

構造電算의 事務化 技法에 關하여

Program Technique of Structural Design

咸性權 / 안산성도건축사사무소
by Ham, Song Kwon



序

構造電算의 사무화기법은 PC 시대가 시작된지 시간이 얼마되지 않아서 아직 국내외로 이렇다할 확고한 기법이 정립되어 있지 않은 것이 현실이다. 本論에서 말하는 構造電算의 技法이라 함은 단순한 力學的, 數學的 過程에 따라 결과치만을 출력하는 SAP 등 구조물의 不靜定解法 프로그램과 같은 것의 일정한 흐름을 말하는 것이 아니라 실제로 실용구조설계를 하는 구조계산자가 그 전산프로그램밍을 함께 있어 계산의 흐름을 어떻게 하고, 어떻게 서로 연결할 것이며 전산의 입출력을 어떻게 하면 부담없이 가장 쉽게 전산 조작이 될 것이며, 또 흔히 있을 수 있는 곤란한 전산 조작성의 실수에 대해서는 어떤 하드웨어/소프트웨어 기법을 쓰는 것이 좋을 것이냐, 또는 프린트되어 나오는 구조계산서 서식은 어떤 것이 가장 간결 명료하고 입출력치가 잘 설명될 것이냐의 문제등으로부터 최후로는 전산결과와 신뢰성의 입증에 관한 문제까지 포함한, 복잡한 實用電算의 事務化技法을 말한다. 이와같은 견지에서 볼 때 구조전산의 사무화기법이란 사실상 기본목적은 서로 같다 하더라도 각 개인마다 그 내용은 각양각색이 될 것이며, 여기에서 희망되는 것은 각개인의 각자의 프로그램조작의 기법을 공개할 때 그것이 서로의 참고가 되어 우리나라 구조기술의 발전에 큰 도움이 될 것이라고 믿어진다.

본인의 경험에 따를 때 구조계산의 사무화기법에 관련되는 要目들은 대체적으로 다음과 같다고 생각되며, 각항에 대하여 소견과 본인이 적용하고 있는 기법을 설명하면 다음과 같으며, 마지막으로 프로그램 예를 소개할까 한다.

1. 사용 電算言語의 選擇
2. 一貫設計 프로그램 問題
3. 對話式 電算과 원활한 자료입력 (Date Supply)
4. 可變事項에 대한 최대한의 對備
5. 화일의 블랙박스(Black-Box) 性에 대한 대처
6. 誤算의 화일화에 대한 防止策

7. 根據가 명확한 構造計算書의 프린팅

電算言語의 選擇

본인은 전산언어로 BASIC 을 택하고 있다. 그 이유는 이 언어는 비록 통속적이라는 인상을 주기는 하지만 퍼스날 프로그래머(Personal Programmer)에 알맞게 프로그램이 타언어에 비하여 용이하고 인간적이고 인터랙티브(Interactive) 즉 대화식 프로그램에 적합하며 오늘날까지 경험한 바 구조전산 프로그램에 있어 계산과 표현에 아무런 부족함이 없다는 점이다. 그리고 더욱이나 구조전산에 BASIC 보다 나은 언어가 더있다 하더라도 하나의 전산언어의 습득에는 많은 연수가 필요하다는 점을 감안할때 그것은 구조기술을 전공하는 자에게는 주객이 전도되는 무리한 일이라고 볼 수도 있다.

BASIC 은 타언어에 비하여 전산기의 RAM 즉 기억용량을 많이 차지하고 실행속도(Running time)도 길다는 단점이 있다고는 하지만 메가급 용량과 고성능 부품을 가진 탁상전산기에서는 전혀 문제가 되지 않고 그 가격도 과히 비싸지 않다. 일반적으로 전산기는 빠르면 빠를수록 좋다고 하지만 대화식 구조계산에 있어서는 전산기의 조작성이 대응할 수 있는 이상의 전산속도는 무의미한 것이고 커피브레이크(Coffee Break)를 위해서는 좀 느린 것이 오히려 바람직하다는 역설도 나오게 된다.

一貫設計 프로그램

構造電算의 理想은 한개의 프로그램 블록으로 구조계산에서부터 구조설계도의 작성까지의 全自動化 즉 一貫構造計算 프로그램의 작성에 있다. 그러나 현단계에서 이와같은 일관프로그램을 평하여 볼 때 일반적으로 건축구조물이 규칙적인 層構造이고 부분적 변화가 얼마 안되는 것에는 그것을 적용하여서 효과가 있을 것이나 구조의 변화가 많거나, 소규모이거나 공장건물 같은 것에는 그와같은 프로그램은

咸性權
1918년 함경남도생.
현재 한양대 건축과 명예교수
본협회 구조분과위원

내용상 적용될 것은 못된다는 것을 짐작할 수 있다. 사실상 이 문제는 일본에서도 그 효용의 한계가 논의된 바 있다고 하며 일관구조설계 프로그램으로 이름난 미국의 GT. STRUDL 도 그 교본을 (Manual) 을 일람하였을 때 막대한 입력규약에 놀라지 않을 수 없었다. 이와같은 사실에 비추어 항상 도시 고층건물만을 다루는 것이 아닌 일반구조설계자에게 알맞는 구조전산프로그램의 형태는 과연 어떤 것일까라는 문제가 제기될 때 이에 대하여 본인은 근래에 완성한 RC. 전산프로그램에서 그 프로그램 형태를 링크(Link) 식 일관프로그램으로 하였다. 이것은 일반사항, 슬라브설계, 보설계 등 개개의 프로그램을 서로 연결하여 일관프로그램의 격식을 갖춘 것으로 선행된 부분에서 입력한 모든 설계변수와(슬라브크기, 보의 스패น(Span 長 등) 설계된 결과들(보의 CVMK, 기둥축력 등) 프로그램의 뒷부분에서 자동적으로 인계하여 이용하는 방법으로서 프로그램에 일관성을 부여하는 동시에 개별프로그램이 연결되는 부분에서 여러가지 인위적인 방향을 잡을 수 있도록 한 것이다. 실무에 수차 이용한 바 시도하여 불만한 가치가 있다고 생각된다.

對話式 電算

전산은 그 프로그램이 작성된대로 흐르고, 전산자의 무드는 그 흐름에 따라 움직이고, 좋은 구조 계산프로그램은 사용자로 하여금 좋은 구조기술적 무드에 잠גיע 할 것이다. 이런 점에서 볼때 전산프로그램이란 단순한 어떤 사무내지 기술적 처리에 그치는 것이 아니라 그 자체에 예술성을 지닌 것이라고 말할 수 있다. 사실상 프로그램사용자에 대한 프로그램의 표현에는 비디오 스크린 위에 배치되는 문자나 도형의 배치상태에서부터 문장, 화술에 이르기까지 여러가지가 있는 것이나 이들중 무엇보다도, 한낱 쇠부치에 지나지 않는 전산기에 인간미를 주게하는 것은 그 대화기능에 있다고 말할 수 있을 것이다. 무미한, 역속된 숫자기호들로서 입력되고,

```

<< MAIN MENU >>
(그림 1)
Code.      Item.
.....
1  < USE GUIDANCE >.
2  General Description, and Loads.
3  Slab, Stair Design.
4  C,V, etc. of Beam.
5  N,k of Column and rc. Wall.
6  2-cycle Stress Analysis + Wind Load of Frame.
7  Energy Methods ( plane, space frame, grillage ).
8  Design, Check of Beam + M,V of Beam.
9  Column, rc. Wall Design.
10 Special Rectangular Slab, Basement Wall.
11 Mat Foundation.
12 Retaining Wall & Flat Slab.
13 Footing Design.
14 DESIGNED MEMBER LIST.
15 Preparation for total print.
16 TOTAL PRINT OF DESIGN.
17 JOB DONE.

enter item code number to use ?

< SUB-MENU >
(그림 2)
Code.      Design item.
.....
1  Rectangular slab.
2  Cantilever slab.
3  Slab or Wall section (given M,V).
4  One-way stair.
5  Cantilever stair.
6  Supplementary Description.
7  PRINT.
8  CANCEL-RESTART.
9  SUB-MENU DONE.

enter code number to use ? 7

TURN ON your printer and enter their code number, one by one. DONE <skip> ? 1
enter print starting number. when print all, <skip> ? 3

Korean standard occupancies for live loads are:
(그림 3)
Code      Standard occupancy
.....
1  dwelling, etc. and their corr., hall, stairs.
2  office, etc. and their corr., hall, stairs.
3  stores, etc.
4  class room, etc.
5  assemblage - fixed seats.
6  assemblage - no seats.
7  garage, car pass.
8  roof - general.
9  roof - heavy live loads.
10 corr., hall, stairs except for dwell., offi.
11 for live loads not specified in Korean code.

enter LL. code to adopt. JOB DONE <skip> ? 2

enter pertaining slabs, one by one. when DONE <skip> ? s2

< INPUT of DATAS >
(그림 4)
enter item. JOB DONE <skip> ? s12
when wall design , enter <y>, otherwise <skip> ?
enter M (t/m) ? 2      enter V (t/m) ? 1.5
when short-time design, enter <y>, otherwise <skip> ?
when steel on one-face of slab only, enter <y>, otherwise <skip> ?
enter % of temp.steel. when <skip>, 0.2% ?

rqd.t = 13.66 cm.(dt=3cm).  enter t,dt (cm) ? 14,3

Rqd. rebar D-x of <s12>:
D: 10 13,10 13 16,13 16 19,16 19 22,19 22 25,22 25 .
main-bars: 6.8 9.3 12.2 15.4 19.0 23.1 27.5 32.2 37.4 40.0
temp-bars: 40.0
! temp. bars above are for both faces of member.
select rebar D-x.  main-bars:  D ? 10      x ? 6.8
                  temp-bars:  D ? 10      x ? 40

enter notes. NONE <skip> ? test.

when INPUT OK, enter <y>, otherwise <skip> ? y

```

```

<그림 5> 275 OPEN"i",4,"s5.d":GOSUB 565:IF AB$="o"THEN OPEN AB$,4,"s5.d"ELSE OPEN"s5.d"FOR APPEND AS 4
280 GOSUB 950:IF M$="end"THEN 15
285 CLS:PRINT TAB(30)HH$:PRINT:PRINT TAB(9);:GOSUB 550:IF A$=""THEN 280 ELSE PRINT      when ";;COLOR 1,0,0:PRINT" wall design ";;COL
OR 1,7,7:PRINT", enter <y>, otherwise <skip> ";;INPUT MM$
290 INPUT      enter M (t/m) ";;P:DC=-P:GOSUB 935:M1$=M$:P=P#1000!:IF MM$=""THEN LOCATE CSRLIN-1,35:INPUT"enter V (t/m) ";;Q:DC=Q:GOSUB
B 935:V$=M$:IF Q=0 THEN Q=.1 ELSE Q=Q#1000! ELSE Q=.1:V$=" 0

335 P1=Z:PRINT TAB(11);:GOSUB 685:PRINT"temp-bars:";;Y=0:Z=P1:N=1E-09:PRINT TAB(11);:GOSUB 685:IF M$=""THEN PRINT ELSE PRINT TAB(9)"# te
mp. bars above are for "PD$" of member.
340 PRINT      "JJ$:LOCATE 19,31:PRINT"main-bars:";K=18:GOSUB 710:A1$=A2$:LOCATE 20,31:PRINT"temp-bars:";K=19:GOSUB 710:PRINT:AB=21:B
C=B2-LEN(S$)-LEN(D$)-LEN(FC$):GOSUB 560:GOSUB 640
345 IF M$=""THEN 285 ELSE WRITE#4,A$,V$+M1$,R,-DT,A1$,A2$,S$+FC$+D$+G$:GOTO 285

565 IF EOF(4)=-1 THEN AB$="o"ELSE AB$="e
570 CLOSE:PRINT:PRINT TAB(23)"-- to continue, press <any key> --":M$=INPUT$(1):RETURN

640 PRINT
645 INPUT;"when INPUT OK, enter <y>, otherwise <skip> ";;M$:RETURN

```

나열된 指數 (Expoment) 숫자로서 출력되는 포트란 (FORTRAN) 식 전산에 비하여 요즘 흔히 Interactive 라는 술어로 애용되고 있는 대화식 프로그래밍은 그야말로 생명없는 전산기에 약동감과 생동감을 일으키게 하였다고 말할 수 있다. 지금 위와같은 취지에 우리들의 구조전산프로그램이 대화식으로 작성되었다고 할 때 그 電算器는 조작자에게 극히 부담을 느끼지 않도록 조심하면서 내적 자료인용과 더불어 그때그때의 필요한 자료의 제시와 함께 계산의 순서대로 필요한 입력자들을 Key-in 하도록 유도할 것이고 마침내 설계결과를 제시하면서 OK 여부를 물을 것이며, 이와같은 과정의 반복으로 구조계산은 만족히 끝날 것이다. 사실상 대화식 구조전산프로그램에서는 그것이 단순한 設計用 數式의 演算에 그치는 것이 아니라 세심한 설계유도와 설계자의 옵션 (Option) 과 구차스러운 문헌참조를 피하기 위하여 비디오상에 그때그때의 충분한 참고 데이터의 디스플레이도 하게 될 것이다.

可變事項에 대한 對備

일관된 프로그램 블록으로 일개 건물의 구조 전체를 연속 전산할 때 예기치 못하게 일어날 수 있는 難題는 그 프로그램 블록으로는 처리할 수 없는 可變事項들이 생길 수 있다는 점이다. 이와같은 可變事項에는 일상구조계산에서 자주 일어날 수 있는 것들과 극히 드물게 일어날 것이라고 생각되는 가변사항도 있다. 예를 들어 특정한 構造部材에 대한 短期荷重에 대한 추가계산 또는 검토, 구조물의 설계가 아닌 체크 (Check) 구조계산의 시행, 슬라브철근이 前斷力에 의한 附着으로 결정되는 경우, 인장과 휨을 받는 기둥의 단면설계등이 지금 여기에서 논의되고 있는 사항들이며 이 정도의 빈도가 많은 가변사항은 그 처리프로그램을 본프로그램에 대비시켜 두는 것이 바람직할 것이고 기타 극히 특이한 가변사항에 대해서는 筆算 등 별도 조치로 구조계산서에 補足할 것인데 그것에 대해서는 프로그램중에 記述적으로

프린팅만을 할 수 있는 Supplementary Description 과 같은 프로그램을 마련하여 두는 것이 바람직하다.

블랙박스성에 대한 대처

일관성을 목표로 한 構造電算에서는 어떤 경우라도 각부분의 계산결과를 잠정적으로나, 또는 전체 구조계산결과를 최종제출용 구조계산서의 프린팅을 위하여 화일에 저장하여 두어야 한다. 그런데 여기에서 문제가 되는 것은 화일이란 원래 블랙박스라는 점에 있다. 계산하여 화일에 보관하여 둔 계산 결과들을 일일이 모두 기억하여 둘 수 없고 기록하여 둔다 하더라도 착오가 일어난다.

예를 들어 방금전에 B1 보에 대한 단면산정을 하였던가, 또는 S3 슬라브를 계산하였는데 그 배근을 어떻게 하였던가, 그리고 전날에 이어 구조계산을 연속할 때 과연 어제 저녁에 어디까지 계산하였던가라는 경우 그때까지의 전체계산을 프린팅하여 본다든 것도 난처하고 막연한 처지에 놓이게 된다. 이와같은 경우 그 해결방안은 Peep Program 으로 화일내의 필요부분을 색출하여 프린트하여 보는 것인데 본인은 이에 대하여 계산항목별 任意 Print out, Designed Member List 법을 쓰고 있다. 전자는 수시로 어떤 항목의 계산화일에서 全部材에 대한 계산결과 또는 먼저까지 引出프린트하여 본 이후의 계산들을 Print out 하여 보는 방법이고 후자는 어떤 시점까지 모든 화일 속에 처리보관하여 놓은 각항목별 部材名들만을 引出프린트하여 보는 방법이다. 참고삼아 이와같은 화일내의 색출에서 부재명만의 인출에 대해서는 Input #에서 부재명의 변수만을 프린팅하면 될 것이고, 먼저 引出프린트한 것 이후의 계산들의 引出프린트에는 화일의 내용에 넘버링 (Numbering) 을 하면 될 것이다. 그리고 이들 기법을 채용함에 있어 유의하여야 할 점은 아직 아무런 계산도 들어가 있지 않은 화일을 Open 하여 Input #할 때 생기는 Error Display 에

대해서는 계산이 들어 있는 화일에서는 eof 가 영이 되고 그렇지 않으면 -1이 된다는 BASIC의 문법을 적용하면 된다. (Sequential 화일 사용을 전제)

誤算의 화일化 防止

上述한바 화일은 一貫的 構造計算에 있어 필수적인 것이며 그 블랙박스성이 문제점으로 지적된다. 그러나 또 하나 이 화일에서 중요한 문제가 되는 것은 일단 화일에 예치된 내용은 쉽게 정정이 되도록 되어 있지 않다는 점이다. 아시다시피 화일에는 Sequential 과 Direct 의 두가지가 있고 그중 Direct 화일은 이와같은 불편이 없으나 일관적 구조계산에는 부적합한 것임으로 Sequential 화일을 들어 그에 대한 적절한 대책이 논의된다. 화일에 입력된 내용이 이와같이 정정이 실용적으로 어려울 때 그 대안의 하나는 잘못된 계산결과를 최대한으로 화일에 입력되지 않도록 방지하는 방안과 또 하나는 잘못 입력된 화일 전체를 말소하고 모든 계산을 다시하여 화일에 입력하는, 원할 바 못되는 방안이 생각되는 데 이때 가능하면 말소의 범위를 극소화 하자는 방안이다. 본인이 쓰고 있는 이와같은 수법들을 요약하면 INPUT OK, VIRTUAL DISK, CANCEL-RESTART 의 세가지 방법으로서 그 내용을 설명하면 다음과 같다. INPUT OK 수법이란 誤算을 최대한으로 화일에 입력되지 않도록 방지하는 방법으로서, 여기에서 誤算이란 부주의등으로 저질러지는 Miss-key-in, 오판에 따른 잘못된 설계, 계산결과와 불만족 등으로 한 設計演算의 최종부분 또는 필요하면 그 도중에도 "When INPUT OK, Enter <y>, Otherwise <Ship>?" 의 if Then Else 문법을 삽입하여, 설계연산이 잘 이루어졌으면 演算이 前進 또는 계산결과가 화일에 입력되도록 하고 그렇지 않으면 그때까지의 연산을 취소하고 적당한 原點에 돌아가 다시 再演算을 하도록 하는 방법이다. 이 방법에 따를 때 뜻하지 않은

Design of Slab(Wall) Section.

V-M: given shear (t/m, when wall. excluded) and bending mt.(t/m) Source of M,V: see note. t,dt(cm). D-x: bar dia. and spacing selected.(mm,ca) main-bars: rqd. tension steel for V,M (or temp.steel). temp-bars: rqd. temp.steel perpendicular to main bars.(0.2% when not mentioned) s-d: short-time design. 2ft,1ft: 2-face, 1-face temp. steel. if any allow. stress differs from originally specified, it is printed.

item	V-M	t-dt	main-bars	temp-bars	note
			D-x	D-x	
fsx	82.61-41.3	110-4	22-12	-	temp.steel=.05% 2ft.
fsy	28.1-20.65	110-4	22-34.9	-	temp.steel=.05% 2ft.

next n = 3

誤算의 화일 入力을 막을 수 있고, 또 다른 큰 이점으로는 電算器의 本領이라고 할 수 있는 반복적 최적연산을 할 수도 있다는 사실을 들 수 있다.

VIRTUAL DISK 라 함은 재래에는 Mem. Disk 또는 Ramdisk 라고 불렀던 것으로 전산기 속의 기억소자인 Ram 속에 가상적 디스크를 설정하고 그것에 드라이브(Drive) 속의 실제 디스크와 마찬가지로 전산결과를 입력, 보존하는 것으로서 일단 작업이 끝나면 그 내용을 드라이브 속의 실제 디스크에 안전하게 옮겨 놓는 것이다. Virtual Disk 의 이점은 디스크가 모터로 회전하여 기능을 발휘하는 것이 아니라 전산기의 기억소자속에 디스크가 있는 것으로 전선속도가 하드 디스크 보다도 월등 빠르며 이것이 誤算의 화일화방지에 도움이 되는 것은 새롭게 시작된 전산이 전반적으로 잘못되었거나 全般的 訂正을 요할 때에는 이때까지의 작업을 포기하고 실제 디스크의 내용을 다시 Virtual Disk 에 옮기면 아무런 걱정없이 먼저한 작업에 이어 일을 다시 시작케 되는 것이다.

CANCEL-RESTART 수법은 전산에 있어 어떤 화일에 오산이 입력되었을때 그 화일을 Kill 하고 다시 Open 하여 모든 작업을 다시하여 화일을 재작성하는 것인데 이것은 화일 내용이 얼마 안되고 위의 Virtual Disk 에 관한 수법을 쓰기에 은곤란할 때에 편리하다.

根據있는 構造計算書의 作成

전산에 의한 構造計算의 내용에는 演算과 같은 것까지 표시할 수는 없다. 그러나 構造 계산의 내용을 알아보고 檢산할 수 있도록 入出力値가 明示되어 있고 설명도 되어 있는 것이 바람직하다. 참고삼아 前述한 바 본인이 작성한 RC 프로그램에서는 構造전산이 완료되었을 때 표지, 목차에서부터 전 구조작업내용이 프린트 되어 그냥 재본될 수 있도록 되어 있고, 매설계항목마다 入出力치가 명시되어 있고 그에 대한 오해가 없도록 설명이 가해져 있으며 내용의 Compact 화를 위하여

구조계산서는 모두 Table 형식으로 프로그램 되었다. 그리고 마지막으로 구조전산내용의 입증으로는 별도로 프로그램式을 마련하였고 다각도로 작성한 構造計算 算算例題과 그 電算結果를 對比하여 놓았다.

프로그램 例

위의 각항에서 설명한 바를 구체적으로 실례로서 설명하여 보면 그림 1~6과 같다. 그림1은 개별적 설계프로그램들의 일관적 연결을 보여주는 Main Menu 이며, 14항이 전산을 쉬었다가 다시 계속할 때 먼저 어디까지 계산하였던가를 既設計된 部材名들로 알아볼 수 있게 하는 프로그램이고 17항은 전산을 잠시 중단하거나 완료하였을 때 Virtual Disk 의 내용을 실제디스크에 옮기는 프로그램이고 15, 16항은 구조계산이 완료되어 표지를 작성하고 목차와 함께 全 구조계산서를 프린트하는 命令部分이다. 그림 2는 각부분의 Sub-Menu 의 예이며 그중에서 6항은 동일 Sub-Menu 내의 각 프로그램에 대하여 補足的 記術 또는 계산이 필요할 때 그것들을 기록하는 프로그램이고 7항은 수시로 전산된 결과를 Print-Out 하여 보는 부분이며 그림2의 下部에서 보는 바와 같이 어느 설계항목을 프린트할 것인가, 화일중의 어느 부분부터 프린트할 것인가를 택하도록 되어 있다. 그리고 8항은 오산이 입력된 화일을 말소하고 그 부분의 전산을 再算코져 하는 부분이다.

그림 3,4는 대화식프로그램에 따른 비디오 디스플레이의 예로서 그림 3은 설계 설계자료를 이용할 수 있도록 한 것이고 그림 4에는 설계요건의 순서적 물음과 연산결과치의 선택의 Option 이 배려되어 있으며 설계의 다변화의 일례로 단기설계여하의 물음이 있고 오산화일의 방지와 최적설계를 위한 INPUT OK 의 물음이 있다.

그림 5의 프로그램 예에서 340, 345행에는 위의 INPUT OK 의 프로그램 기술을 볼 수 있고 275행에는 화일의 EOF 문법을

이용하여 error없이 화일을 Open 하는 프로그램이 기술되어 있다. 마지막으로 그림 6은 설계결과와 Table 식 프린트 예로서 여기에는 모든 入出力치와 함께 그에 대한 단위와 설명이 프린트 되어 있다.

結

지금 본 小論을 마치면서 원하는 바는 우리나라에서도 建築電算이 더욱 활발하여져서 斯界의 동인들이 적극적으로 技法교환을 하여 우리나라의 건축기술에 발전이 이루어지기를 기대하며 줄렬하나마 본문에서 소개한 본인의 기법에 대하여 의문되는 점이 있으시면 연락하여 주시기를 바란다. 그리고 여기에서 附言할 것은 기술용어와 우리나라 말의 프로그램 사용에 어려움이 있어 프로그램이 모두 영어로 된 것을 미안하게 생각하며 원래 이 小論은 韓國科學財團에 대한 연구논문 “鐵筋콘크리트造, 鐵骨造 電算 Program의 作成과 그 大學構造教育에 대한 連關에 관한 연구”의 先行部分임을 밝혀둔다.

□註□

1. Micro Computer Application in Structural Engineering, W.H.Mosley, Mac Millan, 1984
2. Computational Method in Structural and Continuum Mechanics C.T.F. Ross Wiley, 1982
3. Computer Aided Structural Design D. Clarke., Wiley, 1980
4. パソコン・プログラム・パーツ集 [RC] [SS]編 滝野文雄, 彰口社, 1984
5. 建築構造計算 プログラム 1,2,3,4 手越義昭, 井上書院, 1985
6. RC, Program(古稀記念) 威性權 1988 [Program式集・筆芥例題와 그 Input/output・Video/output Message集]