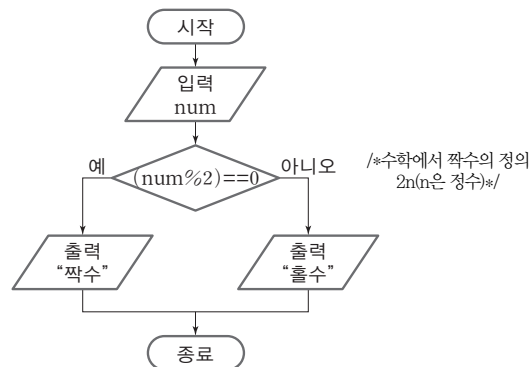
예 한 수를 입력받아 짝수이면 "짝수"라고 출력하고 홀수이면 "홀수"라고 콘솔(모니터)에 출력

알고리즘(자연 언어로 작성)	알고리즘(의사 코드로 작성)
1. 시작 2. 한 수 num 입력. 3. 짝수인지 홀수인지 판단 3-1. 짝수이면 "짝수" 출력 3-2. 홀수이면 "홀수" 출력 4. 프로그램을 종료	Begin input num if (num % 2) == 0 then print "짝수" else print "홀수" End

● 분기형 순서도

조건이 참일 때와 거짓일 때의 처리가 다르다.

• 순서도



위 순서도는 모니터에 출력하는 경우이고 종이문서로 출력하는 경우에는

를 사용한다.

(3) 반복형 순서도

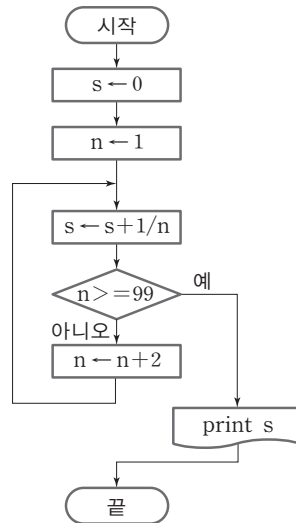
- ① 주어진 조건이 만족할 때까지 특정한 과정을 반복해서 처리한다.
- ② 기억 장소에 저장된 데이터를 교체하면서 같은 명령을 반복적으로 수행하여 처리한다.



예 $s = 1 + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \dots + \frac{1}{99}$ 를 구하여 출력

알고리즘(자연 언어로 작성)	알고리즘(의사 코드로 작성)
<ol style="list-style-type: none"> 1. 시작 2. 1,3,5,7,...,99으로 증가하는 값을 저장할 기억 장소 n과 합을 저장할 기억 장소 이름을 s로 하고 n을 1로 s를 0으로 초기화한다. 3. 분자는 항상 1, 분모는 1에서 99까지 2를 증가하면서 s에 더함 4. n이 99가 되면 s를 출력 5. 프로그램을 종료 	<pre> Begin s=0 n=1 Loop Begin: s=s+ 1/ n if n >=99 then Exit Loop n=n+2 Loop End print s End </pre>

• 순서도



(3) 혼합형 순서도

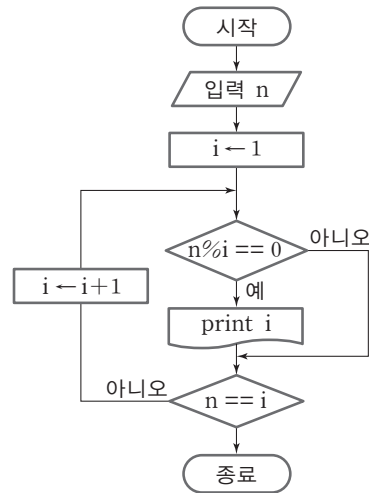
대부분 프로그램은 직선형과 분기형, 반복형이 혼합되어 있다

예 자연수를 입력하여 입력한 수의 약수를 출력

알고리즘(자연 언어로 작성)	알고리즘(의사 코드로 작성)
<ol style="list-style-type: none"> 1. 시작 2. 자연수 입력(n) 3. 나누는 값(i)을 1로 초기화 4. 4-1 입력수(n)를 i로 나눔 4-2 나머지가 0이면 i를 출력 4-3 n과 i가 같으면 루프 나감. 4-4 i를 1 증가 4-5 4-1을 실행 5. 종료 	<pre> Begin input n i=1 Loop Begin: if (n % i) == 0 then print i if n == i then Exit Loop i = i + 1 Loop End End </pre>

■ 대부분의 순서도는 직선형과 분기형, 반복형이 혼합되어 있다.

• 순서도



6 자료의 단위와 표현

- ① 컴퓨터는 2진법에 의한 2진수의 체계로 모든 자료를 표현한다.
(즉 2진수의 0과 1을 사용하여 숫자, 문자, 음성, 영상 정보 등을 표현한다.)
- ② 자료의 표현
 - 수치 자료는 고정 소수점 데이터 형식으로 정수를 표현하고, 부동 소수점 데이터 형식으로 실수를 표현한다.
 - 그래픽 자료는 점의 집합으로 이루어지며, 그 점을 화소(픽셀 : Pixel)라 한다.
 - 소리 자료는 음성, 음향, 음악 등을 샘플링하여 디지털 신호로 변환·조합하여 하나의 소리 데이터로 만든다.
 - 동영상 자료는 정지 영상을 1초당 여러 프레임(보통 30프레임)을 변화시켜 만든다.

7 문자 자료의 표현

(1) 표준 BCD 코드(Standard Binary Coded Decimal Interchange Code)

표준 2진화 10진 코드, 또는 6비트 BCD 코드라고도 하며, 숫자·영문자·특수 문자를 코드화하기 위한 것으로 BCD 코드에다 2비트를 추가하여 나타낸 것이다. 2^6 이므로 64가지의 서로 다른 숫자, 영문자, 특수 문자를 나타낼 수 있다.

(2) EBCDIC 코드(Extended Binary Coded Decimal Interchange Code)

확장 2진화 10진 코드는 8개의 비트를 사용하여 하나의 문자를 나타낸다. 따라서 숫자, 영문자의 대문자와 소문자, 특수 문자, 한글의 자음과 모음 등을 모두 나타낼 수 있도록 BCD에다 4개의 비트를 추가한 것으로 확장된 BCD 코드라고 한다. $256(2^8)$ 가지의 문자를 표현할 수 있어 영문자의 소문자와 여러 가지 특수 문자까지 사용이 가능한 코드이다.

(3) ASCII 코드(American Standard Code for Information Interchange)

ASCII 코드는 현재 많은 컴퓨터에서 사용되고 있으며, 7비트로 구성되어 $128(2^7)$ 가지의 문자를 표현할 수 있다. 개인용 컴퓨터에서는 패리티 비트 1비트를 추가시킨 확장 ASCII 코드(8비트)가 문자 코드로 사용되고 있다.

● 유니코드(Unicode)

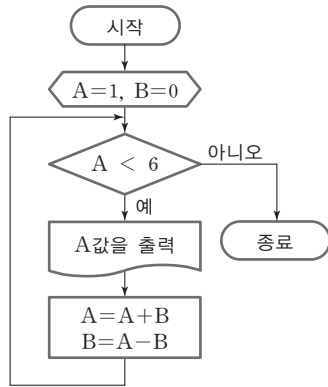
유니코드는 1바이트로 표현하는 ASCII 코드와는 달리 2byte로 전 세계의 모든 문자를 표현하고자 개발된 국제 표준 문자 코드이다.



1

2011학년도 대수능

다음 순서도의 실행 결과는?

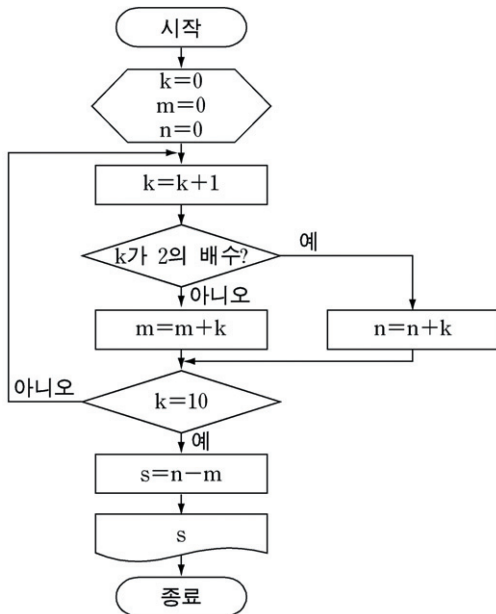


- ① 11111
- ② 11223
- ③ 11235
- ④ 12321
- ⑤ 12345

2

2008년 9월 시행 평가원 모의평가

다음 순서도에서 s의 출력값으로 옳은 것은?



- ① -6 ② -5 ③ 5
- ④ 6 ⑤ 7

3

2009학년도 대수능

다음은 로그인 과정을 순서도로 표현한 것이다. 그림 (A)와 같은 입력에 대하여 그림 (B)가 출력되었다. 순서도의 (가)~(마) 중 그림 (B)에 해당하는 것은?

(A)

아이디와 비밀번호는 대소문자 구분이 없습니다.

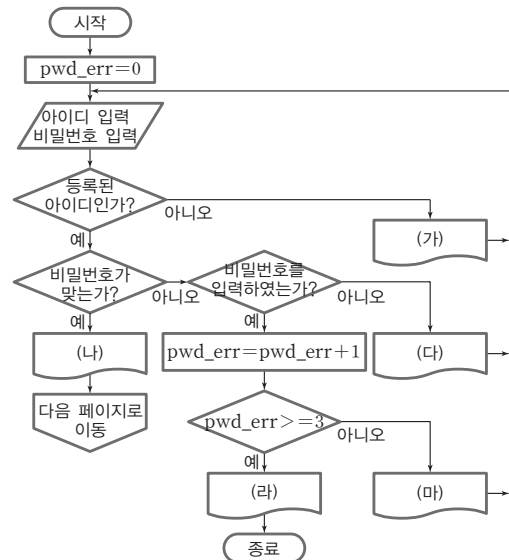
아이디 로그인

비밀번호

(B)

오류

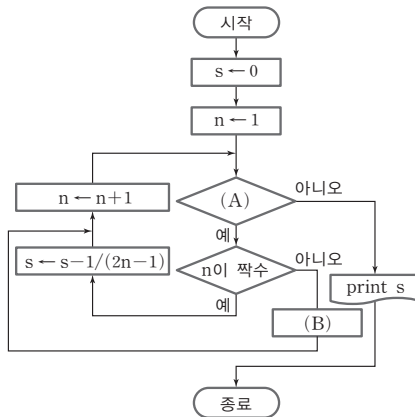
비밀번호가 입력되지 않았습니다.
비밀번호를 입력하세요!



- ① (가) ② (나) ③ (다)
- ④ (라) ⑤ (마)

- 4 다음 수식을 계산하는 프로그램을 작성하기 위해 순서도를 작성하였다. (A)와 (B)에 들어갈 내용으로 옳은 것은?

$$s = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9} - \frac{1}{11} \cdots + \frac{1}{1001}$$

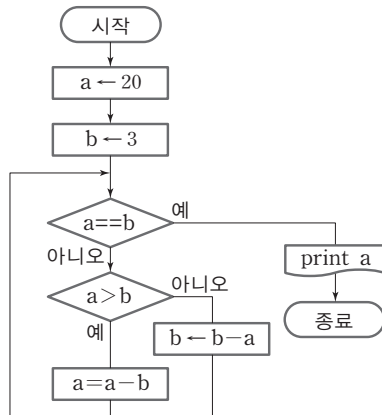


(A)

(B)

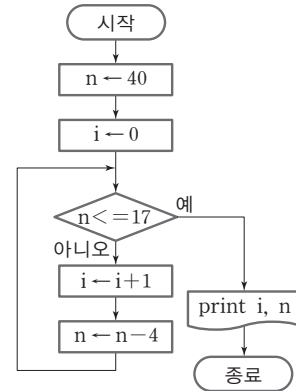
- ① $n \leq 501$ $s = s + 1 / (2n+1)$
 ② $n \leq 501$ $s = s - 1 / (2n-1)$
 ③ $n \leq 501$ $s = s + 1 / (2n-1)$
 ④ $n \leq 1001$ $s = s - 1 / (2n-1)$
 ⑤ $n \leq 1001$ $s = s - 1 / (2n+1)$

- 5 다음 순서도의 출력값으로 옳은 것은?



- ① 1 ② 2 ③ 3
 ④ 4 ⑤ 5

- 6 다음 순서도의 실행 결과는?



- ① 3 13 ② 4 14
 ③ 5 15 ④ 6 16
 ⑤ 7 17