

ANSYS®

ANSYS APDL セミナー

ANSYS®は ANSYS,Inc.の登録商標です。

本書の内容の一部あるいは全部を無断で転載、複写、複製することを禁じます。

本書の内容は、予告なく変更することがあります。

本書の内容は、ANSYS Release 10.0 をもとに作成したものです。

“Copyright © 2006 ANSYS, Inc. All right reserved.

ANSYS is a registered trademark of SAS IP, Inc., a subsidiary of ANSYS, Inc.

Translation copy made by permission of ANSYS, Inc., but ANSYS, Inc. has not approved this translated version.

All issues regarding warranties will be governed by the English version of this documentation provided by ANSYS, Inc.”

著作権表示 Copyright © 2006 ANSYS, Inc. All right reserved.

ANSYS は、ANSYS, Inc. の子会社である SAS IP, Inc.の登録商標です。

翻訳版は ANSYS, Inc.の許可を得て作成しておりますが、(内容については) ANSYS, Inc.の承認を得たものではありません。

保証に関する事項は ANSYS, Inc.の提供する当該文書の英語版に準拠するものとします。

ANSYS APDL セミナー

サイバネットシステム株式会社

目次

第1章	概要	1-1
第2章	パラメータ	
2.1	パラメータとは	2-2
2.2	スカラーパラメータ	2-3
	－ 実習問題 －	
2.3	配列パラメータ	2-11
	・アレイ型配列パラメータ	2-17
	－ 実習問題 －	
	・テーブル型配列パラメータ	2-19
	－ 実習問題 －	
第3章	GET構文	
3.1	*GET、GET関数	3-2
3.2	*VGET	3-5
	－ 実習問題 －	
3.3	GET関数一覧	3-11

目次

第4章 演算・関数

4.1 数値演算4-2
4.2 配列演算4-3
4.3 数値関数4-6
4.4 配列関数4-7
－ 実習問題 －	

第5章 繰り返し・条件分岐

5.1 繰り返し 5-2
－ 実習問題 －	
5.2 条件分岐 5-11
－ 実習問題 －	

目次

第6章 マクロ

6.1 マクロ - 概要6-2
6.2 マクロの作成、使用6-3
6.3 引数付きマクロ6-5
6.4 マクロ - グラフィカルピッキング6-6
6.5 マクロ - メッセージボックス6-7
6.6 マクロ作成時の注意点6-11
－ 実習問題 －	

第7章 実習問題解答例

第8章 例題

例題1 座標の平均値8-3
例題2 パネ要素の生成8-7
例題3 分割比の反転8-9
例題4 圧力分布の定義8-11

目次

例題5 接触面のトータル圧力8-13
例題6 エリアのソート8-17
例題7 引っ張りと圧縮でヤング率の異なる材料8-23
<例題解答例>	

付録

1. ANSYS ツールバー と START.ANS
2. ファイルの入出力
3. 実習問題データ作成用コマンド

第1章 概要

APDLとは

- ANSYS Parametric Design Languageの略
- 有限要素解析を実行する際の、ユーザ入力又は必要な操作の手間の省略を目的とする、ANSYSのコマンド“言語”
- 基本的な機能
 - パラメータ（スカラー、配列）
 - 条件分岐（*IF構文）
 - 繰り返し（*DO構文）
 - マクロファイル
 - 各種数値の取得（*GET、GET関数）

APDLとは

- APDLを組み合わせることにより、応用的な解析を必要最小限の作業量で行うことができる
 - パラメトリックモデリング
 - 感度解析
 - 設計基準の評価チェック
 - ユーザ指定フォーマットでの出力
 - アダプティブメッシュ
 - 最適設計
 - ユーザ定義関数
 - …等

-Notes-

第2章 パラメータ

- キーワード -

スカラーパラメータ

アレイ型配列パラメータ

テーブル型配列パラメータ

パラメータとは

- ・ パラメータとは
 - 数値に対して任意の名称を定義し定数とした物
 - スカラーパラメータと配列パラメータの2種類がある
 - 一回のRUNで5,000個までのパラメータを定義可能
- ・ パラメータの名称規約
 - 文字数は32文字まで
 - アルファベット、数字、下線()が使用可能
 - 最初の文字を数字にしてはならない
 - スカラーパラメータと配列パラメータ間で同じ名前は使用不可
 - **ARG1~ARG9、AR10~AR99**の名称は使用不可(マクロ内で使用)
 - 特殊文字は使用不可
 例 ! @ # \$ % & + - = | { } ¥ / “ ’ ◇ 等
 - コマンド名やETABLE等で指定したラベル名、既に定義したコンポーネント名と重複してはならない

注) 最初の文字を下線()にした場合、パラメータのリスト表示の際に出力されない。それ故そのようなパラメータ名は、主にマクロ内でのみ使用する。

※大文字・小文字の区別は無い

スカラーパラメータ

- スカラーパラメータとは
 - 単独の実数を値として持ち、パラメータ名称で値が参照される
- パラメータの作成
 - 書式

NAME = VALUE

NAME : パラメータ名

VALUE : 代入したい数値、他のパラメータ名、パラメータの演算

または

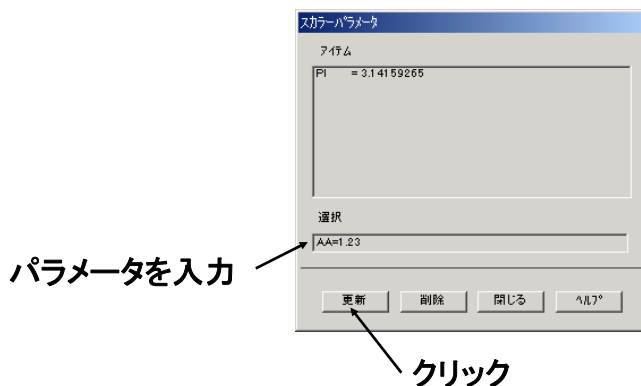
***SET, NAME, VALUE**

※数値が割り当てられていないパラメータ名を入力中に使用すると
ANSYS側で**ゼロに非常に近い数字**を使用する

スカラーパラメータ

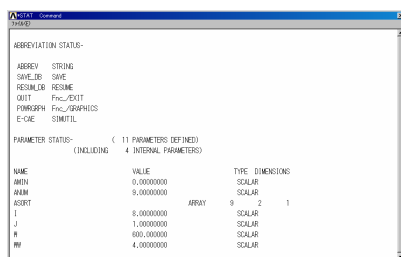
- スカラーパラメータの作成(メニュー)

ANSYS Utility Menu> パラメータ > スカラーパラメータ



パラメータのリスト表示

- 定義したパラメータの種類、パラメータに代入した値を確認
***STAT, パラメータ名**
 パラメータ名を省略すると、定義済みパラメータの一覧が出力
 指定すると、保存されている値が出力
- メニュー
 ANSYS Utility Menu > リスト > その他 > パラメータ
 又は 名前指定で指定



パラメータの保存・読み込み

- データベースをクリアしたりANSYSを再起動した場合、定義した配列も消去される。PARSAVコマンドで、パラメータを保存することができる。
 - パラメータの保存
PARSAV, Lab, Fname, Ext
 Lab : SCALAR ... スカラーパラメータのみ保存
 ALL ... スカラー・アレイ両方保存
 Fname: ファイル名およびディレクトリパス(248文字まで)
 Ext : 拡張子(最大半角8文字)
 - メニュー
 ANSYS Utility Menu > パラメータ > パラメータの保存...

指定を省略した場合、Fnameは“ジョブ名”、
Extは“PARM”となる

パラメータの保存・読み込み

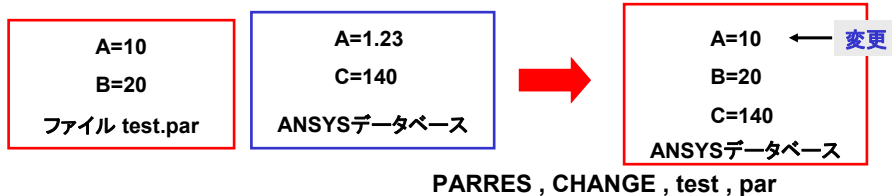
– パラメータの読み込み

PARRES , Lab , Fname , Ext

Lab :NEW – 完全書き換え

CHANGE – 既存のパラメータのみ書き換え

– CHANGE使用例



– メニュー

ANSYS Utility Menu> パラメータ> パラメータの読込...

パラメータの削除

– パラメータの削除(コマンドによる特別な方法)

*** DEL , val1 , val2**

Val1 : ALL – ユーザ定義パラメータを全て削除。または val2の指定に依存し、システムパラメータの全てをも削除。

(空白) – val2の指定による削除。

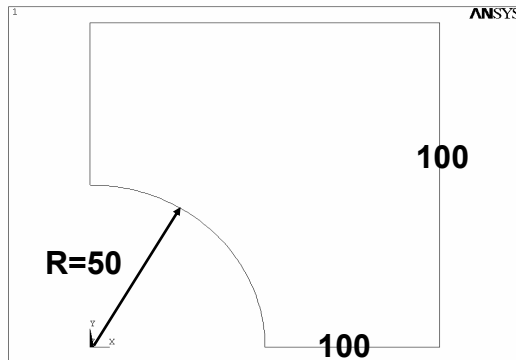
Val2 : _PRM – val1がALLの場合、先頭にアンダーバー () が付いたパラメータを含め全てのパラメータを削除(_STATUS と _RETURN は例外)。

val1が空白の場合、先頭にアンダーバー () が付いたパラメータのみを削除(_STATUS と _RETURN は例外)。

PRM_ – val1が空白の場合、最後にアンダーバー () が付いたパラメータのみを削除(配列パラメータの削除に便利)。

実習:スカラーパラメータ

- 下図のような穴の空いた平板を作成するためのコマンドを、スカラーパラメータを利用して作成する



実習2_1:スカラーパラメータ

- モデルを作成するためのコマンド例は、以下のとおり

/PREP7	!プリプロセッサに入る
RECT,0,100,0,100	!100 × 100の四角形を作成
PCIRC,50	!半径50の円を作成
ASBA,1,2	!四角形から円を減算

- 問1
 - 円を、半径Rをパラメータ化して作成するようなコマンドを作成せよ
<ヒント>PCIRC (円形エリア作成)コマンドを使用
- 問2
 - 円の原点(X_1,Y_1)をパラメータ化し、円を、原点を移動して作成できるようなコマンドを作成せよ
<ヒント>円の作成前にWPOFFS (ワーキングプレーン移動)コマンドを使用

WPOFFS,x方向移動量,y方向移動量,z方向移動量

配列パラメータ

- 配列パラメータとは
 - 複数のスカラーデータからなる集合
 - 数値データには、**アレイ (ARRAY) 型**と**テーブル (TABLE) 型**がある
 - 1,2,3,4又は5次元を持つ(ANSYS8.1より)
 - 配列内の値を参照、値を代入するには、パラメータ名称と配列内の位置を表す**添え字**を指定する
 - 値を特に代入していない箇所には、**ゼロ**が定義される

スカラーパラメータ

A=1

A	1
---	---

アレイパラメータ

B(1,2)=2

B	1	2
	1.5	-1
	40	10

配列パラメータの例

$$AA = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix}$$

1次元配列

(3行1列 配列)

$$BB = \begin{pmatrix} 0 & 2 & -3 \\ 4 & 5 & -6 \\ 1 & 0 & 9 \\ 7 & -8 & 9 \end{pmatrix}$$

2次元配列

(4行3列 配列)

$$CC = \begin{pmatrix} 0 & 9 & 2 \\ 5 & 7 & 1 \\ 4 & 7 & 1 \\ 4 & 1 & 1 \end{pmatrix}$$

3次元配列

(2行2列3面 配列)

配列パラメータの定義

- 配列パラメータの定義

***DIM , Par , Type , IMAX , JMAX , KMAX, Var1, Var2 , Var3**

Par : パラメータの名称
 Type : 配列パラメータのタイプ。ARRAY または TABLE (後述)
 IMAX , JMAX , KMAX : 行、列、面の最大値
 Var1 , Var2 , Var3 : 行、列、面の変数名

※パラメータ名の命名規約はスカラーパラメータと同様

配列パラメータへの値の代入

- 配列パラメータへ値を代入

－ 例

***DIM , test , array , 3 , 2** : アレイ型3行2列パラメータを定義

test(1,1) = 1

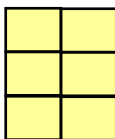
test(2,1) = 5

***set , test(3,1) , -4**

test(1,2)= 3 , -9 , 2

: 1行1列に“1”を代入
 : 2行1列に“5”を代入
 : 3行1列に“-4”を代入
 : 1行2列に“3”、2行2列に“-9”、
 3行2列に“2”を代入

test



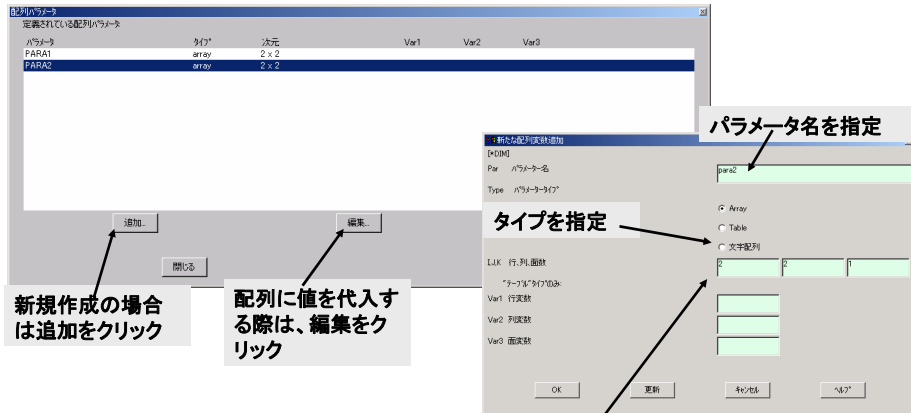
→ test=

1	3
5	-9
-4	2

* DIMコマンドで、配列領域を確保

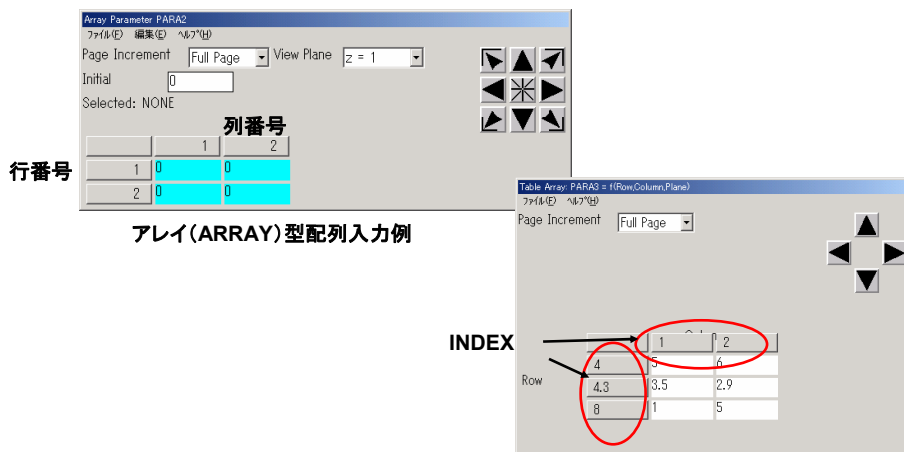
配列パラメータの定義(メニュー)

ANSYS Utility Menu> パラメータ > 配列パラメータ > 定義/編集



行・列・面数を指定

配列パラメータの定義(メニュー)



アレイ (ARRAY) 型配列入力例

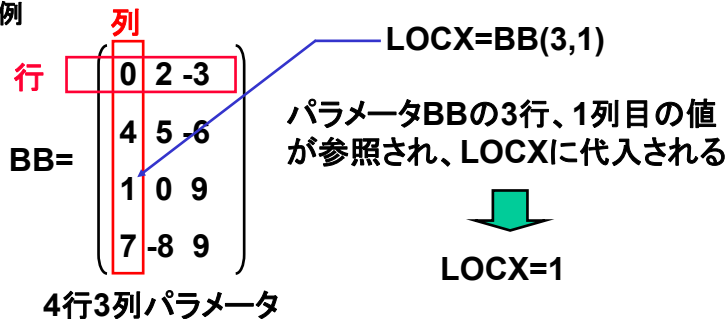
INDEX

テーブル (TABLE) 型配列入力例

アレイ (ARRAY) 型配列パラメータ

- 行/列/面の位置を指定する添え字を用いて値を代入・参照する
- 添え字は全て整数として扱う
- 添え字に実数を使用すると四捨五入される

例



実習2_2: アレイ型配列パラメータ

- 例題データ array1.dbを読み込んだ上で、下記の間を実行せよ
- 問1:
 - 以下のような配列型パラメータを定義するためのコマンドを作成せよ

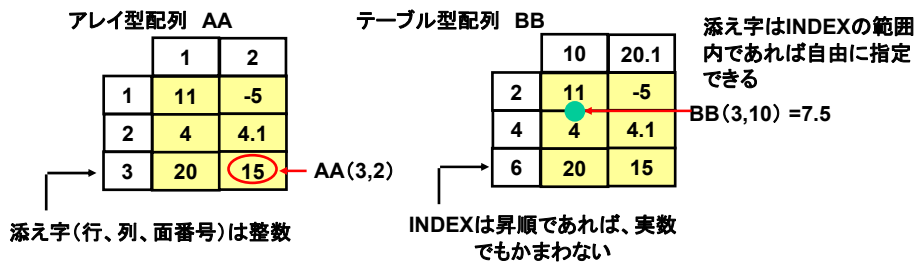
$$fx_a = \begin{pmatrix} 1 & 5000 \\ 46 & 3200 \\ 45 & 1800 \\ 44 & 800 \\ 43 & 200 \\ 37 & 0 \end{pmatrix}$$

- 問2:
 - 上記配列の1列を節点番号、2列を荷重値として、モデル右端の節点のX方向に集中荷重を定義するためのコマンドを作成せよ
 - <ヒント> F コマンド

F, 節点番号, 荷重ラベル, 荷重値

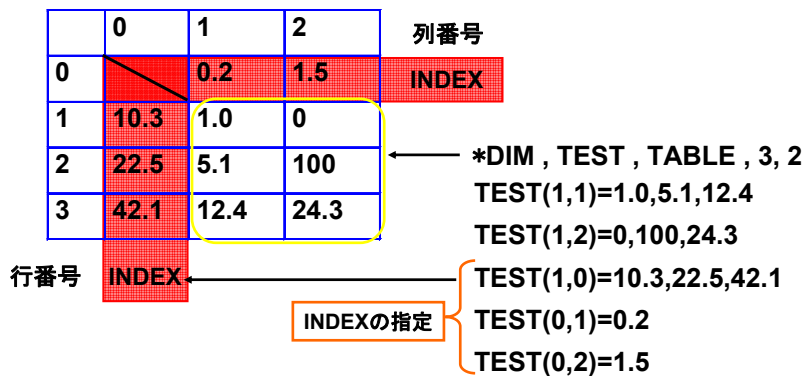
テーブル (TABLE) 型配列パラメータ

- 値の定義方法はアレイ型と同様
- 行番号0と列番号0にINDEXを持つ
- INDEXに依存して添え字が解釈され、配列内で値が補完される
- INDEXは昇順に並んでいなければならない
- INDEXは重複してはいけない

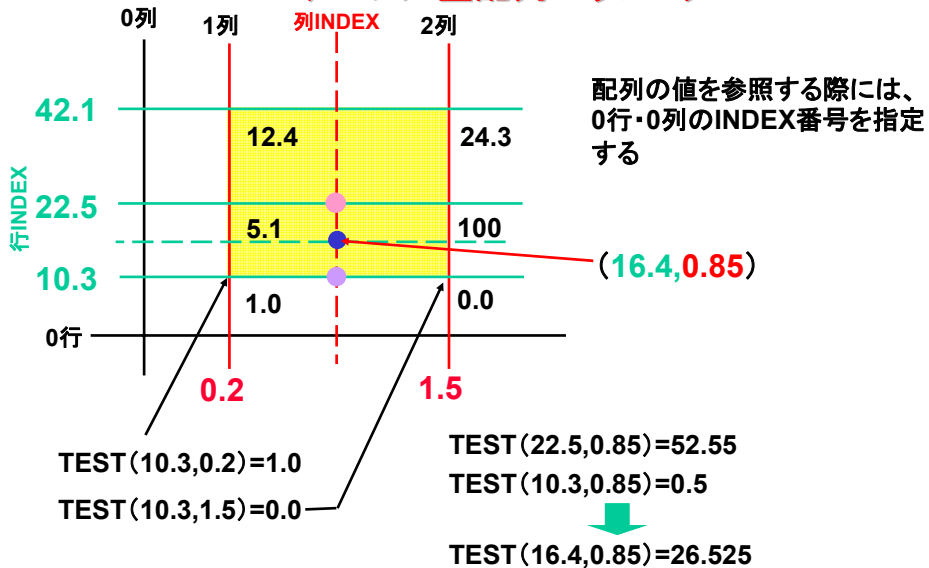


テーブル型配列パラメータ

例: 3行2列のテーブル型パラメータ



テーブル型配列パラメータ



実習2_3: テーブル型配列パラメータ

- 例題データ array1.dbを読み込んだ上で、下記の間を実行せよ
- 問1
 - X=100の位置にある節点のY座標をINDEXとし、荷重値 $FX=0.5Y^2$ となるようなテーブル型配列パラメータを定義するためのコマンドを作成せよ。
(テーブル配列パラメータを用い、節点Y座標に依存する荷重を表現する)

fx_t1 =

	1	INDEX
0	0	
20	200	
40	800	
60	1800	
80	3200	
100	5000	
INDEX	=Y座標	

実習2_3: テーブル型配列パラメータ

- 問2

- 問1で定義したテーブル型配列パラメータを利用して、モデル右端の節点のX方向に集中荷重を定義するためのコマンドを作成せよ。このとき節点のY座標をリストで確認し(NLIST)、目的の節点にY座標に依存した荷重値を定義する。

F,節点番号,FX,fx_t1(Y座標)

- 問3

- いったん要素を削除(ACLEAR,ALL)後、要素サイズを現在の1/2に再定義(ESIZE,,10)、再メッシュ(AMESH,ALL)を実行した後、問1で定義したテーブルパラメータを利用してモデル右端の節点のX方向に集中荷重を定義するためのコマンドを作成せよ

-Notes-

第3章 GET構文

- キーワード -

***GET**

GET関数

***VGET**

*GET、GET関数

- ANSYS内の値を取り出して**パラメータ**に納める機能
- ***GET**構文

***GET, Par, Entity, ENTNUM, Item1, IT1NUM, Item2, IT2NUM**

Par : 値を代入するパラメータ名

Entity以降の引数で参照する値を指定する

－ 例

***GET, kx_loc, KP, 15, LOC, X**

: キーポイント15番のX座標値をkx_locに代入

***GET, n1_temp, NODE, 1, TEMP**

: 節点1番の温度自由度解をn1_tempに代入

***GET, lnext, line, 2, NXTH**

: ライン2番の次に大きいライン番号をlnextに代入

***GET, aaa(1,1), NODE, 20, LOC, X**

: 節点20番のX座標を配列変数aaaに代入

座標値などのパラメータは、アクティブな座標系に依存する

*GET、GET関数

- **GET関数**

- *GETで取り出せるいくつかの項目に対し、より簡単に値をストアできるようにしたもの
- 例

kx_loc = KX(15)

:キーポイント15番のX座標をkx_locに代入

```
n1_temp = TEMP(1)
```

:節点1番の温度自由度解をn1_tempに代入

```
Inext = LSNEXT(2)
```

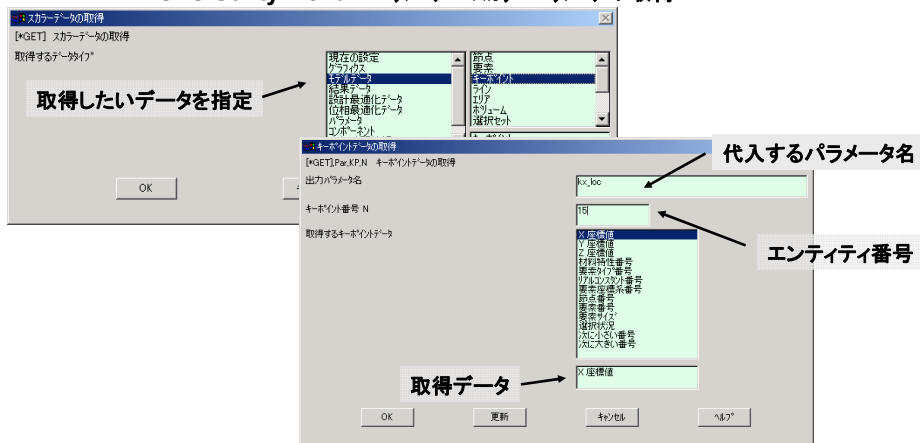
:ライン2番の次に大きいライン番号をlnextに代入

aaa(1,1) = NX(20)

:節点番号20番のX座標を配列変数aaaに代入

*GET(メニュー)

ANSYS Utility Menu> パラメータ > スカラーパラメータの取得



例) キーポイント15番のX座標をパラメータkx locに代入

*VGET

- 配列パラメータに値を一度に代入する(連続したデータの取得)
- コマンド自体がループ機能を持っており、値を配列に順次保存していく
***VGET,Par,Entity,ENTNUM,Item1,IT1NUM,Item2,IT2NUM,KLOOP**

KLOOP: ループすべき入力フィールドを指定するためのキー

0又は2 ENTNUMフィールドをループ(デフォルト)

3 Item1フィールドをループ

4 IT1NUMフィールドをループ

5 Item2フィールドをループ

6 IT2NUMフィールドをループ

*VGET

- *VGET使用例(節点の座標を配列に保存): array1.dbを読み込んで作業

***GET, NCOUNT, node, , COUNT** → 節点数をカウント

***DIM, NLOC, array, NCOUNT, 3** → (節点数)行 * 3列 配列

***VGET, NLOC(1,1), node, 1, LOC, X**

→ 節点番号1~NCOUNTまでのX座標を保存

***VGET, NLOC(1,2), node, 2, LOC, Y**

→ 節点番号2~NCOUNTまでのY座標を保存

***VGET, NLOC(2,3), node, 1, LOC, Z**

→ 節点番号1~(NCOUNT-1)までのZ座標を保存

(ただし、保存開始位置は2行3列目)

*VGET

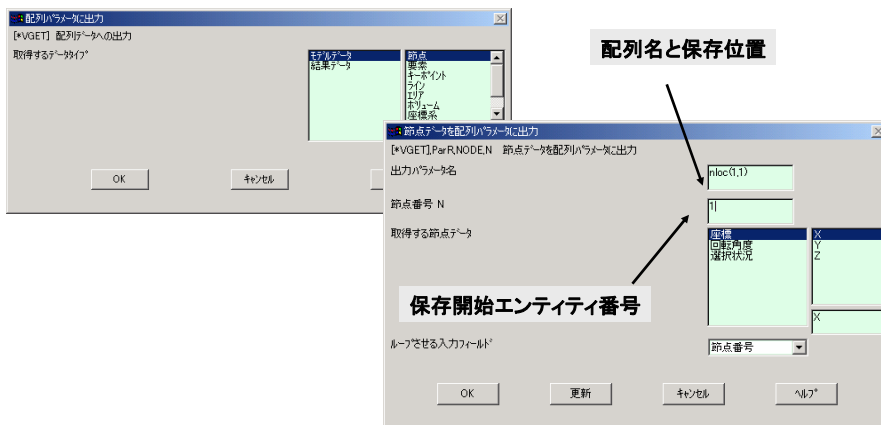
- *VGET使用例(節点の座標を配列に保存)

配列 nloc(行数nnum、列数3)の中身

節点1のX座標	節点2のY座標	0
節点2のX座標	節点3のY座標	節点1のZ座標
...
節点(nnum-1)の X座標	節点nnumの Y座標	...
節点nnumの X座標	0	節点(nnum-1)の Z座標

*VGET(メニュー)

ANSYS Utility Menu> パラメータ> 配列データの取得



※保存する配列はあらかじめ定義しておかなければならない

実習3_1:*GET、*VGET

- 例題データ array1.dbを読み込んだ上で、下記の間を実行せよ
- 問1
 - X=100の位置にある節点のY座標をINDEXとし、荷重値が1列目に入るようなテーブル型配列パラメータを定義するためのコマンドを作成せよ
 <ヒント> *DIMを使用

fx_t2 =

	1	INDEX (0, 1) 列INDEX
0	0	
20	200	
40	800	
60	1800	
80	1000	
100	600	
INDEX =Y座標		
		(1, 0) 行INDEX

6行1列テーブル型配列

実習3_1:*GET、*VGET

- 問2
 - 全節点のY座標の値をアレイ型配列パラメータNOD_Yに保存するためのコマンドを作成せよ
 <ヒント> *VGETを使用
- 問3
 - X=100の位置にある節点のX方向に集中荷重を負荷するためのコマンドを作成せよ。その際、問1・問2の配列を活用すること

GET関数一覧表

3-11

エンティティの状態

NSEL(N)	節点Nの状況 (-1: 非選択 0: 未定義 1: 選択)
ESEL(E)	要素Eの状況 (-1: 非選択 0: 未定義 1: 選択)
KSEL(K)	キーポイントKの状況 (-1: 非選択 0: 未定義 1: 選択)
LSEL(L)	ラインLの状況 (-1: 非選択 0: 未定義 1: 選択)
ASEL(A)	エリアAの状況 (-1: 非選択 0: 未定義 1: 選択)
VSEL(V)	ボリュームVの状況 (-1: 非選択 0: 未定義 1: 選択)

次に大きな番号を持つエンティティ

NDNEXT(N)	選択されている節点群からNよりも大きな節点番号を持つ節点
ELNEXT(E)	選択されている要素群からEよりも大きな要素番号を持つ要素
KPNEXT(K)	選択されているキーポイント群からKよりも大きなキーポイント番号を持つキーポイント
LSNEXT(L)	選択されているライン群からLよりも大きなライン番号を持つライン
ARNEXT(A)	選択されているエリア群からAよりも大きなエリア番号を持つエリア
VLNEXT(V)	選択されているボリューム群からVよりも大きなボリューム番号を持つボリューム

座標値

CENTRX(E)	全体直交座標系における要素Eの要素中心のX座標値
CENTRY(E)	全体直交座標系における要素Eの要素中心のY座標値
CENTRZ(E)	全体直交座標系における要素Eの要素中心のZ座標値
NX(N)	アクティブな座標系における節点NのX座標値
NY(N)	アクティブな座標系における節点NのY座標値
NZ(N)	アクティブな座標系における節点NのZ座標値
KX(K)	アクティブな座標系におけるキーポイントKのX座標値
KY(K)	アクティブな座標系におけるキーポイントKのY座標値
KZ(K)	アクティブな座標系におけるキーポイントKのZ座標値
LX(L, LFRAC)	ラインL上で、端部からLFRAC (0.0~1.0) の位置のX座標値
LY(L, LFRAC)	ラインL上で、端部からLFRAC (0.0~1.0) の位置のY座標値
LZ(L, LFRAC)	ラインL上で、端部からLFRAC (0.0~1.0) の位置のZ座標値

近傍の位置

NODE(X,Y,Z)	アクティブ座標系において座標値X,Y,Zに最も近いアクティブな節点番号
KP(X,Y,Z)	アクティブ座標系において座標値X,Y,Zに最も近いアクティブなキーポイント番号

3-12

距離

DISTND (N1,N2)	節点N1とN2の間の距離
DISTKP (K1,K2)	キーポイントK1とK2の間の距離
DISTEN (E,N)	要素Eの要素中心と節点Nとの間の距離。要素中心の座標値は、その要素内でアクティブな節点から決められる。

角度

ANGLEN (N1,N2,N3)	節点N1を頂点とする3節点間の角度。単位のデフォルトはラジアン
ANGLEK (K1,K2,K3)	キーポイントK1を頂点とする3キーポイント間の角度。単位のデフォルトはラジアン

エンティティに隣接するエンティティ

NNER(N)	節点NIに最も座標値が近い、選択されている節点の番号
KNER(K)	キーポイントKIに最も座標値が近い、選択されているキーポイントの番号
ENEAR(N)	節点NIに最も座標値が近い、選択されている要素の番号。要素位置は、構成節点のうちアクティブなものから算出。

面積

AREAND (N1,N2,N3)	節点N1,N2,N3で構成される三角形の面積
AREAKP (K1,K2,K3)	キーポイントK1,K2,K3で構成される三角形の面積
ARNODE(N)	2D、3Dソリッド要素に対して、節点Nの分担面積

法線方向

NORMNX (N1,N2,N3)	節点N1,N2,N3で構成される平面の法線方向の方向余弦X成分
NORMNY (N1,N2,N3)	節点N1,N2,N3で構成される平面の法線方向の方向余弦Y成分
NORMNZ (N1,N2,N3)	節点N1,N2,N3で構成される平面の法線方向の方向余弦Z成分
NORMKX (K1,K2,K3)	キーポイントK1,K2,K3で構成される平面の法線方向の方向余弦X成分
NORMKY (K1,K2,K3)	キーポイントK1,K2,K3で構成される平面の法線方向の方向余弦Y成分
NORMKZ (K1,K2,K3)	キーポイントK1,K2,K3で構成される平面の法線方向の方向余弦Z成分

コネクティビティ

ENEXTN(N,LOC)	節点NIに接続する要素番号。LOCは、1節点に多数の要素が接続している際の要素リストの順番
NELEM(E,NPOS)	要素Eにおける位置NPOS(1~20)における節点番号

3-13

面情報

ELADJ(E,FACE)	要素Eの要素面(FACE)を共有する要素の番号。該当する要素が複数ある場合は-1が返ってくる。
NDFACE(E,FACE,LOC)	要素Eの要素面(FACE)においてLOCに位置する節点の番号
NMFACE(E)	選択されている節点を含む要素面番号。同一要素面に対して複数の面番号がある場合は、最小の物が返ってくる
ARFACE(E)	2D、3Dソリッド要素に関して、選択されている節点を含む要素面面積。軸対称の場合は、全周の値

自由度解

UX(N)	節点NIにおける並進変位UX
UY(N)	節点NIにおける並進変位UY
UZ(N)	節点NIにおける並進変位UZ
ROTX(N)	節点NIにおける回転変位ROTX
ROTY(N)	節点NIにおける回転変位ROTY
ROTZ(N)	節点NIにおける回転変位ROTZ
TEMP(N)	節点NIにおける温度解
PRES(N)	節点NIにおける圧力
VX(N)	節点NIにおける流速VX
VY(N)	節点NIにおける流速VY
VZ(N)	節点NIにおける流速VZ
ENKE(N)	節点NIにおける乱流エネルギーENKE
ENDS(N)	節点NIにおける乱流エネルギー散逸率ENDS
VOLT(N)	節点NIにおける電気ポテンシャル
MAG(N)	節点NIにおける磁気スカラーポテンシャル
AX(N)	節点NIにおける磁気ベクトルポテンシャル成分AX
AY(N)	節点NIにおける磁気ベクトルポテンシャル成分AY
AZ(N)	節点NIにおける磁気ベクトルポテンシャル成分AZ

その他についてはANSYS Utility Menu> ヘルプ > ヘルプボックス

ANSYSリリース10.0ドキュメント > APDL Programmer's Guide > GET Function Summary

を参照のこと

3-14

-Notes-

-Notes-

第4章 演算・関数

- キーワード -

数値演算

数学関数

*VOPER、*VFUN

数値演算

- パラメータの一般的な演算

演算子の優先順位	シンボル
括弧内の演算	()
指数 (右から左へ)	**
乗算、除算	* または /
正負の符号	+ または -
加算、減算	+ または -
論理演算子	< または >

- 一般的な演算例

A=1 B=3 C=A+B D=((A+B)**C) A=A+1

XYZ=(A<B)+Y

もし A<Bならば XYZ=A+Y

それ以外ならば XYZ=B+Y

配列演算

- 配列パラメータのベクトル演算

***VOPER , ParR , Par1 , Oper , Par2 , CON1 , CON2**

ParR : 結果のパラメータ名 ParR=(Par1 演算子 Par2)

Oper : 演算子

演算子のラベル	記述の内容
ADD , SUB	加算 / 減算
MULT , DIV	乗算 / 除算
MIN , MAX	最小値 / 最大値
LE , LT , EQ , NE , GT , GE	$\leq, <, =, \neq, >, \geq$ 真の場合 1、偽の場合 0 を返値
※この他にも、単積分 (INT1) や一次導関数 (DER1) 等がある。 他の演算子ラベルに関しては、コマンドマニュアルを参照	

※あらかじめ結果を保存する配列パラメータ(ParR)を作成しておく必要がある

配列演算

- *VOPER使用例

***VOPER , C(1) , A(1) , ADD , B(1)**

$$A = \begin{pmatrix} 1 \\ 3 \\ 5 \end{pmatrix}$$

$$B = \begin{pmatrix} 4 \\ 6 \\ 3 \end{pmatrix}$$

$$C = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

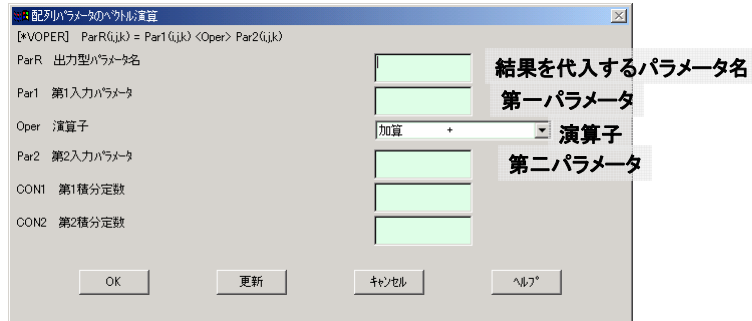
結果を入れる空の配列

$$C = \begin{pmatrix} 5 \\ 9 \\ 8 \end{pmatrix}$$

*VOPER(メニュー)

ANSYS Utility Menu> パラメータ

> 配列パラメータ演算 > ベクトル演算...



※各配列はあらかじめ定義しておく必要がある

数値関数

- 数学関数 ※デフォルトの角度の単位はラジアンである

関数名	記述内容
三角関数 SIN,COS,TAN ASIN,ACOS,ATAN ATAN2(x,y)	角度 角度の逆関数
指数関数 LOG , EXP , LOG10	自然対数、指数、常用対数
双曲線関数 SINH,COSH,TANH	
乱数 RAND(min,max) GDIS(mean,std_dev) ABS , NINT , SQRT	乱数 X: $\min \leq X \leq \max$ ガウス分布からの任意抽出数 絶対値、最も近い整数値、平方根

- 使用例

PI = ACOS(-1) : 円周率を代入

AA = 10 BB = 3 CC = AA/BB { = 3.33333 } の時

DD = NINT(CC) { = 3 }

配列関数

- 配列パラメータに対する関数

***VFUN , ParR , Func , Par1 , CON1 , CON2 , CON3**

ParR : 結果のパラメータ名 ParR=Function(Par1)

Func : 使用する関数のラベル

関数名	記述内容
三角関数 SIN,COS,TAN ASIN,ACOS,ATAN	角度 角度の逆関数
指数関数 LOG , EXP , LOG10	自然対数、指数、常用対数
双曲線関数 SINH,COSH,TANH	
その他のベクトル関数 NINT SQRT COPY NOT PWR	最も近い整数値 平方根 直接コピー 論理否定 各入力値の CON1 累乗

配列関数

- 配列関数の使用例

***VFUN , B(1), PWR , A(1) , 3**

$$A = \begin{pmatrix} 1 \\ 3 \\ 5 \end{pmatrix} \quad B = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

$$B = \begin{pmatrix} 1 \\ 27 \\ 125 \end{pmatrix}$$

※配列Aの各成分の3乗が代入される

*VFUN(メニュー)

ANSYS Utility Menu> パラメータ

> 配列パラメータ演算 > ベクトル関数...



※各配列はあらかじめ定義しておく必要がある

角度の単位の変換

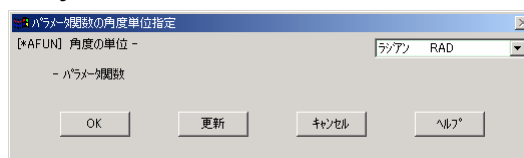
- 関数などで角度を入力する際、デフォルトの単位はラジアンである。度で入力する場合は、*AFUNコマンドを使用する。

***AFUN, Lab**

Lab	RAD	ラジアン
	DEG	度
	STAT	現状の角度単位を確認

- メニュー

ANSYS Utility Menu> パラメータ > 角度単位指定...



実習4_1: 演算・関数

array1.dbを読み込んだ上で下記の間を実行せよ

- 問1
 - 半径50の位置にある節点番号を調べ、**節点番号**と全体座標系のX軸との**角度 θ** をアレイ型配列n_thに保存するためのコマンドを作成せよ。

CSYSコマンド: 全体直交座標系(X,Y,Z) 全体円筒座標系(R, θ , Z)

該当する節点の数をカウントし(*GET)、それを配列の行数とする

- 問2
 - n_thの値を利用し、そのCOS、SINを算出しそれぞれ配列n_cos、n_sinに保存するためのコマンドを作成せよ。

角度の単位に注意

- 問3
 - 上記配列に、荷重値F=10を掛けたものを配列f_nに保存し、対象となる節点に負荷するためのコマンドを作成せよ。

配列の中身を整数倍するには、どのコマンド? (*VFUN? *VOPER?)

-Notes-

第5章 繰り返し・条件分岐

- キーワード -

***DO~*ENDDO**

***IF~*ENDIF**

繰り返し

- コマンドの繰り返し

***REPEAT , NTOT , VINC1 , VINC2 , ... , VINC11**

NTOT : 元となるコマンドを含めた繰り返し回数

VINC~: 元のコマンドの~フィールドへの増分値

- 直前に実行したコマンドを、一定の増分値を与えながら繰り返す

- 例

N , 1 , 1 : (1,0,0) に節点1を作成

*REPEAT , 5 , 2 , 3 : 直前のコマンドを含めて5回繰り返す

Nコマンドの第1フィールド増分が2

第2フィールド増分が3



節点番号1,3,5,7,9 X座標値1,4,7,10,13 計5個の節点を作成される

繰り返し

• DOループ

***DO , Par , IVAL , FVAL , INC**

実行するコマンド群

***ENDDO**

Par :カウンタパラメータ名

IVAL :カウンタの初期値

FVAL :カウンタの終了値

INC :カウンタの増分値

- * DOと*ENDDOで挟まれたコマンド群を一定の回数だけ繰り返す
- ループのネストは最大20レベルまで

繰り返し

• DOループ使用例

/prep7

***do,i,1,10,3**

k,i,0,i*10,0

***enddo**

キーポイント1,4,7,10番を、それぞれY=10,40,70,100の位置に作成する

参考) 暗黙DOループを用いる場合:

K,(1:10:3),0,(10:100:30) は可

K,(1:10:3),0,(1:10:3) * 10 は不可

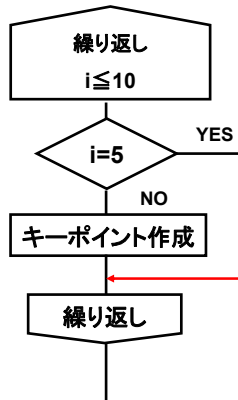
繰り返しの中断

- DOループを中断するコマンドとして、***CYCLE**コマンドと***EXIT**コマンドがある。
 - ***CYCLE**コマンドは、DOループ内の現在のループを途中で終了し、**次のループを開始**する
 - ***EXIT**コマンドは、現在の**DOループ自体を終了**し、***ENDDO**コマンドの次のコマンドから作業を再開する。
- 両コマンドとも、現在のレベルのDOループにのみ適用される。

繰り返しの中断(*CYCLE)

- 例 ***CYCLE**の場合

```
/prep7
*do,i,1,10
  *if,i,eq,5,then
    *CYCLE
  *else
    k,i,i
  *endif
*enddo
```

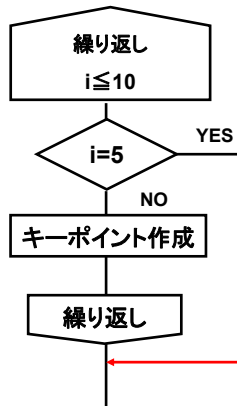


5番を除いた9つのキーポイントが作成される

繰り返しの中断(*EXIT)

－ 例 *EXITの場合

```
/prep7
*do,i,1,10
  *if,i,eq,5,then
    *EXIT
  *else
    k,i,i
  *endif
*enddo
```



4つのキーポイントが作成される

実習5_1: 繰り返し

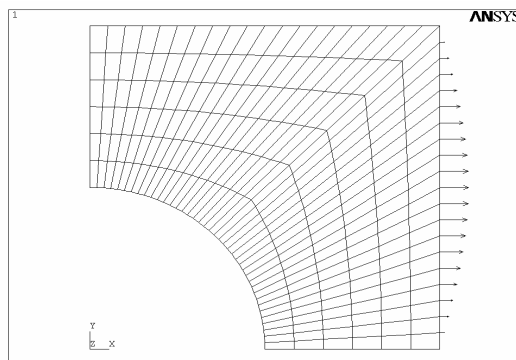
例題データloop1.dbを読み込んだ上で、以下の問に答えよ

右図のように、モデル端部に集中荷重を定義するためのコマンドを作成せよ。

ただし、荷重値は

$$FX = \sin(\pi / 100 * Y \text{座標値})$$

とする。



実習: 繰り返し

* GETで取得

ncount = 選択されている節点の個数
nmin = 選択されている節点の内、最小
の節点番号

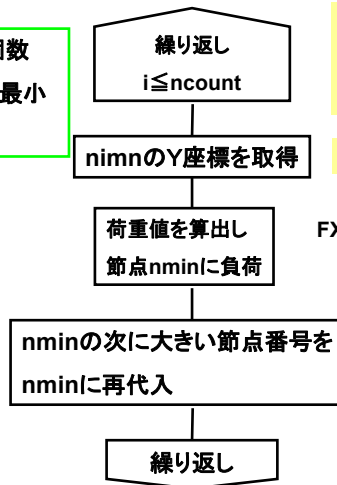
X=100の節点を選択し、その
個数をループ数とすることで、
無駄なループを避けることが
できる

* GETまたはGET関数で取得

$$FX = \sin(\pi / 100 * Y \text{座標値})$$

正確な π の値を求めるには？

* GETまたはGET関数で取得



-Notes-

条件分岐

- IF - THEN - ELSE 構造

***IF** , VAL1 , Oper , VAL2 , BASE

実行コマンド群

***ELSEIF** , VAL1 , Oper , VAL2

実行コマンド群

***ELSE**

実行コマンド群

***ENDIF**

- 論理演算を行い、その真偽によって以降のコマンド作業を決定する
- 最大20レベルまでのネストを構成することができる
- 比較演算子(AND, OR, XOR)が *IF 、*ELSEIF で使用可能
 - 使用例 *IF,A,EQ,B,AND,C,GT,D,THEN
A=B かつ C>D の場合、真となる

条件分岐

- 論理演算子 Oper

LT	<	EQ	=
GT	>	NE	≠
GE	≥	LE	≤
ABLT	< (評価には絶対値を使用)	ABGT	> (評価には絶対値を使用)

その他の分岐

- *IFコマンドを通常の書式で使用方法

***IF , VAL1 , Oper1 , VAL2 , Base1 , VAL3 , Oper2 , VAL4 , Base2**

Base(1,2)

THEN : 通常のIF - THEN - ELSEを使用

CYCLE : DOループ内のみで使用。*CYCLEに相当

EXIT : DOループ内のみで使用。*EXITに相当

STOP : 作業を終了し、ANSYSから抜ける

:LABEL : ユーザ定義のLABEL名を指定する場合

Base1

AND, OR, XOR : Oper1とOper2の2つの論理式の結合を決定

例)

***IF , XX1 , LT , YY1 , AND , XX2 , GT , YY2 , :NEXT**

コマンド群 A XX1<YY1が真かつXX2>YY2が真

:NEXT →コマンド群Aをとばしてコマンド群Bを実行

コマンド群 B

XX1<YY1が真かつXX2>YY2が偽

→コマンド群AとBを実行

その他の分岐

- *GOコマンドによる無条件分岐を使用する場合

***GO , Base**

“ジャンプ”動作:

:label/ — ユーザ定義のラベル(コロン(:)で始まり、最大 8文字)。コマンド読み込みは、ラベル :label で始まる最初の行へ(必要であればファイルの先頭へ戻ってから)スキップする。このラベルオプションは、do ループや if-then-else 構文と組み合わせることはできない。注意を喚起するためにラベルオプションを使用するとワーニングメッセージが常に出力される。

STOP — この行で ANSYS プログラムを終了させる。

例)

***GO, :SKP**

:SKP

■ 実習5_2:条件分岐

- 例題ファイルloop.dbを読み込み 解析を行った上で 下記の問に答えよ
- 下記のような配列を作成するためのコマンドを作成せよ。
 - 節点の相当応力を3つに分類する
 - 応力の値に応じて、3つの配列に節点番号と応力値をストアする
 - 各配列の1行目から詰めて保存していく

* GETで取得

SEQV1

(応力値0以上0.3未満)

節点番号	応力値
節点番号	応力値
節点番号	応力値
節点番号	応力値

SEQV2

(応力値0.3以上0.6未満)

節点番号	応力値
節点番号	応力値
節点番号	応力値
節点番号	応力値

SEQV3

(応力値0.6以上)

節点番号	応力値
節点番号	応力値
節点番号	応力値
節点番号	応力値

実習5_2:条件分岐

ncount = 節点の数

seqv1 = ncount行*2列array配列

seqv2 = ncount行*2列array配列

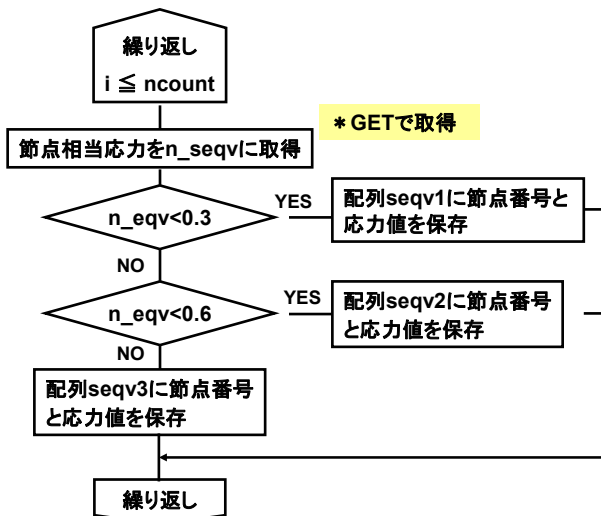
seqv3 = ncount行*2列array配列

counter1 = 添え字用パラメータ

counter2 = 添え字用パラメータ

counter3 = 添え字用パラメータ

n_seqv = 相当応力保存用パラメータ



第6章 マクロ

- キーワード -

***CREATE ~ *END**

引数

メッセージボックス

マクロ - 概要

- マクロとは？
 - ある処理を行うための、APDLを含むANSYSコマンドの一連の流れを
サブルーチン化したもの
 - 通常のANSYSコマンドと同じ様に使用することができる
- マクロで利用可能な機能
 - 引数付きマクロ
 - マウスによるグラフィカルピッキング
 - メッセージボックスの出力
 - ネスト(マクロの中からマクロを呼び出すー最大20まで)

マクロの作成、使用

- ANSYS起動中に作成して、データベースに保存

```
*CREATE, マクロ名
```

```
...
```

```
*END
```

- 作業中に作成して、ファイルとして保存

```
*CREATE, マクロ名, MAC
```

```
...
```

```
*END
```

- あらかじめファイルとして作成しておく

```
テキストエディタにANSYSコマンドを記述し  
「マクロ名.MAC」の書式で保存
```

マクロの作成、使用

- マクロの使用方法
 - *USEコマンドを使用

```
*USE, マクロ名
```
 - マクロ名をそのまま入力(ファイルとして保存している場合)
- マクロファイルはどこに置いておけばよい？
 - 1. v**/ANSYS/apdl (**はリビジョン番号)
 - 2. 環境変数ANSYS_MACROLIBで指定したディレクトリ
 - 3. ログイン時のホームディレクトリ
 - 4. カレントディレクトリ(ワーキングディレクトリ)

引数付きマクロ

- パラメータ名ARG*(* は1～9),AR**(** は10～19)は、マクロの引数として使用可能
 - マクロ名,ARG1,ARG2,...,AR10,AR11,...,AR19
- パラメータ名ARG*(* は1～9),AR**(** は10～99)は、マクロ内のみで有効なパラメータとして使用可能(局所的パラメータ)

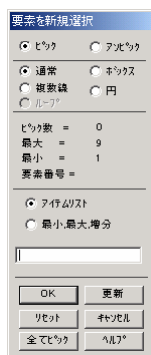
例: プリプロセッサにて...

*CREATE,TEST	!マクロの作成開始(マクロ名:TEST)
WPAV, ARG1, ARG2	!WPの原点を(ARG1,ARG2)に移動
PCIRC, ARG3	!半径ARG3の円を作成
*END	!マクロの終了
*USE, TEST, 0.2, 0.5, 1	!(0.2,0.5)の位置に半径1の円を作成

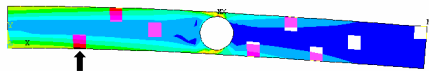
※マクロを*useコマンドを用いて使用する場合、使用できる引数はARG1～AR18までとなる。

マクロ - グラフィカルピッキング

- マクロ内のコマンドで、引数でエンティティ番号を入れる場合、番号の代わりに「P」を使用(P以降の引数は無視される)



ESEL,s,要素番号 → ESEL,s,P



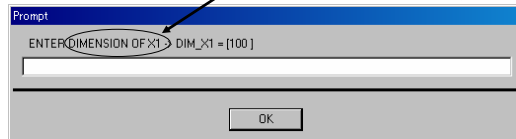
マクロ - メッセージボックス

- 入力にメッセージボックスを使用
 - 単一パラメータへの入力 → ***ASKコマンド**

使用例

***ASK, DIM_X1, DIMENSION OF X1, 100**

※このメッセージボックスに日本語は使用できません



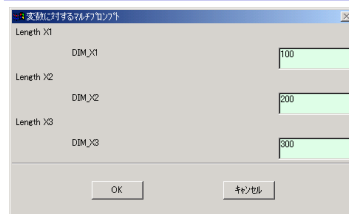
スカラーパラメータ“DIM_X1”を入力するためのダイアログボックスを開く。デフォルトは100。

マクロ - メッセージボックス

- 入力にメッセージボックスを使用（続き）
 - 複数パラメータへの入力 → **MULTIPRO, *CSETコマンド**

使用例

MULTIPRO,'START',3
***CSET,1,3,DIM_X1,'Length X1',100**
***CSET,4,6,DIM_X2,'Length X2',200**
***CSET,7,9,DIM_X3,'Length X3',300**
MULTIPRO,'END'



スカラーパラメータ“DIM_X1”, “DIM_X2”, “DIM_X3”を入力するためのダイアログボックスを開く。デフォルトはそれぞれ100, 200, 300。

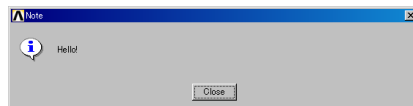
マクロ - メッセージボックス

- 出力にメッセージボックスを使用

- *MSGコマンド

- 使用例

```
*MSG, UI, 'Hello!'
%c
```



出力形式UIで、出力フォーマットを%c
にてメッセージを出力

マクロ - メッセージボックス

- 出力にメッセージボックスを使用(続き)

- 出力形式

```
INFO    : アウトプットウィンドウに出力
NOTE    : アウトプットウィンドウに出力
WARN    : Warningウィンドウに出力
ERROR   : Errorウィンドウに出力
FATAL   : Errorウィンドウに出力 → ANSYS強制終了
UI       : Noteウィンドウに出力
```

- 出力フォーマット

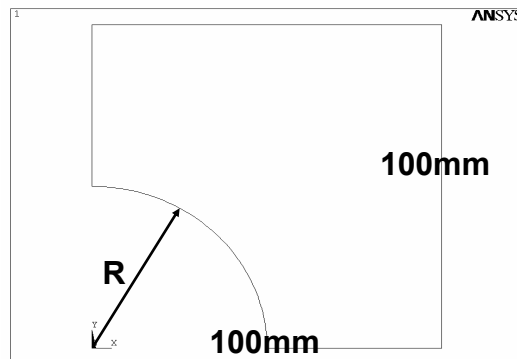
```
%i      : 整数型
%g      : 実数型
%c      : 文字型
%/      : 改行
```

マクロ作成時の注意点

- マクロファイル作成時に考慮しておくこと
 - そのマクロは、単一のモデルに使用するか？或いは類似する他のモデルにも使用するのか？
 - そのマクロは他のユーザーも使用するのか？
- マクロ内に記述しておくべき情報
 - マクロの作成者、日付、対応しているANSYSのバージョン
 - 使用している局所パラメータの説明、制限事項
 - 注釈
- マクロ内に含めるべきでない項目
 - *IF、*GOによるラベルへ(:LABEL)の分岐
 - システム依存のコマンド(/SYS、/SYP)
 - マクロの開始と最後でプロセッサが異なるような操作

実習6_1: マクロ

- 問1
 - 第1章のモデルを作成するためのマクロ「BASEMDL.MAC」を、穴の半径Rを引数として入力できるような形で作成し、実行せよ。



実習6_1: マクロ

- 問2
 - 「BASEMDL.MAC」で、穴の半径Rを引数ではなくメッセージボックスから入力するようなマクロ「BASEMDL2.MAC」を作成し、実行せよ。尚、Rのデフォルト値は50とする。
 - * ASKコマンドを使用
- 問3
 - 「BASEMDL2.MAC」で、穴の位置をマウスピッキングで決めてから穴の半径をメッセージボックスで入力するようなマクロ「BASEMDL3.MAC」を作成し、実行せよ。(マクロファイルの先頭に/pmacroコマンドを書き込んでおく)
 - WPAVEコマンドで中心を指定することが可能。ピックで指定するには？
- 問4
 - 「BASEMDL2.MAC」で、作成不可能な数値を半径に指定した場合にエラーウィンドウを表示して終了するようなマクロ「BASEMDL4.MAC」を作成し、実行せよ。

-Notes-

-Notes-

-Notes-

第7章 実習問題解答例

7-1

実習2_1: 解答例 (スカラーパラメータ)

- 問1
 - ANSYSで円形エリアを作成するコマンドはPCIRCコマンドである。
PCIRC , RAD1 , RAD2 , THETA1 , THETA2
RAD1,RAD2 : 内径、外径
THETA1、THETA2: 始点、終点角度
 - 今回はRAD1またはRAD2をパラメータ化する

解答例

```
/prep7
rect,0,100,0,100
r=50          !スカラーパラメータの定義
pcirc,r       !半径rの円の作成
asba,1,2
```

実習2_1: 解答例(スカラーパラメータ)

- 問2
 - PCIRCコマンドは、ワーキングプレーン座標系の原点を中心とした円しか描画できない
 - 原点をずらすには、WPAVEコマンドを使用する
 WPAVE , X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3
 X1~Z3 : アクティブな座標系における座標値
 X2~Z3の値を入力すると平均した座標値に移動する

解答例

```

/prep7
rect,0,100,0,100
r=50
x_i=20          !X座標値用スカラーパラメータの定義
y_1=35          !Y座標値用スカラーパラメータの定義
wpoffs,x_1,y_1  !WP座標系の原点を移動
pcirc,r
asba,1,2

```

実習2_2: 解答例(アレイ型配列パラメータ)

- 問1 **解答例**

```

*dim,fx_a,array,6,2
fx_a(1,1)=1,46,45,44,43,37
fx_a(1,2)=5000,3200,1800,800,200,0

```
- 問2
 - 節点に対して集中荷重を与えるにはFコマンドを使用する
 F , NODE , Lab , VALUE , VALUE2 , NEND , NINC
 NODE : 節点番号
 Lab : 荷重ラベル(例: FX、FY、FZ、ROTX、ROTY、ROTZ等)
 VALUE: 荷重値
 - 今回は、NODEとVALUEの箇所をパラメータ化する

解答例

```

/solu
f,fx_a(1,1),fx,fx_a(1,2) !節点番号1に5000のFX荷重が負荷される
f,fx_a(2,1),fx,fx_a(2,2) !節点番号46に3200のFX荷重が負荷される
⋮

```

実習2_3: 解答例(テーブル型配列パラメータ)

• 問1 解答例

```
*dim,fx_t1,table,6,1
fx_t1(1,0)=0,20,40,60,80,100
fx_t1(1,1)=0,200,800,1800,3200,5000
```

• 問2

- アレイ型配列パラメータ実習の際の、Fコマンドを使う
- 今回は、VALUEの箇所をパラメータ化する
- 節点の座標は、NLISTで確認する

解答例

```
/solu
f,1,fx,fx_t1(100) !節点番号1に5000のFX荷重が負荷される
f,46,fx,fx_t1(80) !節点番号46に3200のFX荷重が負荷される
:
```

実習2_3: 解答例(テーブル型配列パラメータ)

• 問3

- いったんパラメータを作成しておけば、INDEXの値を変えるだけで線形補完された荷重値が算出される。

解答例

```
/prep7
aclear,all
esize,,10
amesh,all
/solu
f,1,fx,fx_t1(100) !節点番号1に5000のFX荷重が負荷される
f,141,fx,fx_t1(90) !節点番号141に4100のFX荷重が負荷される
:
```

実習3_1:解答例(*GET,*VGET)

問1 解答例

```
*dim,fx_t2,table,6,1      !テーブル型配列パラメータ宣言
fx_t2(1,0)=0,20,40,60,80,100  ! INDEX
fx_t2(1,1)=0,200,800,1800,1000,600  !荷重値
```

問2

- 無駄なサイズの配列を作成することを防ぐために、総節点数を*GETで取得する
- 配列作成後、*VGETで各節点の座標値を取り込む

解答例

```
*GET , ncount , node , , count      !選択されている節点数を取得
*DIM , nod_y , array , ncount      !アレイ型配列パラメータ宣言
*VGET , nod_y(1) , node , 1 , loc , y  !配列変数にY座標値をストア
```

実習3_1:解答例(*GET,*VGET)

問3

- 荷重値入力に必要な節点Y座標は問2で作成した配列から求める
- 配列nod_yの添え字は、節点番号に対応している

解答例

```
/solu
f,1,fx,fx_t2(nod_y(1))      !節点番号1に600のFX荷重が負荷される
f,46,fx,fx_t2(nod_y(46))    !節点番号46に1000のFX荷重が負荷される
⋮
```

※参考

GET関数を使用すると更にスマートになる

```
f , 1 , fx_t2 (ny(1))
ny(1) ← 節点番号1番のY座標値
```

問2の操作を省略することができる

実習4_1:解答例(演算・関数)

• 問1

- まず、全体円筒座標系に変換し $R=50$ ($X=50$)の位置にある節点の番号を調べ、配列に保存する
- *GETまたはGET関数を使用し、そのY座標を配列に保存する

解答例

```

csys , 1
nselect , s , loc , x , 50
*get , ncount , node , , count
*dim , n_th , array , ncount , 2
n_th(1,1) = 38 , 47 , 48 , 49 , 50      !節点番号保存
n_th(6,1) = 12 , 13 , 14 , 15 , 16 , 7
n_th(1,2) = ny(38),ny(47),ny(48),ny(49),ny(50)      !Y座標
n_th(6,2) = ny(12),ny(13),ny(14),ny(15),ny(16),ny(7)

```

実習4_1:解答例(演算・関数)

• 問2

- 関数で使用する角度はデフォルトではラジアンである
- それに対して問1で求めた角度は、度である
- 角度の単位系を変更する必要がある

解答例

```

*afun , deg
*dim , n_cos , array , ncount
*dim , n_sin , array , ncount
*vfun , n_cos(1) , cos , n_th(1,2)
*vfun , n_sin(1) , sin , n_th(1,2)

```

■ 実習4_1:解答例(演算・関数)

• 問3

- *VOPERコマンドを使用し、n_cos、n_sinの10倍を計算し、配列f_nに保存する

解答例

```
*dim , f_n , array , ncount , 2
*voper , f_n(1,1) , n_cos(1) , mult , 10
*voper , f_n(1,2) , n_sin(1) , mult , 10
/solu
f , n_th(1,1) , fx , f_n(1,1)
f , n_th(1,1) , fy , f_n(1,2)
f , n_th(2,1) , fx , f_n(2,1)
f , n_th(2,1) , fy , f_n(2,2)
⋮
```

※本例題は、半径方向の集中荷重を定義する例である。

※節点座標系を回転させることができない場合に有効である。

■ 実習5_1:解答例(繰り返し)

- 節点個数・最小節点番号・次に大きな節点番号は、*GETもしくはGET関数で取得可能である
- 正確な π を求めるため、関数を使用する

解答例

```
nsel , s , loc , x , 100
*get , ncount , node , , count
*get , nmin , node , , num , min
pi = acos(-1)
/solu
*do , i , 1 , ncount
  f , nmin , fx , sin (ny(nmin)*pi/100)
  nmin=ndnext(nmin)
*enddo
allsel
```

■ 実習5_2: 解答例(条件分岐)

解答例

```
/solu
```

```
sfl,3,pres,1
```

```
solve
```

```
*get,ncount,node,,count
```

```
*dim,seqv1,array,ncount,2
```

```
*dim,seqv2,array,ncount,2
```

```
*dim,seqv3,array,ncount,2
```

```
counter1=0
```

```
counter2=0
```

```
counter3=0
```

```
*do,i,1,ncount
```

```
  *get,n_seqv,node,i,s,eqv
```

```
  *if,n_seqv,LE,0.3,then
```

```
    counter1=counter1+1
```

```
    seqv1(counter1,1)=i
```

```
    seqv1(counter1,2)=n_seqv
```

```
  *elseif,n_seqv,LE,0.6
```

```
    counter2=counter2+1
```

```
    seqv2(counter2,1)=i
```

```
    seqv2(counter2,2)=n_seqv
```

```
  *else
```

```
    counter3=counter3+1
```

```
    seqv3(counter3,1)=i
```

```
    seqv3(counter3,2)=n_seqv
```

```
  *endif
```

```
*enddo
```

実習6_1: 解答例(マクロ)

問1

解答例

```
/prep7
```

```
RECT,100,,100
```

```
PCIRC,ARG1
```

```
ASBA,1,2
```

←100*100の四辺形エリア

←引数ARG1を半径とした円形エリア

←ブーリアン演算

実習6_1:解答例(マクロ)

• 問2

解答例

```
/prep7
RECT,100,,100
*ASK,RAD,Radius,50 !半径を入力するためのパネルを出力
PCIRC,RAD
ASBA,1,2
```

• 問3

解答例

```
/pmacro
/prep7
RECT,100,,100
WPAV,P !ワーキングプレーンの原点をマウスピックで指定
*ASK,RAD,Radius,50
PCIRC,RAD
ASBA,1,2
```

■ 実習6_1:解答例(マクロ)

• 問4

解答例

```
/prep7
RECT,100,,100
*ASK,RAD,Radius,50 !半径入力パネル出力
*IF,RAD,LE,0,THEN !無効な半径が入力されたかどうかを判定
*MSG,ERROR,'Small !'
%C
*ELSEIF,RAD,GE,100*SQRT(2)
*MSG,ERROR,'Large !'
%C
*ELSE
PCIRC,RAD
ASBA,1,2
*ENDIF
```

第8章 例題集

例題-目次

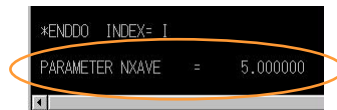
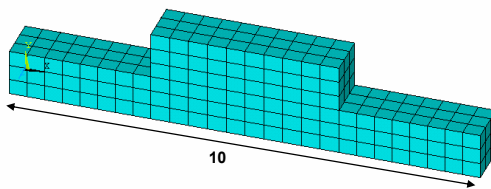
- 例題1 座標の平均値
- 例題2 バネ要素の作成
- 例題3 分割比の反転
- 例題4 圧力分布の定義
- 例題5 接触面のトータル力
- 例題6 エリアのソート
- 例題7 引っ張りと圧縮でヤング率の異なる材料

※解答例は章の最後

例題1 座標の平均値

手順

1. chapter8_1.dbを呼び出す (RESUME)。このファイルにはモデル作成までの入力データがストアされている。
2. (必要であれば) 幾つかの節点を選択 (NSELコマンド)
3. エディタを用いて、節点のX座標値の平均値をパラメータにストアするためのANSYSコマンドを記述。
4. 完成したらファイルを読み込んで (/INPUT)、或いはコピー & ペーストでインプットウィンドウからコマンドを実行して、動作を確認。

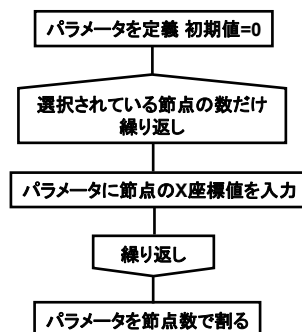


↑
全ての節点を選択
されている場合

例題1 座標の平均値

ヒント

- 平均値を求めるためのフローチャート



例題1 座標の平均値

- ヒント(続き)
 - 現在選択されている節点の数 → *GETコマンド
 - 現在選択されている節点の数だけループするには？
 - 節点番号が常に連続しているとは限らないため、節点番号をループインデックスとして使用することはできない
 - *GETコマンドで、選択されている節点の中から「最小番号」と「次に大きな節点番号」をパラメータにストアできることを利用。詳細は第5章の実習を参照のこと。

-Notes-

例題2 バネ要素の作成

• 手順

1. chapter8_2.dat(下記)を呼び込む(/INPUT)。このファイルにはプリプロセッサにおけるコマンドが記述されている。
2. エディタを用いて、下図のようにばね要素(COMBIN14、chapter8_2.dat内にて定義済)を作成するためのANSYSコマンドを記述。
3. 完成したらファイルを読み込んで(/INPUT)、或いはコピー&ペーストでインプットウィンドウからコマンドを実行して、動作を確認。

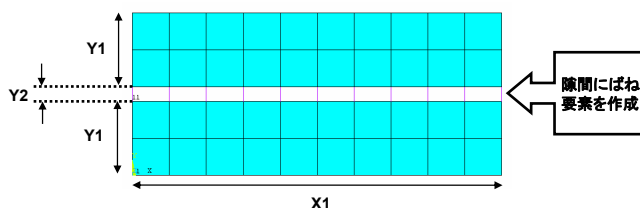
chapter8_2.dat

!*** パラメータの定義

X1=50 !X方向寸法
Y1=10 !Y方向寸法
Y2=2 !隙間の寸法
EDIV=5 !最大要素サイズ

!** モデル作成

/PREP7
ET,1,PLANE42
ET,2,COMBIN14
RECT,0,X1,0,Y1
ESIZE,EDIV
AMESH,ALL
TYPE,2



例題2 バネ要素の作成

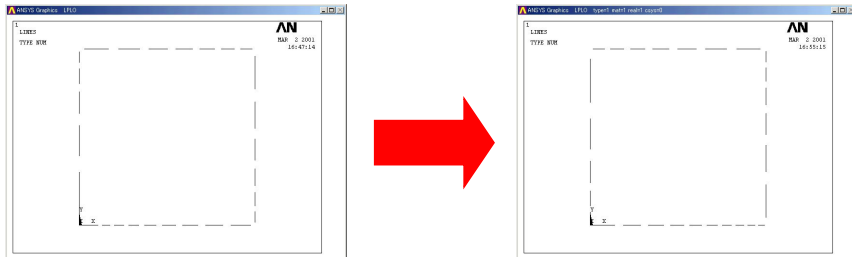
• ヒント

- ループさせる回数は、ばね要素の数。
- ばね要素を作成するためのコマンド → Eコマンド
E, 節点番号1, 節点番号2
- 節点番号をどうやって指定するか？
 - ・ 節点番号は整列されていないので、直接番号を指定することは不可
 - ・ エリアの寸法($X1 * Y1$)、要素分割数(EDIV)は既知 → 節点の位置は既知 → 選択機能、或いは*GETコマンドにて節点を指定可能

例題3 分割比の反転

• 手順

1. chapter8_3.dbを呼び出す (RESUME)。このファイルにはプリプロセッサにおける入力データがストアされている。全てのラインに任意の要素分割数、分割比が定義されている。
2. エディタを用いて、現在選択されている全てのラインの分割比を反転させるためのマクロ「RATIOREV.MAC」を記述。
3. 完成したらマクロを実行して、動作を確認。



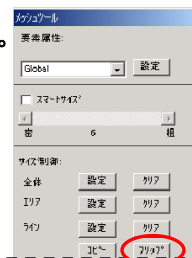
例題3 分割比の反転

• ヒント

- ラインに要素分割数と分割比を与えるためのコマンド
LESIZE,ライン番号,,,分割数,分割比
- 分割比を反転させるには...
LESIZE,ライン番号,,,分割数,1/分割比
- 分割数、分割比は*GETコマンドでパラメータにストア可能
- メッシュツールの「フリップ」を実行後

ANSYS Utility Menu>リスト > ファイル > ログファイル

を参照してみよ。この機能は内部的にマクロを使用している。



例題4 圧力分布の定義

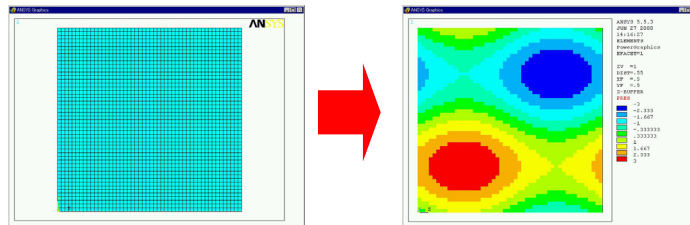
• 手順

1. chapter8_4.dbを呼び出す (RESUME)。このファイルには圧力荷重データを除く全ての入力データがストアされている。
2. エディタを用いて、1×1の平面に、以下の式で表される圧力を-Z方向に負荷するためのANSYSコマンドを記述。

$$\text{Pres} = \sin(2\pi * X) + 2\sin(2\pi * Y)$$

(X,Yは全体直交座標系による座標値)

3. 完成したらファイルを読み込んで (/INPUT) 或いはコピー & ペーストでインプットウィンドウからコマンドを実行して、動作を確認 (/PSF)。



例題4 圧力分布の定義

• ヒント

- 圧力は要素単位で与える (SFE)。
- 圧力が位置 (X,Y) の関数 → 要素の位置 (X,Y) の情報が必要 → *GET コマンドにより、要素中心の X,Y 座標値をとることができる
- ループする回数は、要素の数。要素番号が連続的なため、要素番号をループインデックスとして使用可能。
- 今回使用する荷重キーは“2” (SFE, 要素番号, 2, PRES,, 荷重値)

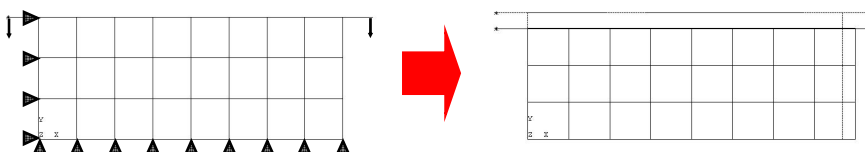
例題5 接触面のトータル力

• 手順

1. chapter8_5.dbを呼び出す (RESUME)。このファイルには全入力データと最終結果データがストアされている。
2. ポストプロセッサに入り、コンタクト要素 (CONTA171、要素タイプ3番) のみを選択する (ESEL ,S,TYPE,,3)。

解析概要:

2次元剛体面-弾性体面の接触解析 ($X=0$ の面にて1/2対称の平面応力問題)。上から剛体板を強制的に押し下げる。ソリッドは下部ラインをY方向拘束。



例題5 接触面のトータル力

• 手順 (続き)

3. エディタを用いて、接触要素にかかっているトータル力をパラメータ "FTOTV" にストアするためのANSYSコマンドを記述。
4. 完成したらファイルを読み込んで (/INPUT)、或いはコピー & ペーストでインプットウィンドウからコマンドを実行して、動作を確認。

例題5 接触面のトータル力

- ヒント

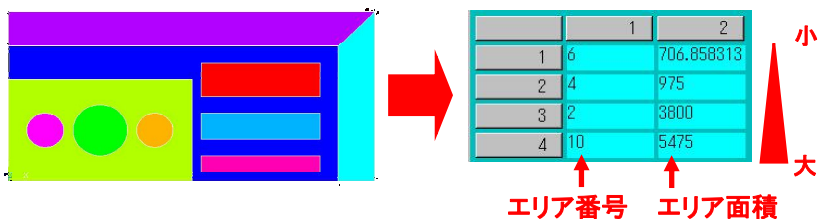
- 接触要素では、解析結果として接触圧および接触面積を要素テーブルから出力することができる(ETABLE)。
- 接触力 = 接触圧 × 要素面積
- トータル接触力 = Σ (接触力)

-Notes-

例題6 エリアのソート

• 手順

1. chapter8_6.dbを呼び出す(RESUME)。このファイルにはソリッドモデル作成までの入力データがストアされている。
2. ASUMコマンドを実行(エリアの面積を計算するため)。
3. エディタを用いて、選択されているエリアをエリア面積が小さい順に、配列パラメータ「ASORT」にストアするためのANSYSコマンドを記述。
4. 完成したらファイルを読み込んで(/INPUT)、或いはコピー＆ペーストでインプットウィンドウからコマンドを実行して、動作を確認。



例題6 エリアのソート

• ヒント

– ソートの例(バブルソート)

3 8 7 1 を番号の小さい順に並べ替える

一巡目(最大値を最後尾に移動)

Step 1 3 8 7 1 → 3 8 7 1

後ろの方が大きいのでそのまま

Step 2 3 8 7 1 → 3 7 8 1

後ろの方が小さいので入れかえる

Step 3 3 7 8 1 → 3 7 1 8

後ろの小さいので入れかえる

最大値が最後尾に移動

例題6 エリアのソート

- ヒント(続き)
 - ソートの例(続き)

二巡目(次に大きな値を最後から2番目に移動)

Step 1 3 7 1 8 \longrightarrow 3 7 1 8

Step 2 3 7 1 8 \longrightarrow 3 1 7 8

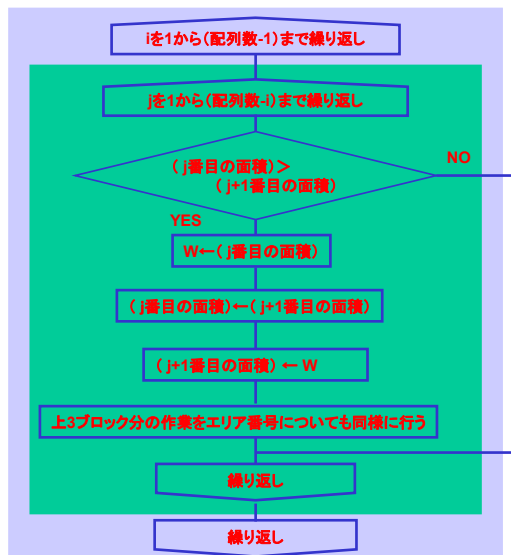
三巡目(次に大きな値を最後から3番目に移動)

Step 1 3 1 7 8 \longrightarrow 1 3 7 8 完成!

- 配列が n 個あった場合、 $1 \sim (n-1)$ 巡目まで行う
- Step数は、「配列数 - 現在の巡目の数」

例題6 エリアのソート

- ヒント(続き)
 - ソートの例(続き)
- フローチャート



例題6 エリアのソート

- ヒント(続き)
 - まず配列パラメータに、エリア番号とその面積をストアする
1. 選択されているエリア数をパラメータにストア(*GET)
 2. エリアの最小番号をパラメータにストア(*GET)
 3. 行数がエリア数、列数が2のアレイ型配列パラメータを宣言(*DIM)
 4. 配列パラメータの一行目にエリア番号を、二行目にエリア面積をストア(*DO, エリア面積は*GETを利用)。
- 次に、ソーティングのアルゴリズムに従い、エリア面積の小さい順に配列を並べ替える。

-Notes-

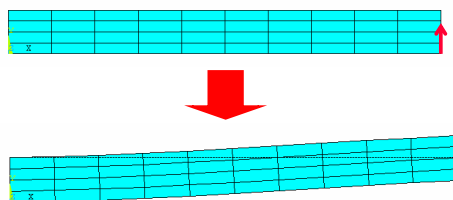
例題7 引っ張りと圧縮でヤング率の異なる材料

• 手順

1. chapter8_7.datを呼び込む(/INPUT)。このファイルには材料変更前の解析が実行されるまでのコマンドが記述されている。

解析概要:

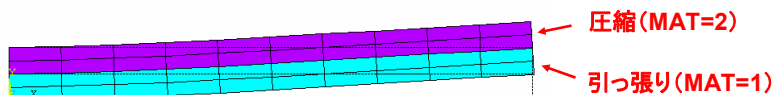
円盤の2次元軸対称問題。円盤の外周下部のラインに+Y方向の力をかける。ヤング率は、1番に $1.5E7$ (引っ張り用)、2番に $2.0E7$ (圧縮用)が定義され、現在は全要素に1番の材料がアサインされている。



例題7 引っ張りと圧縮でヤング率の異なる材料

• 手順(続き)

2. エディタを用いて、要素が引っ張りが支配的か圧縮が支配的かにより異なるヤング率を使用し解析を実行するためのマクロ「BILINEAR.MAC」を作成。なお、引数としてイタレーション数を指定できるようにすること。
3. 完成したらマクロを実行して、動作を確認。



例題7 引っ張りと圧縮でヤング率の異なる材料

- ヒント

- ある要素で、引っ張りが支配的か、それとも圧縮が支配的か？
 - ・ 要素の最大主応力と最小主応力との絶対値の比較によって評価

	最大主応力	最小主応力	状態
要素1	1581.8	519.66	引っ張り
要素2	85.817	-399.56	圧縮

- 状態を判定するために配列パラメータを用意。*VOPERコマンドにて判定。
- 詳細は、解答例の注釈を参照のこと

-Notes-

例題解答例

• 例題1

*GET,NCOUNT,NODE,,COUNT	!現在選択されている節点の数をパラメータNNUMにストア
*GET,NNUM,NODE,,NUM,MIN	!現在選択されている節点の最小節点番号をパラメータNMINにストア
NXSUM=0	!初期パラメータ: 初期値0
*DO,I,1,NCOUNT	!DOループの開始 (Iは1からNNUMまで1ずつ増加)
NXSUM=NXSUM+NX(NNUM)	!パラメータNXSUMに節点NMINのX座標値を代入
NMIN=NDNEXT(NNUM)	!節点番号NMINの次に大きな節点番号をパラメータNMINに代入
*ENDDO	!繰り返し
NXAVE=NXSUM/NCOUNT	!パラメータNXAVEに、NXSUM/NNUMの値を代入

例題解答例

• 例題2

LSEL,S,LOC,Y,Y1,Y1+Y2	!全体直交座標系のY=Y1~Y1+Y2上にあるラインを選択
NSLL,S,1	!現在選択されているラインに属する節点を選択(端部を含む)
*GET,NNUM,NODE,,COUNT	!現在選択されている節点の数をパラメータNNUMにストア
LOCAL,11,0,,Y1	!全体直交座標系の(0,Y1,0)上に11番の局所直交座標系を作成
*DO,I,1,NNUM/2,1	!DOループの開始 (Iは1からNNUM/2まで1ずつ増加)
NODE1=NODE(0,0,0)	!座標系11番の原点に最も近い節点の節点番号をパラメータNODE1にストア
NODE2=NODE(0,Y2,0)	!座標系11番の (0,Y2,0) に最も近い節点の節点番号をパラメータNODE2にストア
/prep7	
E,NODE1,NODE2	!節点NODE1とNODE2から、要素COMBIN14を作成
NSEL,U,,,NODE1	!現在選択されている節点から、節点NODE1を非選択にする
NSEL,U,,,NODE2	!現在選択されている節点から、節点NODE2を非選択にする
*ENDDO	!繰り返し
ALLSEL	!全選択(選択の解除)
CSYS,0	!アクティブな座標系を全体直交座標系に戻す

例題解答例

• 例題3

*GET,_z1,LINE,,COUNT	!ライン数をパラメータにストア
*SET,_z2,0	!スカラーパラメータの宣言(_z2=0)
*DO,_z5,1,_z1	!DOループの開始
*SET,_z2,LSNEXT(_z2)	!_z2に「次に大きなライン番号」をストア
*GET,_z3,LINE,_z2,ATTR,NDNX	!_z3にライン_z2の分割数をストア
*GET,_z4,LINE,_z2,ATTR,SPNX	!_z4にライン_z2の分割比をストア
*GET,_z6,LINE,_z2,ATTR,KYND	!_z6にライン_z2のソフト分割オプションフラグをストア
*IF,_z3,GT,0,THEN	!分岐:_z3が0より大きければ真
*IF,_z4,NE,0,THEN	!分岐:_z4が0でなければ真
/prep7	
LESIZE,_z2,,, _z3,1/_z4,,, _z6	!ラインの分割比を反転
*ENDIF	!分岐終了
*ENDIF	!分岐終了
*ENDDO	!繰り返し

例題解答例

• 例題4

```

PAI=3.141592
A=1
B=2
*GET,MAXELEM,ELEM,,NUM,MAX      !最大要素番号をMAXELEMへ代入
*DO,I,1,MAXELEM,1               !DO文・要素番号1からMAXELEMまで
*GET,ELX,ELEM,I,CENT,X          !要素番号Iの中心X座標をELXへ代入
*GET,ELY,ELEM,I,CENT,Y          !要素番号Iの中心Y座標をELYへ代入
ECP=A*SIN(2*PAI*ELX)+B*SIN(2*PAI*ELY) !圧力値を計算しECPへ代入
/solu
SFE,I,2,PRES,,ECP               !ECPを要素Iの圧力として定義
*ENDDO
/PSF,PRES,3
EPLOT

```

例題解答例

• 例題5

esel,s,type,,3	!コンタクト要素を選択
etab,cpres_i,smisc,1	!!節点の接触圧を要素テーブルにストア
etab,cpres_j,smisc,2	!J節点の接触圧を要素テーブルにストア
etab,carea_i,nmisc,27	!!節点の接触面積を要素テーブルにストア
etab,carea_j,nmisc,28	!J節点の接触面積を要素テーブルにストア
smult,cforce_i,cpres_i,carea_i	!!節点の接触力を計算
smult,cforce_j,cpres_j,carea_j	!J節点の接触力を計算
ssum	!各要素テーブルの総和を計算
*get,ftotv_i,ssum,,item,cforce_i	!!節点のトータル接触力を計算
*get,ftotv_j,ssum,,item,cforce_j	!J節点のトータル接触力を計算
ftotv=ftotv_i+ftotv_j	!トータル接触力を計算
*stat,ftotv	!トータル接触力をリスト表示

例題解答例

• 例題6

*GET,ANUM,AREA,,COUNT	!エリア数をパラメータにストア
*GET,AMIN,AREA,,NUM,MIN	!エリアの最小番号をパラメータにストア
*DIM,ASORT,ARRAY,ANUM,2	!ANUM行2列のアレイ型配列/パラメータの宣言
*DO,I,1,ANUM,1	!DOループの開始(Iは1からANUMまで1ずつ増加)
ASORT(I,1)=AMIN	!!行1列にエリア番号をストア
*GET,ASORT(I,2),AREA,AMIN,AREA	!!行2列にエリア面積をストア
AMIN=ARNEXT(AMIN)	!AMINに、次に大きなエリア番号をストア
*ENDDO	!繰り返し
*DO,I,1,ANUM-1,1	!DOループの開始(Iは1からANUM-1まで1ずつ増加)
*DO,J,1,ANUM-1,1	!DOループの開始(Jは1からANUM-1まで1ずつ増加)
*IF,ASORT(J,2),GT,ASORT(J+1,2),THEN	!分岐: J行2列の値がJ+1行2列の値より大きければ真
W=ASORT(J,2)	!WにJ行2列の値を代入
ASORT(J,2)=ASORT(J+1,2)	!J行2列にJ+1行2列の値を代入
ASORT(J+1,2)=W	!J+1行2列にWの値を代入
WW=ASORT(J,1)	!WWにJ行1列の値を代入
ASORT(J,1)=ASORT(J+1,1)	!J行1列にJ+1行1列の値を代入
ASORT(J+1,1)=WW	!J+1行1列にWWの値を代入
*ENDIF	!分岐終了
*ENDDO	!繰り返し
*ENDDO	!繰り返し

例題解答例

• 例題7

```

!***** 引っ張りと圧縮で異なる材料特性を使用するためのマクロ *****
! 材料1ー引っ張りの材料特性
! 材料2ー圧縮の材料特性
! ARG1ーイタレーション数(デフォルト2)
! このマクロは静解析を実施した後に実行する

/NOP                                !アウトプットウィンドウへの出力をオフにする
_NITER=ARG1                        !_NITERにイタレーション数をストア
*IF,_NITER,LT,2,THEN               !分岐:_NITERが2より小さければ真
_NITER=2                            !_NITERに2を代入
*ENDIF                              !分岐終了
*DO,_ITER,1,_NITER                 !DOループの開始(_ITERを1から_NITERまで)
/POST1
SET,1,1                            !結果データの読み込み
*GET,AR11,ELEM,,COUNT             !要素数をパラメータAR11にストア
*DIM,_S1,,AR11                     ! (AR11)行のアレイ型配列パラメータ_S1の宣言
*DIM,_S3,,AR11                     ! (AR11)行のアレイ型配列パラメータ_S3の宣言
ETABLE,SIGMAX,S,1                  !要素の最大主応力をSIGMAXにストア
ETABLE,SIGMIN,S,3                  !要素の最小主応力をSIGMINにストア
*VGET,_S1(1),ELEM,1,ETAB,SIGMAX   !SIGMAXを_S1にストア
*VGET,_S3(1),ELEM,1,ETAB,SIGMIN   !SIGMINを_S3にストア
*DIM,_MASK,,AR11                   !AR11行のアレイ型配列パラメータ_MASKの宣言
(続く)

```

例題解答例

• 例題7(続き)

```

(続き)
*VOPER,_MASK(1),_S1(1),LT,0        !最大主応力が0より小さければ1を、それ以外は0を_MASKに代入
*VCUM,1                             !次の*VOPERでは、値を追加させる
*VABS,0,1                           !次の*VOPERでは、最小主応力を絶対値で評価する
*VOPER,_MASK(1),_S3(1),GT,_S1(1)   !最小主応力の絶対値が最大主応力よりも大きければ1を、それ以外は0
                                    !を_MASKに代入

FINISH
/PREP7
MAT,1                                !材料特性のデフォルトを1番にする
EMODIF,ALL                          !全ての要素の材料特性を1番に変更
*VPUT,_MASK(1),ELEM,1,ESEL          !要素に_MASKの値を、選択ステータス(-1:非選択、0:未定義、1:選択)
                                    !として入力。← 圧縮状態の要素のみを選択
MAT,2                                !材料特性のデフォルトを2番にする
EMODIF,ALL                          !選択されている全ての要素の材料特性を1番に変更
ESEL,ALL                            !要素の全選択
FINISH
_S1(1)=                             !パラメータの削除
_S3(1)=
_MASK(1)=
/SOLVE
SOLVE                               !解析の実行
FINISH
*ENDDO                              !繰り返し
_NITER=                             !パラメータの削除
_ITER=
/GOP                                !アウトプットウィンドウへの出力をオンにする

```

-Notes-

-Notes-

付録1 ANSYS ツールバー と START.ANS

ANSYS ツールバー - 概要

- ANSYS ツールバーとは
 - ANSYS GUIの1つ
 - 頻繁に使用する機能(コマンド、メニュー、マクロ)をボタンの形で登録。ボタンをピックアップすることで、即座に登録された機能を実行
 - デフォルトで4つのボタンが登録済(リビジョンにより多少異なる)

SAVE_DB	データベースの保存
RESUM_DB	データベースのリジューム
QUIT	ANSYS終了時のメニューを開く
POWRGRPH	パワーグラフィックスの設定用メニューを開く



ANSYS ツールバー - 使用方法

- ボタンの登録方法

- *ABBRコマンド

*ABBR, ボタン名, 登録する機能(コマンド、GUI、マクロ)

例

*ABBR, REPLOT, /REPLOT ! 画面を再表示

*ABBR, MACRO, SAMPLE.MAC ! マクロを実行

注意: この方法で設定した内容は、ANSYSを終了すると消去されてしまう。
常に設定した内容でANSYSを起動させたい場合は、START.ANSF
ファイル(後述)を利用すること。

ANSYS ツールバー - 使用方法

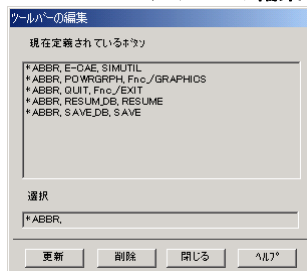
- ボタンの登録方法(続き)

- GUIからツールバー用のエディターを使用

ANSYS Utility Menu> マクロ > ツールバーの編集...

ANSYS Utility Menu> メニュー > ツールバーの編集...

メニューパスは2つある
が、どちらも同じ



注意: この方法で設定した内容は、ANSYSを終了すると消去されてしまう。
常に設定した内容でANSYSを起動させたい場合は、START.ANSF
ファイル(後述)を利用すること。

ANSYS ツールバー - 使用方法

- ボタンの消去
 - *ABBRコマンドの第2フィールドをブランクにする
 *ABBR, SAVE_DB, ! SAVE_DBボタンを消去
- 注意事項
 - ボタン名は8文字まで
 - 機能名は60文字まで、但し、以下の文字は入力できない
 \$, C**, /COM, /GOPR, /NOPR, /QUIT, /UI, *END
 - ボタンは100個まで登録可能

START * *.ANS

- START * *.ANSとは
 - 初期設定用ファイル。ANSYS起動時に読み込まれ、ファイル中に記述されている内容が実行される。
 - * * の箇所にはリビジョン番号が入る(例: ANSYS10.0ならば start100.ansとなる)
- START * *.ANSに記述できる内容
 - ユーティリティコマンド (/VIEW,,1,1,1 !モデルビューの設定)
 - ツールバーの設定 (*ABBRコマンド)
 - メニューの初期設定 (/MSTARTコマンド)
 - パラメータの設定 (PI=ACOS(-1) !円周率をPIに定義)

START * *.ANS

- どこにSTART * *.ANSを入れておけば良いか？
 - 1.カレントディレクトリ
 - 2.ホームディレクトリ
 - 3./ANSYS/apdlディレクトリ
- START * *.ANSファイルのサンプル
 - /ANSYS/apdlディレクトリには、デフォルトでstart100.ansファイルが存在している。このファイルの中には一般的な設定項目が記述され、各行の先頭に“！”マークをつけて無効にしているが、“！”をはずせば有効にすることができる。

START * *.ANS - サンプル

```

start100.ans - メモ帳
ファイル(F) 編集(E) 書式(O) 表示(V) ヘルプ(H)

*****

Sample START.ANS file provided by Ansys, Inc.
for the ANSYS Program
8/07/2002

All of the commands included in this file are commented. If you want to
have any command executed at ANSYS start-up, simply remove the ! from the
beginning of the line.

/NOPR                                ! Suppresses printout of the START.ANS file

Suggested ANSYS abbreviations for the Toolbar:

!*ABBR, REPLOTT ,/REPLOTT            ! Replots the last graphics display
!*ABBR, ISO ,/VIEW,,1,1              ! Changes the view to isometric
!*ABBR, FRONT ,/VIEW,,0,0,1          ! Changes the view to front view
!*ABBR, VECTOR ,/DEVICE,VECT,ON      ! Specifies vector mode for graphics displays
!*ABBR, RASTER ,/DEVICE,VECT,OFF     ! Specifies raster mode for graphics displays
!*ABBR, NOERASE ,/NOERASE            ! Does not erase the screen between plots
!*ABBR, ERASE ,/ERASE                ! Erases the screen between plots
!*ABBR, HIDE_LINE,Fnc_PL_Hidden      ! Brings up dialog box for hidden-line options
!*ABBR, INFO-ON ,/PLOPTS,INFO,ON     ! Turn the legend information on
!*ABBR, INFO-OFF,/PLOPTS,INFO,OFF    ! Turn the legend information off
!*abbr, RESU_ABR,Fnc_ABBRESU        ! Brings up a dialog box to read an
                                     ! abbreviation file

```

付録2 ファイルの入出力

ファイルの入出力 - 概要

- ファイルの入出力は、以下のコマンドで行う
 - ファイルへの出力
 - /OUTPUT** { アウトプットウィンドウの内容を出力 (PC)
 リストウィンドウの内容を出力 (UNIX)
 - *VWRITE** パラメータの出力
 - ファイルから入力
 - *VREAD** パラメータの入力
 - *TREAD** パラメータの入力 (テーブル型パラメータのみ)

ファイルへの出力 - /OUTPUT

- /OUTPUTコマンド

- /OUTPUT, ファイル名, 拡張子, ディレクトリ名
- PCではアウトプットウィンドウの内容を出力し、UNIXではリストウィンドウの内容を出力する
- /OUTPUTのみで、標準出力に戻る

注意: パッチランの場合はPCもリストウィンドウの内容を出力

例: 解析情報をファイルに出力
(PC)

```
/OUTPUT, SOLVE, OUT
SOLV
/OUTPUT
```

節点応力解をファイルに出力
(UNIX)

```
/OUTPUT, NSCOMP, OUT
PRNSOL,S,COMP
/OUTPUT
```

注意: UNIXではアウトプットウィンドウの内容はデフォルトで
Jobname.outに出力されている

注意: PCではリストウィンドウから
ファイル(F) > 保存 でファイルに出力

ファイルへの出力 - *VWRITE

- *VWRITEコマンド

- スカラー、配列パラメータを外部ファイルに出力する
- *VWRITEコマンドの後にFORTRAN編集子を()でくくって指定
- デフォルトでは、アウトプットウィンドウに出力。ファイルに出力したい場合は前後に/OUTPUTコマンドが必要

- FORTRAN編集子

- F: 浮動小数点
- E: 倍数型浮動小数点(指数表記)
- A: 文字列

例:

0.12345 → (F7.5) 全部で7カラム、小数点以下は5カラム

0.12345E6 → ±0.12345E±06 → (E12.5) 全部で12カラム、小数点以下は5カラム

(2X,3F6.1) 空白が2カラム、3回繰り返し、浮動小数点で6カラム、小数点以下は1カラム

ファイルへの出力 - *VWRITE

例1

- 下図の3行4列のアレイ型配列パラメータ「TEST1」をファイル「test1.out」に出力

```
/OUTPUT, test1, out
```

```
*VWRITE, TEST1(1,1), TEST1(1,2), TEST1(1,3), TEST1(1,4)
```

```
(4F10.3) ! 4回繰り返し、浮動小数点で10カラム、小数点以下は3カラム
```

```
/OUTPUT
```

TEST1

	1	2	3	4
1	-12.866	41.097	10.465	-64.771
2	81.304	6.17	-11.577	28.896
3	-9.677	76.445	35.2	59.253

行(縦)方向に書き出す

test1.out

	10	20	30	40
1	-12.866	41.097	10.465	-64.771↓
2	81.304	6.170	-11.577	28.896↓
3	-9.677	76.445	35.200	59.253↓

ファイルへの出力 - *VWRITE

例2

- 解析後、各節点のX方向変位をアレイ配列パラメータ「TEST2」にストアしファイル「test2.out」に出力

```
*DIM,TEST2,ARRAY,25
```

! パラメータの宣言。節点の数を25個とする。

```
*VGET,TEST2(1,1),NODE,1,U,X
```

! 節点のX方向変位をパラメータにストア

```
/OUTPUT, test2, out
```

```
*VWRITE, TEST2(1,1)
```

```
(E12.4)
```

! 倍数型浮動小数点で12カラム、小数点以下は4カラム

```
/OUTPUT
```

test2.out

	10
1	0.0000E+00↓
2	-0.4073E-09↓
3	-0.2420E-09↓
4	-0.3552E-09↓
5	-0.4005E-09↓

ファイルからの入力 - *VREAD

- *VREADコマンド
 - 外部ファイルのデータをスカラー、配列パラメータとして読み込む
 - *VREADコマンドの後にFORTRAN編集子を()でくくって指定
- FORTRAN編集子
 - F: 浮動小数点
 - E: 倍数型浮動小数点(指数表記)
 - A: 文字列

ファイルからの入力 - *VREAD

- 例1
 - 以下のファイル「test3.txt」内にあるデータをアレイ型配列パラメータ「TEST3」に読み込む

```
*DIM,TEST3,ARRAY,5
*vread,TEST3(1),test3.txt
(F5.2)    ! 浮動小数点で5カラム、小数点以下は2カラム
```

test3.txt

0
1.25↓
-2.43↓
0.04↓
8.31↓
-5.99↓



TEST3

	1
1	1.25
2	-2.43
3	0.04
4	8.31
5	-5.99



行(縦)方向に読み込む

ファイルからの入力 - *VREAD

例2

- 以下のファイル「test4.txt」内にあるデータをアレイ型配列パラメータ「TEST4」に読み込む

```
*DIM,TEST4,ARRAY,5,3
*VREAD,TEST4(1,1),test4.txt
(F1.0)
*VREAD,TEST4(1,2),test4.txt
(2X,E6.1)
*VREAD,TEST4(1,3),test4.txt
(9X,F4.2)
```

test4.txt

	10	10
1	2.0E11	0.3↓
2	2.1E11	0.27↓
3	1.9E10	0.33↓
4	1.8E11	0.31↓
5	2.1E10	0.25↓



TEST4

	1	2	3
1	1	2000000000	0.3
2	2	2100000000	0.27
3	3	1900000000	0.33
4	4	1800000000	0.31
5	5	2100000000	0.25

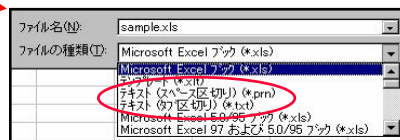
ファイルからの入力 - *TREAD

*TREADコマンド

- 外部ファイルのデータをテーブル型配列パラメータとして読み込む
- FORTTRAN編集子は必要なし
- 外部ファイル中の各データはスペースかタブで区切っておく
- 表計算ソフトからのデータを読み込む場合は、スペースかタブ区切りでテキストファイルに出力しておく必要がある

Microsoft Excel - sample.xls

	A	B	C	D	E
1	0	2	4	6	8
2	1	54	435	6	432
3	3	43	543	543	34
4	5	6	4	46	5436
5	7	76	123	543	75
6	9	54	87	6	543



ファイルからの入力 - *TREAD

例1

- 以下のファイル「test5.txt」内にあるデータをテーブル型配列パラメータ「TEST5」に読み込む

```
*DIM,TEST5,TABLE,7
*TREAD,TEST5,test5.txt
```

test5.txt

1-	0↓
2-	10↓
3-	30↓
4-	60↓
5-	100↓
6-	150↓
7-	210↓



TEST5

	0
1	0
2	10
3	30
4	60
5	100
6	150
7	210

注意: 1次元配列の場合、0行をテキストに入れておく必要はない

ファイルからの入力 - *TREAD

例2

- 以下のファイル「test6.txt」内にあるデータをテーブル型配列パラメータ「TEST6」に読み込む

```
*DIM,TEST6,TABLE,5,4
*TREAD,TEST6,test6.txt
```

test6.txt

0-	2-	4-	6-	8↓
1-	10-	50-	15-	20↓
3-	30-	60-	17-	40↓
5-	50-	70-	19-	60↓
7-	70-	80-	21-	80↓
9-	90-	90-	23-	100↓



TEST6

	2	4	6	8
1	10	50	15	20
3	30	60	17	40
5	50	70	19	60
7	70	80	21	80
9	90	90	23	100

注意: 2次元配列の場合、0行0列にダミーの値が必要

ファイルからの入力 - *TREAD

例3

- 以下のファイル「test7.txt」内にあるデータをテーブル型配列パラメータ「TEST7」に読み込む

```
*DIM,TEST7,TABLE,2,3,3
```

```
*TREAD,TEST7,test7.txt
```

注意:3次元配列の場合、0行0列に面のインデックス番号を入力する

test7.txt

100	1	2	3
10	2	10	20
20	4	15	30
↓			
200	1	2	3
10	100	50	20
20	200	40	10
↓			
300	1	2	3
10	50	200	60
20	100	400	30

TEST7

100	200	300	
	1	2	3
10	2	10	20
20	4	15	30

-Notes-

-Notes-

-Notes-

付録3 実習問題データ作成用コマンド

array1.db 作成用コマンド

```
/prep7  
rect,,100,,100  
pcirc,,50  
asba,1,2  
ldiv,5,,,2  
l,1,3  
asbl,all,all  
et,1,42  
esize,,5  
amesh,all  
save,array1,db
```

loop1.db 作成用コマンド

```
/prep7                                finish
rect,,100,,100                        save,loop,db
pcirc,,50
asba,1,2
ldiv,5,2
l,1,3
asbl,all,all
lselect,s,,,1,3
lselect,a,,,5
lesize,all,,,20
allselect
lesize,all,,,6
et,1,42
mp,ex,1,21000
mp,nuxy,1,0.3
amesh,all
lselect,s,,,9,10
dl,all,,symm
allselect
```

-Notes-

ANSYS APDL セミナー

2006 年 2 月 1 日 初版発行

発行者 サイバネットシステム株式会社

発行所 サイバネットシステム株式会社

本社／ 〒112-0012 東京都文京区大塚 2 丁目 15 番 6 号
ニッセイ音羽ビル

電話 (03) 5978-5423(直通)

FAX (03) 5978-5960

西日本支社／ 〒540-0028 大阪市中央区常磐町 1 丁目 3 番 8 号
中央大通 FNビル

電話 (06) 6940-3631(直通)

FAX (06) 6940-3601

中部支社／ 〒460-0003 愛知県名古屋市中区錦1丁目6番26号
富士ソフトABC 名古屋ビル

電話 (052) 219-5900

FAX (052) 219-5970
