



1. 프로그램과 순서도

주요 내용

- 1.1 개요
- 1.2 프로그램 작성 단계
- 1.3 순서도의 개념
- 1.4 순서도의 기호
- 1.5 순서도의 종류
- 1.6 순서도 작성 방법

1.1 개 요

□ 순서도(flowchart)

- ◎ 순서도란 어떤 업무를 처리하는 과정을 순서적으로 나타내는 것으로서 필요한 데이터나 제품 등 이동 과정도 함께 나타낼 수 있는 도표를 의미.
- ◎ 반드시 컴퓨터의 이용을 전제로 하는 것은 아님.
- ◎ 컴퓨터의 이용을 전제로 한 경우에 제기된 문제를 분석하고, 문제 해결을 위한 순서와 방법을 나타내는 순서도 작성 방법에 대하여 고찰

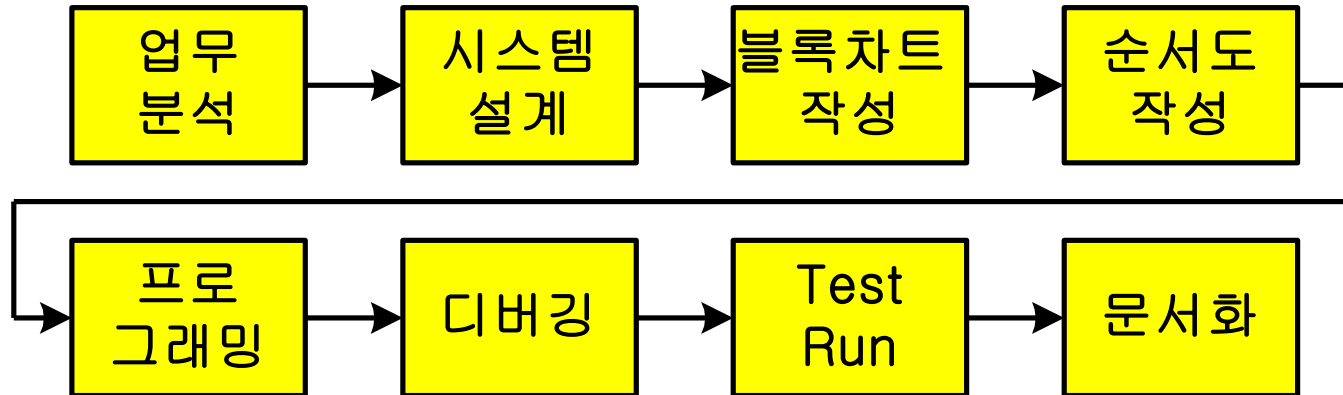
□ 컴퓨터 프로그램(computer program)

- ◎ 컴퓨터가 이해할 수 있는 정해진 규칙에 따라 표현하여야 하며, 이러한 규칙을 프로그램 언어(program language)라 한다.

□ 프로그래밍(programming)

- ◎ 컴퓨터로 어떤 일을 처리하기 위하여 프로그램 언어로 프로그램을 작성하는 것
- ◎ 프로그램을 작성하는 사람을 프로그래머(programmer)라 한다.

1.2 프로그램 작성 단계



□ 업무 분석

- ◎ 처리하려는 업무를 분석하여 컴퓨터로 처리하는 것이 타당한지를 검토하며, 문제와 처리의 범위를 결정한다. 업무 분석 시에는 그 업무를 부분적인 면 뿐 아니라, 전체적인 요소를 잘 분석해서 부분과 전체와의 관계를 항상 고려해야 한다. 또한 입력과 출력 매체를 결정하고 입출력 데이터의 형식을 결정한다. 그리고 어떠한 방법으로 처리하는 것이 경제적이고 능률적인가를 고려하여 적용하는 기법을 결정하며 처리 순서를 정한다.

1.2 프로그램 작성 단계(Cont'd)

□ 시스템 설계

- ◎ 업무의 분석이 끝나면 여러 단계를 통하여 수 작업으로 처리하고 있는 작업을 컴퓨터로 처리할 방안을 설계해야 한다.

□ 블록 차트 작성

- ◎ 프로그램은 여러 개의 명령들이 모인 것이므로 문제 처리를 위하여 논리를 세우고 이를 보다 쉽게 파악하고 검토할 수 있도록 표준 기호를 사용하여 그림으로 나타낸다. 블록 차트(block chart)는 업무의 처리 순서를 크게 나누어 처리의 흐름을 개략적인 도표로 표시한 것이며 순서도를 작성하기 위한 기초 자료로도 이용되고 시스템의 설계자와 현 업무 담당자와의 대화용으로도 사용된다.

1.2 프로그램 작성 단계(Cont'd)

□ 순서도 작성

- ◎순서도는 업무의 처리 내용을 컴퓨터의 사고 체계에 의하여 세밀한 요소까지 분석하고 각 요소간의 관계를 도표로서 표시한 것이다.
- ◎프로그램 논리는 대부분 순서도에서 검토하고 개선하며 순서도에 따라 프로그램 언어를 사용하여 원시 프로그램을 작성할 수 있다.
- ◎그러므로 프로그램 논리는 순서도에서 충분히 검토하여 완전하고 효율이 좋은 흐름이 되었을 때 프로그램을 입력한다.

□ 프로그래밍(programming)

- ◎프로그램 논리가 순서도로 표현되면 프로그래밍 언어를 사용하여 명령어로 표현한다. 적절한 언어를 선택하여 그 언어의 문법에 준해 기술한다.
- ◎이때 일정한 언어로 작성하는 과정을 프로그래밍 또는 코딩이라 한다. 프로그램을 프로그래밍 언어로 표현한 것을 원시 프로그램이라 한다.

1.2 프로그램 작성 단계(Cont'd)

□ 수정(debugging)

- ◎입력된 프로그램은 언어 번역 프로그램에 의하여 기계어로 번역.
- ◎이 때 문법적으로 잘못된 부분이 있으면 그 내용을 인쇄 장치나 화면에 출력.
- ◎프로그램이 일단 완성되면 작성된 프로그램이 정확한가 또는 논리적으로 잘못 작성된 부분이 있는가를 테스트하여야 하는데, 이와 같이 프로그램의 정확성을 테스트하는 과정을 디버깅(debugging)이라 한다.
- ◎보통 디버깅은 테스트 상의 체크, 기계를 사용하는 테스트, 실제 데이터를 사용하는 테스트로 구분된다.

□ Test Run

- ◎디버깅이 완료되고 프로그램 상에 이상이 없게 되면 모의 데이터를 사용하여 프로그램의 이상 여부를 최종 테스트한다.
- ◎Test Run의 결과 올바른 결과를 얻지 못하면 다시 디버깅 과정을 반복하게 되며, Test Run이 완전히 끝난 다음에 실제의 Run에 착수한다. 이와 같이 하여 완전한 프로그램이 마련되면 실제 데이터를 처리하게 되는데 이 과정을 러닝(running)이라 한다.

1.2 프로그램 작성 단계(Cont'd)

□ 문서화(documentation)

- ◎문서화는 반드시 필요하며 프로그램을 가장 빨리 파악할 수 있도록 작성
- ◎담당자가 변경되어도 쉽게 파악할 수 있도록 상세하게 작성되어야 한다.

1.3 순서도의 개념

□ 알고리즘(algorithm)

- ◎어떤 문제를 해결하려고 할 때 존재하는 규칙들을 나열하여 수학에서의 용어를 컴퓨터의 용어로서 사용하는 것.
- ◎즉, 알고리즘이란 어떤 주어진 문제를 유한 단계 내에서 어떤 해를 구할 수 있게끔 나타낸 명확한 단계의 연속을 의미.

□ 알고리즘 작성기법

- ◎하향식 설계(top-down design) 기법 :
- ◎상향식 설계(bottom-up) 기법 :

□ 순서도

- ◎처리하고자 하는 문제를 분석하여 그 처리 순서를 단계화시켜 상호간의 관계를 일정한 기호를 사용하여 일목요연하게 나타낸 그림

1.3 순서도의 개념 (Cont'd)

□ 순서도 작성으로 인하여 얻을 수 있는 효과

- ① 문제 처리의 과정을 논리적으로 파악할 수 있어 정확성 여부에 대한 쉬운 판단 가능.
- ② 문제를 해석하고 분석하며, 타인에게 전달하는 것이 용이.
- ③ 프로그램의 보관, 보수, 유지의 자료로서 활용이 가능.
- ④ 프로그램 코딩의 기본 자료로서 활용 가능 .

1.4 순서도의 기호

□ 순서도 기호의 종류

- 기본 기호(basic symbol)
- 프로그래밍 관련 기호(symbols related to programming)
- 시스템 관련 기호(symbols related to systems)


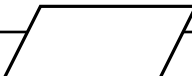
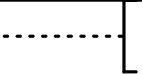

□ 순서도의 일반적인 작성 규칙

- ① 기호의 내부에 처리할 내용을 기입한다.
- ② 흐름도의 방향을 위에서 아래로, 좌에서 우로 하는 것을 원칙으로 하되 반대일 경우에는 반드시 화살표를 사용한다.
- ③ 흐름선은 교차시켜도 아무런 논리적 관계를 갖지 않는다.
- ④ 흐름선 두 개 이상이 모여 한 줄이 될 수 있다.
- ⑤ 기호의 모형상 가로, 세로의 비율은 정확하게 정해지지는 않았으나, 잘 구분될 수 있도록 해야 한다.

1.4 순서도의 기호 (Cont'd)



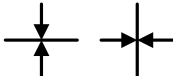
□ 기본 기호(basic symbol)

◎ 시스템 순서도와 프로그램 양 쪽에서 공동으로 사용되는 기본적인 기호

기 호	이 름	용 도
	처리 (process)	지정된 동작, 각종 연산, 값이나 기억 장소의 변화, 데이터의 이동 등의 모든 처리 과정을 나타낸다.
	입출력 (input/output)	일반적인 입력과 출력 처리를 나타낸다.
	주해 (comment annotation)	표현된 기호를 보다 구체적으로 설명하며, 점선은 해당 기호까지 연결한다.
	연결자 (connector)	흐름이 다른 곳으로 연결됨과 다른 곳에서 연결되는 입구를 나타내며, 화살표와 기호 내에 쓰여진 이름의 연결 관계를 나타낸다.

1.4 순서도의 기호 (Cont'd)

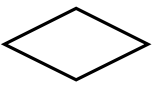



□ 기본 기호(basic symbol)

기 호	이 름	용 도
	페이지 연결자 (offpage connector)	흐름이 다른 페이지로 연결됨과 다른 페이지에서부터 연결되는 입구를 나타내며, 연결자와 같은 방법으로 사용한다.
	흐름선 (flow line)	처리의 흐름을 나타내며, 선이 연결되는 순서대로 진행됨을 나타낸다.
		여러 개의 흐름이 한 곳으로 모여서 하나가 됨을 나타낸다.

1.4 순서도의 기호 (Cont'd)

□ 프로그래밍 관련 기호



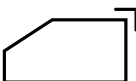

◎프로그램의 논리를 나타내는 기호로서 기본 기호와 함께 프로그램 순서도를 작성하는 경우에 사용

기 호	이 름	용 도
	의사 결정 (decision)	주어진 조건에 따라 비교하여 해당되는 조건을 찾아서 흐름이 결정되게 하며, 왼쪽이나 오른쪽 또는 아래로 흐름이 분기하게 된다.
	정의된 처리 (predefined process)	이미 다른 곳에서 표현된 논리를 인용함을 나타내며, 서브프로그램은 다른 곳에 표현된 것이므로 이 기호를 사용하여 호출함을 나타낸다.
	준비 (preparation)	기억 장소의 할당, 초기값 설정, 설정된 스위치의 변화, 인덱스 레지스터의 변화, 순환 처리를 위한 준비 등을 나타낸다.
	단자 (terminator)	프로그램 순서도의 시작과 끝을 나타낸다.

1.4 순서도의 기호 (Cont'd)

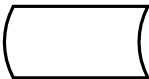
□ 시스템 관련 기호

◎시스템을 분석하고 설계할 때, 데이터가 어느 매체에서부터 처리되어 어느 매체로 변환되어 해당되는 곳으로 흐르는지를 나타내는 경우에 사용되는 기호들로서 기본 기호와 함께 시스템 순서도를 작성하는 데에 사용

기 호	이 름	용 도
	펀치 카드 (punched card)	펀치 카드 매체에 의한 입력과 출력을 나타낸다.
	카드 뭉치 (card deck)	펀치 카드가 모여 있음을 나타낸다.
	카드 파일 (card file)	펀치 카드에 레코드가 모여서 파일을 구성하고 있음을 나타낸다.
	서류 (document)	각종 원시 데이터가 기록된 서류나 종이 매체에 출력되는 결과 및 문서화된 각종 서류를 나타낸다.



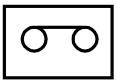


1.4 순서도의 기호 (Cont'd)

□ 시스템 관련 기호

기 호	이 름	용 도
	자기 테이프 (magnetic tape)	자기 테이프 매체에 의한 입력과 출력을 나타낸다.
	종이 테이프 (Paper Tape)	종이 테이프 매체에 의한 입력과 출력을 나타낸다.
	키 작업 (keying)	자판(keyboard)을 통한 키 펀칭이나 검사 등의 동작을 나타낸다.
	온라인 기억장치 (on-line storage)	온라인 상태의 각종 보조기억장치에 의한 입력과 출력을 나타낸다.
	자기 드럼 (magnetic drum)	자기 드럼 매체를 통한 입력과 출력을 나타낸다.




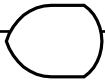
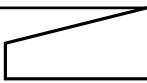
1.4 순서도의 기호 (Cont'd)

□ 시스템 관련 기호

기 호	이 름	용 도
	자기 디스크 (magnetic disk)	자기 디스크 매체를 통한 입력과 출력을 나타낸다.
	디스켓 (diskette)	디스켓(floppy disk) 매체를 통한 입력과 출력을 나타낸다.
	카세트 테이프 (cassette tape)	카세트 테이프 매체를 통한 입력과 출력을 나타낸다.
	오프라인 기억장치 (off-line storage)	오프라인 상태의 기억 매체에 레코드들이 기록됨을 나타낸다.
	병합 (merge)	정렬된 2개 이상의 파일을 하나로 합치는 작업을 나타낸다.



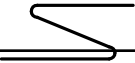
1.4 순서도의 기호 (Cont'd)

□ 시스템 관련 기호

기 호	이 름	용 도
	대합 (collate)	2개 이상의 파일을 합쳐서 다른 2개 이상의 파일을 만드는 작업을 나타낸다.
	정렬 (sort)	일정한 순서 없이 배열된 데이터를 일정한 기준에 따라 순서대로 배열하는 작업을 나타낸다.
	추출 (extract)	만들어진 파일에서 필요한 부분만을 분리하여 새로운 파일을 만드는 작업을 말한다.
	화면 표시 (display)	결과나 필요한 메시지를 온라인으로 연결된 CRT, 콘솔 등에 출력하는 것을 나타낸다.
	수동 입력 (manual input)	각종 정보를 온라인으로 연결된 자판 (keyboard) 또는 스위치 등을 통하여 수동으로 입력하는 것을 나타낸다.

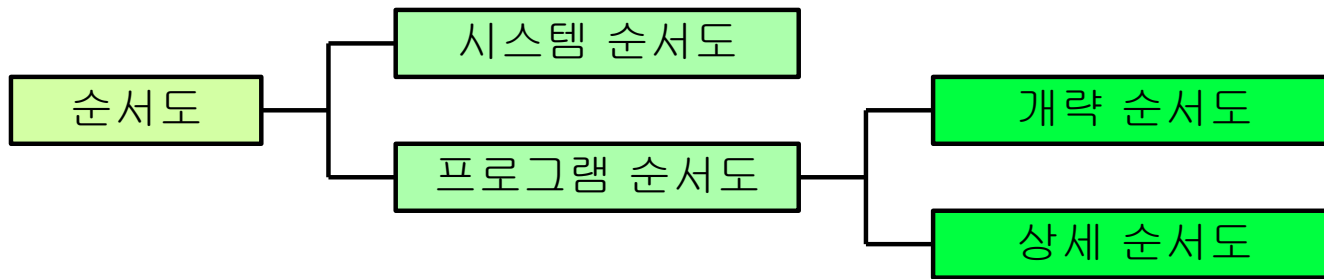
1.4 순서도의 기호 (Cont'd)

□ 시스템 관련 기호

기 호	이 름	용 도
	수동 조작 (manual operation)	오프라인 상태에서 기계에 의하지 않고 손으로 조작되는 각종 데이터 처리 작업을 나타낸다.
	보조 조작 (auxiliary operation)	오프라인 상태에서 직접 중앙처리장치의 통제를 받지 않은 장치에서 행해지는 작업을 말한다.
	통신 연결 (communication link)	전화선이나 무선 등의 각종 통신회선과 연결됨을 나타낸다.

1.5 순서도의 종류

□ 순서도의 종류 및 분류



□ 시스템 순서도(system flowchart)

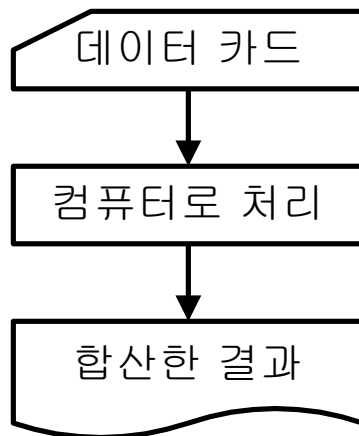
- ◎ 시스템 분석가가 전체 시스템의 관계를 표현
- ◎ 시스템 설계 단계에서 작성하는 순서도
- ◎ 원시 데이터의 발생에서부터 최종 결과에 이르기까지 전체 과정을 데이터의 흐름을 중심으로 표현
- ◎ 컴퓨터 외부에서 수행되는 모든 과정을 표현하며, 데이터의 흐름이 중심이기 때문에 데이터가 처리되는 작업 단위로 나타내고 데이터가 변환되는 매체들을 표현하며 프로그램 논리는 표현하지 않는다.

1.5 순서도의 종류 (Cont'd)

□ 시스템 순서도의 작성 목적

- ◎ 대상 업무에 관한 처리 과정의 전체적인 연관성을 파악한다.
- ◎ 작성한 프로그램의 수를 파악한다.
- ◎ 작성된 프로그램의 작업 순위를 안다.
- ◎ 대상 업무의 전체적인 실행 번호, 입력 번호, 출력 번호를 정한다.
- ◎ 기계 처리 시간을 산출한다.

□ 시스템 순서도의 예



두 개의 값(A, B)을 읽어서 두 수의 합(S)을 구하는 경우의 시스템 순서도

1.5 순서도의 종류 (Cont'd)

□ 프로그램 순서도(program flowchart)

- ◎ 프로그래밍 단계에서 작성하는 순서도.
- ◎ 컴퓨터 내부에서의 처리 과정을 표현하는 것으로 프로그램 논리를 나타냄.
- ◎ 일반적으로 프로그래머가 작성하는 순서도는 프로그램 순서도가 대부분.

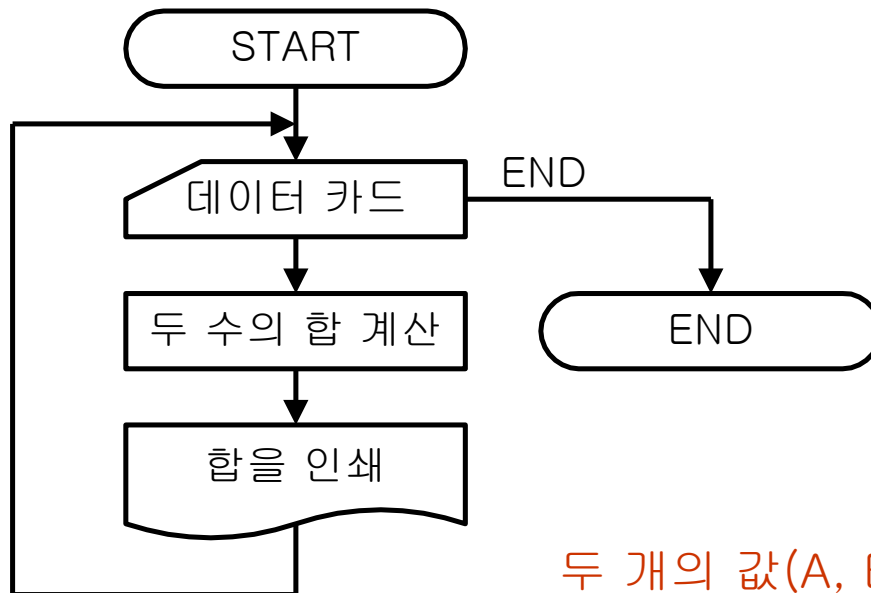
□ 프로그램 순서도의 사용 목적

- ◎ 문제 해결을 위한 논리적인 순서의 검토와 구성
- ◎ 프로그램 서류(document)로서의 기록, 전달용(후일의 사양 변경 등에서 사용하는 보조 수단)

1.5 순서도의 종류 (Cont'd)

□ 개략 순서도(general flowchart)

- ◎ 처리 대상 업무의 전반에 대하여 종합적인 사항과 일반적인 사항을 나타낸 순서도
- ◎ 처리되는 업무의 전체적인 구조와 기능을 용이하게 파악할 수 있음.



두 개의 값(A, B)을 읽어서 두 수의 합(S)을 구하는 경우의 개략 순서도

1.5 순서도의 종류 (Cont'd)

□ 상세 순서도(detail flowchart)

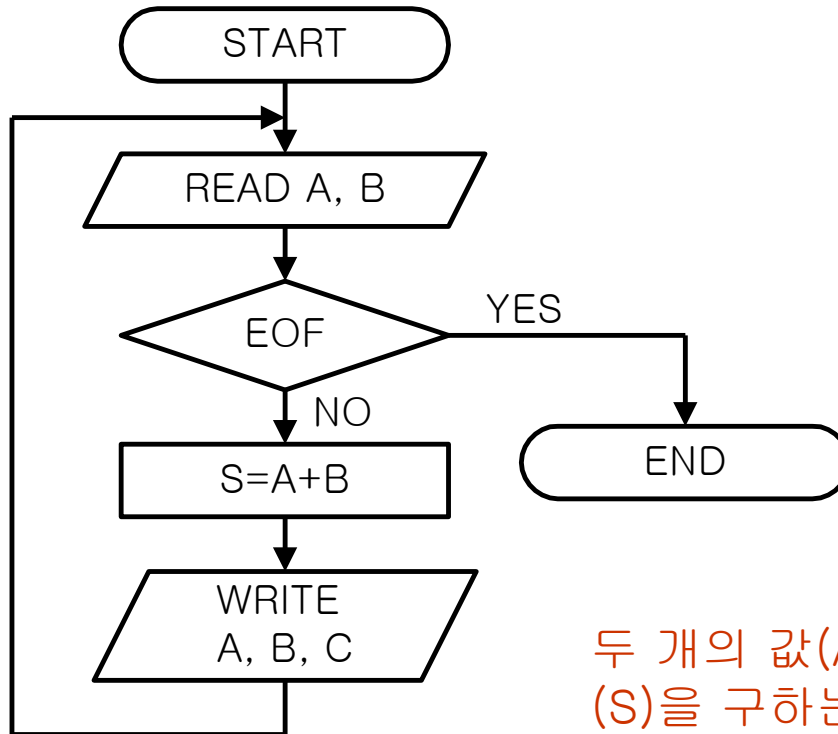
- ◎ 프로그램을 작성할 때 순서도의 내용 그대로 직접 프로그램을 작성할 수 있도록 세부적으로 표현하는 순서도
- ◎ 일반적으로 프로그램 설계 과정에서 작성
- ◎ 컴퓨터의 모든 조작과 데이터의 이동 과정을 순서대로 나타낸 것으로서 이 내용을 그대로 코딩할 수 있도록 세밀하게 작성된 순서도

□ 상세 순서도의 용도

- ◎ 프로그램 작성 상 최종 검토 자료
- ◎ 코딩 자료
- ◎ 프로그램의 논리적 모순 검토 자료
- ◎ 문서 자료

1.5 순서도의 종류 (Cont'd)

□ 상세 순서도의 예



두 개의 값(A, B)을 읽어서 두 수의 합 (S)을 구하는 경우의 상세 순서도

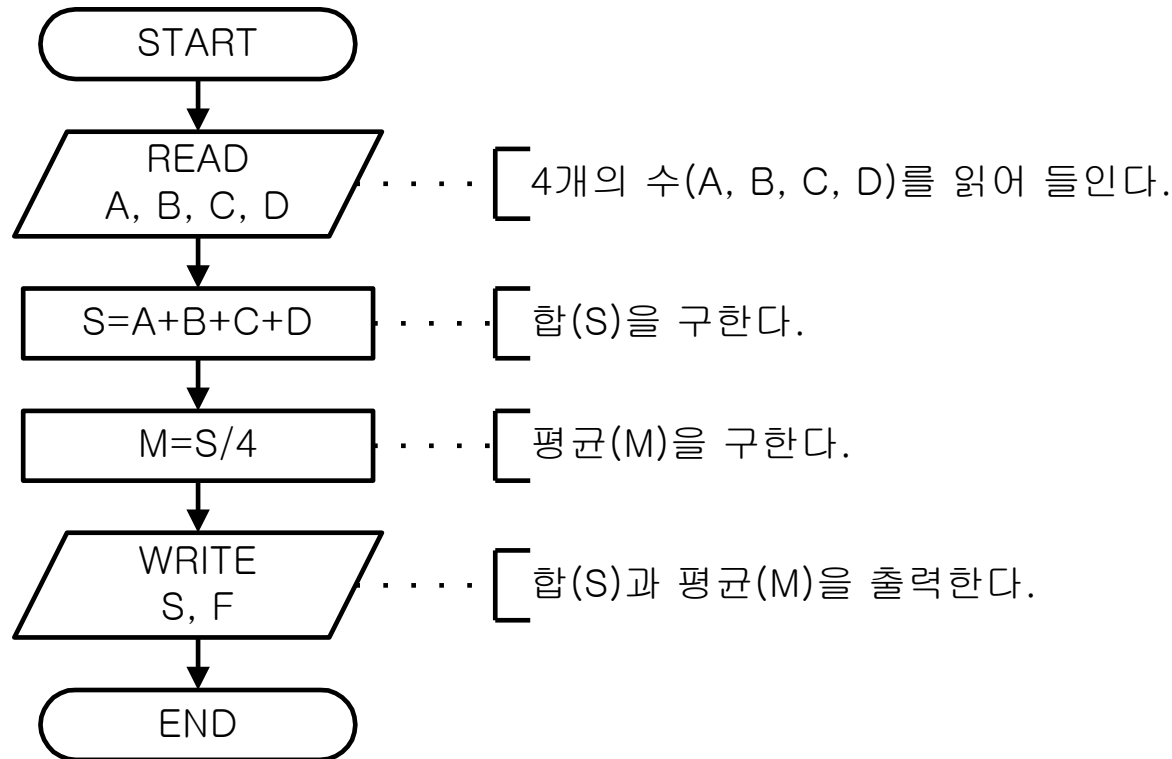
1.6 순서도 작성 방법

□ 프로그램 순서도 작성시 고려사항

- ◎ 처리되는 과정은 모두 표현한다.
- ◎ 간단하고 명료하게 표현한다.
- ◎ 전체의 흐름을 명확히 알아볼 수 있도록 작성한다.
- ◎ 과정이 길거나 복잡하면 나누어서 작성하고 연결자로 연결한다.
- ◎ 통일된 기호를 사용한다.

1.6 순서도 작성 방법(Cont'd)

□ 단순 처리 순서도의 예



4개의 수(A, B, C, D)를 읽어 합과 평균을 구하는 과정