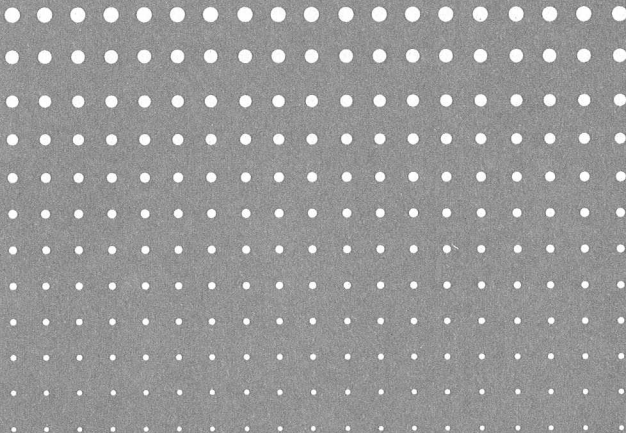


*fx-4800P*  
操作マニュアル



本書はお読みになった後も大切に保管してください。

この装置は、第二種情報装置（住宅地域またはその隣接した地域において使用されるべき情報装置）で住宅地域での電波障害防止を目的とした情報処理装置等電波障害自主規制協議会（VCCI）基準に適合しております。しかし、本装置をラジオ、テレビジョン受信機に近接してご使用になると、受信障害の原因となることがあります。取扱説明書に従って正しい取り扱いをしてください。

## 安全上のご注意

このたびは本機をお買い上げいただきまして、誠にありがとうございます。ご使用になる前に、この「安全上のご注意」をよくお読みの上、正しくお使いください。

### 絵表示の意味



記号は「してはいけないこと」を意味しています(左の例は分解禁止)。



記号は「しなければならないこと」を意味しています(左の例は電源プラグをコンセントから抜く)。



### 注意

この表示を無視して誤った取り扱いをすると、人が傷害を負う可能性が想定される内容および物的損害のみの発生が想定される内容を示しています。

### 電池について



● 本機で使用しているボタン電池を取り外した場合は、誤ってボタン電池を飲むことがないようにしてください。特に小さなお子様にご注意ください。

● 電池は小さなお子様の手の届かない所へ置いてください。万一、お子様が飲み込んだ場合は、ただちに医師と相談してください。



● 電池は、充電や分解、ショートする恐れのあることはしないでください。また、加熱したり火の中へ投入したりしないでください。



## 注意

- ● 電池は使い方を誤ると液もれによる周囲の汚損や、破裂による火災・けがの原因となることがあります。次のことは必ずお守りください。
  - ・極性(＋と－の向き)に注意して正しく入れてください。
  - ・長期間使用しないときは、本体から電池を取り出しておいてください。また、5年に1度は動作電池およびメモリー保護用電池を交換してください。
- ⊘ ● 本機で指定されている電池以外は使用しないでください。

## メモリー保護

- ● 本機に記憶させた内容は、ノートに書くなどして、本機とは別に必ず控えを残してください。本機の故障、修理や電池消耗などにより、記憶内容が消えることがあります。
- ● 電池交換を行なう際は、取扱説明書をよくお読みになり、正しく行なってください。電池交換のしかたを誤ると、データが消えたり、変化したりすることがあります。

## 火中に投入しないでください

- ⊘ ● 本機を火中に投入しないでください。破裂による火災・けがの原因となることがあります。

お買い上げ後、初めて本機を使用する際は必ずリセット操作を行なってください(8ページ参照)。

- 本書の内容に関しては、将来予告なしに変更することがあります。
- 本書の内容については万全を期して作成いたしました。が、万一ご不審な点や誤りなど、お気づきのことがありましたらご連絡ください。
- 本書使用による損害および逸失利益等につきましては、当社では一切その責任を負えませんので、あらかじめご了承ください。
- 本書の一部または全部を無断で複製することは禁止されています。また、個人としてご利用になる場合は、著作権法上、当社に無断では使用できませんのでご注意ください。
- 本機使用により生じた損害、逸失利益または第三者からのいかなる請求につきましても、当社では一切その責任を負えませんので、あらかじめご了承ください。
- 故障、修理、電池交換等に起因するデータの消去による損害および逸失利益等につきましては、当社では一切その責任を負えませんので、あらかじめご了承ください。

## その他の使用上のご注意

- 極端な温度条件下での使用や保管は避けてください。  
低温では表示の応答速度が遅くなったり、点灯しなくなったり、電池寿命が短くなったりします。また、直射日光の当たる場所や窓際または暖房器具の近くなど、極端に温度が高くなる場所には置かないでください。ケースの変色や変形、または電子回路の故障の原因になります。
- 湿気やほこりの多い場所での使用や保管は避けてください。  
水が直接かかるような使用は避けるとともに、湿気やほこりにも十分ご注意ください。電子回路の故障の原因になります。
- 落としたり、強いショックを与えないでください。
- 「ひねり」や「曲げ」を与えないでください。
- 分解しないでください。
- お手入れの際は、乾いた柔らかい布をご使用ください。  
特に汚れがひどい場合は、中性洗剤液に浸した布を固くしぼっておふきください。なお、シンナーやベンジンなどの揮発性溶剤は使用しないでください。キーの上の文字が消えたり、ケースにシミをつけてしまう恐れがあります。

## 電源および電池交換について

本機の電源には、動作用電池とメモリー保護用電池の2種類を使います。

- a. 動作用電池 …… ボタン電池<CR2032>1個
- b. メモリー保護用電池 …… ボタン電池<CR2032>1個

動作用電池が消耗しますと、次のようなメッセージが表示されます。そのときは本機の使用を一時中断して、ただちに電池を交換してください。

**\*\*Low battery!\*\***

電池を交換しないでそのまま使用を続けると、メモリーを保護するために自動的に電源がOFFになります。

この状態では **AC** を押しても、電源はONになりません。また、この状態のまま放置した場合には、メモリー保護の保証はできません。この場合、動作用電池を交換すると、通常の状態に戻ります。

### ⚠ 注意

- 動作用電池とメモリー保護用電池は、絶対同時にはずさないでください。データが消えたり、変化したりすることがあります。万一両方の電池を同時にはずした場合は、電池を正しくセットした後、リセット操作を行ってください(8ページ参照)。
- 長時間使用しないときは、本体から電池を取り出しておいてください。また、5年に1度は動作用電池およびメモリー保護用電池を交換してください。

最初の電池は工場出荷時に組み込まれたモニター用電池\*ですので、記載された電池寿命に満たないうちに切れることがあります。  
\*モニター用電池とは製品の機能や性能をチェックするための電池のことです。

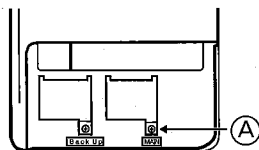
## ■ 動作電池の交換の仕方

### 動作電池交換時の諸注意

- メモリー保護用電池は、絶対同時に取りはずさないでください。
- 電池交換は、必ず電源をOFFにして行なってください。電源がONの状態では電池を交換すると、データが消えてしまいます。
- 動作電池を取りはずした状態や正しく入っていない状態で、電源をONにしたりしないでください。記憶されていたデータが消えたり、正常に動作しなくなります。万一このような状態になった場合は、電池を正しくセットした後、リセット操作(8ページ参照)を行なってください。
- 電池の液もれによる故障を防ぐため、5年に1度は必ず電池を交換してください。

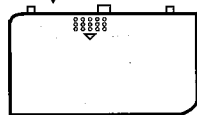
① **SHIFT OFF**と操作して、電源をOFFにします。

② 裏ボタンを矢印の方向にスライドさせて、取りはずします。



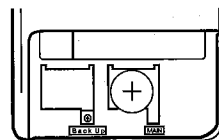
③ ネジ④をはずして、電池押さえ板を取りはずします。

④ 古い電池を取り出します。



⑤ 新しい電池の表面を乾いた布でよく拭いてから「+」側を上に入れて入れます。

⑥ 電池押さえ板で電池を押さえながら、ネジ止めして電池押さえ板を閉じます。



⑦ 裏ボタンをはめ込み、**AC/ON**を押して電源をONにします。

動作電池交換時にはメモリー保護用電池が働いていますので、本体内のデータが消えてしまうことはありません。

※ 動作電池を取りはずしたまま、長時間放置しないでください。記憶されていたデータが消えてしまいます。

※ 動作電池交換後に **AC/ON**を押して電源をONにしたとき、表示が濃すぎたり薄すぎたりして見にくい場合は、コントラスト(表示濃度)を調整してください(29ページ参照)。

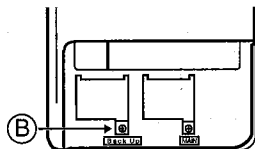
## ■ メモリー保護用電池の交換の仕方

### メモリー保護用電池交換時の諸注意

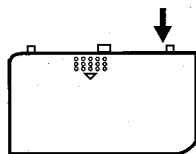
- 電池交換は、動作用電池の寿命が切れていないことを確認してから行なってください。もし、“Low battery!”と表示された場合は、動作用電池を先に交換してください。
- 動作用電池は、絶対同時に取りはずさないでください。
- 電池交換は、必ず電源をOFFにして行なってください。
- 電池寿命は約2年ですが、メモリー保護のために早めの電池交換をお勧めします。

① **SHIFT OFF**と操作して、電源をOFFにします。

② 裏ボタンを矢印の方向にスライドさせて、取りはずします。

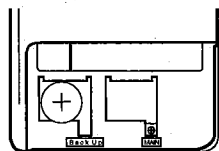


③ ネジ④をはずして、電池押さえ板を取りはずします。



④ 古い電池を取り出します。

⑤ 新しい電池の表面を乾いた布でよく拭いてから、「+」側を上にして入れます。



⑥ 電池押さえ板で電池を押さえながら、ネジ止めして電池押さえ板を閉じます。

⑦ 裏ボタンをはめ込み、**AC ON**を押して電源をONにします。

メモリー保護用電池交換時には動作用電池が働いていますので、本体内のデータが消えてしまうことはありません。



## ⚠ 注意

- 電池は使い方を誤ると液もれによる周囲の汚損や、破裂による火災・けがの原因となることがあります。次のことは必ずお守りください。

・ 極性(＋と－の向き)に注意して正しく入れてください。



- 電池は、充電や分解、ショートする恐れのあることはしないでください。また、加熱したり火の中へ投入したりしないでください。



電池は小さなお子様の手の届かない所に置いてください。万一、お子様が飲み込んだ場合は、ただちに医師と相談してください。

## ■ オートパワーオフ(自動電源OFF)機能

ムダな電力消費を防ぐため、操作完了後約6分で自動的に電源がOFFになります。計算機を再びご使用になるときは、**[AC/ON]**を押すと電源ONになります。この場合でも、メモリー内容およびモード設定は保護されています。

## ■ リセット操作

本機を初期状態に戻したいときは、以下のリセット操作を行なってください。なお、この操作を行なうと記憶されていたデータが消えてしまいますので、注意してください。

- ① **MODE** **8** と操作して、リセット確認表示にします。

**MODE** **8**

```
**** RESET ****
Reset all?
YES:[EXE]
NO :[EXIT]
```

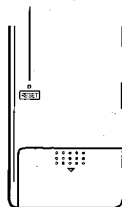
- ② リセットを実行します。

**EXE**

```
*****
RESET
ALL MEMORIES!
*****
```

- ※ リセット確認表示を解除したいときは、**EXIT** を押してください。
- ※ リセット確認表示が現われたら、速やかに **EXE** または **EXIT** を押してください。また、キーを押す前にオートパワーオフ機能により電源OFFになった場合は、最初からリセット操作をやり直してください。
- ※ リセット操作を行なった後、表示が濃すぎたり薄すぎたりして見にくいときは、コントラスト(表示濃度)を調整してください(29ページ参照)。
- ※ 記憶されていたデータをリセット操作後もお使いになる場合は、リセット操作前にノートなどに控えを書き留めておくことをお勧めします。
- ※ 何らかの要因により本機が正常に動作しなくなった場合は、本体裏面にあるリセットボタンを先の細い棒などで軽く押ししてください。リセット確認表示になりますので、続けて上記のリセット操作を行なってください。

リセットボタン



リセット操作を行なうと、本機は次のように初期化されます。

1. COMPメニュー選択
2. Degモード設定
3. Norm1モード設定
4. Dec指定(BASEメニュー選択時)
5. 変数メモリー／増設メモリークリアー
6. アンサーメモリー (Ans) クリアー
7. 統計計算専用メモリークリアー
8. 数式記憶専用メモリークリアー
9. 漸化式メモリークリアー
10. プログラムクリアー
11. 入力バッファ／ACリプレイクリアー

#### 注 意

演算実行中(表示が消えて、計算機内部で演算を行なっている状態)に誤ってリセット操作を行なうと、記憶していたメモリー内容が消えてしまいます。リセット操作は、必ず演算終了後に行なってください。

# 目次

■ 安全上のご注意 .....	1
■ その他の使用上のご注意 .....	3
■ 電源および電池交換について .....	4
動作用電池の交換の仕方 .....	5
メモリー保護用電池の交換の仕方 .....	6
オートパワーオフ(自動電源OFF)機能 .....	7
リセット操作 .....	8
操作の流れ (必ずお読みください) .....	15
1. キーの見方 .....	15
2. モードの指定 .....	15
3. 表示の見方 .....	17
4. 故障かと思われる前に .....	20

## 第1章 本体構成と使い方

1-1. 各部の名称とその説明 .....	22
キーの働き .....	23
コントラスト(表示濃度)調整 .....	29
1-2. 計算を始める前に .....	30
機能メニュー表示 .....	30
計算式の入力(書式通り入力方式) .....	34
計算の優先順位 .....	35
乗算記号(×)の省略 .....	36
スタック数 .....	36
入出力桁数と演算範囲 .....	37
桁オーバーとエラー .....	37
入力文字数(バイト数) .....	38
訂正の仕方 .....	39
メモリー .....	40
内蔵関数(MATH)メニュー .....	48

## 第2章 マニュアル計算

2-1. 基本計算の仕方 .....	52
--------------------	----

加減乗除計算 .....	52
カッコ計算 .....	53
小数点以下指定、有効桁数指定、および指数表示範囲指定 .....	54
パーセント計算 .....	56
メモリー計算 .....	57
<b>2-2. 特別機能 .....</b>	<b>59</b>
アンサー(Ans)機能 .....	59
連続演算機能 .....	60
リプレイ機能 .....	60
エラー位置表示機能 .....	61
マルチステートメント機能 .....	62
<b>2-3. 関数計算の仕方 .....</b>	<b>63</b>
角度変換( $^{\circ}$ , r, g) .....	63
三角関数(sin, cos, tan)、逆三角関数( $\sin^{-1}$ , $\cos^{-1}$ , $\tan^{-1}$ ) .....	64
対数関数(log, ln)、指数関数( $10^x$ , $e^x$ , $\wedge(x^y)$ , $x\sqrt{\quad}$ ) .....	65
双曲線関数(sinh, cosh, tanh)、逆双曲線関数( $\sinh^{-1}$ , $\cosh^{-1}$ , $\tanh^{-1}$ ) .....	66
その他の関数( $\sqrt{\quad}$ , $x^2$ , $x^{-1}$ , $x!$ , $\sqrt[3]{\quad}$ , Ran#, Abs, Int, Frac, Intg) .....	67
座標変換(Pol, Rec) .....	69
順列( $nPr$ )、組み合わせ( $nCr$ ) .....	70
分数計算 .....	71
Eng記号計算 .....	72
<b>2-4. 科学定数計算 .....</b>	<b>73</b>
<b>第3章 微分/2次微分計算・積分計算・<math>\Sigma</math>計算</b>	
<b>3-1. 微分計算 .....</b>	<b>78</b>
微分計算の実例 .....	79
微分計算の応用 .....	80
微分計算時の諸注意 .....	81
<b>3-2. 2次微分計算 .....</b>	<b>82</b>
2次微分計算の実例 .....	83
2次微分計算の応用 .....	84
2次微分計算時の諸注意 .....	84

<b>3-3. 積分計算</b> .....	<b>85</b>
積分計算の実例 .....	86
積分計算の応用 .....	87
積分計算時の諸注意 .....	87
<b>3-4. <math>\Sigma</math>計算</b> .....	<b>89</b>
$\Sigma$ 計算の実例 .....	90
$\Sigma$ 計算の応用 .....	91
$\Sigma$ 計算時の諸注意 .....	91

## **第4章 複素数計算**

<b>4-1. 複素数計算を行なう前に</b> .....	<b>94</b>
<b>4-2. 複素数計算の仕方</b> .....	<b>95</b>
加減乗除計算 .....	95
逆数、平方根、自乗計算 .....	95
絶対値／偏角計算 .....	95
共役複素数を求める .....	96
実数部／虚数部の抽出 .....	97
<b>4-3. 複素数計算時の諸注意</b> .....	<b>97</b>

## **第5章 漸化式計算**

<b>5-1. 漸化式計算を行なう前に</b> .....	<b>100</b>
<b>5-2. 漸化式計算の実例</b> .....	<b>102</b>
<b>5-3. 漸化式計算時の諸注意</b> .....	<b>105</b>

## **第6章 2進・8進・10進・16進計算**

<b>6-1. 2進・8進・10進・16進計算を行なう前に</b> .....	<b>108</b>
<b>6-2. 基数の指定 (2進・8進・10進・16進数指定)</b> .....	<b>110</b>
入力数値の基数指定 .....	110
各基数で扱える数値 .....	111
<b>6-3. 加減乗除計算</b> .....	<b>111</b>
<b>6-4. 負数計算および論理演算</b> .....	<b>112</b>
負数計算 .....	112
論理演算 .....	113
<b>6-5. 2進・8進・10進・16進計算時の諸注意</b> .....	<b>114</b>

## 第7章 統計計算

7-1. 1変数統計計算 (標準偏差計算) .....	116
7-2. $t$ 検定計算 .....	120
7-3. 2変数統計計算 (回帰計算) .....	122
2変数統計計算(直線回帰計算)の仕方 .....	122
2変数統計計算(直線回帰計算)の応用 .....	127

## 第8章 数式記憶機能

8-1. 数式記憶機能の実例 .....	136
8-2. コメント文の表示 .....	139
8-3. テーブル機能 .....	140
8-4. ソルブ機能 .....	142
8-5. 数式の保存/呼び出し .....	145
数式を保存する .....	145
数式を呼び出す .....	146

## 第9章 プログラム機能

9-1. プログラム機能とは .....	148
9-2. プログラミングする前に .....	148
9-3. プログラミングの実例 .....	149
ファイル名を登録する .....	150
プログラムを実行する .....	152
9-4. プログラムのデバッグ (間違いを直す) .....	154
9-5. バイト数の数え方 .....	155
9-6. ファイル名の検索 .....	157
ファイル一覧表示から検索する(シーケンシャルサーチ) .....	157
ファイル名の読み入力により検索する(ダイレクトサーチ) .....	158
9-7. ファイル名またはプログラムの訂正 .....	160
ファイル名を訂正する .....	160
プログラムを訂正する .....	161
9-8. ファイル名 (プログラム) の消去 .....	165
指定したファイル名のみ消去する .....	165
すべてのファイル名を消去する .....	166

<b>9-9. 便利なプログラム命令</b> .....	<b>167</b>
プログラム命令メニュー表示 .....	167
変数入力命令 .....	169
変数確定命令 .....	169
ジャンプ命令 .....	169
サブルーチン .....	174
ポーズ命令 .....	178

## **第10章 ライブラリー編**

---

1. 素因数分解 .....	180
2. 最大公約数 .....	183
3. 最小損失整合 .....	186

## **巻末資料**

---

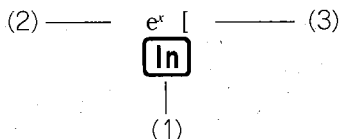
1. エラーメッセージ一覧表 .....	190
2. 関数の入力範囲と精度 .....	192
3. 仕 様 .....	195
■ 保証・アフターサービスについて .....	<b>197</b>
■ サービスセンター .....	<b>198</b>
■ カシオ電卓保証書 .....	<b>199</b>



# 操作の流れ (必ずお読みください)

## 1. キーの見方

本機では、1つのキーに対して複数の機能を持つキーがいくつかあります。たとえば、次のキーは3つの機能を持っています。



機能の使い方は、以下のとおりです。

機 能	使 い 方
ln	直接押します。
e <sup>x</sup>	<b>[SHIFT]</b> を押してから押します。
[	<b>[ALPHA]</b> を押してから押します。

また、キーに印刷されている機能は、以下のように色分けされています。

色	使 い 方
オレンジ色	<b>[SHIFT]</b> を押してから使う機能
赤 色	<b>[ALPHA]</b> を押してから使う機能

## 2. モードの指定

本機では、まず初めに分類されたメニューから計算モードを指定します。

(1) **[MODE]**を押して、メインメニュー表示を呼び出します。

**[MODE]**

1. COMP	2. BASE-N
3. SD	4. LR
5. PROG	6. an
7. CONT	8. RESET

(2) 指定したいモードに該当する数字キー(①～⑧)を押します。数字キーを押したモードの状態表示になります。

各モードの意味は、以下のとおりです。

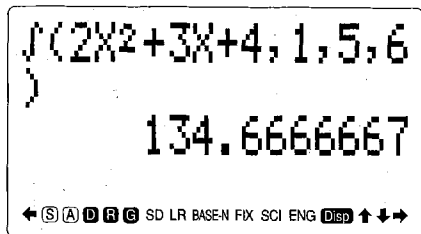
モード	意 味
COMP	関数計算を含む一般の計算を行なうときに指定します。
BASE-N	2進・8進・10進・16進の計算や変換、および論理演算を行なうときに指定します。
SD	1変数統計計算(標準偏差計算)を行なうときに指定します。
LR	2変数統計計算(回帰計算)を行なうときに指定します。
PROG	プログラミングエリアにファイル名の登録・プログラムの書き込み・実行を行なうときに指定します。
an	漸化式計算を行なうときに指定します。
CONT	表示濃度(コントラスト)を濃くしたり薄くするときに指定します。
RESET	本機を初期化(リセット)するときに指定します。

<b>COMP</b>	:COMPute(計算する)の略
<b>SD</b>	:Standard Deviation(標準偏差)の略
<b>LR</b>	:Linear Regression(直線回帰)の略
<b>PROG</b>	:PROGram(プログラム)の略
<b>CONT</b>	:CONTRast(表示濃度)の略

### 3. 表示の見方

#### ■表示シンボル

計算機が現在どのような状態にあるのかを、表示シンボルにより確認することができます。



- (S) : [SHIFT] が押されたことを示します。
- (A) : [ALPHA] が押されたことを示します。
- (D) : 角度単位が「度」に指定されていることを示します。
- (R) : 角度単位が「ラジアン」に指定されていることを示します。
- (G) : 角度単位が「グレード」に指定されていることを示します。
- SD : SDモードに指定されていることを示します。
- LR : LRモードに指定されていることを示します。
- BASE-N : BASE-Nモードに指定されていることを示します。
- FIX : 小数点以下指定を行なっていることを示します。
- SCI : 有効桁数指定を行なっていることを示します。
- ENG : ENGモードに指定されていることを示します。
- (Disp) : 演算の途中結果を表示していることを示します。
- $\uparrow$   $\downarrow$  : ファイル一覧表示または数式記憶機能の変数入力状態において、画面の下にファイル名または変数があるときは“ $\downarrow$ ”を、画面の上にファイル名または変数があるときは“ $\uparrow$ ”を点灯します。
- $\leftrightarrow$  : 1行に入力した文字が表示桁(16文字)を超えていることを示します。左に文字があるときは“ $\leftarrow$ ”を、右に文字があるときは“ $\rightarrow$ ”を点灯します。

## ■指数表示

計算結果は通常10桁で表示します。しかし、計算途中または答えがある範囲を超えると、自動的に指数表示となります。

指数表示の範囲は次の2種類があり、切り替えることができます。

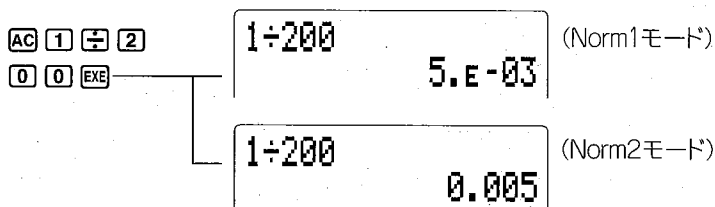
(A)  $10^{-2}$  (0.01)  $> |x|, |x| \geq 10^{10} \dots$  Norm1モード

(B)  $10^{-9}$  (0.000000001)  $> |x|, |x| \geq 10^{10} \dots$  Norm2モード

指数表示範囲の切り替えは **FUNCTION** **6** (DSP/CLR) **3** (Norm) と操作した後、**1** または **2** を押します。

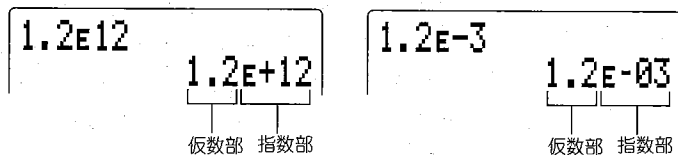
**1** を押すとNorm1モードに、**2** を押すとNorm2モードに指定されます(33ページ参照)。

現在の状態はシンボルにて表示しませんので、次の操作例を参考にして、確認してください。



これから先の計算例は、特に断りがない限りNorm1モードで行ないます。

次に、指数表示の見方を説明します。



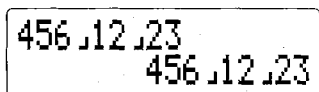
→  $1.2 \times 10^{12}$  → 1,200,000,000,000

→  $1.2 \times 10^{-3}$  → 0.0012

## ■特殊な表示

分数計算時や16進数計算時などは、通常の表示とは異なった表示の仕方を行います。

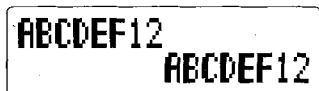
### ●分数表示例



456 12 23  
456 12 23

[ $456 \frac{12}{23}$  を表示]

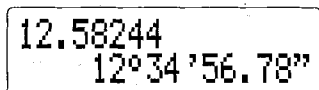
### ●16進数表示例



ABCDEF12  
ABCDEF12

[ $ABCDEF12_{(16)}$   
 $= -1412567278_{(10)}$  を表示]

### ●60進数表示例



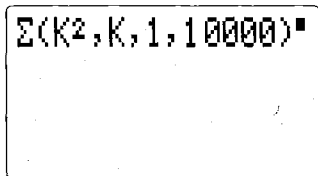
12.58244  
12°34'56.78"

[ $12^{\circ}34'56.78''$  を表示]

※ これ以外に本機では限られた画面上でさまざまな数式を表現するため、本機独自の記号など準備しています。詳しくは、それぞれの操作説明の項に記載します。

## ■演算実行中表示

複雑なプログラムを実行したときなど、実行結果を表示するのに時間がかかる場合は、表示右上に演算実行中を示すシンボル“■”が点灯します。



Σ(K², K, 1, 10000)■

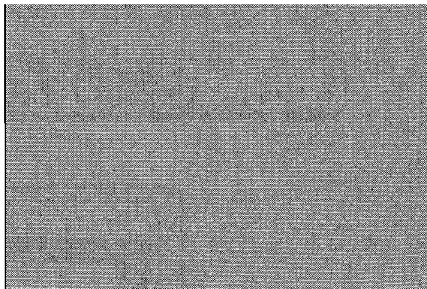
## 4. 故障かと思われる前に

予期せぬ演算結果が表示されたりエラーが表示された場合は以下の操作を行なって、一度標準状態(初期状態)に戻してください。

- (1) **MODE** **1** と操作して、COMPメニューを指定します。
- (2) **FUNCTION** **5** (DRG) と操作して角度単位メニュー表示を呼び出した後、**1** (Deg) を押してDegモードを指定します。
- (3) **FUNCTION** **6** (DSP/CLR) と操作して表示形式/クリアメニュー表示を呼び出した後、**3** (Norm) **1** と操作してNormモードを指定します。

その後、計算内容を再度確認し、正しいモードを指定してから再度演算を行なってください(15ページ参照)。

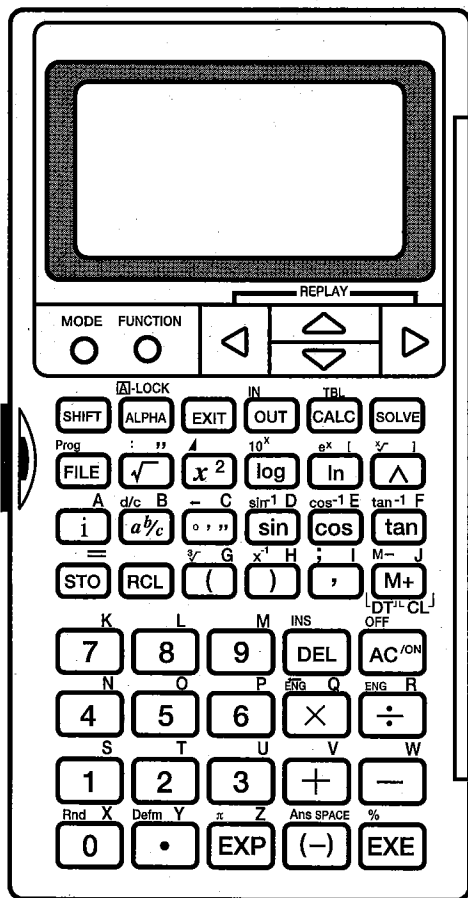
# 1



## 本体構成と使い方



# 1-1. 各部の名称とその説明





## キーの働き

MODE

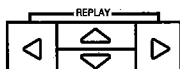
### ○ モードキー

計算機のモード状態を指定するとき、**[1]**～**[8]**の数字キーと合わせて押します。詳しくは「モードの指定」(15ページ)をご覧ください。

FUNCTION

### ○ 機能メニューキー

機能メニュー表示を呼び出します。



### カーソル移動/リプレイキー

- カーソルを上下左右に移動させます。  
**[▲]**は上に、**[▼]**は下に、**[◀]**は左に、**[▶]**は右にカーソルを移動させます。これらのキーを押し続けると、カーソルを連続的に移動させることができます。
- 計算式や数値を入力して**[EXE]**を押した後に**[◀]**または**[▶]**を押すと、計算式が表示され、カーソルが点滅します。この状態から、再び計算したり数値などを変えて計算し直すことができます。詳しくは「リプレイ機能」(60ページ)をご覧ください。

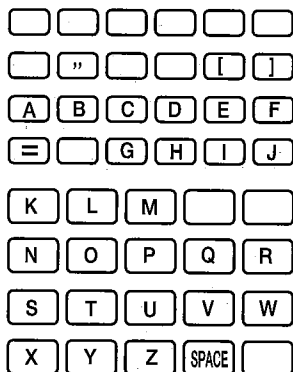
### **[SHIFT]** シフトキー

キーパネル上にオレンジ色で記されている関数命令や機能を使うときに押します。再度**[SHIFT]**を押すと、**[SHIFT]**を押す前の状態に戻ります。

**[ALPHA]** LOCK

### **[ALPHA]** アルファベットキー

- アルファベットや特殊記号を表示させるときに押します。再度**[ALPHA]**を押すと、アルファ文字入力状態は解除されます。
- **[SHIFT]**に続けて押すとアルファ文字入力状態が固定され、連続してアルファベット文字が入力できます。固定状態を解除するときには、再度**[ALPHA]**を押してください。



### **EXIT** エグジットキー

次のような表示から抜けるときに押します。

- ・ 機能メニュー表示から抜けるとき。
- ・ プログラム書き込み表示から抜けるとき。
- ・ 数式記憶機能／テーブル機能／ソルブ機能／漸化式計算機能を終了するとき。

### <sup>IN</sup>**OUT** 数式記憶機能キー

数式記憶機能を使って計算するときに使います。詳しくは「数式記憶機能」(136ページ)をご覧ください。

### <sup>TBL</sup>**CALC** 数式記憶機能／テーブル機能キー

- 記憶した数式を使って計算するときに押します。詳しくは「数式記憶機能」(136ページ)をご覧ください。
- 数式記憶機能の数式の1つの変数に対してレンジ(変数の条件)を設定して、繰り返し計算を行なうとき、**SHIFT**に続けて押します。詳しくは「テーブル機能」(140ページ)をご覧ください。

### **SOLVE** ソルブ機能キー

数式記憶機能の数式を使って、ニュートン法を基に変数の解を求めるときに使います。詳しくは「ソルブ機能」(142ページ)をご覧ください。

Prog

**FILE** ファイル／プログラム命令キー

- 登録されたファイル名を呼び出します。
- COMP/BASE-N/SD/LRモード指定時に **SHIFT** **Prog** "ファイル名" **EXE** と操作すると、プログラムを実行します。詳しくは「プログラムを実行する」(152ページ)をご覧ください。

**√** ルート／マルチステートメントキー

- 数値の前に押すと、数値の平方根を求めます。
- **SHIFT** に続けて押すと、プログラム計算や連続計算において計算式や命令の区切りに使う記号「:」(コロン)を入力します。この記号を書き込むことをマルチステートメントといいます(62ページ参照)。

**x<sup>2</sup>** 2乗／出力命令キー

- 数値の後に押すと、数値の2乗(自乗)を求めます。
- **SHIFT** に続けて押すと、プログラム計算や連続計算の計算結果を出力(表示)する命令「▲」を入力します。

10<sup>o</sup>**log** ログ／10の $x$ 乗キー

- 数値の前に押すと、数値の常用対数(10を底とする対数)を求めます。
- 数値の前に **SHIFT** に続けて押すと、10の数値乗を求めます。

e<sup>o</sup>**ln** エルエヌ／ $e$ の $x$ 乗キー

- 数値の前に押すと、数値の自然対数( $e$ を底とする対数)を求めます。
- 数値の前に **SHIFT** に続けて押すと、 $e$ の数値乗を求めます。

x<sup>o</sup>**△** べき乗／べき乗根キー

- $x$ (任意の数)と $y$ (任意の数)の間で押すと、 $x$ の $y$ 乗を求めます。
- $x$ (任意の数)と $y$ (任意の数)の間で **SHIFT** に続けて押すと、 $y$ の $x$ 乗根を求めます。

## **i** 虚数置数キー

複素数の虚数単位「i」を置数するときに使います。

## **d/c** **B** **2/3** 分数キー

● 分数および帯分数を置数するときに使います。

(例)  $\frac{23}{45} \rightarrow 23$  **2/3** 45、 $2\frac{3}{4} \rightarrow 2$  **2/3** 3 **2/3** 4

● 分数を表示しているときに押すと、分数を小数表示にします。

● 分数を表示しているときに **SHIFT** に続けて押すと、帯分数を仮分数表示にします。

## **° ' "** **C** **2/3** 度・分・秒(60進数)置数/10進数⇒60進数変換キー

● 度・分・秒(時・分・秒)のような60進数を置数するときに使います。

(例)  $78^{\circ}45'12'' \rightarrow 78$  **2/3** 45 **2/3** 12 **2/3**

● **SHIFT** に続けて押すと、10進数の数値を度・分・秒(60進数)に変換します。

## **sin<sup>-1</sup>** **D** **cos<sup>-1</sup>** **E** **tan<sup>-1</sup>** **F** **sin** **cos** **tan** 三角関数/逆三角関数キー

● 数値の前に押すと、数値の三角関数(sin, cos, tan)を求めます。

● 数値の前に **SHIFT** に続けてそれぞれ押すと、数値の逆三角関数(sin<sup>-1</sup>, cos<sup>-1</sup>, tan<sup>-1</sup>)を求めます。

◎ BASE-Nメニューの16進数指定時に **i** ~ **tan** を押すと、A<sub>(16)</sub> ~ F<sub>(16)</sub> (10<sub>(10)</sub> ~ 15<sub>(10)</sub>)が入力されません。

## **STO** ストアメモリーキー

計算式を実行した結果を変数メモリーに記憶したいとき、アルファベットの前に押します。

## **RCL** リコールメモリーキー

変数メモリーに記憶した数値を表示したいとき、アルファベットの前に押します。

## <sup>M G X H</sup> [ ] [ ] カッコ/立方根/逆数キー

- 計算式の開きカッコの位置で [ ] を、閉じカッコの位置で [ ] を押し  
ます。
- 数値の立方根を求めるときは数値の前に [SHIFT] に続けて [ ] を、数値の逆  
数を求めるときは数値の後に [SHIFT] に続けて [ ] を押し  
ます。

## [ ] カンマ/セミコロンキー

- 「,」(カンマ)を入力します。
- [SHIFT] に続けて押すと、「;」(セミコロン)を入力します。

## <sup>M- J</sup> <sup>LDP LCL</sup> [ ]+ [ ]- メモリープラス/メモリーマイナス/データ入力/データ削除キー

- 表示されている数値をメモリーに加算(合計)したいとき押します。なお、  
計算式が表示されているときは、計算結果がメモリーに記憶されます  
([EXE]の働きを兼ねそなえています)。
- 表示されている数値をメモリーから減算(マイナス合計)したいとき、  
[SHIFT] に続けて押します。
- SDモード([MODE] [3] (SD))・LRモード([MODE] [4] (LR))のときは、データ入  
力/データ削除キーとして働きます。
  - ・ [DT] ..... <データ[DT]>と押します。
  - ・ [SHIFT] [CL] ..... 入れ間違えたデータを削除するとき、上例  
の[DT]の代わりに[SHIFT] [CL]と押します。

## <sup>Rnd X</sup> [0] ~ <sup>M</sup> [9]、<sup>Defm Y</sup> [.] 置数キー(数字キー、小数点キー)

- 置数(計算機に数値を入れること)のとき、数の上桁より順に押します。  
[.] は、小数点の位置で押します。  
★置数は最大10桁までです。
- [SHIFT] に続けて [0] または [.] を押すと、以下の機能になります。
  - ・ [SHIFT] [0] (Rnd) ..... 内部数値丸め  
内部数値の有効桁数11桁目を四捨五入して丸めます(Ansメモ  
リー内容も同様に丸めます)。また、FIX・SCIモードのときは内部数  
値を切り捨てし、設定により表示されている数値(または有効桁数  
分)と同じ数値にします。

- **SHIFT** **□** (Defm) .....メモリーの増設  
メモリーの数を標準状態の26個より増やします。

INS

## **DEL** デリート／インサートキー

- カーソルと文字が点滅している箇所を削除します。カーソルが末尾に点滅しているときは、カーソルの左側の文字を削除します。
- **SHIFT** に続けて押すと、カーソルが「**□**」(インサートカーソル)に切り替わり、その位置に文字(命令)を挿入することができます。

OFF

## **AC/ON** オールクリアー／電源ON／電源OFFキー

- 表示されている計算式や数値をすべて消したり、エラーを解除します。
- 電源をONにします。
- **SHIFT** に続けて押すと、電源をOFFにします。

V

W ENG Q ENG R

## **+** **-** **×** **÷** 四則計算／エンジニアリングキー

- 加減乗除を計算するとき、書式通りにそれぞれの位置で押します。
- **-** は負符号として使うことができます。
- **SHIFT** に続けて押すと、以下の機能になります。
  - **SHIFT** **×** (ENG) ..... エンジニアリング(指数表示変換)  
表示されている数値を、指数部が3の倍数(正の数)となる指数表示に変換します。  
(3の倍数<正の数):  $10^3 = k$ ,  $10^6 = M$ ,  $10^9 = G$
  - **SHIFT** **÷** (ENG) ..... エンジニアリング(指数表示変換)  
表示されている数値を、指数部が3の倍数(負の数)となる指数表示に変換します。  
(3の倍数<負の数):  $10^{-3} = m$ ,  $10^{-6} = \mu$ ,  $10^{-9} = n$ ,  $10^{-12} = p$

$\pi$  Z

## **EXP** エキスポーネント(指数部置数)／パイキー

- 指数部を置数するとき、仮数部を置数した後に押します。  
(例)  $2.56 \times 10^{34} \rightarrow 2.56 \text{EXP} 34$
- ★ 指数部の置数は、最大±99までです。それ以上入力すると、エラー「Syn ERROR」となります。

- **SHIFT** に続けて押すと、「 $\pi$ 」(円周率)を入力します。

Ans SPACE

## **←** 負符号/アンサーキー

- 負数を置数するとき、数値の前に押します。
- **SHIFT** に続けて押すと、**EXE** を押して求めた最新の演算結果を呼び戻します。

## **%** **EXE** エグゼキュート/パーセントキー

- 計算の答えを求めます。また、データ入力時や、結果表示後に次の実行に移ります。
- パーセント計算を行なうとき、**SHIFT** に続けて押します。

## コントラスト(表示濃度)調整

表示が薄くて見にくいときは以下のように操作して、コントラスト(表示濃度)を調整してください。

(1) メインメニュー表示から **7** (CONT) を押します(15ページ参照)。

- (2) **◀** または **▶** を押します。
- ・ **◀** ..... 表示が薄くなります。
  - ・ **▶** ..... 表示が濃くなります。

```
*** CONTRAST ***  
  
LIGHT          DARK  
[←]           [→]
```

なお、上記のキーを押し続けることにより、連続して濃度を調整することができます。

※ **MODE** を押すとコントラスト調整は解除され、メインメニュー表示に戻ります。

## 1-2. 計算を始める前に

### 機能メニュー表示

本機では、計算を行なう前に角度単位／表示形式モードを指定する必要があります。

この指定は **FUNCTION** を押して、機能メニュー表示を呼び出して行ないます。

#### 例 1 COMPモード指定時の機能メニュー表示

```
1.MATH 2.COMPLX
3.PROG 4.CONST
5.DRG 6.DSP/CLR
```

#### 例 2 SD/LRモード指定時の機能メニュー表示

```
1.MATH 2.COMPLX
3.PROG 4.CONST
5.DRG 6.DSP/CLR
7.STAT 8.RESULTS
```

※ 現在指定しているモードによって、表示内容は異なります。詳しくは各モードの章をご覧ください。

#### (1) 内蔵関数 (MATH) メニュー

キーやパネル上に印刷されていない関数命令を呼び出します。詳しくは48ページで説明します。

#### (2) 複素数計算 (COMPLX) メニュー

複素数計算に使う命令を呼び出します。詳しくは94ページで説明します。

#### (3) プログラム命令 (PROG) メニュー

特別なプログラム命令を書き込みます。詳しくは167ページで説明します。

#### (4) 科学定数計算 (CONST) メニュー

科学定数計算に使う定数を呼び出します。詳しくは73ページで説明します。



#### (5) 角度単位(DRG)メニュー

どの角度単位で計算するかを指定します。

#### (6) 表示形式/クリアー(DSP/CLR)メニュー

表示される桁数を指定したり、Eng記号を使った表示を指定します。また、メモリーエリアを指定して、メモリーに記憶されている内容を消去します。

#### (7) 統計計算(STAT)メニュー

データの統計処理に使う命令を呼び出します。詳しくは117ページで説明します。

#### (8) 統計計算演算結果(RESULTS)メニュー

1変数/2変数統計計算の演算結果を表示します。詳しくは118ページで説明します。

### ■ 角度単位(DRG)メニュー

⑤ (DRG)

1. Deg	2. Rad
3. Gra	4. °
5. r	6. g

メニューの意味は、以下のとおりです。

- ・「1. Deg」..... 角度単位を「度数法(ディグリー単位)」に指定。
- ・「2. Rad」..... 角度単位を「弧度法(ラジアン単位)」に指定。
- ・「3. Gra」..... 角度単位は「グラード単位」に指定。
- ・「4. °」..... 入力数値の角度単位を「度数法」に変換。
- ・「5. r」..... 入力数値の角度単位を「弧度法」に変換。
- ・「6. g」..... 入力数値の角度単位を「グラード単位」に変換。

※ 各角度単位については、次の表を参照してください。

度数法(ディグリー単位)	360°	90°
弧度法(ラジアン単位)	$2\pi$	$\frac{\pi}{2}$
グラード単位	400	100

**例** 角度単位を「度数法 (ディグリー単位)」に指定する。

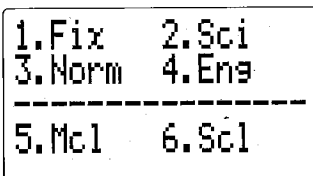
**5** (DRG) **1** (Deg)



シンボルが点灯します。

## ■ 表示形式 / クリアー (DSP / CLR) メニュー

**6** (DSP / CLR)



メニューの意味は、以下のとおりです。

- ・「**1. Fix**」..... 小数点以下の桁数を何桁まで表示するのかを指定。
- ・「**2. Sci**」..... 有効桁数 (上位より何桁) を指定。
- ・「**3. Norm**」..... 指数表示の切り替え範囲を指定。
- ・「**4. Eng**」..... Eng記号を使って演算結果を表示。
- ・「**5. Mcl**」..... すべての変数メモリー内容を消去。
- ・「**6. Scl**」..... 統計計算用メモリー内容を消去。

※ 詳しくは「小数点以下指定、有効桁数指定、および指数表示範囲指定」(54ページ)をご覧ください。

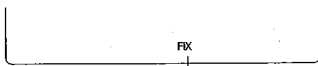
### ● 小数点以下指定 (Fix モードの指定)

**例** 小数点以下を 2 桁に指定する。

**6** (DSP / CLR)

**1** (Fix) **2**

指定したい小数点以下桁数 ( $n = 0 \sim 9$ )



シンボルが点灯します。

※ 小数点以下指定を行なうと、指定桁の次の桁を四捨五入して表示されます。

※ 小数点以下指定は、Normモードに指定するまで解除されません。

## ● 有効桁数指定 (Sci モードの指定)

例 有効桁数を 3 桁に指定する。

6 (DSP/CLR)

2 (Sci) 3

指定したい有効桁数 ( $n=0\sim 9$ )



シンボルが点灯します。

※ 有効桁数指定を行なうと、指定桁の次の桁を四捨五入して表示されます。

※  $n=0$ のときは、有効桁数は10桁になります。

※ 有効桁数指定は、Normモードに指定するまで解除されません。

## ● 指数表示範囲指定 (Norm1 / Norm2 モードの指定)

2つの指数表示範囲があります。

・ Norm1モード .....  $10^{-2}$ 未満または $10^{10}$ 以上の数値を指数表示にします。

・ Norm2モード .....  $10^{-9}$ 未満または $10^{10}$ 以上の数値を指数表示にします。

この範囲は、以下のように操作して指定します(シンボルは点灯しません)。

6 (DSP/CLR) 3 (Norm) 1

指定したいモード ( $n=1\sim 2$ )

## ● Eng 記号表示指定 (Eng モードの指定)

以下の操作をするごとに、

Engモード → NON-Engモード → Engモード → ……

と交互に切り替わります。

6 (DSP/CLR)

4 (Eng)



シンボルが点灯します。

※ 使われるEng記号は、次の9種類です。

Eng記号	単 位
T(テラ)	$10^{12}$
G(ギガ)	$10^9$
M(メガ)	$10^6$
k(キロ)	$10^3$
m(ミリ)	$10^{-3}$
μ(マイクロ)	$10^{-6}$
n(ナノ)	$10^{-9}$
p(ピコ)	$10^{-12}$
f(フェムト)	$10^{-15}$

※ Eng記号は、仮数部が1以上1000未満になる記号を選択して表示します。

### ●すべての変数メモリー（A～Z）の消去

**6** (DSP/CLR)

Mcl

**5** (Mcl) **EXE**

0

※ 変数メモリー(A～Z)および増設メモリーの値をクリアします。

### ●統計計算用メモリー（P、Q、R、U、V、W）の消去

**6** (DSP/CLR)

Sc1

**6** (Sc1) **EXE**

0

※ SDモードではU,V,Wのメモリーをクリアします。

## 計算式の入力(書式通り入力方式)

本機は紙に書いた通りに計算式を入力し、**EXE**を押すだけで計算できる「書式通り入力方式」を採用しています。加減乗除、関数、およびカッコの優先順位は、計算機が自動的に判断します。

**例**  $2(5+4) \div (4 \times 3) =$

**AC** **2** **(** **5** **+** **4** **)** **÷**

**(** **4** **×** **3** **)** **EXE**

$2(5+4) \div (4 \times 3)$

1.5

# 計算の優先順位

計算には「優先順位」という規則があり、加算・減算より乗算・除算の方を先に計算することになっています。また、関数の方が優先され、関数の中でも前置関数より後置関数の方が優先されます。

本機は、以下のような優先順位を自動的に判別します。

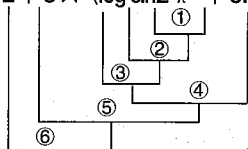
- (1) 座標変換・ $t$ 検定  $\text{Pol}(x, y), \text{Rec}(r, \theta), t(\text{微分} \cdot \text{二次微分} \cdot \text{積分} \cdot \Sigma \text{計算}) \quad d/dx, d^2/dx^2, \int dx, \Sigma$
- (2) 後置関数  $x^2, x^1, x!$   
60進数置数 “ ”  
Eng記号
- (3) べき乗・べき乗根  $\wedge(x^y), \sqrt[n]{x}$
- (4) 分数  $a \frac{b}{c}$
- (5)  $\pi$  やメモリ、変数の直前の省略乗算、漸化式、科学定数  $2\pi, 5A, \pi R, 2mp$  など
- (6) 前置関数  $\sqrt{\quad}, \sqrt[3]{\quad}, \log, \ln, e^x, 10^x, \sin, \cos, \tan, \sin^{-1}, \cos^{-1}, \tan^{-1}, \sinh, \cosh, \tanh, \sinh^{-1}, \cosh^{-1}, \tanh^{-1}, (-),$   
(以下BASE-Nモードのみ)  $d, h, b, o, \text{Neg}, \text{Not}$
- (7) 前置関数の直前の省略乗算  $2\sqrt{3}, A\log 2$  など
- (8) 順列・組み合わせ  $nPr, nCr$
- (9)  $\times, \div$
- (10)  $+, -$
- (11) and    BASE-Nモードのみ
- (12) or, xor, xnor

※ 同順位の関数が連続しているときは右側から左側へ ( $e^{\ln \sqrt{120}} \rightarrow e^{\ln(\sqrt{120})}$ )、他は左側から右側へ計算されます。

※ 複合関数は右側から左側へ計算されます。

※ カッコを使うと、カッコ内が最優先されます。

**例**  $2 + 3 \times (\log \sin 2 \pi^2 + 6.8) = 22.07101691$  (Radモード指定時)



## 乗算記号(×)の省略

以下の場合、乗算記号(×)が省略できます。

### (1) 次の関数の前

sin, cos, tan,  $\sin^{-1}$ ,  $\cos^{-1}$ ,  $\tan^{-1}$ , sinh, cosh, tanh,  $\sinh^{-1}$ ,  $\cosh^{-1}$ ,  $\tanh^{-1}$ ,  
log, ln,  $10^x$ ,  $e^x$ ,  $\sqrt[3]{\quad}$ , Pol(x,y), Rec(r,  $\theta$ )

例  $2\sin 30$ ,  $2\sqrt{3}$  など

### (2) 定数、変数、メモリーの前

例  $2\pi$ , 2AB, 3Ans など

### (3) 開きカッコの前

例  $3(5+6)$ ,  $(A+1)(B-1)$  など

## スタック数

本機には、優先順位の低い計算数値や計算命令(関数など)を一時的に記憶するためのスタックメモリーと呼ばれるメモリーがあります。

数値用スタックは10段、命令用スタックは26段まであります。

このスタック数を超えるような複雑な計算式を実行すると、エラー「Stk ERROR」となります。

例

スタックの数え方

$$2 \times ( ( 3 + 4 \times ( 5 + 4 ) \div 3 ) \div 5 ) + 8 =$$



数値用スタック

①	2
②	3
③	4
④	5
⑤	4
⋮	

命令用スタック

①	×
②	(
③	(
④	+
⑤	×
⑥	(
⑦	+
⋮	

※ 計算は優先順位の高い順に実行され、スタックの中から消えていきます。

## 入出力桁数と演算範囲

本機により入出力できる範囲(入出力範囲)は、仮数部10桁、指数部2桁です。

ただし、内部では仮数部15桁、指数部2桁で計算を行なっています。

例  $3 \times 10^5 \div 7 - 42857 =$

AC 3 EXP 5  $\div$  7 EXE

3 EXP 5  $\div$  7 =

4 2 8 5 7 EXE

3E5÷7  
42857.14286  
3E5÷7-42857  
0.1428571428

## 桁オーバーとエラー

計算範囲を超えて使用されたり誤った入力を行なうと、表示窓に“○○○ERROR”とエラーメッセージが表示されます。

エラーメッセージが表示されるのは、以下のときです。

- (1) 計算途中や答え、またはメモリー内の数値が $\pm 9.999999999 \times 10^{99}$ を超えたとき。
- (2) 関数計算において、192ページの被演算数の範囲を超えて計算しようとしたとき。
- (3) 統計計算において、適切でない操作が行なわれたとき。

例  $n=0$ で $\bar{x}$ や $x\sigma n$ を求めようとしたとき。

- (4) 数値用スタックや命令用スタックを超えて計算しようとしたとき。

例  $\square$ を続けて25回押した後、2 $\square$ 3 $\square$ 4 $\square$ の計算をしたとき。

- (5) 書式上誤った入力をして、EXEを押したとき。

例 5 $\square$  $\square$ EXEと操作したとき。

- (6) メモリーを増設していないのに、Z[2]のように誤ったメモリー指定をしたとき(メモリーの増設については、45ページを参照してください)。
- (7) 引き数を必要とする命令や関数で、適切でない引き数の値を入力したとき。

例 プログラム実行中に、Fix、Sciで0~9以外の値を入力したとき。

以上のようなとき、次のようなエラーメッセージを表示します。

(1)~(3)のときは	.....	Ma ERROR
(4)のときは	.....	Stk ERROR
(5)のときは	.....	Syn ERROR
(6)のときは	.....	Mem ERROR
(7)のときは	.....	Arg ERROR

エラーメッセージが表示されると、計算ができなくなります。この状態を解除するときは、**AC**を押してください。

その他のエラーメッセージとして、「**Ne ERROR**」と「**Go ERROR**」があります。詳しくは「エラーメッセージ一覧」(190ページ)をご覧ください。

## 入力文字数(バイト数)

本機は一度に計算できる数値や計算命令を覚えておくためのエリアとして、127バイトのエリアを持っています。

数字や **+** **-** **×** **÷**などは、1つのキー操作で1バイトと数えます。また、2つ以上のキー操作をしても1機能のものは、1バイトと数えます(**SHIFT** **☒**など)。

※ 文字(命令)は127バイトまでしか入力できません。121バイト目以降の入力になると、カーソルが“—”から“■”に変わります。もし入力していてカーソルが“■”になったときは、区切りの良いところで一度計算を終わらせてください。

※ 数値や計算命令を入力していきまると、数値や計算命令は表示窓の左から表示されます。ただし、計算結果は右詰めで表示されます。



## 訂正の仕方

### ● 計算式を入力しているとき、押し間違いに気付いた場合

◀ または ▶ を使って間違った箇所にカーソルを合わせ、正しいキーを押します。

例 sin 60 を cos 60 と押してしまった。

cos 6 0

cos 60\_

◀ ◀ ◀

cos 60

sin

sin 60

※ 押し間違いを訂正した後に **EXE** を押すと、答えを求めることができます。

### ● 間違っって不要なキーを押してしまったとき

◀ または ▶ で不要な箇所にカーソルを合わせ、**DEL** を押します。

例  $36 \times 2$  を  $36 \times \times 2$  と押してしまった。

3 6 ~~x~~ ~~x~~ 2

36xx2\_

◀ ◀ **DEL**

36x2

### ● 入力した計算式の途中で挿入したいとき

◀ または ▶ で挿入したい箇所にカーソルを合わせ、**SHIFT** **INS** と操作します。

例  $2^2$  を  $\sin 2^2$  としたい。

2  $x^2$

2<sup>2</sup>\_

◀ ◀

2<sup>2</sup>

**SHIFT** **INS**

2<sup>2</sup>

sin

sin 2<sup>2</sup>

- ※ **SHIFT** **INS**と操作すると、「□」(インサートカーソル)が点滅します。このときキーを押すと、キー内容が□の位置に挿入されます。  
この状態を解除するときは再度**SHIFT** **INS**と操作するか、**◀▶**または**EXE**を押してください。

## メモリー

本機で使えるメモリーは標準状態で26個あり、仮数部15桁、指数部2桁までの数値を記憶します。

このメモリーには、以下の2種類があります。

- 変数メモリー ..... **STO**、**RCL**とアルファベットA~Zを組み合わせて使用するメモリー
- 独立メモリー ..... **M+**、**SHIFT M-**、**RCL M**を組み合わせて使用するメモリー

メモリーは両方とも不揮発性ですので、電源をOFFにしてもメモリー内容は保護されます。

※ 変数メモリーMと独立メモリーは同じメモリーエリアを使用しています。

### ■ 変数メモリー

数値を同時に26個記憶させることができ、必要に応じて呼び出すことができます。

**例 1**    Aメモリーに123を記憶する。

**AC** **1** **2** **3** **STO** **A**

123	
A=	123

**AC** **RCL** **A**

A=	123
----	-----

計算式を入力したときは、計算式の結果がメモリー内に記憶されます。

**例 2** Bメモリーに  $123 \times 456$  の計算結果を記憶する。

AC 1 2 3 X 4 5 6

123x456\_

STO B

123x456  
B=  
56088

AC RCL B

B=  
56088

変数式を入力したときは、まず、その変数式に使用した変数メモリーに記憶された値により計算されます。その後、その計算結果が指定のメモリー内に記憶されます。

**例 3** Cメモリーに  $A \times B$  の計算結果を記憶する。

AC ALPHA A X ALPHA B

AxB\_

STO C

AxB  
C=  
6898824

AC RCL C

C=  
6898824

※ メモリーへの式の代入(たとえば、 $C=A \times B$ )などの操作を行なったときにエラー「Syn ERROR」となった場合、代入しようとした変数メモリーに記憶されていた内容は保護されます。

メモリーへの代入式を「変数＝計算式」の形式で入力したときは、計算式の計算結果が指定された変数のメモリー内に記憶されます。

**例 4** 「 $S = \log 2$ 」を記憶する。

AC ALPHA S ALPHA  $\equiv$  log 2 EXE

S=log 2  
0.3010299957

AC RCL S

S=  
0.3010299957

### 【注意】

以下の変数メモリーは、微分／2次微分計算時、積分計算時、および統計計算時に使われます。

計 算 内 容	変数メモリー
微分／2次微分計算	F、G、H
積分計算	K、L、M、N
1変数統計計算(SDモード)	U、V、W
2変数統計計算(LRモード)	P、Q、R、U、V、W

これらの変数メモリーは上記計算時には使用できませんので、注意してください。

### ■ 独立メモリー

メモリー内に直接加減(累計)することができます。個々の答えを求めながら同時に合計を求める合計計算に非常に便利です。

**例** 独立メモリーに 123 を記憶する。

AC 1 2 3 M+

123  
123

メモリー内容呼び出す。

AC RCL M

M=	123
----	-----

25を加え、12を引く。

2 5 M+ 1 2 SHIFT M-

25	25
12	12

※ 2 5 - 1 2 M+ と  
操作してもかまいません。

メモリー内容呼び出す。

AC RCL M

M=	136
----	-----

※ メモリー内容をクリアーするときは、0 STO M と操作してください。

※ SDモード、LRモードでは、M+、SHIFT M- によるメモリーの加減算を行なうことはできません。

### ● STO M と M+・SHIFT M- の違いについて

STO M も M+・SHIFT M- も、計算途中で操作して、その答えを直接メモリー内に記憶します。ただし、STO M の場合は、現在のメモリー内容をクリアーしてから記憶するのに対して、M+・SHIFT M- の場合は、現在のメモリー内容に加減算します。

例 1) メモリーMに123が記憶されているとき、456を STO M によりメモリーMに記憶する。

AC 1 2 3 STO M

123	
M=	123

AC 4 5 6 STO M

456	
M=	456

AC RCL M

M= 456

例 2

メモリーMに123が記憶されているとき、456を  $\boxed{M+}$  によりメモリーMに加算する。

AC 1 2 3 STO M

123  
M= 123

AC 4 5 6  $\boxed{M+}$

456  
456

AC RCL M

M= 579

## ■ 配列メモリー

配列メモリーとはA、B、X、Yのようにアルファベット1文字をメモリー名として使うものとは異なり、メモリー名(アルファベット1文字A~Z)に[1]、[2]のような添字を付けて使うメモリーのことを呼びます。

今までのメモリー

A  
B  
C

配列メモリー

A[0] B[-1]  
A[1] B[0]  
A[2] B[1]

配列メモリーを使うと添字(番号)によってメモリーを指定できるので、プログラムを短くかつ簡単にすることができます。

## ■ メモリーの増設

プログラムを記憶するエリアをメモリーに変換することによって、変数メモリーを最大476個まで増やすことができます。メモリーの増設には、1メモリーにつき10バイトを必要とします。

メモリー数	26	27	28	……	476
フリーバイト数	4500	4490	4480	……	0

※ プログラムバイト数については、155ページをご覧ください。

- メモリーの増設は、以下のように操作します。

**SHIFT** **Defm** (増設個数) **EXE**

**例**      メモリーを10個増やして、36個にする。

**SHIFT** **Defm** **10** **EXE**

```
MEMORY :    36
PROGRAM:     0

4400 Bytes Free
```

- メモリー数を標準状態の26個に戻すときは増設個数を「0」として、増設する場合と同じ操作をします。

※ メモリーを増設したい個数ほどバイト数が残っていないときは、エラー「Arg ERROR」となります。

※ 現在の増設メモリー数を確認したいときは、**SHIFT** **Defm** **EXE**と操作してください。

※ 「Defm(増設個数)」は、プログラムに書き込むこともできます。

## ■ 増設されたメモリーの使い方

増設されたメモリーは、

Z[1]=Y[2]=X[3]=...=A[26] (Defm1)

Z[2]=Y[3]=X[4]=...=A[27] (Defm2)

⋮                    ⋮                    ⋮                    ⋮

と使います。

呼び名は、変数Z[n]～変数A[n+25]  ただし、nは増設されたメモリー数となります。

例えば、メモリーを5個増設したときは26個の変数メモリーに加えて、Z[1]～Z[5]の増設メモリーが使えます。

このメモリーの使い方は、配列メモリーの場合と同じです。

**例**                    増設されたメモリーZ [2] に 123 を記憶する。

SHIFT Defm 2 EXE

```
MEMORY :        28
PROGRAM:        0

4480 Bytes Free
```

AC ALPHA Z ALPHA L 2 ALPHA 1

ALPHA = 1 2 3 EXE

```
Z[2]=123

                  123
```

メモリー内容を読み出す。

AC ALPHA Z ALPHA L 2 ALPHA 1

EXE

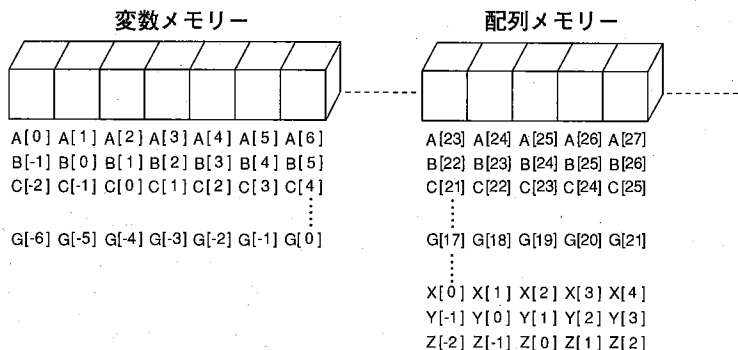
```
Z[2]

                  123
```



## ■ 配列メモリーの使用上の注意

配列メモリーはメモリー名をA～Zから選び、添字を付けることにより使い分けます。ただし、メモリーエリアとしては変数メモリーA～Zと同じエリアを使っていますので、配列メモリーを使うときはメモリーエリアが重複しないように気を付けてください。



## 内蔵関数(MATH)メニュー

COMP/SD/LR/anモード指定時は、キーおよびパネル上に印刷している関数以外に、機能メニュー表示から内蔵関数(MATH)メニューを呼び出して使う関数が備えられています。

FUNCTION

1. MATH	2. COMPLX
3. PROG	4. CONST
5. DRG	6. DSP/CLR

※COMPモード指定時

1 (MATH)

1. $\int dx$	2. $d/dx$
3. $d^2/dx^2$	4. $\Sigma()$
5. $X!$	6. Ran#
7. nPr	8. nCr

▼ | ↑  
| | |  
| | |  
▲

1. m	2. $\mu$	3. n
4. p	5. f	6. k
7. M	8. G	9. T

### ■ 積分/微分/ $\Sigma$ (シグマ)/確率計算

積分、微分/二次微分、 $\Sigma$ 計算、順列、組み合わせ、階乗、および乱数を求めることができます。

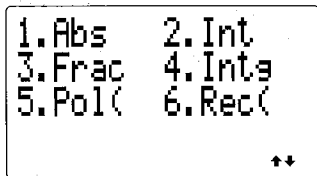
1. $\int dx$	2. $d/dx$
3. $d^2/dx^2$	4. $\Sigma()$
5. $X!$	6. Ran#
7. nPr	8. nCr

メニューの意味は、以下のとおりです。

- 「1.  $\int dx$ 」 ..... 積分計算をするときに使用。→85ページ参照
- 「2.  $d/dx$ 」 ..... 微分計算をするときに使用。→78ページ参照
- 「3.  $d^2/dx^2$ 」 ..... 2次微分計算をするときに使用。  
→82ページ参照
- 「4.  $\Sigma(\ )$ 」 .....  $\Sigma$  (シグマ)計算をするときに使用。  
→89ページ参照
- 「5.  $x!$ 」 ..... 数値の階乗を求めるとき、数値の後に使用。
- 「6. **Ran#**」 ..... 0以上1未満の擬似乱数(小数点以下10桁の値)を発生させたいときに使用。
- 「7.  $nPr$ 」 ..... 順列計算をするときに使用。
- 「8.  $nCr$ 」 ..... 組み合わせ計算をするときに使用。

## ■ 数値計算

絶対値計算、整数部や小数部の抽出、および直交座標と極座標の相互変換ができます。



メニューの意味は、以下のとおりです。

- 「1. **Abs**」 ..... 数値の絶対値を求めるとき、数値の前に使用。
- 「2. **Int**」 ..... 数値の整数部分を取り出すとき、数値の前に使用。
- 「3. **Frac**」 ..... 数値の小数部分を取り出すとき、数値の前に使用。
- 「4. **Intg**」 ..... 数値の値を超えない最大の整数値を求めるとき、数値の前に使用。
- 「5. **Pol(**」 ..... 直交座標を極座標に変換するときに使用。
- 「6. **Rec(**」 ..... 極座標を直交座標に変換するときに使用。

## ■ ハイパボリック計算

数値の双曲線関数および逆双曲線関数を求めることができます。

1. sinh	2. cosh
3. tanh	4. sinh <sup>-1</sup>
5. cosh <sup>-1</sup>	6. tanh <sup>-1</sup>

↑↓

メニューの意味は、以下のとおりです。

- 「1. sinh」..... 数値の双曲線関数「sinh」を求める。
- 「2. cosh」..... 数値の双曲線関数「cosh」を求める。
- 「3. tanh」..... 数値の双曲線関数「tanh」を求める。
- 「4. sinh<sup>-1</sup>」..... 数値の逆双曲線関数「sinh<sup>-1</sup>」を求める。
- 「5. cosh<sup>-1</sup>」..... 数値の逆双曲線関数「cosh<sup>-1</sup>」を求める。
- 「6. tanh<sup>-1</sup>」..... 数値の逆双曲線関数「tanh<sup>-1</sup>」を求める。

## ■ Eng記号計算

Eng記号を使った計算ができます。

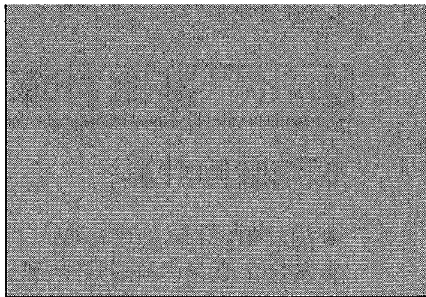
1. m	2. μ	3. n
4. p	5. f	6. k
7. M	8. G	9. T

↑

メニューの意味は、以下のとおりです。

- 「1. m」..... 「m(ミリ)」(10<sup>-3</sup>)を入力。
- 「2. μ」..... 「μ(マイクロ)」(10<sup>-6</sup>)を入力。
- 「3. n」..... 「n(ナノ)」(10<sup>-9</sup>)を入力。
- 「4. p」..... 「p(ピコ)」(10<sup>-12</sup>)を入力。
- 「5. f」..... 「f(フェムト)」(10<sup>-15</sup>)を入力。
- 「6. k」..... 「k(キロ)」(10<sup>3</sup>)を入力。
- 「7. M」..... 「M(メガ)」(10<sup>6</sup>)を入力。
- 「8. G」..... 「G(ギガ)」(10<sup>9</sup>)を入力。
- 「9. T」..... 「T(テラ)」(10<sup>12</sup>)を入力。

# 2



## マニュアル計算

---

## 2-1. 基本計算の仕方

### 加減乗除計算

- 加減乗除計算は、書式通りにキーを押します。
- 負数は、数値の前に(←)を押します。

例題	操作	表示窓
$23+4.5-53 = -25.5$	23 <b>+</b> 4.5 <b>-</b> 53 <b>EXE</b>	-25.5
$56 \times (-12) \div (-2.5) = 268.8$	56 <b>×</b> (←) 12 <b>+</b> (←) 2.5 <b>EXE</b>	268.8
$12369 \times 7532 \times 74103 =$ $6.903680613 \times 10^{12}$ (6903680613000)	12369 <b>×</b> 7532 <b>×</b> 74103 <b>EXE</b>	6.903680613E+12
$(4.5 \times 10^{75}) \times (-2.3 \times 10^{-79})$ $= -1.035 \times 10^{-3}$ (-0.001035)	4.5 <b>EXP</b> 75 <b>×</b> (←) 2.3 <b>EXP</b> (←) 79 <b>EXE</b>	-1.035E-03 (Norm 1モード指定時)
$(2+3) \times 10^2 = 500$	( <b>[</b> ) 2 <b>+</b> 3 <b>]</b> <b>×</b> 1 <b>EXP</b> 2 <b>EXE</b>	500
※ ( <b>[</b> ) 2 <b>+</b> 3 <b>]</b> <b>EXP</b> 2では正しい答えは得られません。必ず上のように ( <b>[</b> ) と <b>EXP</b> の間で <b>×</b> 1 と操作してください。		

- 四則の混合計算は、加減より乗除優先で計算されます。

例題	操作	表示窓
$3+5 \times 6 = 33$	3 <b>+</b> 5 <b>×</b> 6 <b>EXE</b>	33
$7 \times 8 - 4 \times 5 = 36$	7 <b>×</b> 8 <b>-</b> 4 <b>×</b> 5 <b>EXE</b>	36
$1+2-3 \times 4 \div 5+6 = 6.6$	1 <b>+</b> 2 <b>-</b> 3 <b>×</b> 4 <b>+</b> 5 <b>+</b> 6 <b>EXE</b>	6.6

# カッコ計算

例題	操作	表示窓
$100 - (2 + 3) \times 4 = 80$	$100 \text{ [=]} \text{ [ ( ] } 2 \text{ [+]} 3 \text{ [ ) ] } \text{ [ \times ] } 4 \text{ [ = ] EXE}$	<b>80</b>
$2 + 3 \times (4 + 5) = 29$	$2 \text{ [+]} 3 \text{ [ \times ] } \text{ [ ( ] } 4 \text{ [+]} 5 \text{ [ = ] EXE}$	<b>29</b>
※ [EXE] を押す直前の閉じカッコは、何重であっても省略できます。 $(7 - 2) \times (8 + 5) = 65$	$\text{ [ ( ] } 7 \text{ [-]} 2 \text{ [ ) ] } \text{ [ ( ] } 8 \text{ [+]} 5 \text{ [ = ] EXE}$	<b>65</b>
※ カッコの直前の [ \times ] は、省略できます。 $10 - (2 + 7 \times (3 + 6)) = -55$	$10 \text{ [-]} \text{ [ ( ] } 2 \text{ [+]} 7 \text{ [ \times ] } \text{ [ ( ] } 3 \text{ [+]} 6 \text{ [ = ] EXE}$	<b>-55</b>
※ 以下、本書では省略せずに記します。 $\frac{2 \times 3 + 4}{5} = (2 \times 3 + 4) \div 5 = 2$	$\text{ [ ( ] } 2 \text{ [ \times ] } 3 \text{ [+]} 4 \text{ [ ) ] } \text{ [ \div ] } 5 \text{ [ = ] EXE}$	<b>2</b>
$\frac{6}{4 \times 5} = 0.3$	$6 \text{ [ \div ] } \text{ [ ( ] } 4 \text{ [ \times ] } 5 \text{ [ ) ] } \text{ [ = ] EXE}$	<b>0.3</b>
※ 上例は、 $6 \text{ [ \div ] } 4 \text{ [ \div ] } 5 \text{ [ = ] EXE}$ としても同様です。		

## 小数点以下指定、有効桁数指定、および指数表示範囲指定

表示形式／クリアー(DSP/CLR)メニューを呼び出して行ないます(32ページ参照)。

**FUNCTION** **6** (DSP/CLR)

1. Fix	2. Sci
3. Norm	4. Eng
-----	
5. Mcl	6. Sc1

- 小数点以下の桁数指定は、**1** (Fix) $n$  ( $n=0\sim 9$ )と操作します。
  - 有効桁数指定は、**2** (Sci) $n$  ( $n=0\sim 9$ 、ただし0=10桁指定)と操作します。
- ※ 小数点以下設定、有効桁数設定を行なっても、内部演算は仮数部15桁を記憶しています。この数値を設定された桁数と同じにしたいときは、**SHIFT** **Rnd**と操作してください。
- 指数表示範囲の指定は、**3** (Norm) $n$  ( $n=1\sim 2$ )と操作します。**1**を押すとNorm1モードに、**2**を押すとNorm2モードに指定されます。
    - ・ Norm1モード\* .....  $10^{-2}$ 未満または $10^{10}$ 以上の数値を指数表示にします。
    - ・ Norm2モード\* .....  $10^{-9}$ 未満または $10^{10}$ 以上の数値を指数表示にします。
  - **SHIFT** **ENG**または**SHIFT** **ENG**と操作すると、表示されている数値を指数部が3の倍数の指数表示にします。



例題	操作	表示窓
100÷6 =16.66666666...	100 $\div$ 6 <b>EXE</b>	16.66666667
(小数点以下4桁指定) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>1</b> (Fix) <b>4</b>		16.6667
(指定解除) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>3</b> (Norm) <b>1</b>		16.66666667
(有効桁数5桁指定) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>2</b> (Sci) <b>5</b>		1.6667E+01
(指定解除) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>3</b> (Norm) <b>1</b>		16.66666667
※指定を行なうと、指定桁の次の桁が四捨五入されて表示されます。		
200÷7×14=400	200 $\div$ 7 $\times$ 14 <b>EXE</b>	400
(小数点以下3桁指定) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>1</b> (Fix) <b>3</b>		400.000
(表示10桁で計算を続ける) 200 $\div$ 7 <b>EXE</b>		28.571
	<b>X</b>	Ans $\times$ _
	14 <b>EXE</b>	400.000
同じ計算を指定桁で計算すると		
	200 $\div$ 7 <b>EXE</b>	28.571
(内部数値丸め) <b>SHIFT</b> <b>Rnd</b>		28.571
	<b>X</b>	Ans $\times$ _
	14 <b>EXE</b>	399.994
(指定解除) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>3</b> (Norm) <b>1</b>		399.994
1÷1000=0.001 =1×10 <sup>-3</sup>	(Norm1モードが指定されているとき) 1 $\div$ 1000 <b>EXE</b>	1.E-03
(Norm2モード指定) <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>3</b> (Norm) <b>2</b>		0.001
※ Norm1/Norm2モードの指定は、Norm1/Norm2モード表示状態のとき <b>FUNCTION</b> <b>6</b> (DSP/CLR) <b>3</b> (Norm) <i>n</i> ( <i>n</i> =1~2)と操作するまで変更されません。		
123m × 456=56088m =56.088km	123 $\times$ 456 <b>EXE</b> <b>SHIFT</b> <b>ENG</b>	56088 56.088 E+03
78g × 0.96=74.88g =0.07488kg	78 $\times$ 0.96 <b>EXE</b> <b>SHIFT</b> <b>ENG</b>	74.88 0.07488E+03

# パーセント計算

例題	操作	表示窓
<p>●割合 1,500円の26%は</p>	<p>1500 <input type="button" value="X"/> 26 <input type="button" value="SHIFT"/> <input <="" p="" type="button" value="%"/> </p>	<p><b>390</b> (円)</p>
<p>●割増し 3,620円の15%増しは</p>	<p>3620 <input type="button" value="X"/> 15 <input type="button" value="SHIFT"/> <input <input="" type="button" value="+"/></p>	<p><b>4163</b> (円)</p>
<p>●割引き 4,750円の4%引きは</p>	<p>4750 <input type="button" value="X"/> 4 <input type="button" value="SHIFT"/> <input <input="" type="button" value="-"/></p>	<p><b>4560</b> (円)</p>
<p>●比率 75個は250個の何%か</p>	<p>75 <input type="button" value="+"/> 250 <input type="button" value="SHIFT"/> <input <="" p="" type="button" value="%"/> </p>	<p><b>30</b> (%)</p>
<p>●変化率(増減比率) 141万円は120万円の 何%アップか</p>	<p>141 <input type="button" value="-"/> 120 <input type="button" value="SHIFT"/> <input <="" p="" type="button" value="%"/> </p>	<p><b>17.5</b> (%)</p>
<p>240円は300円の 何%引きか</p>	<p>240 <input type="button" value="-"/> 300 <input type="button" value="SHIFT"/> <input <="" p="" type="button" value="%"/> </p>	<p><b>-20</b> (%)</p>

# メモリー計算

## ■ 独立メモリー

- メモリー内へ直接加減(累計)することができます。個々の答えを求めながら、同時に合計を求める合計計算に非常に便利です。

例題	操作	表示窓
$23+9=32$	$23$ $\oplus$ $9$ $\text{STO}$ $\text{M}$	<b>32</b>
$53-6=47$	$53$ $\ominus$ $6$ $\text{M}+$	<b>47</b>
$\rightarrow 45 \times 2 = 90$	$45$ $\times$ $2$ $\text{SHIFT}$ $\text{M}$ $\ominus$	<b>90</b>
$99 \div 3 = 33$	$99$ $\div$ $3$ $\text{M}+$	<b>33</b>
(合計) 22	$\text{RCL}$ $\text{M}$	<b>22</b>
<p>※独立メモリーに最初に数値を入れるときは <math>\text{STO}</math> <math>\text{M}</math> を使います            (したがってあらかじめメモリーをクリアしておく必要はありません)。            また、上記のように <math>\text{M}+</math>、<math>\text{SHIFT}</math> <math>\text{M}</math><math>\ominus</math> は、<math>\text{EXE}</math> の働きを兼ねそなえています。</p>		
$7+7+7+(2 \times 3)+(2 \times 3)$ $+ (2 \times 3) - (2 \times 3) = 33$	$7$ $\text{STO}$ $\text{M}$ $\text{M}+$ $\text{M}+$ $2$ $\times$ $3$ $\text{M}+$ $\text{M}+$ $\text{M}+$ $\text{SHIFT}$ $\text{M}$ $\ominus$ $\text{RCL}$ $\text{M}$	<b>33</b>

## ■ 変数メモリー

- 変数メモリーは標準状態で26個あります。このメモリーには、データや定数、答えなどの数値を自由に保存することができます。

例題	操作	表示窓
$193.2 \div 23 = 8.4$	193.2 <b>STO</b> <b>A</b> <b>÷</b> 23 <b>EXE</b>	<b>8.4</b>
$193.2 \div 28 = 6.9$	<b>RCL</b> <b>A</b> <b>÷</b> 28 <b>EXE</b>	<b>6.9</b>
$193.2 \div 42 = 4.6$	<b>RCL</b> <b>A</b> <b>÷</b> 42 <b>EXE</b>	<b>4.6</b>
$\frac{9 \times 6 + 3}{(7-2) \times 8} = 1.425$	9 <b>×</b> 6 <b>+</b> 3 <b>STO</b> <b>B</b>	<b>57</b>
	<b>(</b> 7 <b>-</b> 2 <b>)</b> <b>×</b> 8 <b>STO</b> <b>C</b>	<b>40</b>
	<b>ALPHA</b> <b>B</b> <b>÷</b> <b>ALPHA</b> <b>C</b> <b>EXE</b>	<b>1.425</b>
<p>※ 上記例題は <math>(9 \times 6 + 3) \div ((7 - 2) \times 8) =</math> とすれば、メモリーを使用せずに</p> <p><b>(</b> 9 <b>×</b> 6 <b>+</b> 3 <b>)</b> <b>÷</b> <b>(</b> 7 <b>-</b> 2 <b>)</b> <b>×</b> 8 <b>)</b> <b>EXE</b></p> <p>で求めることができます。</p>		

## 2-2. 特別機能

### アンサー(Ans)機能

本機は最新の計算結果を記憶しておくアンサー機能を備えています。この機能は、数値や数式などを入力した後 **EXE** を押して求めた最新の計算結果(数式ならば答え)を記憶するものです。

最新の計算結果を確認したいときや、最新の計算結果を使ってさらに新しい計算を行ないたいときなどに使うと便利です。

※ Ansはメモリーと同じ扱いになりますので、以後Ansメモリーと呼びます。

**例**  $123 + 456 = 579$   
 $789 - 579 = 210$

**AC** **1** **2** **3** **+**

**4** **5** **6** **EXE**

**7** **8** **9** **-**

**SHIFT** **Ans** **EXE**

123+456	579
---------	-----

789-Ans	210
---------	-----

Ansメモリーには仮数部15桁、指数部2桁が記憶されます。この内容は **AC** を押しても電源をOFFにしても消えません。

※ **EXE**、**%**、**M+**、**SHIFT** **M-**、**STO**  $\alpha$  ( $\alpha = A \sim Z$ ) と押して求めた計算結果は、新たにAnsメモリーに記憶されます。

※ 変数メモリーに記憶させた数値を呼び出した直後(たとえば **RCL**  $\alpha$  ( $\alpha = A \sim Z$ ) と操作した直後) や変数入力状態から変数に数値を入力した直後は、Ansメモリーの内容は更新されません。

※ 演算実行中にエラーとなったとき、前回のAnsメモリーの内容は保護されます。

## 連続演算機能

**EXE**を押して求めた演算結果に対して、さらに計算を続けることができます。この場合、Ansメモリーに記憶された値を使って計算が行なわれます。

**例**  $1 \div 3 =$  続けて  $\times 3 =$

**AC** **1** **÷** **3** **EXE**

1÷3  
0.3333333333

(続けて) **×** **3** **EXE**

Ans×3  
1

この機能は、後置関数 ( $x^2$ ,  $x^{-1}$ ,  $x! : 35$ ページ参照)、 $+$ 、 $-$ 、 $^{\wedge}(x^y)$ 、 $\sqrt{\quad}$ 、 $''$ ”についても使うことができます。

## リプレイ機能

演算終了後に **▶** または **◀** を押すことにより、実行した計算式を表示させることができます。計算式をチェックしたり、数値や計算命令を変更して新たな計算を行いたいときなどに使うと便利です。

- **▶** を押したときは、計算式の先頭から表示します。また、カーソルが計算式の先頭に点滅します。
- **◀** を押したときは、計算式の最後から表示します。また、カーソルが計算式の最後の後に点滅します。

**例**  $4.12 \times 6.4 = 26.368$ 、  
 $4.12 \times 7.1 = 29.252$

**AC** **4** **.** **1** **2** **×** **6** **.** **4** **EXE**

4.12×6.4  
26.368



4.12×6.4



4.12×7.1



4.12×7.1

29.252

※ リプレイ機能により記憶される文字数は、127バイトです(38ページ参照)。

※ リプレイ機能により記憶された計算式は、新たな演算やモード変更によりクリアされます。

※ AC後のリプレイについて

**AC**を押してもリプレイ内容はクリアされませんので、AC状態でリプレイすると再び計算式を呼び出すことができます。ただし、メインメニュー表示から他のメニューを選択すると、リプレイ内容はクリアされます。

## エラー位置表示機能

演算実行時にエラーが生じたときに **◀** または **▶** を押すとエラー状態が解除され、エラーが生じた位置にカーソルが表示されます。

例

14 ÷ 10 × 2.3 を間違えて、14 ÷ 0 × 2.3 と入力してしまった。

**AC** **1** **4** **÷** **0**

**×** **2** **.** **3** **EXE**

14÷0×2.3

Ma ERROR

**◀** (または **▶**)

14÷0×2.3

ここでエラーが発生しました。

**◀** **SHIFT** **INS** **1**

14÷10×2.3

**EXE**

14÷10×2.3

3.22

## マルチステートメント機能

- 区切りコード(「:」、「▲」)によって計算式を区切り、複数の計算式を連続して実行させることができます。この計算式は **EXE** を押すことにより、先頭から順番に実行されます。
- 区切りコードは、以下のように作動します。
  - ・「:」..... 複数の計算式を最後まで実行します。計算式の途中結果は表示されません。
  - ・「▲」..... 末尾に「▲」が書き込まれている計算式を実行して、計算結果を表示します。**EXE** を押すと、次の計算式を実行します。

※ 最後の計算式の末尾には「▲」を書き込まなくても、計算結果は表示されます。

例

$$6.9 \times 123 = 848.7$$

$$123 \div 3.2 = 38.4375$$

AC 1 2 3 STO A  
6 . 9 X ALPHA A SHIFT ▲  
ALPHA A ÷ 3 . 2 EXE

123  
6.9×A  
A÷3.2  
848.7  
STO

EXE

123  
6.9×A  
A÷3.2  
38.4375  
0

※ マルチステートメント文の中では、連続演算機能は使えません。

$$123 \times 456 : \times 5$$

↑  
不可



## 2-3. 関数計算の仕方

### 角度変換(°、r、g)

- 角度単位(DRG)メニューの **4** (°)、**5** (r)、**6** (g) を使うと、基本となる角度単位を設定し直すことなく他の角度単位に値を変換することができます(31ページ参照)。
- ※ 基本となる角度単位を一度設定すると、他の角度単位に設定し直さない限り、電源をOFFにしても変更されません。
- BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
度数法(ディグリー単位)が設定されているとき、4.25ラジアンを度数法に変換する。	$4.25$ <b>FUNCTION</b> <b>5</b> (DRG) <b>1</b> (Deg)	<b>243.5070629</b>
	$4.25$ <b>FUNCTION</b> <b>5</b> (DRG) <b>5</b> (r) <b>EXE</b>	
$47.3^\circ + 82.5\text{rad}$ $= 4774.20181^\circ$	$47.3$ <b>+</b> $82.5$ <b>FUNCTION</b> <b>5</b> (DRG)	<b>4774.20181</b>
	<b>5</b> (r) <b>EXE</b>	

## 三角関数(sin, cos, tan)、逆三角関数(sin<sup>-1</sup>, cos<sup>-1</sup>, tan<sup>-1</sup>)

- 三角・逆三角関数の計算を行なうときは、角度単位を確実に設定してください(31ページ参照)。
- BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
$\sin 63^\circ 52'41'' = 0.897859012$	[FUNCTION] [5] (DRG) [1] (Deg) [sin] 63 [***] 52 [***] 41 [***] [EXE]	<b>0.897859012</b>
$\cos\left(\frac{\pi}{3}\text{rad}\right) = 0.5$	[FUNCTION] [5] (DRG) [2] (Rad) [cos] [◀] [SHIFT] [7] [π] [÷] 3 [)] [EXE]	<b>0.5</b>
$\tan(-35\text{grad}) = -0.6128007881$	[FUNCTION] [5] (DRG) [3] (Gra) [tan] [(-)] 35 [EXE]	<b>-0.6128007881</b>
$2 \cdot \sin 45^\circ \times \cos 65^\circ = 0.5976724775$	[FUNCTION] [5] (DRG) [1] (Deg) 2 [×] [sin] 45 [×] [cos] 65 [EXE]	<b>0.5976724775</b>
$\sin^{-1}0.5 = 30^\circ$ ( $\sin x = 0.5$ の $x$ を求める)	[SHIFT] [sin] 0.5 [EXE]	<b>30</b>
	省略可 ↓ .5でも可	
$\cos^{-1} \frac{\sqrt{2}}{2} = 0.7853981634\text{rad}$ $= \frac{\pi}{4}\text{rad}$	[FUNCTION] [5] (DRG) [2] (Rad) [SHIFT] [cos] [◀] [√] 2 [÷] 2 [)] [EXE]	<b>0.7853981634</b>
	[÷] [SHIFT] [7] [π] [EXE]	<b>0.25</b>
$\tan^{-1}0.741 = 36.53844577^\circ$ $= 36^\circ 32' 18.4''$	[FUNCTION] [5] (DRG) [1] (Deg) [SHIFT] [tan] 0.741 [EXE]	<b>36.53844577</b>
	[SHIFT] [***]	<b>36° 32' 18.4''</b>
※ 度、分、秒の各桁の合計が11桁を超えた場合は、上位(度および分)を優先して表示します。それ以下は表示されませんが、計算機内には10進数として全部残っています。		

# 対数関数(log、ln)、指数関数( $10^x$ 、 $e^x$ 、 $\wedge(x^y)$ 、 $x\sqrt{\quad}$ )

● BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
$\log 1.23(\log_{10} 1.23)=$ $8.990511144 \times 10^{-2}$	$\boxed{\log} 1.23 \boxed{\text{EXE}}$	<b>0.08990511144</b>
$\ln 90(\log_e 90)=4.49980967$	$\boxed{\ln} 90 \boxed{\text{EXE}}$	<b>4.49980967</b>
$\log 456 \div \ln 456$ $=0.4342944819$ (log/ln比=定数M)	$\boxed{\log} 456 \boxed{\div} \boxed{\ln} 456 \boxed{\text{EXE}}$	<b>0.4342944819</b>
$4^x=64(x$ を求める) $x \cdot \log 4 = \log 64$ $x = \frac{\log 64}{\log 4} = 3$	$\boxed{\log} 64 \boxed{\div} \boxed{\log} 4 \boxed{\text{EXE}}$	<b>3</b>
$10^{1.23} = 16.98243652$ (常用対数1.23の真数を求める)	$\boxed{\text{SHIFT}} \boxed{10^x} 1.23 \boxed{\text{EXE}}$	<b>16.98243652</b>
$e^{4.5} = 90.0171313$ (自然対数4.5の真数を求める)	$\boxed{\text{SHIFT}} \boxed{e^x} 4.5 \boxed{\text{EXE}}$	<b>90.0171313</b>
$10^4 \cdot e^{-4} + 1.2 \cdot 10^{2.3}$ $= 422.5878667$	$\boxed{\text{SHIFT}} \boxed{10^x} 4 \boxed{\times} \boxed{\text{SHIFT}} \boxed{e^x} \boxed{(-)} 4 \boxed{+}$ $1.2 \boxed{\times} \boxed{\text{SHIFT}} \boxed{10^x} 2.3 \boxed{\text{EXE}}$	<b>422.5878667</b>
$5.6^{2.3} = 52.58143837$	$5.6 \boxed{\wedge} 2.3 \boxed{\text{EXE}}$	<b>52.58143837</b>
$\sqrt[7]{123} (= 123^{\frac{1}{7}})$ $= 1.988647795$	$7 \boxed{\text{SHIFT}} \boxed{\sqrt[x]{\quad}} 123 \boxed{\text{EXE}}$	<b>1.988647795</b>
$2 + 3 \times \sqrt[3]{64} - 4 = 10$	$2 \boxed{+} 3 \boxed{\times} 3 \boxed{\text{SHIFT}} \boxed{\sqrt[x]{\quad}} 64 \boxed{-} 4 \boxed{\text{EXE}}$	<b>10</b>

※  $\wedge(x^y)$ および $\sqrt[x]{\quad}$ は $\times$ 、 $\div$ より優先して計算されます。

# 双曲線関数 (sinh, cosh, tanh)、逆双曲線関数 (sinh<sup>-1</sup>, cosh<sup>-1</sup>, tanh<sup>-1</sup>)

● BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
sinh 3.6=18.28545536	[FUNCTION] [1] (MATH) [▼] [▼] [1] (sinh) 3.6 [EXE]	<b>18.28545536</b>
cosh 1.23=1.856761057	[FUNCTION] [1] (MATH) [▼] [▼] [2] (cosh) 1.23 [EXE]	<b>1.856761057</b>
tanh 2.5=0.9866142982	[FUNCTION] [1] (MATH) [▼] [▼] [3] (tanh) 2.5 [EXE]	<b>0.9866142982</b>
cosh 1.5 - sinh 1.5 =0.2231301601 =e <sup>-1.5</sup> (cosh x ± sinh x = e <sup>±x</sup> の証明)	[FUNCTION] [1] (MATH) [▼] [▼] [2] (cosh) 1.5 [-] [FUNCTION] [1] (MATH) [▼] [▼] [1] (sinh) 1.5 [EXE]	<b>0.2231301601</b>
	(続けて) [In] [SHIFT] [Ans] [EXE]	<b>-1.5</b>
sinh <sup>-1</sup> 30=4.094622224	[FUNCTION] [1] (MATH) [▼] [▼] [4] (sinh <sup>-1</sup> ) 30 [EXE]	<b>4.094622224</b>
cosh <sup>-1</sup> ( $\frac{20}{15}$ ) =0.7953654612	[FUNCTION] [1] (MATH) [▼] [▼] [5] (cosh <sup>-1</sup> ) [20] [÷] [15] [EXE]	<b>0.7953654612</b>
tanh 4x=0.88のとき、xは? $x = \frac{\tanh^{-1} 0.88}{4}$ =0.3439419141	[FUNCTION] [1] (MATH) [▼] [▼] [6] (tanh <sup>-1</sup> ) 0.88 [÷] [4] [EXE]	<b>0.3439419141</b>

# その他の関数( $\sqrt{\quad}$ 、 $x^2$ 、 $x^{-1}$ 、 $x!$ 、 $\sqrt[3]{\quad}$ 、Ran#、Abs、Int、Frac、Intg)

● BASE-Nモード指定時は、この操作はできません。

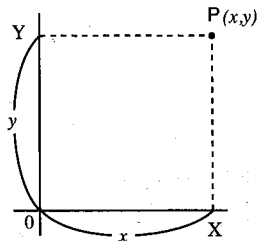
例題	操作	表示窓
$\sqrt{2} + \sqrt{5} = 3.65028154$	$\sqrt{\square} 2 \square + \sqrt{\square} 5 \square \text{EXE}$	<b>3.65028154</b>
$2^2 + 3^2 + 4^2 + 5^2 = 54$	$2 \square x^2 \square + 3 \square x^2 \square + 4 \square x^2 \square$ $+ 5 \square x^2 \square \text{EXE}$	<b>54</b>
$\frac{1}{\frac{1}{3} - \frac{1}{4}} = 12$	$\square \square 3 \square \text{SHIFT} \square x^{-1} \square - \square \square 4 \square \text{SHIFT} \square x^{-1} \square \square$ $\square \text{SHIFT} \square x^{-1} \square \text{EXE}$	<b>12</b>
$8!(=1 \times 2 \times 3 \times \dots \times 8)$ $=40320$	$8 \square \text{FUNCTION} \square 1 \square (\text{MATH}) \square 6 \square (x!) \square \text{EXE}$	<b>40320</b>
$\sqrt[3]{36 \times 42 \times 49} = 42$	$\square \text{SHIFT} \square \sqrt[3]{\square} \square \square 36 \square \square \square 42 \square \square \square$ $49 \square \square \text{EXE}$	<b>42</b>
乱数発生(0以上1未満の擬似乱数。 小数点以下10桁で表示します。)	$\square \text{FUNCTION} \square 1 \square (\text{MATH})$ $\square 6 \square (\text{Ran}\#) \square \text{EXE}$	(例) <b>0.4810497011</b>
$\sqrt{1 - \sin^2 40^\circ}$ $= 0.7660444431$ $= \cos 40^\circ$	$\square \text{FUNCTION} \square 5 \square (\text{DRG}) \square 1 \square (\text{Deg})$ $\square \square 1 \square - \square \square \sin 40 \square \square \square x^2 \square$ $\square \square \text{EXE}$	<b>0.7660444431</b>
$(\cos \theta = \pm \sqrt{1 - \sin^2 \theta})$ の証明	(続けて) $\square \text{SHIFT} \square \cos \square \square \text{SHIFT} \square \text{Ans} \square \text{EXE}$	<b>40</b>
$\frac{3}{4}$ の常用対数の絶対値は? $ \log \frac{3}{4}  = 0.1249387366$	$\square \text{FUNCTION} \square 1 \square (\text{MATH}) \square \square$ $\square 1 \square (\text{Abs}) \square \log \square \square 3 \square \square \square 4 \square \square \text{EXE}$	<b>0.1249387366</b>
$\frac{7800}{96}$ の整数部は.....81	$\square \text{FUNCTION} \square 1 \square (\text{MATH}) \square \square$ $\square 2 \square (\text{Int}) \square \square 7800 \square \square \square 96 \square \square \text{EXE}$	<b>81</b>

例題	操作	表示窓
$\frac{7800}{96}$ の小数部は.....0.25	FUNCTION 1 (MATH) ▼ 3 (Frac) ( 7800 ÷ 96 ) EXE	<b>0.25</b>
2512549139 ÷ 2141 は割り切れるか?	2512549139 ÷ 2141 EXE FUNCTION 1 (MATH) ▼ 3 (Frac) ( 2512549139 ÷ 2141 ) EXE	<b>1173540</b>  <b>0.99953292</b>
-3.5を超えない整数は?	FUNCTION 1 (MATH) ▼ 4 (Intg) ( - ) 3.5 EXE	<b>-4</b>

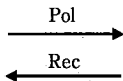
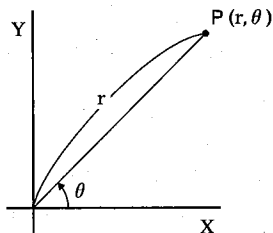
# 座標変換 (Pol、Rec)

直交座標と極座標の相互変換を行なうことができます。

●直交座標 (Rectangular)



●極座標 (Polar)



● Polにより計算される  $\theta$  の範囲は、 $-180^\circ < \theta \leq 180^\circ$  です (ラジアン、グレードでの演算範囲も同様)。

● 演算結果は、自動的に変数メモリー I と変数メモリー J に記憶されます。

	I	J
Pol	r	$\theta$
Rec	x	y

● BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
$x=14, y=20.7$ のとき $r$ および $\theta$ は?	FUNCTION 5 (DRG) 1 (Deg) FUNCTION 1 (MATH) ▾ 5 (Pol()) 14 20.7 EXE (続けて) ALPHA J EXE SHIFT $\leftrightarrow$	$r = 24.98979791$ $\theta = 55.92839019$ $55^\circ 55' 42.2''$
$r=25, \theta=56^\circ$ のとき $x$ および $y$ は?	FUNCTION 5 (DRG) 1 (Deg) FUNCTION 1 (MATH) ▾ 6 (Rec()) 25 56 EXE	$x = 13.97982258$ $y = 20.72593931$

## 順列( $nPr$ )、組み合わせ( $nCr$ )

● 順列の総数

$$nPr = \frac{n!}{(n-r)!}$$

● 組み合わせの総数

$$nCr = \frac{n!}{r!(n-r)!}$$

● BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
10個の中から4個を取り出して 並べるとき、その並べ方は何通り あるか? $10P4 = 5040$	10 <b>FUNCTION</b> <b>1</b> (MATH) <b>7</b> ( $nPr$ ) <b>4</b> <b>EXE</b>	<b>5040</b>
10個の中から4個を取り出すと き、その取り出し方は何通りあ るか? $10C4 = 210$	10 <b>FUNCTION</b> <b>1</b> (MATH) <b>8</b> ( $nCr$ ) <b>4</b> <b>EXE</b>	<b>210</b>



# 分数計算

- 分数の置数は、整数、分子、分母の順に行ないます。表示もこの順となります。
- BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
$\frac{2}{5} + 3\frac{1}{4} = 3\frac{13}{20}$ $= 3.65$	$2 \text{ [a/b]} 5 \text{ [+]} 3 \text{ [a/b]} 1 \text{ [a/b]} 4 \text{ [EXE]}$ (小数変換) $\text{[a/b]}$	<b>3J13J20</b>  <b>3.65</b>
※ 分数は小数に変換することができます。また、分数を小数に変換した直後は、小数を分数に戻せます。		
$3\frac{456}{78} = 8\frac{11}{13} \quad (\text{約分})$	$3 \text{ [a/b]} 456 \text{ [a/b]} 78 \text{ [EXE]}$ $\text{[SHIFT] [d/c]}$	<b>8J11J13</b>  <b>115J13</b>
※ 約分可能な分数および仮分数は、計算命令キーを押せば既約分数になります。また、 $\text{[SHIFT] [d/c]}$ と操作すると、仮分数となります。		
$\frac{1}{2578} + \frac{1}{4572}$ $= 6.066202547 \times 10^{-4}$	$1 \text{ [a/b]} 2578 \text{ [+]} 1 \text{ [a/b]} 4572 \text{ [EXE]}$	<b>6.066202547E-04</b> (Norm1モード指定時)
※ 整数、分子、分母、区切りシボルの合計桁数が10桁を超えた場合は、自動的に小数表示となります。		
$\frac{1}{2} \times 0.5 = 0.25$	$1 \text{ [a/b]} 2 \text{ [X]} 0.5 \text{ [EXE]}$	<b>0.25</b>
※ 分数と小数の計算は、小数で求められます。		
$\frac{1}{\frac{2}{3}} = \frac{1}{6}$	$\text{[1] [a/b]} 2 \text{ [D]} \text{ [a/b]} 3 \text{ [EXE]}$	<b>1J6</b>
$\frac{1}{\frac{1}{3} + \frac{1}{4}} = 1\frac{5}{7}$	$1 \text{ [a/b]} \text{[ ( ]} 1 \text{ [a/b]} 3 \text{ [+]} 1 \text{ [a/b]} 4 \text{ [D]} \text{ [EXE]}$	<b>1J5J7</b>
※ 分子、分母内で ( ) を使うと、繁分数の計算を行なうことができます。		

# Eng記号計算

- Eng記号を使って計算を行なうことができます。また、演算結果をEng記号を使って表示することができます。

詳しくは「表示形式/クリアー (DSP/CLR) メニュー」(32ページ)および「Eng記号表示指定」(33ページ)をご覧ください。

- BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
999k(キロ) +25k(キロ) =1.024M(メガ)	FUNCTION 6 (DSP/CLR) 4 (Eng) 999 FUNCTION 1 (MATH) ▼ ▼ ▼ 6 (k) + 25 FUNCTION 1 (MATH) ▼ ▼ ▼ 6 (k) EXE FUNCTION 6 (DSP/CLR) 4 (Eng)	1.024M 1024000
100m(ミリ)× 5μ(マイクロ) =500n(ナノ)	FUNCTION 6 (DSP/CLR) 4 (Eng) 100 FUNCTION 1 (MATH) ▼ ▼ ▼ 1 (m) X 5 FUNCTION 1 (MATH) ▼ ▼ ▼ 2 (μ) EXE FUNCTION 6 (DSP/CLR) 4 (Eng)	500.n 5.E-07
9÷10=0.9 =900m(ミリ)	FUNCTION 6 (DSP/CLR) 4 (Eng) 9 ÷ 10 EXE SHIFT ENG SHIFT ENG SHIFT ENG SHIFT ENG SHIFT ENG SHIFT ENG	900.m 0.9 0.0009k 0.9 900.m 900000.μ 900.m

## 2-4. 科学定数計算

本機は20種類の科学定数を記憶しています。COMP/SD/LRモード指定時に以下のように操作して、呼び出すことができます。

- (1) 機能メニュー表示を呼び出します。

**FUNCTION**

```
1.MATH 2.COMPLX
3.PROG 4.CONST
5.DRG 6.DSP/CLR
```

- (2) 科学定数メニュー表示No.1を呼び出します。

**4** (CONST)

```
1.mP 2.F 3.a0
4.c 5.h 6.G
7.e 8.me 9.u
0.NA
```

- (3) **▼**を押すと、科学定数メニュー表示No.2になります。

**▼**

```
1.k 2.g 3.R
4.a0 5.M0 6.ME
7.h 8.Mn 9.R0
0.d
```

- (4) 科学定数メニュー表示No.1/No.2から、呼び出したい科学定数を指す数字キーを押します。

※ **▲**を押すと、科学定数メニュー表示No.1に戻ります。

● 科学定数は、以下の20種類になります。

※ 下記のデータは、「ISO規格(1992)」および「CODATA Bulletin 63(1986)」のデータに準拠しています。

記号	名称	数値	単位
mp	陽子の静止質量	1.6726231E-27	kg
F	ファラデー定数	96485.309	C/mol
a <sub>0</sub>	ボーア半径	5.29177249E-11	m
c	真空中の光速度	299792458	m/s
h	プランク定数	6.6260755E-34	J·s
G	万有引力定数	6.67259E-11	Nm <sup>2</sup> /kg <sup>2</sup>
e	電気素量	1.60217733E-19	C
m <sub>e</sub>	電子の静止質量	9.1093897E-31	kg
u	原子質量単位	1.6605402E-27	kg
N <sub>A</sub>	アボガドロ定数	6.0221367E+23	mol <sup>-1</sup>
k	ボルツマン定数	1.380658E-23	J/K
g	重力加速度	9.80665	m/s <sup>2</sup>
R	モル気体定数	8.314510	J/(mol·K)
ε <sub>0</sub>	真空の誘電率	8.854187818E-12	F/m
μ <sub>0</sub>	真空の透磁率	1.256637061E-06	H/m
μ <sub>B</sub>	ボーア磁子	9.2740154E-24	A·m <sup>2</sup>
ħ	換算プランク定数	1.05457266E-34	J·s
m <sub>n</sub>	中性子の静止質量	1.6749286E-27	kg
R <sub>∞</sub>	リュードベリ定数	10973731.53	m <sup>-1</sup>
σ	ステファン-ボルツマン定数	5.67051E-08	W/(m <sup>2</sup> ·K <sup>4</sup> )

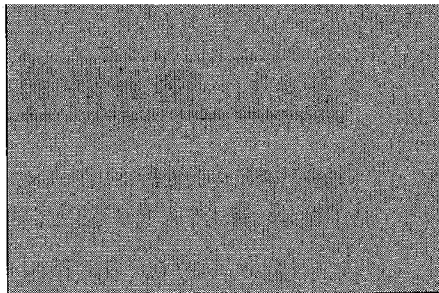
※数値はNorm1モード指定時のものです。

●BASE-Nモード指定時は、この操作はできません。

例題	操作	表示窓
<p>① 光速度[c] 質量2gの物質が消滅して全部エネルギーに変わったときのエネルギーを求めよ。</p>	<p>2 EXP (←) 3 (X) FUNCTION 4 (CONST) 4 (c) (X²) EXE</p>	<p>1.797510357E+14</p>
<p>② プランク定数[h] ある原子が波長 <math>\lambda = 5.0 \times 10^{-7}</math>mの光子1個を出したとき失うエネルギーを求めよ。</p>	<p>FUNCTION 4 (CONST) 5 (h) (X) FUNCTION 4 (CONST) 4 (c) (÷) 5 EXP (←) 7 EXE</p>	<p>3.972894922E-19</p>
<p>③ 万有引力定数[G] 体重が60kgと80kgの人が70cm離れているとすると、2人の引き合う力はいくらか。</p>	<p>FUNCTION 4 (CONST) 6 (G) (X) 60 (X) 80 (÷) 0.7 (X²) EXE</p>	<p>6.536414694E-07</p>
<p>④ 素電荷[e]、電子の質量[me] 3cm隔てである平行な電極間に200Vの電圧をかけた場合に、この中の電子が受ける力と加速度を求めよ。</p>	<p>FUNCTION 4 (CONST) 7 (e) (X) 200 (÷) 0.03 EXE (続けて) (÷) FUNCTION 4 (CONST) 8 (me) EXE</p>	<p>1.06811822E-15 1.172546411E+15</p>
<p>⑤ 1原子量の質量[u] 水素原子の質量は1.00783amuで電子の質量はその1/1800である。水素原子核(陽子)の質量は?</p>	<p>(1) 1.00783 (÷) 1.00783 (÷) 1800 (1) (X) FUNCTION 4 (CONST) (9) (u) EXE</p>	<p>1.672612484E-27</p>
<p>⑥ アボガドロ定数[NA] 水分子1個の質量を求めよ。</p>	<p>18 (÷) FUNCTION 4 (CONST) (0) (NA) EXE</p>	<p>2.988972336E-23</p>
<p>⑦ ボルツマン定数[k] 0°における理想気体の分子1個がもつ並進運動のエネルギーの平均を求めよ。</p>	<p>3 (÷) 2 (X) FUNCTION 4 (CONST) (▽) (1) (k) (X) 273 EXE</p>	<p>5.65379451E-21</p>

例題	操作	表示窓
<p>⑧ 重力加速度[g]</p> <p>小石を池に落とすと、池の表面に着くまで1.5秒かった。小石を落とした位置から池の表面までの距離はいくらか。</p>	<p>FUNCTION 4 (CONST) ▾ 2 (g)</p> <p>× 1.5 2 EXP</p>	<p>11.03248125</p>
<p>⑨ 真空の誘電率[ε<sub>0</sub>]</p> <p>面積が700cm<sup>2</sup>で2mmの距離だけ離れた2枚の銅板でできたコンデンサーがある。相対誘電率が5の油の中にこのコンデンサーを入れたとき、コンデンサーの容量はいくらになるか。</p>	<p>FUNCTION 4 (CONST) ▾ 4 (ε<sub>0</sub>)</p> <p>× 5 × 700 EXP (←) 4 (←)</p> <p>2 EXP (←) 3 EXE</p>	<p>1.549482868E-09</p>
<p>⑩ 真空の透磁率[μ<sub>0</sub>]</p> <p>真空中で、2つの長い導体が1.1m離れて置いてある。この2つの導体にそれぞれ2A、3Aの電流が逆方向に流れるとき、導体2mあたりに働く力を求めよ。</p>	<p>FUNCTION 4 (CONST) ▾ 5 (μ<sub>0</sub>)</p> <p>× 3 × 2 (←) SHIFT 7 (←)</p> <p>(←) 1.1 EXE</p>	<p>2.181818182E-06</p>

# 3



微分 / 2 次微分計算・  
積分計算・ $\Sigma$  計算

---

## 3-1. 微分計算

微分計算は内蔵関数 (MATH) メニューから **2** ( $d/dx$ ) を押して、次の書式で微分計算式を入力することにより求められます。

$$\mathbf{2} (d/dx)f(x) \mathbf{)} \mathbf{a} \mathbf{)} \Delta x \mathbf{)}$$

$\swarrow$   $\searrow$   
 微分係数を求める点       $x$  の増減分

$$d/dx(f(x), a, \Delta x) \Rightarrow \frac{d}{dx} f(a)$$

微分計算とは、微分の定義

$$f'(a) = \lim_{\Delta x \rightarrow 0} \frac{f(a + \Delta x) - f(a)}{\Delta x}$$

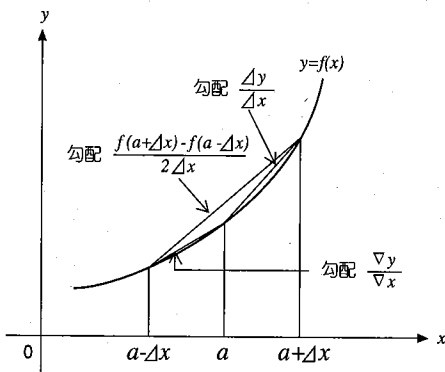
において「無限小」の代わりに「十分小さな  $\Delta x$ 」を用いて、 $f'(a)$  に近似した値

$$f'(a) \doteq \frac{f(a + \Delta x) - f(a)}{\Delta x}$$

を求めるものです。

本機ではより精度の高い答えを求めるために、中心差分を用いて微分計算を行ないます。

※ 中心差分とは





関数  $y=f(x)$  の点  $a$  と点  $a+\Delta x$  および点  $a$  と点  $a-\Delta x$  の勾配は、それぞれ

$$\frac{f(a+\Delta x)-f(a)}{\Delta x} = \frac{\Delta y}{\Delta x}, \quad \frac{f(a)-f(a-\Delta x)}{\Delta x} = \frac{\nabla y}{\nabla x}$$

となります ( $\Delta y/\Delta x$  を前方差分、 $\nabla y/\nabla x$  を後方差分といいます)。  
 微分係数を求める場合に上記の  $\Delta y/\Delta x$  の値と  $\nabla y/\nabla x$  の値の平均を取ると、より精度の高い微分係数の値を求めることができます。  
 この平均を中心差分といい、

$$f'(a) = \frac{1}{2} \left( \frac{f(a+\Delta x)-f(a)}{\Delta x} + \frac{f(a)-f(a-\Delta x)}{\Delta x} \right) \\ = \frac{f(a+\Delta x)-f(a-\Delta x)}{2\Delta x}$$

で表わされます。

## 微分計算の実例

**例** 関数  $y=x^3+4x^2+x-6$  の点  $x=3$  における微分係数を求める。ただし、 $x$  の増減分を  $\Delta x=1E-5$  とする。

AC FUNCTION 1 (MATH) 2 (d/dx)

ALPHA X ^ 3 + 4 ALPHA X X^2

+ ALPHA X - 6 ,

(関数  $f(x)$  の入力)

d/dx(X^3+4X^2+X-6  
, -

3 ,

(微分係数を求める点  $a$  の入力)

d/dx(X^3+4X^2+X-6  
, 3, -

1 EXP (-) 5 )

( $x$  の増減分  $\Delta x$  の入力)

d/dx(X^3+4X^2+X-6  
, 3, 1E-5) -

EXE

d/dx(X^3+4X^2+X-6  
,3,1E-5)

52

- ※ 関数  $f(x)$  には  $X$  の式しか使うことができません。 $X$  以外 ( $X$  を除く  $A \sim Z$ ) の変数は定数と見なされ、その変数メモリーに記憶されている数値を使って計算されます。
- ※  $\Delta x$ 、閉じカッコは省略することができます。なお、 $\Delta x$  を省略すると、 $\Delta x$  は微分係数を求める点  $x=a$  の値に応じた適切な値が自動的に設定されます。
- ※ 計算精度は、原則として表示結果の最下位桁目  $\pm 1$  です。

## 微分計算の応用

- 微分同士の四則計算ができます。

$$\frac{d}{dx} f(a) = f'(a), \quad \frac{d}{dx} g(a) = g'(a)$$

とすると、

$$f'(a) + g'(a), \quad f'(a) \times g'(a) \quad \text{など}$$

- 微分計算の結果に対して、四則計算や関数計算ができます。  
 $2 \times f'(a)$ 、 $\log(f'(a))$  など
- 微分計算の書式の各項 ( $f(x)$ ,  $a$ ,  $\Delta x$ ) に、関数式を用いることができます。

$$\frac{d}{dx} (\sin x + \cos x, \sin 0.5) \quad \text{など}$$

- ※ 微分計算の書式の各項の中に、微分計算式、積分計算式、または  $\Sigma$  計算式を入力することはできません。

## 微分計算時の諸注意

- 微分計算中(カーソルが消えている状態)に **AC** を押すと、微分計算は中止されます。
- 三角関数の微分計算は、Radモードで行なってください。
- 微分計算では、F~Hの3本のメモリーが使用されます。計算内容はこれらのメモリー内容を読み出して、確認することができます。これらの変数メモリーは微分計算時に使用できませんので、注意してください。

使用メモリー	F	G	H
データ	$a$	$\Delta x$	$f'(a)$

また、微分計算実行後、変数メモリーXには微分係数  $a$  の値が記憶されません。

## 3-2. 2次微分計算

微分計算は内蔵関数(MATH)メニューから③( $d^2/dx^2$ )を押して、次の書式で2次微分計算式を入力することにより求められます。

$$\text{③ } (d^2/dx^2)f(x) \text{ ④ } a \text{ ⑤ } n \text{ ⑥}$$

└─ 終了条件( $n=1\sim 15$ )

※  $n$  は省略できます。

└─ 微分係数を求める点

$$d^2/dx^2(f(x), a, n) \Rightarrow \frac{d^2}{dx^2}f(a)$$

2次微分計算は、ニュートンの補間多項式を基にした2階数値微分公式

$$f''(x) = \{-f(x-2h) + 16f(x-h) - 30f(x) + 16f(x+h) - f(x+2h)\} / (12h^2)$$

により微分の近似値を計算します。

この式において、「十分小さな  $x$  の増分」

$$h = 1/5^m$$

の値を、 $m=1, 2, 3, \dots$ と順次変更して計算していきます。

そして、前回の  $m$  の値による  $h$  の計算結果を使った  $f''(x)$  の値と、今回の  $m$  の値による  $h$  の計算結果を使った  $f''(x)$  の値が、指定した上位  $n$  桁目まで同じになったとき、計算を終了します。

• 基本的には終了条件  $n$  を省略した書式で計算を行なってください( $n$  を省略すると、自動的に  $n=7$  と設定して計算されます)。

答えの精度が出ないときのみ、 $n$  の値を変化させて計算を行なうことをお勧めします。

• 終了条件  $n$  の値を大きくすると、答えの精度が高くなるとは限りません。

## 2次微分計算の実例

**例** 関数  $y = x^3 + 4x^2 + x - 6$  の点  $x = 3$  における 2 次微分係数を求める。

ただし、ここでは終了条件に  $n = 6$  を入力する。

AC FUNCTION 1 (MATH) 3 ( $d^2/dx^2$ )

ALPHA X ^ 3 + 4 ALPHA X

X^2 + ALPHA X - 6 ,

(関数  $f(x)$  の入力)

$d^2/dx^2 (X^3 + 4X^2 + X - 6, _$

3 ,

(微分係数を求める点  $a$  の入力)

$d^2/dx^2 (X^3 + 4X^2 + X - 6, 3, _$

6 )

(終了条件  $n$  の入力)

$d^2/dx^2 (X^3 + 4X^2 + X - 6, 3, 6) _$

EXE

$d^2/dx^2 (X^3 + 4X^2 + X - 6, 3, 6)$

26

※ 関数  $f(x)$  には  $X$  の式しか使うことができません。 $X$  以外 ( $X$  を除く  $A \sim Z$ ) の変数は定数と見なされ、その変数メモリーに記憶されている数値を使って計算されます。

※ 終了条件  $n$ 、閉じカッコは省略することができます。

※ 計算精度は、原則として表示結果の最下位桁目  $\pm 1$  です。

## 2次微分計算の応用

- 2次微分同士の四則計算ができます。

$$\frac{d^2}{dx^2} f(a) = f''(a), \quad \frac{d^2}{dx^2} g(a) = g''(a)$$

とすると、

$$f''(a) + g''(a), \quad f''(a) \times g''(a) \quad \text{など}$$

- 2次微分計算の結果に対して、四則計算や関数計算ができます。

$$2 \times f''(a), \quad \log(f''(a)) \quad \text{など}$$

- 2次微分計算の書式の各項( $f(x), a, n$ )に、関数式を用いることができます。

$$\frac{x^2}{dx^2} (\sin x + \cos x, \sin 0.5) \quad \text{など}$$

- ※ 2次微分計算の書式の各項の中に、微分/2次微分計算式、積分計算式、またはΣ計算式を入力することはできません。

## 2次微分計算時の諸注意

- $n$ には1~15までの整数を入力してください。それ以外の値を入力したときは、エラー「Ma ERROR」となります。
- 2次微分計算中(カーソルが消えている状態)に **AC** を押すと、2次微分計算は中止されます。
- 三角関数の2次微分計算は、Radモードで行なってください。
- 2次微分計算では、F~Hの3本のメモリーが使用されます。計算内容はこれらのメモリー内容を読み出して、確認することができます。これらの変数メモリーは2次微分計算時に使用できませんので、注意してください。

使用メモリー	F	G	H
データ	$a$	$n$	$f''(a)$

また、2次微分計算実行後、変数メモリーXには微分係数  $a$  の値が記憶されます。

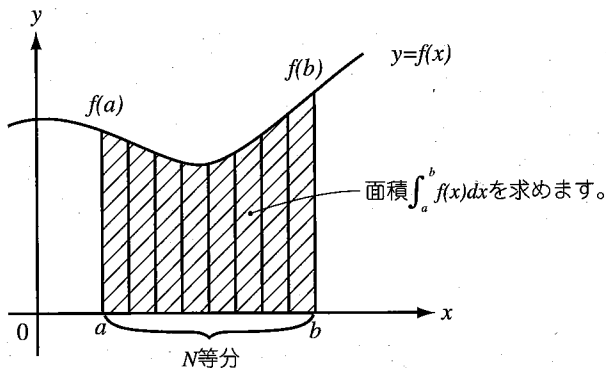
### 3-3. 積分計算

積分計算は内蔵関数(MATH)メニューから  $\boxed{1}$  ( $\int dx$ ) を押して、次の書式で積分計算式を入力することにより求められます。

$$\boxed{1} (\int dx) f(x) \boxed{a} \boxed{b} \boxed{n} \boxed{}$$

始点  $a$       終点  $b$       分割数  $N=2^n$  の  $n$  の値

$$f(f(x), a, b, n) \Rightarrow \int_a^b f(x) dx, N=2^n$$



積分計算とは、上の図に示すように区間  $a \leq x \leq b$  において常に  $f(x) \geq 0$  であり、かつ連続な関数  $y=f(x)$  の  $a$  から  $b$  までの積分値、すなわち斜線の部分の面積を求めるものです。

※ 区間  $a \leq x \leq b$  において  $f(x) < 0$  の場合、積分計算結果は負の値(すなわち面積の値に  $-1$  をかけたもの)となります。

## 積分計算の実例

例 1  $\int_1^5 (2x^2 + 3x + 4) dx$  を求める。ただし、分割数の  $n = 6$  とする。

AC FUNCTION 1 (MATH) 1 (f dx)

2 ALPHA X  $x^2$  + 3 ALPHA X

+ 4 )

(関数  $f(x)$  の入力)

$\int(2X^2+3X+4, _$

1 , 5 )

(積分区間  $[a, b]$  の入力)

$\int(2X^2+3X+4, 1, 5, _$

6 )

( $n$  の入力)

$\int(2X^2+3X+4, 1, 5, 6$   
 $) _$

EXE

$\int(2X^2+3X+4, 1, 5, 6$   
 $)$   
134.6666667

- ※ 関数  $f(x)$  には、 $X$  の式しか使うことができません。 $X$  以外 ( $X$  を除く  $A \sim Z$ ) の変数は定数と見なされ、その変数メモリーに記憶されている数値を使って計算されます。
- ※  $n$ 、閉じカッコは省略することができます。なお、 $n$  を省略すると、 $n$  は式に応じた適切な値が自動的に設定されます。
- ※ 積分計算はシンプソン法を用いるため、計算に時間がかかることがあります。
- ※ 計算精度は、原則として表示結果の最下位桁目  $\pm 1$  です。



## 積分計算の応用

- 積分同士の四則計算ができます。

$$\int_a^b f(x) dx + \int_c^d g(x) dx \text{ など}$$

- 積分計算の結果に対して、四則計算や関数計算ができます。

$$2 \times \int_a^b f(x) dx \text{ など}$$

$$\log\left(\int_a^b f(x) dx\right) \text{ など}$$

- 積分計算の書式の各項( $f(x)$ 、 $a$ 、 $b$ 、 $n$ )に、関数式を用いることができます。

$$\int_{\sin 0.5}^{\cos 0.5} (\sin x + \cos x) dx$$

$$= \int(\sin x + \cos x, \sin 0.5, \cos 0.5, 5)$$

※ 積分計算の書式の各項の中に、微分計算式、積分計算式、または  $\Sigma$  計算を入力することはできません。

## 積分計算時の諸注意

- 積分計算中(カーソルが消えている状態)に **AC** を押すと、積分計算は中止されます。
- 三角関数の積分計算は、Radモードで行なってください。
- 積分計算では、K~Nの4本のメモリーが使用されます。計算内容はこれらのメモリー内容呼び出して、確認することができます。これらの変数メモリーは積分計算時に使用できませんので、注意してください。

使用メモリー	K	L	M	N
データ	$a$	$b$	$N=2^n$	$\int_a^b f(x) dx$

また、積分計算実行後、変数メモリーXには積分区間の始点  $a$  の値が記憶されます。

例 2 例 1 の計算内容を確認する。

ALPHA K EXE

( $a$  の値)

K

1

ALPHA L EXE

( $b$  の値)

L

5

ALPHA M EXE

( $N=2^n$  の  $N$  の値)

M

64

ALPHA N EXE

( $\int_a^b f(x) dx$  の値)

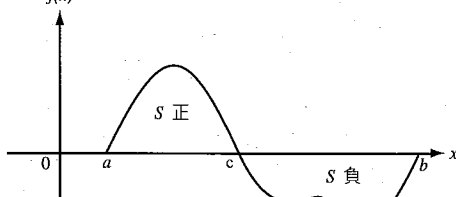
N

134.6666667

- 積分する関数の種類、積分区間における関数の正・負、または積分したい区間によっては求めた積分値の誤差が大きくなることがあります。また、計算結果の有効桁数が1桁以下のときは、エラー「Ma ERROR」となります。

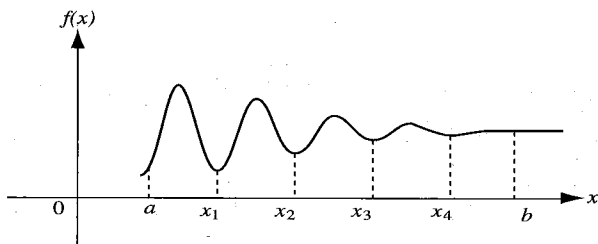
正確な積分値を求めるために、以下の点にご注意ください。

1. 周期関数や、積分区間によって関数  $f(x)$  の値が正・負になる場合は → 1 周期ごと、または正の部分と負の部分に分けて積分値を求め、各々を加算します。



$$\int_a^b f(x) dx = \underbrace{\int_a^c f(x) dx}_{\text{正の部分 (S正)}} + \underbrace{\left(-\int_c^b f(x) dx\right)}_{\text{負の部分 (S負)}}$$

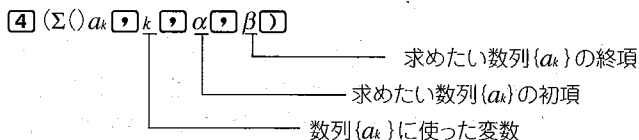
2. 積分区間の微小移動により、積分値が大きく変動する場合は  
 → 積分区間を分割して(変動の大きい所をより細かく分割する)積分値を求め、各々を加算します。



$$\int_a^b f(x) dx = \int_a^{x_1} f(x) dx + \int_{x_1}^{x_2} f(x) dx + \dots + \int_{x_4}^b f(x) dx$$

## 3-4. $\Sigma$ 計算

$\Sigma$ (シグマ)計算は内蔵関数(MATH)メニューから **4** ( $\Sigma$ )を押して、次の書式で  $\Sigma$ 計算式を入力することにより求められます。



$$\Sigma(a_k, k, \alpha, \beta) \Rightarrow \sum_{k=\alpha}^{\beta} a_k$$

$\Sigma$ 計算とは、数列  $\{a_k\}$  の部分 and

$$S = a_{\alpha} + a_{\alpha+1} + \dots + a_{\beta} = \sum_{k=\alpha}^{\beta} a_k$$

を求めるものです。

## Σ計算の実例

例  $\sum_{k=2}^6 (K^2 - 3K + 5)$  を求める。

AC FUNCTION 1 (MATH) 4 (Σ)  
ALPHA K  $x^2$  = 3 ALPHA K  
+ 5 )  
(数列  $\{a_k\}$  の入力)

Σ(K<sup>2</sup>-3K+5, \_

ALPHA K )  
(数列  $\{a_k\}$  に使った変数の入力)

Σ(K<sup>2</sup>-3K+5, K, \_

2 ) 6 )  
(求めたい数列  $\{a_k\}$  の初項  $\alpha$  と  
終項  $\beta$  の入力)

Σ(K<sup>2</sup>-3K+5, K, 2, 6)  
-

EXE

Σ(K<sup>2</sup>-3K+5, K, 2, 6)  
55

- ※ 数列  $\{a_k\}$  の関数式に、変数は1つしか使うことはできません。
- ※ 求めたい数列  $\{a_k\}$  の初項  $\alpha$  と終項  $\beta$  には、整数を入力するようにしてください。
- ※ 閉じカッコは、省略することができます。

## Σ計算の応用

- Σ計算式同士の四則計算ができます。

$$S_n = \sum_{k=1}^n a_k, \quad T_n = \sum_{k=1}^n b_k$$

とすると、

$$S_n + T_n, S_n - T_n \text{ など}$$

- Σ計算の結果に対して、四則計算や関数計算ができます。

$$2 \times S_n, \log(S_n) \text{ など}$$

- Σ計算の書式の各項( $a_k, \alpha, \beta$ )に、関数式を用いることができます。

$$\Sigma(x^2+x, x, 2^2, 5^2+1) \text{ など}$$

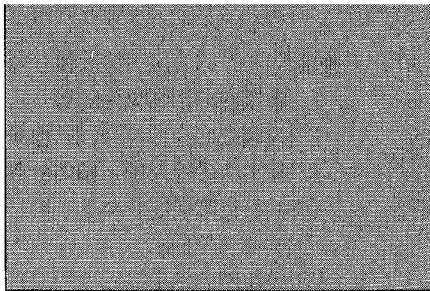
- ※ Σ計算の書式の各項の中に、微分計算式、積分計算式、またはΣ計算式を入力することはできません。

## Σ計算時の諸注意

- 終項  $\beta$  には、初項  $\alpha$  より大きな値を入力してください。終項  $\beta$  の値が初項  $\alpha$  の値より小さいときは、エラー「Ma ERROR」となります。
- Σ計算中(カーソルが消えている状態)に **AC** を押すと、Σ計算は中止されます。



# 4



## 複素数計算

---

複素数を使って、以下の計算ができます。

- (1)複素数の加減乗除計算
- (2)複素数の逆数、平方根、自乗計算
- (3)複素数の絶対値／偏角計算
- (4)共役複素数を求める
- (5)実数部の抽出
- (6)虚数部の抽出

## 4-1. 複素数計算を行なう前に

以下のように操作して、複素数計算 (COMPLX) メニュー表示を呼び出して  
ください。

FUNCTION [2] (COMPLX)

1. Abs	2. Arg
3. Conjg	4. ReP
5. ImP	

メニューの意味は、以下のとおりです。

- ・[1. Abs]..... 複素数の絶対値を求める。
- ・[2. Arg] ..... 複素数の偏角を求める。
- ・[3. Conjg] ..... 共役複素数を求める。
- ・[4. ReP]..... 複素数の実数部分を抽出。
- ・[5. ImP] ..... 複素数の虚数部分を抽出。



## 4-2. 複素数計算の仕方

### 加減乗除計算

複素数計算はマニュアル計算と同じように加減乗除計算、カッコ計算、メモリ計算などができます。

例 1  $(1+2i) + (2+3i) =$

AC ( 1 + 2 i ) +  
( 2 + 3 i ) EXE

(1+2i)+(2+3i)  
3+5i

例 2  $(2+i) \times (2-i) =$

AC ( 2 + i ) ×  
( 2 - i ) EXE

(2+i)×(2-i)  
5

### 逆数、平方根、自乗計算

複素数の逆数計算、平方根、自乗計算ができます。

例  $\sqrt{3+i} =$

AC  $\sqrt{\quad}$  ( 3 + i ) EXE

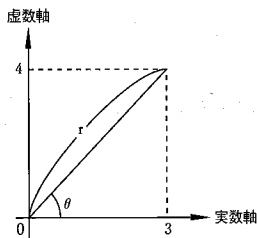
$\sqrt{3+i}$   
1.755317302  
+0.2848487846i

### 絶対値／偏角計算

$[Z=a+bi]$ の形で表わされる複素数を複素平面(ガウス平面)上の座標と見なして、絶対値(|Z|)と偏角(arg)を求めます。

例

複素数「 $3+4i$ 」の絶対値 ( $r$ ) と偏角 ( $\theta$ ) を求める  
(Deg モード指定時)。



AC FUNCTION 2 (COMPLX)  
1 (Abs) ( 3 + 4 i ) EXE  
(絶対値の算出)

Abs (3+4i)  
5

AC FUNCTION 2 (COMPLX)  
2 (Arg) ( 3 + 4 i ) EXE  
(偏角の算出)

Arg (3+4i)  
53.13010235

※ 偏角の計算結果は、現在設定されている角度単位モード(度・ラジアン・  
グラード)によって異なります。

## 共役複素数を求める

複素数「 $a+bi$ 」に対して、共役複素数は「 $a-bi$ 」となります。

例

複素数「 $2+4i$ 」の共役複素数を求める。

AC FUNCTION 2 (COMPLX)  
3 (Conj) ( 2 + 4 i ) EXE

Conj (2+4i)  
2-4i

## 実数部／虚数部の抽出

複素数「 $a+bi$ 」の実数部は「 $a$ 」、虚数部は「 $b$ 」となります。

**例** 複素数「 $2+5i$ 」の実数部および虚数部を求める。

AC FUNCTION 2 (COMPLX)

4 (ReP) ( 2 + 5 i ) EXE

(実数部の抽出)

ReP (2+5i)

2

AC FUNCTION 2 (COMPLX)

5 (ImP) ( 2 + 5 i ) EXE

(虚数部の抽出)

ImP (2+5i)

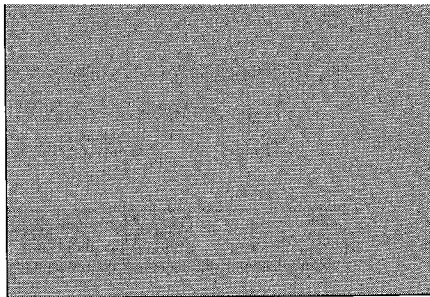
5

### 4-3. 複素数計算時の諸注意

- 虚数部の表示桁数は、仮数部9桁、指数部2桁です。ただし、指数部が表示されないときは、仮数部10桁となります。
- 複素数の実数部または虚数部が16桁を超える場合は1行で結果表示せず、実数部と虚数部を分けて2行で表示します。
- 複素数の実数部または虚数部が0のとき、実数部または虚数部は表示しません。
- 複素数を変数メモリーに記憶させると、虚数部記憶用に10バイト必要になります(40ページ参照)。
- 複素数計算において使うことができる機能は、以下のとおりです。
  - $\sqrt{\quad}$ ,  $x^2$ ,  $x^{-1}$
  - Int, Frac, Rnd, Intg, Fix, Sci, ENG,  $\overleftarrow{\text{ENG}}$ ,  $\overleftarrow{\text{ON}}$ ,  $\overleftarrow{\text{OFF}}$ ,  $a^b/c$ ,  $d/c$



# 5



## 漸化式計算

---

以下の2種類のタイプの漸化式を計算することができます。

- (1)  $a_n$ 、 $n$ で構成される数列  $\{a_n\}$  の一般項
- (2)  $a_{n+1}$ 、 $a_n$ 、 $n$ で構成される線型2項間漸化式

## 5-1. 漸化式計算を行なう前に

初めに、計算を行なう漸化式のタイプを指定してください。

- (1) **MODE** **6** ( $a_n$ )と操作します。



$a_n = \_$

※数列  $\{a_n\}$  の一般項を指定時

※ 漸化式が入力されている場合は、その漸化式が表示されます。

- (2) **FUNCTION** を押して、機能メニュー表示を呼び出します。

### 1. 数列 $\{a_n\}$ の一般項が指定されている場合



1.n      2.MATH  
3.TYPE

メニューの意味は、以下のとおりです。

- ・「1.n」.....  $n$  を入力。
- ・「2.MATH」..... 内蔵関数メニュー表示を呼び出す。  
→48ページ参照
- ・「3.TYPE」..... 漸化式タイプ指定表示を呼び出す。

## 2. 線型2項間漸化式が指定されている場合

1. n	2. an
3. MATH	4. TYPE

メニューの意味は、以下のとおりです。

- 「1. n」.....  $n$  を入力。
- 「2. an」.....  $a_n$  を入力。
- 「3. MATH」..... 内蔵関数メニュー表示を呼び出す。  
→48ページ参照
- 「4. TYPE」..... 漸化式タイプ指定表示を呼び出す。

(3) **3** (TYPE) または **4** (TYPE) を押して漸化式タイプ指定表示を呼び出し、漸化式タイプを指定します。

<SELECT an TYPE>
1. an=An+B
2. an+1=Aan+Bn+C

メニューの意味は、以下のとおりです。

- 「1. an=An+B」..... 数列  $\{a_n\}$  の一般項に指定。
- 「2. an+1=Aan+Bn+C」..... 線型2項間漸化式に指定。

※ 画面上の「an=An+B」は、 $\{a_n\}$  の一般項「 $a_n=Axn+B$ 」を表わしています。

## 5-2. 漸化式計算の実例

次の例題を計算してみましょう。

**例 1**  $a_n = n + 2$  を入力し、変数  $n$  の値を 2 から 6 まで変化させたときの  $a_n$  および  $\Sigma a_n$  の値を計算する。

(1) 漸化式タイプを数列  $\{a_n\}$  の一般項に指定します。

①

$a_n =$  \_

(2) 式を入力します。

FUNCTION

1.n    2.MATH  
3.TYPE

① (n) ⊕ ②

$a_n = n + 2$  \_

(3) EXE を押します。数表レンジ設定画面になり、変数  $n$  の初期値が表示されます。

EXE

```
**TABLE Range**  
n  
Start? 1
```

数表レンジ設定画面には変数  $n$  の初期値が表示されます。この条件をもとに変数  $n$  の値を1ずつ増加させて計算します。

(4) 変数  $n$  の初期値を変更します。

②

```
**TABLE Range**  
n  
Start? 2_
```



(5) **EXE** を押して、 $a_n$  および  $\Sigma a_n$  を計算します。

**EXE**

$a_n=n+2$	
$n=$	2
$a_n=$	4
$\Sigma a_n=$	7

**EXE**

$a_n=n+2$	
$n=$	3
$a_n=$	5
$\Sigma a_n=$	12

⋮

**EXE**

$a_n=n+2$	
$n=$	6
$a_n=$	8
$\Sigma a_n=$	33

※ **EXE** を押し続けると、変数  $n=9999999999$  まで計算することができます。

※ 計算終了後は、**EXIT** を押してください。(2)の画面に戻ります。

### 【参考】

$\{a_n\}$  の一般項には  $n$  の1次式以外に指数式 ( $\lceil a_n=2^n-1 \rceil$  など)、分数式 ( $\lceil a_n=(n+1) \div n \rceil$  など)、無理式 ( $\lceil a_n=\sqrt{n}-\sqrt{(n-1)} \rceil$  など)、または三角関数式 ( $\lceil a_n=\sin 2n\pi \rceil$  など) などを入力して、計算することもできます。

**例 2**

$a_{n+1}=a_n+5$  を入力し、変数  $n$  の値を1から5まで変化させたときの  $a_n$  および  $\Sigma a_n$  の値を計算する。ただし、 $a_1=2$  とする。

(1) 漸化式タイプを線型2項間漸化式に指定します。

**2**

$a_{n+1} =$  \_

(2) 式を入力します。

**FUNCTION**

1. n      2.  $a_n$   
3. MATH 4. TYPE

**2** ( $a_n$ ) **+** **5**

$a_{n+1} = a_n + 5$  \_

(3) **EXE**を押します。数表レンジ設定画面No.1になり、数列 $\{a_n\}$ の第1項 $a_1$ の値が表示されます。

**EXE**

\*\*TABLE Range\*\*

$a_1$  ?

1

(4) 第1項 $a_1$ の値を変更します。

**2**

\*\*TABLE Range\*\*

$a_1$  ?

2 \_

(5) **EXE**を押します。数表レンジ設定画面No.2になり、変数 $n$ の初期値が表示されます。

**EXE**

\*\*TABLE Range\*\*

n  
Start?

1

数表レンジ設定画面No.2には変数 $n$ の初期値が表示されます。この条件をもとに変数 $n$ の値を1ずつ増加させて計算します。

(6) **EXE** を押して、 $a_n$  および  $\Sigma a_n$  を計算します。

**EXE**

$a_{n+1}=a_n+5$	
$n=$	1
$a_n=$	2
$\Sigma a_n=$	2

**EXE**

$a_{n+1}=a_n+5$	
$n=$	2
$a_n=$	7
$\Sigma a_n=$	9

⋮

**EXE**

$a_{n+1}=a_n+5$	
$n=$	5
$a_n=$	22
$\Sigma a_n=$	60

※ **EXE** を押し続けると、変数  $n=9999999999$  まで計算することができます。

※ 計算終了後は、**EXIT** を押してください。(2)の画面に戻ります。

## 5 - 3. 漸化式計算時の諸注意

● 数値レンジの値を入力するとき、以下の点に注意してください。

・**Start**の値が負の数、小数、分数のとき

負の数は正の数に、小数や分数は整数部のみを取り出して計算します。

・**Start**の値が大きいとき

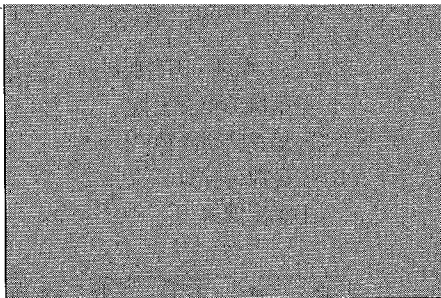
計算に時間がかかる場合があります。

● 計算結果がエラーとなるときは、エラー「Ma ERROR」が表示されます。

● 三角関数などを含む漸化式の計算結果が表示されている状態から角度単位を指定し直しても、計算結果は変更されません。角度単位を変更して計算し直したいときは、COMPモードに戻って角度単位を指定し直した後、**MODE** **6** ( $a_n$ )と操作して計算し直してください。



# 6



2進・8進・10進・16進計算

---

2進・8進・10進・16進数を使って、以下の計算ができます。

- (1) 基数の相互変換
- (2) 加減乗除計算
- (3) 負数の計算
- (4) 論理演算

## 6 - 1. 2進・8進・10進・16進計算を行なう前に

(1) メインメニュー表示からBASE-Nモードを指定します。

**MODE** **2** (BASE-N)

現在の基数

**\*\*DEC MODE\*\***

(2) 機能メニュー表示を呼び出します。

**FUNCTION**

1.BASE-N 2.PROG  
3.Mcl

メニューの意味は、以下のとおりです。

- ・「1.BASE-N」.....  $n$  進計算メニュー表示No.1を呼び出す。
- ・「2.PROG」..... プログラム命令メニュー表示を呼び出す。  
→167ページ参照
- ・「3.Mcl」..... すべての変数メモリー内容を消去。  
→34ページ参照

(3)  $n$  進計算メニュー表示No.1を呼び出します。

**1** (BASE-N)

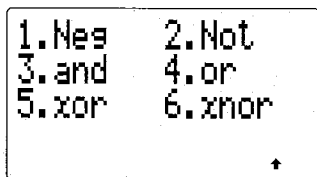
1.Dec	2.Hex
3.Bin	4.Oct
5.d	6.h
7.b	8.o

↓

メニューの意味は、以下のとおりです。

- 「1. Dec」..... 10進計算状態になる。
- 「2. Hex」..... 16進計算状態になる。
- 「3. Bin」..... 2進計算状態になる。
- 「4. Oct」..... 8進計算状態になる。
- 「5. d」..... 入力数値の基数を10進数に指定。
- 「6. h」..... 入力数値の基数を16進数に指定。
- 「7. b」..... 入力数値の基数を2進数に指定。
- 「8. o」..... 入力数値の基数を8進数に指定。

(4) ▼を押すと、*n*進計算メニュー表示No.2になります。



メニューの意味は、以下のとおりです。

- 「1. Neg」..... 置数の負数を求める。
- 「2. Not」..... 論理演算の否定を求める。
- 「3. and」..... 論理演算の論理積を求める。
- 「4. or」..... 論理演算の論理和を求める。
- 「5. xor」..... 論理演算の排他的論理和<不一致>を求める。
- 「6. xnor」..... 論理演算の排他的論理和の否定<一致>を求める。

※ ▲を押すと、*n*進計算メニュー表示No.1に戻ります。

## 6-2. 基数の指定 (2進・8進・10進・16進数指定)

基数の指定は、 $n$  進計算メニュー表示No.1から **1** (Dec)、**2** (Hex)、**3** (Bin)、**4** (Oct)を押します。

なお、結果表示中に基数を指定し直すと、指定した基数に数値が変換されます。

**例**  $22_{(10)}$  を 2 進数、8 進数に変換する。

**AC** **FUNCTION** **1** (BASE-N)

**1** (Dec) **2** **2** **EXE**

22  
22

**FUNCTION** **1** (BASE-N)

**3** (Bin)

22  
0000000000000000  
00000000000010110

**FUNCTION** **1** (BASE-N)

**4** (Oct)

22  
00000000026

### 入力数値の基数指定

数値の先頭に記号「d」、「h」、「b」、「o」を付けると、入力する数値の基数を現在指定されている基数と異なる基数に指定することができます。

**例**  $123_{(10)}$ 、 $1010_{(2)}$  を 16 進数に変換する。

**AC** **FUNCTION** **1** (BASE-N)

**2** (Hex)

**FUNCTION** **1** (BASE-N) **5** (d)

**1** **2** **3** **EXE**

d123  
0000007B



FUNCTION 1 (BASE-N) 7 (b)

1 0 1 0 EXE

b1010

0000000A

## 各基数で扱える数値

それぞれの基数では扱える数字は、以下のとおりです。

基数	有効数値
2進数	0,1
8進数	0,1,2,3,4,5,6,7
10進数	0,1,2,3,4,5,6,7,8,9
16進数	0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F

画面上ではアルファベット文字と区別して表示します。

※ 有効数値以外の数値を使うときは、数値の先頭に記号「b」、「o」、「d」、「h」を付けないと、計算時にエラー「Syn ERROR」となります。

## 6-3. 加減乗除計算

例 1  $10111_{(2)} + 11010_{(2)}$  を求める。

AC FUNCTION 1 (BASE-N) 3 (Bin)

1 0 1 1 1 +

1 1 0 1 0 EXE

```

10111+11010
000000000000000000
000000000000110001
    
```

例 2  $123_{(8)} \times ABC_{(16)}$  を 10 進数と 16 進数で求める。

AC FUNCTION 1 (BASE-N) 1 (Dec)

FUNCTION 1 (BASE-N) 8 (o)

1 2 3 X

FUNCTION 1 (BASE-N) 6 (h)

A B C EXE

o123xhABC

228084

FUNCTION 1 (BASE-N) 2 (Hex)

o123xhABC

00037AF4

## 6 - 4. 負数計算および論理演算

### 負数計算

2進、8進、16進の負数は、2の補数を取ります。

例  $110010_{(2)}$  の負数を求める。

AC FUNCTION 1 (BASE-N) 3 (Bin)

FUNCTION 1 (BASE-N) ▼

1 (Neg) 1 1 0 0 1 0 EXE

Neg 110010

1111111111111111

11111111111001110

## 論理演算

論理演算は、論理積 (and)、論理和 (or)、排他的論理和<不一致> (xor)、排他的論理和の否定<一致> (xnor)、否定 (Not) について行なうことができます。

例 1  $120_{(16)}$  and  $AD_{(16)}$  を求める。

AC FUNCTION 1 (BASE-N)

2 (Hex) 1 2 0

FUNCTION 1 (BASE-N) ▼

3 (and) A D EXE

120andAD  
00000020

例 2  $36_{(8)}$  or  $1110_{(2)}$  を 8 進数で求める。

AC FUNCTION 1 (BASE-N)

4 (Oct) 3 6

FUNCTION 1 (BASE-N) ▼ 4 (or)

FUNCTION 1 (BASE-N) 7 (b)

1 1 1 0 EXE

36orb1110  
00000000036

例 3  $2FFFED_{(16)}$  の否定を求める。

AC FUNCTION 1 (BASE-N) 2 (Hex)

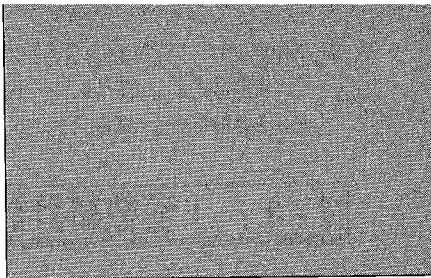
FUNCTION 1 (BASE-N) ▼

2 (Not) 2 F F F E D EXE

Not 2FFFE  
FFD00012



# 7



## 統計計算



統計計算では、1変数および2変数の統計計算が行なえます。

- (1) 1変数統計計算 (SDモード) ..... 標準偏差計算
- (2) 2変数統計計算 (LRモード) ..... 回帰計算

## 7-1. 1変数統計計算 (標準偏差計算)

SDモードを指定すると、母標準偏差、標本標準偏差、データの平均、データの2乗和、データの総和、データ数を求めることができます。

### ■ データの入力

- (1) **FUNCTION** **6** (DSP/CLR) **6** (ScI) **EXE** と操作して、統計計算用メモリーU、V、Wの内容を消去します。

※ 計算を行なう前に、必ず上記操作によりメモリー内容を消去してください。

- (2) 以下のように操作して、データを入力します。

[データの値] **DT** (= **M+**)

**例** 入力するデータ : 10、20

キー操作 : 10 **DT** 20 **DT**

※ 続けて同じデータを入力するときは、以下のように操作してください。

[データの値] **DT** **DT**

※ 同じデータを複数個入力するときは、以下のように操作してください。

[データの値] **SHIFT** **;** [データの個数] **DT**

### ■ データの削除

- (1) 入力する前 (**DT** を押す前) のデータを削除するときは……

**AC** を押してください。

- (2) 入力されたデータを削除するときは……

[削除するデータの値] **SHIFT** **CL** と操作してください。

例) 削除するデータ : 10、20

キー操作 : 10 **SHIFT** **CL**

20 **SHIFT** **CL**

※ 同じデータを複数個削除するときは、以下のように操作してください。

[削除するデータの値] **SHIFT** **;** [削除するデータの個数] **SHIFT** **CL**

## ■統計計算の実行

- データの統計処理は以下のように操作して、1変数統計計算メニュー表示を呼び出して行ないます。

**FUNCTION** **7** (STAT)

1. $\bar{x}$	2. $x\sigma_n$
3. $x\sigma_{n-1}$	4. $\Sigma x^2$
5. $\Sigma x$	6. $n$
7. $t()$	

メニューの意味は、以下のとおりです。

- ・「1.  $\bar{x}$ 」..... $\bar{x}$ ( $x$ データの平均)を求める。
- ・「2.  $x\sigma_n$ 」..... $x\sigma_n$ ( $x$ データの母標準偏差)を求める。
- ・「3.  $x\sigma_{n-1}$ 」..... $x\sigma_{n-1}$ ( $x$ データの標本標準偏差)を求める。
- ・「4.  $\Sigma x^2$ 」..... $\Sigma x^2$ ( $x$ データの2乗和)を求める。
- ・「5.  $\Sigma x$ 」..... $\Sigma x$ ( $x$ データの総和)を求める。
- ・「6.  $n$ 」.....データ数を求める。
- ・「7.  $t()$ 」..... $t$ 検定を計算するとき使用。

- 以下のように操作して、データの統計処理結果を一覧表示することができます。

FUNCTION 8 (RESULTS)

```

Deviation
x̄ = 53.375
σn = 1.316956719
σn-1 = 1.407885953
  
```

▼ ↓ ↑ ▲

```

Sum
Σx² = 22805
Σx = 427
n = 8
  
```

※統計処理結果は、最大12桁まで表示されます。

- ※  $\Sigma x^2$ 、 $\Sigma x$ 、 $n$ の値はそれぞれ変数メモリーU、V、Wに記憶されます。これらの変数メモリーは統計計算時に使用できませんので、注意してください。
- ※ 平均と標準偏差は、それぞれ下記の式によって計算されます。

〈平均〉

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n} = \frac{\Sigma x}{n}$$

〈標準偏差〉

$$\sigma_n = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}} = \sqrt{\frac{\Sigma x^2 - (\Sigma x)^2/n}{n}}$$

有限母集団全部のデータを使い、その集団の標準偏差を求める。

$$\sigma_{n-1} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}} = \sqrt{\frac{\Sigma x^2 - (\Sigma x)^2/n}{n-1}}$$

集団中のサンプルデータを使い、その集団の標準偏差を推定する。



例

データが 55、54、51、55、53、53、54、52 のとき、そのデータの統計処理を行なう。また、データの不偏分散および各データと平均との差を求める。

**FUNCTION** **6** (DSP/CLR) **6** (ScI) **EXE**  
(メモリーのクリアー)

ScI  
0

55 **DT** 54 **DT** 51 **DT** 55 **DT**  
53 **DT** **DT** 54 **DT** 52 **DT**  
(データの入力)

52  
X= 52  
f= 1  
n= 8  
SD

**FUNCTION** **8** (RESULTS)  
(統計処理結果の表示)

Deviation  
 $\bar{x}$  = 53.375  
 $\sigma_n$  = 1.316956719  
 $\sigma_{n-1}$  = 1.407885953  
↓

▽

Sum  
 $\sum x^2$  = 22805  
 $\sum x$  = 427  
n = 8  
↑

**EXIT** **FUNCTION** **7** (STAT)

**3** ( $x_{0n-1}$ )  **$x^2$**  **EXE**

(データの不偏分散)

$x_{0n-1}^2$   
1.982142857

55 **⇐** **FUNCTION** **7** (STAT)

**1** ( $\bar{x}$ ) **EXE**

(平均との差)

55- $\bar{x}$   
1.625

54 **⇐** **FUNCTION** **7** (STAT)

**1** ( $\bar{x}$ ) **EXE**

54- $\bar{x}$   
0.625

(以下省略)

## 7-2. $t$ 検定計算

統計データの平均(標本平均)と標本標準偏差より、 $t$  検定のための  $t$  の値を求めることができます。

—  $t$  検定とは —

標本平均  $\bar{x}$  と標本標準偏差  $x\sigma_{n-1}$  から、母平均  $\mu$  がある値だといえるか否かを一定の危険率の範囲内で判定することをいいます。

※  $t$  の値は、下記の式によって計算されます。

$$t = \frac{(\bar{x} - \mu)}{\frac{x\sigma_{n-1}}{\sqrt{n}}}$$

$\bar{x}$  :  $x$  データの平均  
 $x\sigma_{n-1}$  :  $x$  データの標本標準偏差  
 $n$  : データ数  
 $\mu$  : 仮定の母平均

**例** サンプルデータが 55、54、51、55、53、53、54、52 のとき、その母平均は 53 といえるか。危険率 5% で  $t$  検定する。

**FUNCTION** **6** (DSP/CLR)

**6** (ScI) **EXE**

(メモリーのクリアー)

55 **DT** 54 **DT** 51 **DT** 55 **DT**

53 **DT** **DT** 54 **DT** 52 **DT**

(データの入力)

**FUNCTION** **7** (STAT) **7** (t)

53 **]** **EXE**

( $t$  の値を算出)

ScI

0

52

$\bar{x}$ =

52

f=

1

n=

8

SD

t(53)

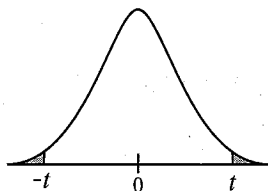
0.7533708035

上記表示より、 $t$  の値として  $t(53) = 0.7533708035$  を得ました。

$t$  分布表によれば危険率5%、自由度  $n-1=8-1=7$ 、両側検定時の  $t$  の値は約2.365となります。算出された  $t$  の値はこの値より小さいため、母平均  $\mu=53$  とする仮説は採択されることとなります。

※  $t$  分布表

表の上側の数字は、ある自由度に対して、 $t$  の絶対値が表中の値を超える確率(両側確率)



P(確率) 自由度	0.2	0.1	0.05	0.01
1	3.078	6.314	12.706	63.657
2	1.886	2.920	4.303	9.925
3	1.638	2.353	3.182	5.841
4	1.533	2.132	2.776	4.604
5	1.476	2.015	2.571	4.032
6	1.440	1.943	2.447	3.707
7	1.415	1.895	2.365	3.499
8	1.397	1.860	2.306	3.355
9	1.383	1.833	2.262	3.250
10	1.372	1.812	2.228	3.169
15	1.341	1.753	2.131	2.947
20	1.325	1.725	2.086	2.845
25	1.316	1.708	2.060	2.787
30	1.310	1.697	2.042	2.750
35	1.306	1.690	2.030	2.724
40	1.303	1.684	2.021	2.704
45	1.301	1.679	2.014	2.690
50	1.299	1.676	2.009	2.678
60	1.296	1.671	2.000	2.660
80	1.292	1.664	1.990	2.639
120	1.289	1.658	1.980	2.617
240	1.285	1.651	1.970	2.596
$\infty$	1.282	1.645	1.960	2.576

## 7-3. 2変数統計計算 (回帰計算)

LRモードを指定すると、回帰計算を行なうことができます。

### 2変数統計計算(直線回帰計算)の仕方

回帰式を  $y = A+Bx$  と定義して、計算します。

#### ■ データの入力

- (1) **FUNCTION** **6** (DSP/CLR) **6** (ScI) **EXE** と操作して、統計計算用メモリーP、Q、R、U、V、Wの内容を消去します。

※ 計算を行なう前に、必ず上記操作によりメモリー内容を消去してください。

- (2) 以下のように操作して、データを入力します。

[x データの値] **□** [y データの値] **DT**

**例** 入力するデータ : 10 / 20、20 / 30

キー操作 : 10 **□** 20 **DT**

20 **□** 30 **DT**

※ 続けて同じデータを入力するときは、以下のように操作してください。

[x データの値] **□** [y データの値] **DT DT**

※ 同じデータを複数個入力するときは、以下のように操作してください。

[x データの値] **□** [y データの値] **SHIFT** **□**

[データの個数] **DT**

#### ■ データの削除

- (1) 入力する前(**DT**を押す前)のデータを削除するときは……

**AC** を押してください。

- (2) 入力されたデータを削除するときは……

[削除する x データの値] **□** [削除する y データの値] **SHIFT** **CL**

と操作してください。

例

削除するデータ : 10 / 20、20 / 30

キー操作 : 10  $\square$  20  $\square$   $\square$   $\square$

20  $\square$  30  $\square$   $\square$   $\square$

※ 同じデータを複数個削除するときは、以下のように操作してください。

[削除する  $x$  データの値]  $\square$  [削除する  $y$  データの値]

$\square$  [削除するデータの個数]  $\square$   $\square$

## ■ 統計計算の実行

- データの統計処理は、2変数統計計算メニュー表示を呼び出して行ないます。

(1) 以下のように操作して、2変数統計計算メニュー表示No.1を呼び出します。

$\square$   $\square$  (STAT)

1. $\bar{x}$	2. $x\sigma_n$
3. $x\sigma_{n-1}$	4. $\bar{y}$
5. $y\sigma_n$	6. $y\sigma_{n-1}$

メニューの意味は、以下のとおりです。

- ・[1.  $\bar{x}$ ] .....  $\bar{x}$  ( $x$  データの平均) を求める。
- ・[2.  $x\sigma_n$ ] .....  $x\sigma_n$  ( $x$  データの母標準偏差) を求める。
- ・[3.  $x\sigma_{n-1}$ ] .....  $x\sigma_{n-1}$  ( $x$  データの標本標準偏差) を求める。
- ・[4.  $\bar{y}$ ] .....  $\bar{y}$  ( $y$  データの平均) を求める。
- ・[5.  $y\sigma_n$ ] .....  $y\sigma_n$  ( $y$  データの母標準偏差) を求める。
- ・[6.  $y\sigma_{n-1}$ ] .....  $y\sigma_{n-1}$  ( $y$  データの標本標準偏差) を求める。

(2) **▼**を押すと、2変数統計計算メニュー表示No.2になります。



1. $\Sigma x^2$	2. $\Sigma x$
3. $n$	4. $\Sigma y^2$
5. $\Sigma y$	6. $\Sigma xy$

↑↓

メニューの意味は、以下のとおりです。

- ・「1.  $\Sigma x^2$ 」.....  $\Sigma x^2$  ( $x$ データの2乗和)を求める。
- ・「2.  $\Sigma x$ 」.....  $\Sigma x$  ( $x$ データの総和)を求める。
- ・「3.  $n$ 」..... データ数を求める。
- ・「4.  $\Sigma y^2$ 」.....  $\Sigma y^2$  ( $y$ データの2乗和)を求める。
- ・「5.  $\Sigma y$ 」.....  $\Sigma y$  ( $y$ データの総和)を求める。
- ・「6.  $\Sigma xy$ 」.....  $\Sigma xy$  ( $x$ データ・ $y$ データの積和)を求める。

※ **▲**を押すと、2変数統計計算メニュー表示No.1に戻ります。

(3) **▼**を押すと、2変数統計計算メニュー表示No.3になります。



1. $A$	2. $B$
3. $r$	4. $\hat{x}$
5. $\hat{y}$	

↑

メニューの意味は、以下のとおりです。

- ・「1.  $A$ 」..... 回帰式の定数項 $A$ を求める。
- ・「2.  $B$ 」..... 回帰式の回帰係数 $B$ を求める。
- ・「3.  $r$ 」..... 相関係数 $r$ を求める。
- ・「4.  $\hat{x}$ 」.....  $\hat{x}$  ( $x$ の推定値)を求める。
- ・「5.  $\hat{y}$ 」.....  $\hat{y}$  ( $y$ の推定値)を求める。

※ **▲**を押すと、2変数統計計算メニュー表示No.2に戻ります。

- 以下のように操作して、データの統計処理結果を一覧表示することができます。

FUNCTION 8 (RESULTS)

```

Regression
A =      997.4
E =      0.56
r = 0.982607368
  
```

▼ ↓ ↑ ▲

```

Deviation x
x̄ =      20
σn = 7.071067811
σn-1 = 7.90569415
  
```

▼ ↓ ↑ ▲

```

Deviation y
ȳ =     1008.6
σn = 4.029888335
σn-1 = 4.50555213
  
```

▼ ↓ ↑ ▲

```

Sum x
Σx² =     2250
Σx =     100
n =      5
  
```

▼ ↓ ↑ ▲

```

Sum y
Σy² =    5086451
Σy =     5043
Σxy =    101000
  
```

※ 統計処理の結果は、10桁まで表示されます。(指数表示のときは仮数部6桁、指数部2桁で表示されます。)

※  $\Sigma x^2, \Sigma x, n, \Sigma y^2, \Sigma y, \Sigma xy$ の値はそれぞれ変数メモリー-U, V, W, P, Q, Rに記憶されます。これらの変数メモリーは統計計算時に使用できませんので、注意してください。

※ 回帰式の定数項A、回帰係数B、相関係数  $r, x$  および  $y$  の推定値はそれぞれ下記の式によって計算されます。

$$A = \frac{\Sigma y - B \cdot \Sigma x}{n} \qquad B = \frac{n \cdot \Sigma xy - \Sigma x \cdot \Sigma y}{n \cdot \Sigma x^2 - (\Sigma x)^2}$$

$$r = \frac{n \cdot \Sigma xy - \Sigma x \cdot \Sigma y}{\sqrt{\{n \cdot \Sigma x^2 - (\Sigma x)^2\} \{n \cdot \Sigma y^2 - (\Sigma y)^2\}}}$$

$$\hat{y} = A + Bx \qquad \hat{x} = \frac{y - A}{B}$$

**例** 下記のデータより回帰式と相関係数を求め、回帰式をもとに温度 18°C のときの気圧と 1000 ヘクトパスカルの温度を推定する。

温度(°C)	気圧(hpa)
10	1003
15	1005
20	1010
25	1011
30	1014

FUNCTION **6** (DSP/CLR)

**6** (Scl) EXE

(メモリーのクリアー)

Scl 0

10  $\rightarrow$  1003 **DT** 15  $\rightarrow$  1005 **DT**

20  $\rightarrow$  1010 **DT** 25  $\rightarrow$  1011 **DT**

30  $\rightarrow$  1014 **DT**

(データの入力)

30, 1014  
 X= 30  
 Y= 1014  
 n= 5  
 LR



**FUNCTION** **8** (RESULTS)

(統計処理結果の表示)

```
Regression
A = 997.4
B = 0.56
r = 0.982607368
  ↓
```

**EXIT** 18 **FUNCTION** **7** (STAT) **▼** **▼**

**5** (y) **EXE**

(18°Cのときの気圧)

```
180
1007.48
```

1000 **FUNCTION** **7** (STAT) **▼** **▼**

**4** (x) **EXE**

(1000hpaのときの温度)

```
10000
4.642857143
```

## 2変数統計計算(直線回帰計算)の応用

直線回帰( $y=A+Bx$ )の応用として、対数回帰、指数回帰、べき乗回帰も計算できます。

### ■ 対数回帰計算

回帰式を  $y=A+B \cdot \ln x$  と定義して、計算します。

#### ● データの入力

(1) **FUNCTION** **6** (DSP/CLR) **6** (ScI) **EXE**と操作して、統計計算用メモリーP、Q、R、U、V、Wの内容を消去します。

(2) 以下のように操作して、データを入力します。

**ln** (xデータの値) **▶** (yデータの値) **DT**

※ 同じデータを複数個入力するときは、直線回帰計算でのデータ入力の仕方と同じ手順で行なうことができます。

## ● データの削除

入力されたデータの削除は、直線回帰計算でのデータ削除の仕方と同じ手順で行なうことができます。

$\boxed{\ln} \boxed{10} \boxed{\blacktriangleright} \boxed{20} \boxed{\text{SHIFT}} \boxed{\text{CL}}$

## ● 計算の仕方

下記の式に合わせて計算することにより、答えを求めることができます。

・  $x$  の推定値  $= e^{y^2}$

・  $y$  の推定値  $= \ln x^2$

回帰式  $y = A + B \cdot \ln x$  は  $\ln x = x$  とおくと、直線回帰式  $y = a + bx$  と考えられます。このため、定数項  $A$ 、回帰係数  $B$ 、相関係数  $r$ 、 $x$  および  $y$  の推定値は直線回帰と同じ式によって計算できます。

ただし、下記の計算結果は直線回帰と異なります。

直線回帰	対数回帰
$\Sigma x$	$\Sigma \ln x$
$\Sigma x^2$	$\Sigma (\ln x)^2$
$\Sigma xy$	$\Sigma \ln x \cdot y$

例

下記のデータを対数回帰して、回帰式と相関係数を求める。  
また、回帰式をもとに  $x_i = 80$ 、 $y_i = 73$  のときの  $y$ 、 $x$  をそれぞれ推定する。

$x_i$	$y_i$
29	1.6
50	23.5
74	38.0
103	46.4
118	48.9

**FUNCTION** **6** (DSP/CLR)

**6** (ScI) **EXE**

(メモリーのクリアー)

ScI  
0

**In** 29 **DT** 1.6 **DT** **In** 50 **DT** 23.5 **DT**

**In** 74 **DT** 38.0 **DT** **In** 103 **DT** 46.4 **DT**

**In** 118 **DT** 48.9 **DT**

(データの入力)

**In** 118,48.9  
X= 4.770684624  
Y= 48.9  
n= 5  
LR

**FUNCTION** **8** (RESULTS)

(統計処理結果の表示)

Regression  
A = -111.1283976  
B = 34.0201475  
r = 0.994013946  
↓

**EXIT** **In** 80 **FUNCTION** **7** (STAT)

**▼** **▼** **5** ( $\hat{y}$ ) **EXE**

( $x_i=80$ のときの  $\hat{y}$  の値)

**In** 80  
37.94879482

73 **FUNCTION** **7** (STAT) **▼** **▼**

**4** ( $\hat{x}$ ) **EXE** **SHIFT** **e<sup>x</sup>**

**SHIFT** **Ans** **EXE**

( $y_i=73$ のときの  $\hat{x}$  の値)

73 $\hat{x}$   
5.412333901  
eAns  
224.1541313  
LR

## ■ 指数回帰計算

回帰式を  $y = A \cdot e^{Bx}$  ( $\ln y = \ln A + Bx$ ) と定義して、計算します。

### ● データの入力

- (1) **FUNCTION** **6** (DSP/CLR) **6** (ScI) **EXE** と操作して、統計計算用メモリーP、Q、R、U、V、Wの内容を消去します。
- (2) 以下のように操作して、データを入力します。

[ x データの値 ] **□** **ln** [ y データの値 ] **DT**

※ 同じデータを複数個入力するときは、直線回帰計算でのデータ入力の仕方と同じ手順で行なうことができます。

### ● データの削除

入力されたデータの削除は、直線回帰計算でのデータ削除の仕方と同じ手順で行なうことができます。

10 **□** **ln** 20 **SHIFT** **CL**

### ● 計算の仕方

下記の式に合わせて計算することにより、答えを求めることができます。

- ・定数項  $A = e^a$
- ・xの推定値  $= \ln y / B$
- ・yの推定値  $= e^{a + Bx}$

回帰式  $y = A \cdot e^{Bx}$  ( $\ln y = \ln A + Bx$ ) は  $\ln y = y$ 、 $\ln A = a$  とおくと、直線回帰式  $y = a + bx$  と考えられます。このため、定数項A、回帰係数B、相関係数  $r$ 、 $x$  および  $y$  の推定値は直線回帰と同じ式によって計算できます。

ただし、下記の計算結果は直線回帰と異なります。

直線回帰	指数回帰
$\Sigma y$	$\Sigma \ln y$
$\Sigma y^2$	$\Sigma (\ln y)^2$
$\Sigma xy$	$\Sigma x \cdot \ln y$

例

下記のデータを指数回帰して、回帰式と相関係数を求める。  
また、回帰式をもとに  $x_i = 16$ 、 $y_i = 20$  のときの  $y$ 、 $x$  をそれぞれ推定する。

$x_i$	$y_i$
6.9	21.4
12.9	15.7
19.8	12.1
26.7	8.5
35.1	5.2

FUNCTION **6** (DSP/CLR)

**6** (ScI) EXE

(メモリーのクリアー)

ScI 0

6.9  $\rightarrow$  In 21.4  $\rightarrow$  DT 12.9  $\rightarrow$  In 15.7  $\rightarrow$  DT

19.8  $\rightarrow$  In 12.1  $\rightarrow$  DT 26.7  $\rightarrow$  In 8.5  $\rightarrow$  DT

35.1  $\rightarrow$  In 5.2  $\rightarrow$  DT

(データの入力)

35.1, ln 5.2  
X= 35.1  
Y= 1.648658625  
n= 5  
LR

FUNCTION **8** (RESULTS)

(統計処理結果の表示)

Regression  
A = 3.417647579  
B = -0.049203708  
r = -0.997247351  
↓

EXIT 16 FUNCTION **7** (STAT)  $\nabla$   $\nabla$

**5** (y) EXE

SHIFT  $\leftarrow$  SHIFT Ans EXE

( $x_i = 16$  のときの  $y$  の値)

16 $\rightarrow$   
2.630388247  
eAns  
13.87915739  
LR

**ln** 20 **FUNCTION** **7** (STAT) **▼** **▼**

**4** (x) **EXE**

( $y_i=20$ のときの  $x$  の値)

**ln** 20 $\frac{2}{3}$

8.574868047

## ■べき乗回帰計算

回帰式を  $y=A \cdot x^B$  ( $\ln y = \ln A + B \ln x$ ) と定義して、計算します。

### ● データの入力

(1) **FUNCTION** **6** (DSP/CLR) **6** (ScI) **EXE** と操作して、統計計算用メモリーP、Q、R、U、V、Wの内容を消去します。

(2) 以下のように操作して、データを入力します。

**ln** (x データの値) **▶** **ln** (y データの値) **DT**

※ 同じデータを複数個入力するときは、直線回帰計算でのデータ入力の仕方と同じ手順で行なうことができます。

### ● データの削除

入力されたデータの削除は、直線回帰計算でのデータ削除の仕方と同じ手順で行なうことができます。

**ln** 10 **▶** **ln** 20 **SHIFT** **CL**

### ● 計算の仕方

下記の式に合わせて計算することにより、答えを求めることができます。

- ・定数項  $A = e^a$
- ・ $x$  の推定値  $= e^{\ln y / B}$
- ・ $y$  の推定値  $= e^{B \ln x + a}$

回帰式  $y = A \cdot x^B$  ( $\ln y = \ln A + B \ln x$ ) は  $\ln y = y$ 、 $\ln A = a$ 、 $\ln x = x$  とおくと、直線回帰式  $y = a + bx$  と考えられます。このため、定数項A、回帰係数B、相関係数  $r$ 、 $x$  および  $y$  の推定値は直線回帰と同じ式によって計算できます。

ただし、下記の計算結果は直線回帰と異なります。

直線回帰	べき乗回帰
$\Sigma x$	$\Sigma \ln x$
$\Sigma x^2$	$\Sigma (\ln x)^2$
$\Sigma y$	$\Sigma \ln y$
$\Sigma y^2$	$\Sigma (\ln y)^2$
$\Sigma xy$	$\Sigma \ln x \cdot \ln y$

例

下記のデータをべき乗回帰して、回帰式と相関係数を求める。また、回帰式をもとに  $x_i = 40$ 、 $y_i = 1000$  のときの  $y$ 、 $x$  をそれぞれ推定する。

$x_i$	$y_i$
28	2410
30	3033
33	3895
35	4491
38	5717

FUNCTION **6** (DSP/CLR)

**6** (ScI) **EXE**

(メモリーのクリアー)

**ln** 28 **▸** **ln** 2410 **DT**

**ln** 30 **▸** **ln** 3033 **DT**

**ln** 33 **▸** **ln** 3895 **DT**

**ln** 35 **▸** **ln** 4491 **DT**

**ln** 38 **▸** **ln** 5717 **DT**

(データの入力)

ScI

0

**ln** 38, **ln** 5717  
 $X = 3.637586159$   
 $Y = 8.651199471$   
 $n = 5$

LR

FUNCTION 8 (RESULTS)

(統計処理結果の表示)

```
Regression
A = -1.432124422
B = 2.771866157
r = 0.998906255
```

※ 上記の定数項Aの値は $a$ として求まっておりますので、 $A \cdot x^B$ 型のAを求めるときは、以下のように操作して、答えを求めてください。

EXIT SHIFT  $e^x$  FUNCTION 7 (STAT)

▼ ▼ 1 (A) EXE

```
ea
0.2388010685
```

In 40 FUNCTION 7 (STAT)

▼ ▼ 5 (y) EXE

SHIFT  $e^x$  SHIFT Ans EXE

( $x_i=40$ のときの  $y$  の値)

```
In 40y
8.792955696
eAns
6587.674589
LR
```

In 1000 FUNCTION 7 (STAT)

▼ ▼ 4 (x) EXE

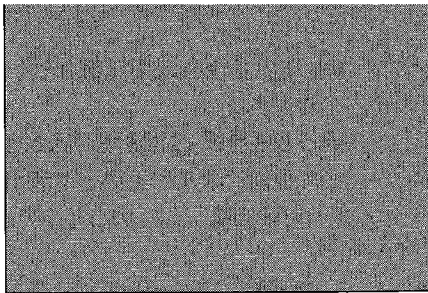
SHIFT  $e^x$  SHIFT Ans EXE

( $y_i=1000$ のときの  $x$  の値)

```
In 1000x
3.008759885
eAns
20.26225681
LR
```



# 8



**数式記憶機能**

---

同じ数式を使い、変数に異なった値を入力して複数の答えを求めることができます。

同じ数式を繰り返し計算するときなどに便利です。

この機能では、以下のキーを使います。

- **SHIFT** **IN** ..... 表示されている数式を数式記憶専用メモリーに記憶します。
- **OUT** ..... 数式記憶専用メモリーに記憶した数式を呼び出します。
- **CALC** ..... 数式記憶専用メモリーに記憶した数式を使い、変数に数値を入力して答えを求めます。

## 8-1. 数式記憶機能の実例

**例 1** 数式「 $Y = AX^2 + 6X - 9$ 」を記憶し、計算する。

(1) 数式を入力します。

**AC** **SHIFT** **ALPHA** **Y** **=** **A** **X**  
**ALPHA** **x<sup>2</sup>** **+** **6** **ALPHA** **X** **-** **9**

$Y=AX^2+6X-9$ \_

(2) 数式を記憶します。

**SHIFT** **IN**

-

(3) 数式を計算します。

**CALC**

入力できる変数  
 ▼ ▲ で移動

$Y=AX^2+6X-9$   
 ■ **A** = 0  
**X** = 0

現在、変数メモリーが記憶している数値

**2** **EXE**

(変数Aへの値の入力)

**5** **EXE**

(変数Xへの値の入力)

**CALC**

(結果の表示)

$Y=AX^2+6X-9$	
A=	2
X=	5

$Y=AX^2+6X-9$	
	71
REPEAT:[EXE]	

**EXE**

(数式の繰り返し計算)

$Y=AX^2+6X-9$	
A=	2
X=	5

(以下省略)

※ 最下段の変数の横に“■”があるときは、**CALC**の代わりに**EXE**を押しても数式を計算します。

※ マルチステートメントを使用し、連続して数式の計算を行ない、その計算時間が長いときに途中で計算を止めたいときは、**AC**を押してください。“Calculation Stopped”と表示されますので、続けて**AC**を押してください。

数式を表示した後、数式を訂正することができます。

**例 2**

数式「 $Y = AX^2 + 6X - 9$ 」を「 $Y = AX^2 + 3X - 9$ 」に訂正する。

(1) 数式を呼び出します。

**OUT**

$Y=AX^2+6X-9$
---------------

(2) 訂正したい箇所にカーソルを移動します。

◀◀◀◀

$Y=AX^2+6X-9$
---------------

(3) 訂正した後、記憶します。

$$Y=AX^2+3X-9$$

**3**

**SHIFT IN**

※ 記憶していた数式をクリアしたいときは、**AC SHIFT IN**と操作してください。

### 【諸注意】

- 記憶できる数式は1つです(マルチステートメントで続く数式も1つの式と見なします)。
- 記憶容量は127バイト以内です。
- 数式を記憶したとき、演算時のモード(COMP、SD、LR、BASE-N)も同時に記憶されます。
- 数式内に配列メモリーを使うことはできません。この場合、配列メモリーが記憶している数値が使われます。
- 以下のような場合は、“BASE-N Mode ERROR!”と表示されます。
  - (1) BASE-Nモード指定時に、BASE-Nモード以外を演算時のモードとして記憶した数式を呼び出そうとした場合。
  - (2) BASE-Nモード以外を指定時に、BASE-Nモードを演算時のモードとして記憶した数式を呼び出そうとした場合。
- オートパワーオフ(自動電源OFF)機能を含めて電源OFFになっても、記憶されている内容は保持されます。ただし新たに数式を記憶させると、以前記憶されていた内容はクリアされます。

## 8-2. コメント文の表示

数式を使った変数メモリーの後に「"」(ダブルクォーテーション)で囲んだ文を書き込むと、書き込んだ文を変数に対するコメント文として表示させることができます。

※ コメント文は、最大15文字まで表示します。

**例** 数式「S"MENSEKI" = 3.14 × R<sup>2</sup>」を記憶し、計算する。

(1) 数式を入力します。

AC SHIFT ALPHA S " M E N S E  
K I " = ALPHA 3 . 1 4 X  
ALPHA R X<sup>2</sup>

```
S"MENSEKI"=3.14X  
R2_
```

SHIFT IN

(2) 数式を計算します。

CALC

```
S"MENSEKI"=3.14X  
R= 0
```

3 EXE

(変数Rへの値の入力)

CALC

```
MENSEKI=  
28.26  
REPEAT:[EXE]
```

(以下省略)

## 8-3. テーブル機能

数式記憶専用メモリーに記憶した数式の1つの変数に対してレンジ(変数の条件)を設定し、繰り返し計算をすることができます。

**例** 数式「 $Y = AX^2 + 6X - 9$ 」の変数 $X$ の値を0から2ずつ変化させて、繰り返し計算を行なう。ただし、変数 $A = 2$ とする。

(1) 数式を入力します。

AC SHIFT ALPHA Y = A X  
ALPHA X<sup>2</sup> + 6 ALPHA X - 9

Y=AX<sup>2</sup>+6X-9\_

SHIFT IN

(2) **CALC** を押した後、変数 $A$ に値を入力します。

CALC 2 EXE

Y=AX<sup>2</sup>+6X-9  
A= 2  
■X= 0

(3) **SHIFT TBL** と操作します。レンジ設定表示になります。

SHIFT TBL

\*\*TABLE Range\*\*  
X  
Start? 1

(4) 変数 $X$ の初期値を入力し、**EXE** を押します。

0

Start?  
0\_

EXE

```
**TABLE Range**
X
Pitch?
1
```

(5) 変数Xの変化の度合いを入力し、EXEを押します。数式が計算されます。

2

```
Pitch?
2_
```

EXE

```
Y=AX2+6X-9
-9
X=
0
```

EXE

(数式の繰り返し計算)

```
Y=AX2+6X-9
11
X=
2
```

(以下省略)

※ 変数の変化の度合いが正ならば変数の値は増加を、負ならば変数の値は減少をします。

※ 途中で計算を止めたいときは、ACまたはEXITを押してください。

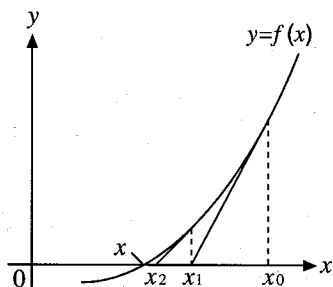
### 【諸注意】

- 変数の初期値/変化の度合いは、値「1」にあらかじめ設定されています。
- 以下のようなときは、2行目にエラー「Ma ERROR」が表示されます。
  - (1) 計算結果がエラーとなるとき。
  - (2) 計算結果が、演算範囲を超えたとき。
  - (3) 計算結果が虚数となったとき。

## 8-4. ソルブ機能

数式記憶専用メモリーに記憶した数式に対して、ニュートン法を基に数式に使われている任意の変数の解を求めることができます。

※ニュートン法とは



「 $f(x)$ は、ごく狭い範囲において1次式で近似できる」という考えをもとに、式の根の近似値を微分を使って求める方法です。

まず、ある初期値(予想値)  $x_0$ を与え、その初期値をもとに近似値  $x_1$ を求めた後、左辺と右辺の計算結果を比較します。次に、求められた近似値を新たな初期値  $x_1$ として、次の近似値  $x_2$ を求める計算を行ないます。この計算を、右辺と左辺の結果の差がある微小な値より小さくなるまで続けます。

求める計算を行ないます。この計算を、右辺と左辺の結果の差がある微小な値より小さくなるまで続けます。

例

数式「 $Y = AX^2 + 6X - 9$ 」において、変数  $Y = 0$ 、 $A = 2$  のときの変数  $X$  の値を計算する。

(1) 数式を入力します。

AC SHIFT ALPHA Y = A X  
 ALPHA  $x^2$  + 6 ALPHA X = 9  
 SHIFT IN

$Y = AX^2 + 6X - 9$

(2) **SOLVE** を押します。

**SOLVE**

$Y = AX^2 + 6X - 9$   
 ■ Y = 0  
 A = 2  
 X =



(3) 値を入力する変数の横に“■”を移動し、変数に値を入力します。

0 **EXE**

(変数Yへの値の入力)

2 **EXE**

(変数Aへの値の入力)

$Y=AX^2+6X-9$		
Y=		0
A=		2
■X=		0

(4) 答えを求める変数の横に“■”を移動し、**SOLVE**を押します。

**SOLVE**

$Y=AX^2+6X-9$		
X=	1.098076211	
Lft=		0
Rgt=	-8.E-14	

※ Lft/Rgtは、求めた近似値により左辺と右辺を計算した結果を表わしています。この2つの値の差が0に近い程、答えはより精度が高いものであるといえます。

**SOLVE**

(数式の繰り返し実行)

$Y=AX^2+6X-9$		
■Y=		0
A=		2
X=	1.098076211	

(以下省略)

※ ソルブ機能はニュートン法を基に計算を行なうため、以下のような場合があります。

- 求めたい変数の初期値(予想値)によっては、解が求められない場合があります。その場合は、変数の初期値に解に近いと思われる数値を入力してから、再度計算し直してください。
- 解の存在する式でも、解が求められない場合があります。

※ ニュートン法の性質上、次のような関数は解を求めにくい傾向にあります。

- 周期関数 ( $y = \sin x$  など)
- グラフを描いたとき、急勾配の部分を持つ関数 ( $y = e^x$ 、 $y = 1/x$  など)
- 不連続な関数 ( $y = \sqrt{x}$  など)

※ ソルブ機能により求めた解は、実際の真値に対して誤差を含んでいる可能性があります。

※ ソルブ実行状態を抜きたいときは、**EXIT** を押してください。

### 【諸注意】

- 答えを求める途中の収束結果が表示されたときは、1行目に“Try again:[EXE]”と表示されます。このとき **EXE** を押すと、今の収束結果を基に計算を続けることができます。
- 答えが求められなかったときは、右のような表示が現われます。そのときは、**AC** または **EXIT** を押してください。

```
**Can't solve**  
Adjust  
initial value.  
Then try again.
```

## 8-5. 数式の保存／呼び出し

数式記憶専用メモリーに記憶した数式を、後述しますプログラミングエリアにファイル名を付けて保存し、自由に呼び出すことができます。

### 数式を保存する

**例** 数式記憶専用メモリーの数式「 $Y=AX^2+6X-9$ 」を、ファイル名「NIJISIKI」としてプログラミングエリアに保存する。

- (1) **MODE** **5** (PROG)と操作して、プログラミングメニュー表示を呼び出します。

**MODE** **5** (PROG)

```
Program menu
1.NEW    2.RUN
3.EDIT  4.DELETE
4300 Bytes Free
```

- (2) **1** (NEW)を押した後、ファイル名を入力します。

**1** (NEW)

**N I J I S I K I**

```
Filename?
[NIJISIKI_ ]
```

- (3) **EXE**を押して、ファイル名を登録します。

**EXE**

```
PGM:NIJISIKI
1.COMP  2.BASE-N
3.SD    4.LR
5.Save formula
```

- (4) **5**を押して、数式保存確認表示にします。

**5**

```
PGM:NIJISIKI
Save formula?
YES:[EXE]
NO :[EXIT]
```

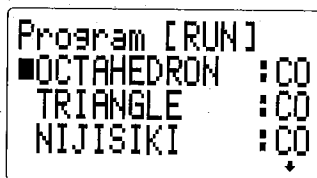
- (5) **[EXE]**を押します。数式が保存され、プログラミングメニュー表示になります。

※数式の保存を止めたいときは、**[EXIT]**を押してください。

## 数式を呼び出す

**例** ファイル名「NIJISIKI」としてプログラミングエリアに保存した数式「 $Y=AX^2+6X-9$ 」を、数式記憶専用メモリーに呼び出す。

- (1) COMP/BASE-N/SD/LRモード指定時に**[FILE]**を押すか、プログラミングメニュー表示から**[2]**(RUN)を押します。

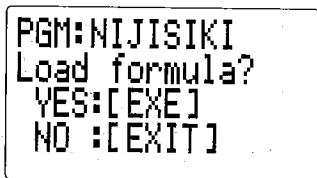


```
Program [RUN]
■OCTAHEDRON :CO
  TRIANGLE  :CO
    NIJISIKI :CO
              ↓
```

※ファイル名が複数登録されている場合

- (2) **[▼]** **[▲]**を押して、数式の保存されているファイル名の横に“■”を移動します。
- (3) **[SHIFT]** **[IN]**と操作して、数式呼び出し確認表示にします。

**[SHIFT]** **[IN]**

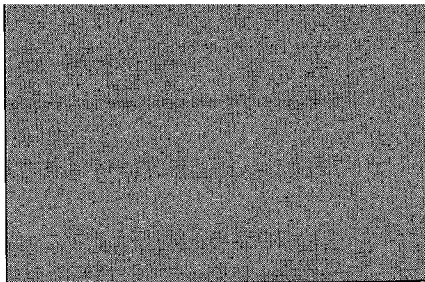


```
PGM:NIJISIKI
Load formula?
YES:[EXE]
NO :[EXIT]
```

- (4) **[EXE]**を押します。数式記憶専用メモリーに数式が呼び出されます。

※ 数式の呼び出しを止めたいときは、**[EXIT]**を押してください。

# 9



## プログラム機能

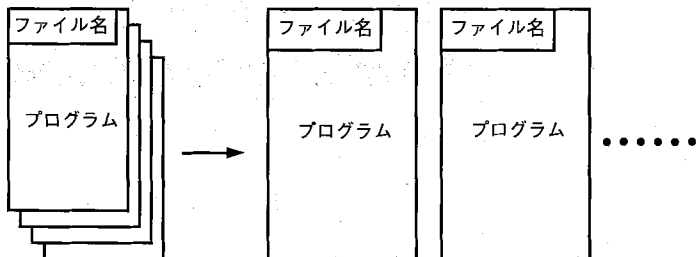


## 9-1. プログラム機能とは

本機には、繰り返し計算や複雑な計算に便利なプログラム機能が内蔵されています。

このプログラム機能とは、マニュアル計算の一つである「マルチステートメント機能」と同様に、計算式を連続して実行する機能です。

本機のプログラム機能は、ファイル名によりプログラムをファイル管理するシステムになっております。このシステムにより、プログラムの訂正や検索が簡単にできます。また、プログラム容量(4500バイト)以内であればいくつでもプログラムを記憶することができます。



## 9-2. プログラミングする前に

プログラミングする前に、メインメニュー表示からPROGモードを指定してください。次のようなプログラミングメニュー表示になります。

MODE [5] (PROG)

メモリーの残り容量

```
Program menu
1.NEW    2.RUN
3.EDIT  4.DELETE
4300 Bytes Free
```

メニューの意味は、以下のとおりです。

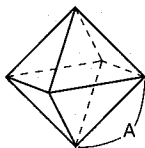
- ・「1. NEW」.....プログラムを新規作成。
- ・「2. RUN」.....登録されているプログラムを実行。  
→152ページ参照
- ・「3. EDIT」.....登録されているプログラムを編集。  
→161ページ参照
- ・「4. DELETE」.....登録されているプログラムを削除。  
→165ページ参照

※プログラムが記憶されていないときは、最下行に「No file」と表示されます。

## 9-3. プログラミングの実例

まず、例題とともにプログラムを見てみましょう。

- 例題 1** 図のような正8面体の一辺の長さが7cm、10cm、15cmのときの、表面積と体積を求めよ。  
なお、ファイル名は「OCTAHEDRON」とする。



一辺の長さ(A)	表面積(S)	体積(V)
7cm	( )cm <sup>2</sup>	( )cm <sup>3</sup>
10cm	( )cm <sup>2</sup>	( )cm <sup>3</sup>
15cm	( )cm <sup>2</sup>	( )cm <sup>3</sup>

※カッコ内を求める

一辺の長さをAとすると、正8面体の表面積S、体積Vは次の式から求められます。

$$S = 2\sqrt{3}A^2$$

$$V = \frac{\sqrt{2}}{3} A^3$$

では、この例題をプログラミングしてみましょう。

プログラミングは、ファイル名の登録、プログラム実行時のモード指定、プログラムの書き込みの順番となります。

## ファイル名を登録する

(1) プログラミングメニュー表示から **F1** (NEW) を押します。

**F1** (NEW)

```
Filename?  
[ _ ]
```

(2) ファイル名を入力します。

**O C T A H E D R O N**

```
Filename?  
[ OCTAHEDRON_ ]
```

※ カーソルはアルファベット大文字入力状態になります。

※ ファイル名には12文字まで入力できます。入力できる文字は、  
A~Z、空白(SPACE)、0~9、. (小数点)、[, ], +, -, ×, ÷  
です。

※ カーソルを移動させて **DEL** を押すと、カーソル上の文字が削除されます。

(3) **EXE** を押します。ファイル名が登録され、プログラム実行時のモード表示になります。      ファイル名 —

**EXE**

```
PGM: OCTAHEDRON  
1. COMP    2. BASE-N  
3. SD      4. LR  
5. Save formula
```

※ ファイル名の登録に、メモリーを17バイト使用します。もし、メモリーの残り容量が16バイト以下のときは、ファイル名を登録することはできません。そのときは余分なプログラムを消去してください(165ページ参照)。

※ 同じファイル名が登録されているときに **EXE** を押すと、“Already exists” と表示されます。

※ ファイル名を入力せずに **EXE** を押すと、無効となります。

※ ファイル名を登録する前に **EXIT** を押すと、プログラムメニュー表示に戻ります。



## ■ プログラム実行時のモードを指定する

プログラム実行時のモード表示から数字キー(①～④)を押して、プログラム実行時のモードを指定してください。

```
PGM: OCTAHEDRON
1.COMP  2.BASE-N
3.SD    4.LR
5.Save formula
```

メニューの意味は、以下のとおりです。

- ・「1.COMP」.....COMPモードに指定。
- ・「2.BASE-N」.....BASEモードに指定。
- ・「3.SD」.....SDモードに指定。
- ・「4.LR」.....LRモードに指定。
- ・「5.Save formula」.....数式記憶専用メモリーに記憶した数式を保存します。→145ページ参照

ここでは、① (COMP)を押します。COMPモードに指定され、プログラム書き込み表示になります。

① (COMP)

-

## ■ プログラムを書き込む

プログラムの基本は、マニュアル計算です。マニュアルで計算するときの操作方法を考えてから、プログラムを書き込んでください。

この例題は、マニュアル計算で

表面積S ..... ②   ③  [Aの数値]   EXE

体積V .....  ②  ③  [Aの数値]  ③ EXE

と操作しますので、一辺の長さAとして次のようにプログラムを書き込みます。

表面積S ..... ②   ③  ALPHA A   EXE

体積V .....  ②  ③  ALPHA A  ③ EXE

本機のプログラムはマニュアルで計算するときの操作内容のみ書き込んで実行させると、最後まで命令を実行して止まりません。そこで、下記の出力命令「▲」が必要となります。

よく使う命令として、次の2種類があります。

▲	プログラムを一時停止し、直前の計算式の答えやアルファベット文字を表示させる命令です。マニュアル計算で[EXE]を押す箇所に書き込みます。
:	マルチステートメントで計算式や他の命令を区切るときに使う命令です。この命令は「▲」と異なり、プログラム実行時の途中結果を表示しません。

では、プログラムに「▲」を加え、実際に書き込んでみましょう。

2 [X] [✓] 3 [X] [ALPHA] [A] [x²] [SHIFT] [▲]

2×√3×A²  
-

[✓] 2 [÷] 3 [X] [ALPHA] [A] [^] 3

2×√3×A²  
√2÷3×A³

[EXIT] [EXIT]

Program menu  
1. NEW    2. RUN  
3. EDIT   4. DELETE  
4467 Bytes Free

## プログラムを実行する

(1) プログラミングメニュー表示から [2] (RUN) を押します。

現在指定されている  
ファイル名

Program [RUN]  
■OCTAHEDRON :CO

プログラム実行時のモード

※ 表示右横に表示される“CO”はCOMPモードを、“BN”はBASE-N モードを表わします。

- (2)  $\blacktriangledown$   $\blacktriangle$  を押して、実行したいプログラムのファイル名の位置に“■”を移動します。

では、例題1のプログラムを実行してみましょう。

一辺の長さ(A)	表面積(S)	体積(V)
7cm	(169.7409791)cm <sup>2</sup>	(161.6917506)cm <sup>3</sup>
10cm	(346.4101615)cm <sup>2</sup>	(471.4045208)cm <sup>3</sup>
15cm	(779.4228634)cm <sup>2</sup>	(1590.990258)cm <sup>3</sup>

- (3)  $\text{EXE}$  を押します。プログラムが実行されます。

$\boxed{7}$   $\text{EXE}$

(Aの値)

A?	0
7	0
$2 \times \sqrt{3} \times A^2$	169.7409791
	<small>Disp</small>

※“ $\text{Disp}$ ”は $\blacktriangle$ により答えを表示中を意味します。

$\text{EXE}$

$2 \times \sqrt{3} \times A^2$	169.7409791
$\sqrt{2 \times 3} \times A^3$	161.6917506

$\text{EXE}$

$\sqrt{2 \times 3} \times A^3$	161.6917506
A?	7

1 0 EXE

```
10
2*√3*A^2
346.4101615
Disp
```

EXE

```
2*√3*A^2
346.4101615
√2÷3*A^3
471.4045208
```

(以下省略)

※ 最後の計算結果を表示した後 **EXE** を押すと、プログラムは再実行されます。

※ COMP/BASE-N/SD/LRモード指定時に以下のように操作しても、プログラムを実行することができます。

- (1) **FILE** を押します。
- (2) **▼ ▲** を押して実行したいプログラムのファイル名の位置に“■”を移動し、**EXE** を押します。
- **SHIFT Prog ALPHA 〃** (ファイル名) **ALPHA 〃 EXE** と操作します。

## 9-4. プログラムのデバッグ (間違いを直す)

プログラムが意図したように実行されなかったとき、その原因であるプログラムの間違いを「バグ(BUG)」といいます(バグとは虫のことを意味します)。また、この虫を取り除く作業のことを「デバッグ(DEBUG)」といいます。次のような場合はプログラム中に間違い(バグ)があると考えられますので、その間違いを訂正(デバッグ)してください。

- (1) プログラムを実行しても、エラーメッセージが表示されるとき。
- (2) 思うような計算結果(答え)が得られないとき。

## ■エラーメッセージによるデバッグ

エラーが発生したとき、次のようなエラーメッセージが表示されます。

```
(1) ———| in Ma ERROR |———— (2)  
           | TRIANGLE
```

- (1) ファイル名 .... エラーが発生したプログラムのファイル名を示します。
- (2) エラー内容 .... どのようなエラーが発生したかを示します。

エラーメッセージが表示されたときは、 または  を押してください。エラーが発生した箇所にカーソルが点滅します。その後「エラーメッセージ一覧表」(190ページ)を参照の上、正しいプログラムに直してください。

## ■プログラム実行結果に対するデバッグ

プログラム実行時に思うような結果(答え)が得られなかった場合は、間違いを直したいプログラムのファイル名を検索し、正しいプログラムに直してください。

詳しくは、「ファイル名またはプログラムの訂正」(160ページ)をご覧ください。

## 9-5. バイト数の数え方

本機は全部で4500バイトのプログラム容量を持っています。このバイトとは、プログラムを記憶できる許容量を表わす単位のことです。

バイト数は、原則として1機能1バイトと数えます。ただし、1機能2バイトと数える命令もあります。

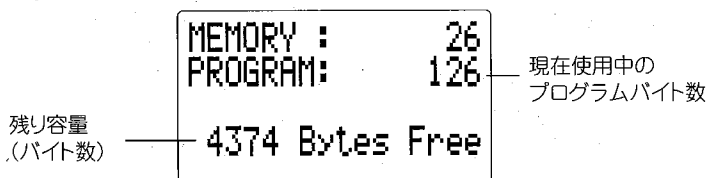
- ・1機能1バイト ..... sin, cos, tan, log, (,), A, B, C, 1, 2など
- ・1機能2バイト ..... Lbl 1, Goto 2など

バイトは、カーソルにより確認できます。カーソルはカーソル移動キー( )を押すことにより、1バイトずつ移動します。

※ プログラムを書き込んでいるときに残りバイト数が5バイト以下になると、カーソルが“\_”から“■”に変わります。もし5バイト以上のプログラムを記憶させたいときは不要なプログラムや増設メモリーなどを消去して、残りバイト数を増やしてください。

## ■プログラムの残り容量の確認

COMP/BASE-N/SD/LRモード指定時に **[SHIFT]** **[Defm]** **[EXE]** と操作してください。



## 9-6. ファイル名の検索

本機に書き込んだプログラムのファイル名を検索する方法は、以下の2種類があります。

- (1) ファイル一覧表示から検索する(シーケンシャルサーチ)
- (2) ファイル名の読み入力により検索する(ダイレクトサーチ)

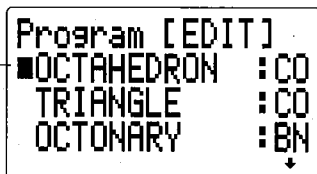
### ファイル一覧表示から検索する(シーケンシャルサーチ)

ファイル一覧表示からファイル名を検索する方法です。

- (1) プログラミングメニュー表示から **3** (EDIT) を押します。

**3** (EDIT)

**3** を押す前に指定していた  
ファイル名



```
Program [EDIT]
■OCTAHEDRON :CO
  TRIANGLE  :CO
  OCTONARY   :BN
  ↓
```

※ファイル名が複数登録されている場合

- (2) **▲** **▼** を押してファイル名の横に“■”を移動し、**EXE** を押します。  
記憶されているプログラムの内容が表示されます。

### 【参考】

COMP/BASE-N/SD/LRモード指定時に **FILE** を押して、ファイル一覧表示に呼び出すことができます。

ただし、記憶されているプログラムの内容を表示させることはできません。

## ファイル名の読み入力により検索する(ダイレクトサーチ)

ファイル名の読みを入力して検索する方法です。

- (1) プログラミングメニュー表示から **3** (EDIT) を押します。

```
Program [EDIT]
■OCTAHEDRON :CO
  TRIANGLE  :CO
  OCTONARY  :BN
                ↓
```

- (2) **FUNCTION** を押して、ファイルコマンド表示を呼び出します。

```
File Commands
1. SEARCH
2. RENAME
```

- (3) **1** (SEARCH) を押します。

```
Search for file
[ _ ]
```

- (4) 検索したいファイル名の頭文字を入力して、**EXE** を押します。該当するファイル名をすべて表示します。

**O C T**

```
Search for file
[OCT_ ]
```

**EXE**

```
Program [EDIT]
■OCTAHEDRON :CO
  OCTONARY  :BN
```

※ 同じ頭文字のファイルが4件以上あるときに **▽** **▲** を押すと、ファイル名がスクロールして隠れているファイル名が現われます。



※ 該当するファイル名がないときは、“No file”と表示されます。  
その場合は **EXIT** を押してください。

- (5) **▼ ▲** を押してファイル名の横に“■”を移動し、**EXE** を押します。  
記憶されているプログラムの内容が表示されます。

### 【参考】

COMP/BASE-N/SD/LRモード指定時に以下のように操作して、ファイル名を検索することができます。ただし、記憶されているプログラムの内容を表示させることはできません。

- (1) 検索したいファイル名の頭文字を入力します。
- (2) **FILE** を押します。該当するファイル名をすべて表示します。

## 9-7. ファイル名またはプログラムの訂正

### ファイル名を訂正する

- (1) プログラミングメニュー表示から **3** (EDIT) を押し、訂正したいファイル名を検索します(ファイル名の横に“■”を移動します)。

```
Program [EDIT]
OCTAHEDRON :CO
■ TRIANGLE :CO
```

- (2) **FUNCTION** を押して、ファイルコマンド表示を呼び出します。

```
File Commands
1. SEARCH
2. RENAME
```

- (3) **2** (RENAME) を押します。現在のファイル名が表示されます。

```
Rename file
[ TRIANGLE ]
```

- (4) 新しいファイル名に訂正します。

**DEL DEL DEL**

```
Rename file
[ ANGLE ]
```

- (5) **EXE** を押します。訂正したファイル名が登録され、ファイル名一覧表示になります。

※ 訂正したファイル名がすでに登録されているときは、“Already exists”が表示されます。そのときは、以下のように操作してください。

- ◀ または ▶ を押します。エラーが解除され、**EXE** を押す前の状態に戻ります。
- AC** を押します。入力されていたファイル名がクリアされ、ファイル名を再入力できる状態になります。

# プログラムを訂正する

(1) プログラミングメニュー表示から **3** (EDIT) を押し、訂正したいプログラムのファイル名を検索します(ファイル名の横に“■”を移動します)。

(2) **EXE** を押します。プログラムの内容が表示され、カーソルがプログラムの先頭に点滅します。

(3) プログラムを訂正します。

※ プログラムの訂正の仕方は、マニュアル計算と同じです。詳しくは、「訂正の仕方」(39ページ)をご覧ください。

※ プログラム訂正時に以下のキーを使うと便利です。

• **SHIFT** **▲** ..... カーソルをプログラムの先頭に移動します。

```

2×√3×A²
√2÷3×A³
    
```

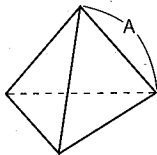
• **SHIFT** **▼** ..... カーソルをプログラムの末尾に移動します。

```

2×√3×A²
√2÷3×A³_
    
```

では実際に、例題1を次の例題2に合わせて訂正してみましょう。

**例題2** 図のような正4面体の一辺の長さが7cm、10cm、15cmのときの、表面積と体積を求めよ。なお、ファイル名は「TETRAHEDRON」とする。



一辺の長さ(A)	表面積(S)	体積(V)
7cm	( )cm <sup>2</sup>	( )cm <sup>3</sup>
10cm	( )cm <sup>2</sup>	( )cm <sup>3</sup>
15cm	( )cm <sup>2</sup>	( )cm <sup>3</sup>

※カッコ内を求める

一辺の長さをAとすると、正4面体の表面積S、体積Vは次の式から求められます。

$$S = \sqrt{3}A^2$$

$$V = \frac{\sqrt{2}}{12} A^3$$

この例題のプログラムは、以下のようになります。

表面積S .....  3  ALPHA A  SHIFT



体積V .....  2   1 2  ALPHA A

このプログラムを例題1のプログラム

表面積S .....     ALPHA A

体積V .....  2    ALPHA A

と比較してみると、例題1のプログラムを次のように訂正すると、例題2のプログラムになることがわかります。

-  の部分 (   ) を削除する。
-  の部分 (  を   ) に訂正する。

では、実際に訂正してみましょう。

(1) ファイル名を訂正します。

MODE 5 (PROG) 3 (EDIT)

```
Program [EDIT]
■OCTAHEDRON :CO
```

FUNCTION

```
File Commands
1.SEARCH
2.RENAME
```

2 (RENAME)

T E T R A H E  
D R O N

```
Rename file
[TETRAHEDRON_]
```

EXE

Program [EDIT]  
■TETRAHEDRON :CO

(2) プログラムを書き替えます。

EXE

$2 \times \sqrt{3} \times A^2$  ,  
 $\sqrt{2} \div 3 \times A^3$

DEL DEL

$\sqrt{3} \times A^2$  ,  
 $\sqrt{2} \div 3 \times A^3$

▼ ► ► ►  
SHIFT INS 1 2

$\sqrt{3} \times A^2$  ,  
 $\sqrt{2} \div 12 \sqrt{3} \times A^3$

DEL

$\sqrt{3} \times A^2$  ,  
 $\sqrt{2} \div 12 \sqrt{3} \times A^3$

EXIT EXIT

Program menu  
1.NEW 2.RUN  
3.EDIT 4.DELETE  
4468 Bytes Free

では、例題2のプログラムを実行してみましょう。

一辺の長さ(A)	表面積(S)	体積(V)
7cm	(84.87048957) cm <sup>2</sup>	(40.42293766) cm <sup>3</sup>
10cm	(173.2050808) cm <sup>2</sup>	(117.8511302) cm <sup>3</sup>
15cm	(389.7114317) cm <sup>2</sup>	(397.7475644) cm <sup>3</sup>

2 (RUN) EXE

A?

0

7 EXE

(Aの値)

$$\begin{array}{r} 7 \\ \sqrt{3 \times A^2} \\ 84.87048957 \end{array}$$

EXE

$$\begin{array}{r} \sqrt{3 \times A^2} \\ 84.87048957 \\ \sqrt{2 \div 12 \times A^3} \\ 40.42293766 \end{array}$$

EXE

$$\begin{array}{r} \sqrt{2 \div 12 \times A^3} \\ 40.42293766 \\ A? \\ 7 \end{array}$$

1 0 EXE

$$\begin{array}{r} 10 \\ \sqrt{3 \times A^2} \\ 173.2050808 \end{array}$$

EXE

$$\begin{array}{r} \sqrt{3 \times A^2} \\ 173.2050808 \\ \sqrt{2 \div 12 \times A^3} \\ 117.8511302 \end{array}$$

(以下省略)

## 9-8. ファイル名 (プログラム) の消去

ファイル名(記憶されているプログラム)を消去する方法は、以下の2種類があります。

- (1) 指定したファイル名のみ消去する。
- (2) すべてのファイル名を消去する。

ファイル名の消去は、プログラミングメニュー表示から次のようなプログラム消去メニュー表示を呼び出して行ないます。

④ (DELETE)

```
Delete Program
1. ONE PROGRAM
2. ALL PROGRAMS
```

メニューの意味は、以下のとおりです。

- ・「1. ONE PROGRAM」..... 指定したファイル名(プログラム)を消去。
- ・「2. ALL PROGRAMS」..... すべてのファイル名(プログラム)を消去。

### 指定したファイル名のみ消去する

例

ファイル名「TRIANGLE」に記憶したプログラムを消去する。

(1) プログラム消去メニューから① (ONE PROGRAM)を押します。

① (ONE PROGRAM)

①を押す前に指定していた  
ファイル名

```
Program [DELETE]
■OCTAHEDRON :CO
  TRIANGLE   :CO
  OCTONARY   :BN
  ↓
```

(2) 消去するファイル名の横に“■”を移動し、**EXE**を押します。

**▼** **EXE**

```
PGM: TRIANGLE
Delete?
YES:[EXE]
NO :[EXIT]
```

(3) **EXE**を押します。指定したプログラムが消去されます。

**EXE**

```
Program [DELETE]
OCTAHEDRON :CO
■OCTONARY :BN
SOINSUU :CO
↓
```

※ **EXIT**を押すと、消去は中止されます。

## すべてのファイル名を消去する

(1) プログラム消去メニューから **2** (ALL PROGRAMS)を押します。

**2** (ALL PROGRAMS)

```
**All programs**
Delete?
YES:[EXE]
NO :[EXIT]
```

(2) **EXE**を押します。すべてのプログラムが消去されます。

**EXE**

```
Program menu
1.NEW 2.RUN
3.EDIT 4.DELETE
*** No file ***
```

※ **EXIT**を押すと、消去は中止されます。



## 9-9. 便利なプログラム命令

本機は単一の計算式だけでなく、判断によりいくつかの計算式を使い分けたり、繰り返し実行ができる特別なプログラム命令を準備しています。

### プログラム命令メニュー表示

主な特別なプログラム命令は、プログラム命令メニュー表示を呼び出して書き込みます。

(1) 以下のように操作して、プログラム命令メニュー表示No.1を呼び出します。

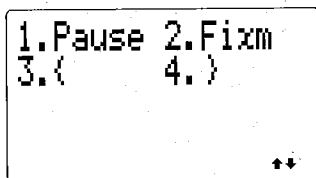
FUNCTION 3 (PROG)

1.⇒	2.キ⇒
3.△	4.Goto
5.Lbl	6.Dsz
7.Isz	

メニューの意味は、以下のとおりです。

- ・「1.⇒」.....条件ジャンプの成立コード「⇒」を書き込む。
- ・「2.キ⇒」.....条件ジャンプの不成立コード「キ⇒」を書き込む。
- ・「3.△」.....条件ジャンプの終了コード「△」を書き込む。
- ・「4.Goto」.....無条件ジャンプの命令「Goto」を書き込む。
- ・「5.Lbl」.....無条件ジャンプの命令「Lbl」を書き込む。
- ・「6.Dsz」.....メモリー内の値を1ずつ減算して判断する命令「Dsz」を書き込む。
- ・「7.Isz」.....メモリー内の値を1ずつ加算して判断する命令「Isz」を書き込む。

(2) **▽**を押すと、プログラム命令メニュー表示No.2になります。

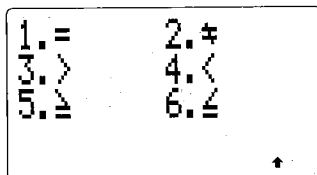


メニューの意味は、以下のとおりです。

- 「1. Pause」.....ポーズ命令「Pause」を書き込む。
- 「2. Fixm」.....変数確定命令「Fixm」を書き込む。
- 「3. {」.....変数入力命令の開きカッコ「{」を書き込む。
- 「4. }」.....変数入力命令の閉じカッコ「}」を書き込む。

※ **▲**を押すと、プログラム命令メニュー表示No.1に戻ります。

(3) **▽**を押すと、プログラム命令メニュー表示No.3になります。



メニューの意味は、以下のとおりです。

- 「1. =」.....条件ジャンプの関係演算子「=」を書き込む。
- 「2. キ」.....条件ジャンプの関係演算子「キ」を書き込む。
- 「3. >」.....条件ジャンプの関係演算子「>」を書き込む。
- 「4. <」.....条件ジャンプの関係演算子「<」を書き込む。
- 「5. ≥」.....条件ジャンプの関係演算子「≥」を書き込む。
- 「6. ≤」.....条件ジャンプの関係演算子「≤」を書き込む。

※ **▲**を押すと、プログラム命令メニュー表示No.2に戻ります。

## 変数入力命令

プログラム中の変数に一度値を入力すると、その値は確定した値として変数に記憶されます。そのため、後述しますジャンプ命令を使ったプログラム実行中に同じ変数が再び出てきても、変数入力を促す状態にはなりません。

もし、変数に新たな値を再入力する必要があるときは、変数入力命令「{ }」を書き込んでください。確定された変数の値を未確定状態にし、変数に新たな値を入力できる状態に戻します。

プログラムには、変数(A~Z)を「{ }」で囲んで書き込んでください。

例

{A} ..... 変数Aの確定状態を解除  
{AB}、{A, B}、{A B} ..... 変数A、Bの確定状態を解除

※「{ }」は1つの文として扱います。

※ 変数に配列変数を用いることはできません。

## 変数確定命令

プログラム中に変数確定命令「Fixm」を書き込むと、この命令以降のすべての変数(A~Z)は、既に値が入力されたものとして確定されます。

この命令が実行されると変数に値が入力できる状態にならず、すでに変数に記憶されている値を使ってプログラムが実行されます。

※「Fixm」は1つの文として扱います。

※ 変数入力命令「{ }」は、「Fixm」より優先されます。

## ジャンプ命令

ジャンプ命令とは、プログラムの実行の流れを変える命令のことです。同じ式を繰り返して実行したいときや、別の式を実行したいときなどに威力を発揮します。

ジャンプ命令には、次の3種類があります。

- (1) 無条件ジャンプ ..... 無条件に指定された箇所へジャンプする。
- (2) 条件ジャンプ ..... 条件を与えて、その条件が正しいか正しくないかを判断し、分岐先にジャンプする。
- (3) カウントジャンプ ..... メモリー内の数値を1ずつ加算または減算して、0になったらジャンプする。

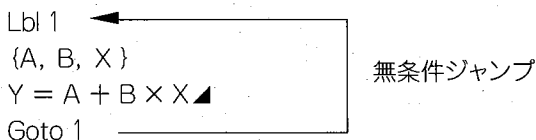
## ■ 無条件ジャンプ

無条件ジャンプとは、「Goto  $n$  ( $n=0\sim 9, A\sim Z$ )」と「Lbl  $n$  ( $n$  は「Goto  $n$ 」の  $n$  と同じ値)」から構成されます。プログラムを実行すると、「Goto  $n$ 」から「Lbl  $n$ 」へ命令が移ります。

この無条件ジャンプは単純にプログラムの先頭に分岐して実行を繰り返したり、プログラムの途中から繰り返すときに使います。また、条件ジャンプやカウントジャンプと組み合わせて使うこともできます。

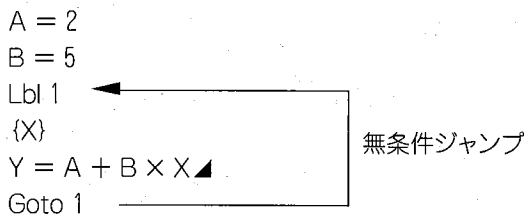
- 例 1**  $y = a + bx$  を計算するとき、 $x$  の値はその都度変化し、 $a$ 、 $b$  の値は計算によって変化する場合のプログラムを組む。

変数入力命令「{ }」を使って、プログラムは以下のように書き込みます。



- 例 2**  $y = a + bx$  を計算するとき、 $x$  の値のみ変化するプログラムを組む。ただし、 $a$  の値は 2、 $b$  の値は 5 とする。

以下のように書き込むと数値を代入した変数は確定状態となり、プログラム実行時に  $x$  の値のみ入力すれば結果が求められます。



※「Goto  $n$ 」に対する「Lbl  $n$ 」が見つからないときは、エラー「Go ERROR」となります。

## ■ 条件ジャンプ

条件ジャンプとは2つの変数メモリー（または算術式）を比較して、その判断結果により「⇒」の次の文、または「キ⇒」以降の文へジャンプして実行する命令です。

条件ジャンプは、次のような書式で構成されます。

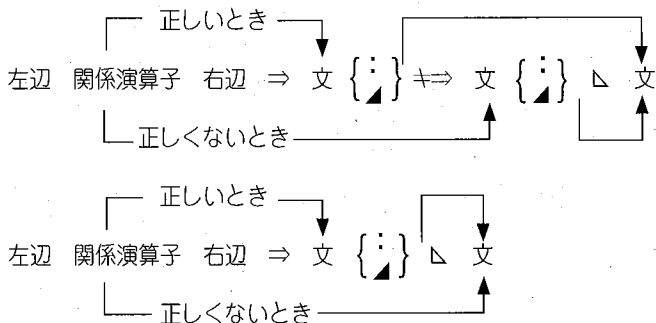
1. 左辺 関係演算子 右辺 ⇒ 文 { ⚡ } キ⇒ 文 { ⚡ } △文
2. 左辺 関係演算子 右辺 ⇒ 文 { ⚡ } △文

左辺と右辺には、変数(A~Z)または数値定数や変数式(A×2、B-Cなど)が入ります。

関係演算子には、次の6種類の比較記号が入ります。

- 左辺 = 右辺 ..... 左辺と右辺が等しい
- 左辺 キ 右辺 ..... 左辺と右辺が等しくない
- 左辺 > 右辺 ..... 左辺より右辺が小さい
- 左辺 < 右辺 ..... 左辺より右辺が大きい
- 左辺 ≥ 右辺 ..... 左辺より右辺が小さいか等しい
- 左辺 ≤ 右辺 ..... 左辺より右辺が大きいか等しい

判断結果が正しい(Yes)場合は、「⇒」以降の文から「キ⇒」または「△」までを実行し、その後ジャンプ終了コード「△」以降の文を実行します。  
 判断結果が正しくない(No)場合は「キ⇒」以降の文から「△」までを実行し、その後ジャンプ終了コード「△」以降の文を実行します。



※「⇒」や「キ⇒」以降の文には、マルチ文も使用できます。

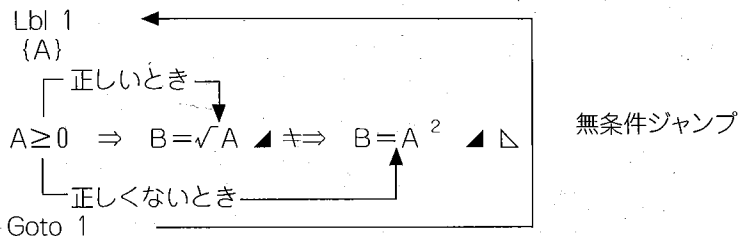
※ 1つの条件ジャンプの中に書き込める条件の数は、最大15段です。

※ 条件ジャンプは改行「↵」のない1つの文で構成されますので、改行「↵」が書き込まれている場合はエラー「Syn ERROR」となります。

**例**

入力した値が0または0より大きければその値の平方根を表示し、0より小さければその値の2乗を表示する計算を繰り返して行なう。

入力する値をA、計算した値をBとすると、プログラムは以下のようになります。



このプログラムでは、まず変数Aに値を入力します。

次に変数Aの値が0または0より大きければ「⇒」と「▲」の間の文(計算式)を実行し、変数Aの値が0より小さければ「⇐」と「▲」の間の文(計算式)を実行します。

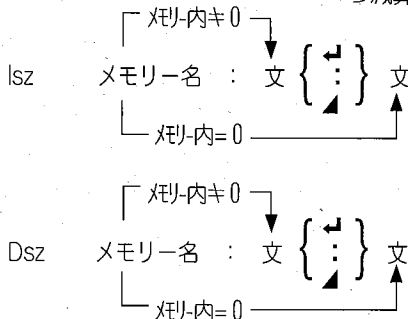
その後「Goto 1」により「Lbl 1」に戻り、入力を繰り返します。

## ■カウントジャンプ

カウントジャンプとは指定されたメモリー内の値に対して1ずつ加算または減算し、数値が0でなければメモリー名の次の文を実行し、数値が0になれば「:」、「▲」、または「⇐」の次の文を実行します。

カウントジャンプ命令には、次の2つがあります。

- Isz (Increment and Skip on Zero) .....メモリー内の数値に1ずつ加算して判断する命令
- Dsz (Decrement and Skip on Zero) .....メモリー内の数値から1ずつ減算して判断する命令



メモリー名には、変数メモリー名(A~Z)や増設メモリー(A[5]など)を書き込みます。

メモリーAに1ずつ加算する ..... Isz A

メモリーBから1ずつ減算する ..... Dsz B


例

10個の数値を入力して、平均値を求める。

入力する値をBとすると、プログラムは以下のようになります。

```
A=10
C=0
Lbl 1 ←
{B}
C=B+C
Dsz A
Goto 1
C÷10
```

無条件ジャンプ



このプログラムではメモリーAをカウンター(数を数える道具)と見なし、「Dsz」により1ずつカウントダウンさせます。またメモリーCは合計を取るメモリーですので、0を入れて消去します。

「B?」により入力された数値はメモリーBに記憶され、計算式「C=B+C」により、メモリーCに合計値が記憶されます。またそのたび、「Dsz A」によりメモリーA内の数値から1が減算されていきます。

もし、メモリーAの内容が0でないときは、「Goto 1」により「Lbl 1」に戻ります。

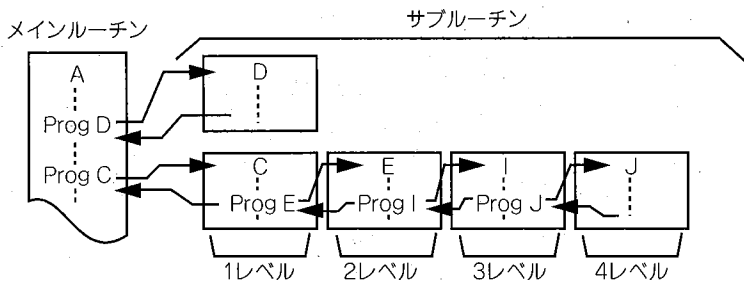
もしメモリーAの内容が0のときは、「Goto 1」の次の計算式「C÷10」を実行します。

## サブルーチン

プログラムで一連の計算を行なう流れを、「メインルーチン」と呼びます。このメインルーチンの中で必要に応じて使うことができるように独立させた部分を、「サブルーチン」と呼びます。ひとつのサブルーチンはメインルーチンのどこで何回使ってもよく、また違うメインルーチンに使うこともできます。

つまりプログラム中に何度も出てくる同じ計算をサブルーチンとして独立させると、必要に応じてメインルーチンで呼び出して計算することができます。





サブルーチン命令は「Prog」(SHIFT Prog)を使い、プログラムのファイル名を「”」(ダブルクォーテーション)で囲んで指定します。

**例** Prog "ABC" ……ファイル名「ABC」のプログラムにジャンプする。

- ・ジャンプ後は、指定したプログラムエリアのプログラムを先頭から実行します。  
実行終了後は、再び元のプログラムの「Prog "ファイル名"」の次の命令に戻ります。
  - ・サブルーチンとして組み込まれているプログラムから、さらに他のサブルーチンへジャンプすることができます。これを「ネスティング」と呼びます。ネスティングは、最高10レベル(10段)まで可能です。このレベルを超えて使うと、エラー「Ne ERROR」となります。
  - ・サブルーチン内の「Goto n」、「Lbl n」は、そのサブルーチン内でのみ有効です。
- ※「Prog "ファイル名"」により指定したプログラムが見つからないときは、エラー「Go ERROR」となります。

**例** 正8面体と正4面体の表面積と体積を求めるプログラムを同時に組む。ただし、答えは小数点以下3桁まで求める。

正8面体:ファイル名.....OCTAHEDRON

プログラム.....Fix 3

$$S=2 \times \sqrt{3} \times A^2 \blacktriangle$$

$$V=\sqrt{2} \div 3 \times A^3$$

正4面体:ファイル名.....TETRAHEDRON

プログラム.....Fix 3

$$S=\sqrt{3} \times A^2 \blacktriangle$$

$$V=\sqrt{2} \div 12 \times A^3$$

2つのプログラムを比較してみると、       を引いた部分が同じ内容となっています。

また、「OCTAHEDRON」のプログラムの        部分を4で割れば、「TETRAHEDRON」のプログラムの        の部分と同じ内容となります。

したがって、この部分を共通して使えば、プログラムは簡単にすることができます。

では、       を引いた部分をファイル名「S.SUB」と、「OCTAHEDRON」の        を引いた部分をファイル名「V.SUB」として、サブルーチンプログラムを組んでみましょう。

「S.SUB」のプログラム.....Fix 3

$$S=\sqrt{3} \times A^2$$

「V.SUB」のプログラム..... $V=\sqrt{2} \div 3 \times A^3$

また、共通部分を除いたメインルーチンプログラムは、以下のようになります。

正8面体:ファイル名.....OCTAHEDRON

プログラム.....Prog "S.SUB"

$$S = \text{Ans} \times 2 \blacktriangle$$

Prog "V.SUB"

正4面体:ファイル名.....TETRAHEDRON

プログラム.....Prog "S.SUB"  $\blacktriangle$

Prog "V.SUB"

$$V = \text{Ans} \div 4$$

このようにプログラムを組むと、「OCTAHEDRON」または「TETRAHEDRON」のプログラムを実行した後、「S. SUB」のプログラムにジャンプします。その後「S. SUB」のプログラムにより小数点以下3桁指定を行ない、一辺の長さAの値を入力して表面積を計算します。

正8面体の場合は「 $2\times$ 」が抜けていますので、「OCTAHEDRON」のプログラムに戻った後「 $S=Ans\times 2$ 」と操作して答えを求めます。

正4面体の場合はそのまま答えが求められていますので、「TETRAHEDRON」のプログラムに戻って答えを表示させます。

体積計算も同様に「V. SUB」のプログラムにジャンプして計算を行なった後、メインルーチンの「OCTAHEDRON」または「TETRAHEDRON」のプログラムに戻ってきます。

正8面体の場合は、そのまま答えを表示させます。

正4面体の場合は「 $\div 4$ 」を取り除いたので、「V. SUB」のプログラムで求められた答えをさらに4で割って答えを求めます。

このようにサブルーチンを使ってプログラムを組んだ方がバイト数も少なくてすみ、プログラムの構造もはっきりします。

## ポーズ命令

「Pause  $n$  ( $n$  は0~9の整数)」をプログラムに書き込むことにより、最大4.5秒までプログラムを停止させることができます。ポーズ停止中は、その時点でプログラムの計算結果(Ans内容)が表示されます。

「Pause  $n$  ( $n$  は0~9の整数)」で停止できる時間は、次のようになります(時間はおよその目安です)。

$n$	0	1	2	⋯	8	9
停止時間(秒)	0	0.5	1	⋯	4	4.5

※「Pause  $n$ 」は1つの文として扱います。

例

変数Aに1を次々加える計算を行なう。ただし、変数Aの初期値は1とする。

プログラムは次のようになります。

```
Lbl 1 ←
A = A + 1
Pause 3 (1.5秒結果表示)
Goto 1
```

無条件ジャンプ

このプログラムは、変数Aに一度値を入力すれば式「 $A=A+1$ 」を計算し、「Pause 3」により約1.5秒計算結果を表示します。その後、「Goto 1」により「Lbl 1」に戻るといふ実行を無限に繰り返します。

# 10



ライブラリー編

---

## 1. 素因数分解

### 内容計算式等

任意の正の整数 $m$ の素因数を選び出す。

但し、 $1 < m < 10^{10}$ で、

素数は小さな方から選び出し、“END”を表示したら終りとする。

〈考え方〉

$m$ を順次2および $d = 3, 5, 7, 9, 11, 13, \dots$  (奇数のすべて)の数列で割り、割り切れるかどうかを調べる。

$d$ が素数だった場合  $m_i = m_{i-1}/d$  とし、

$\sqrt{m_i} + 1 \leq d$  まで割算を繰り返す。

### 例題

〈例1〉

$$119 = 7 \times 17$$

〈例2〉

$$630 = 2 \times 3 \times 3 \times 5 \times 7$$

〈例3〉

$$262701 = 3 \times 3 \times 17 \times 17 \times 101$$

### 準備および操作

- ファイル名を登録します。
- プログラム実行時のモードを指定します。
- プログラムを計算機に書き込みます。
- 操作手順に従ってキーを押します。



# 操作手順

手順	キー操作	表示	手順	キー操作	表示
1	MODE (I) (COMP) FACTON (E) (DSP/CLR) (E) (McI) (EXE)	McI 0	12	(EXE)	PRIME FACTOR= 7
2	(FILE)	Program [RUN] ■SOINSUU :CO	13	(EXE)	END 630
3	(EXE)	M? 0	14	(EXE)	M? 7
4	119 (EXE)	PRIME FACTOR= 7	15	262701 (EXE)	PRIME FACTOR= 3
5	(EXE)	PRIME FACTOR= 17	16	(EXE)	PRIME FACTOR= 3
6	(EXE)	END 119	17	(EXE)	PRIME FACTOR= 17
7	(EXE)	M? 17	18	(EXE)	PRIME FACTOR= 17
8	630 (EXE)	PRIME FACTOR= 2	19	(EXE)	PRIME FACTOR= 101
9	(EXE)	PRIME FACTOR= 3	20	(EXE)	END 262701
10	(EXE)	PRIME FACTOR 3	21		
11	(EXE)	PRIME FACTOR= 5	22		
メモリー 内容	A	$mi$	H	O	V
	B	$d$	I	P	W
	C	$\sqrt{m} + 1$	J	Q	X
	D		K	R	Y
	E		L	S	Z
	F		M	T	
	G		N	$m$	U



## 2. 最大公約数

### 内容計算式等

ユークリッドの互除法で、 $a$ 、 $b$  2つの整数の最大公約数を求める。  
但し、 $|a|$ 、 $|b| < 10^9$  正の場合は  $< 10^{10}$  とする。

〈考え方〉

$$n_0 = \max(|a|, |b|)$$

$$n_1 = \min(|a|, |b|)$$

$$n_k = n_{k-2} - \left\lfloor \frac{n_{k-2}}{n_{k-1}} \right\rfloor n_{k-1}$$

$$k = 2, 3, \dots$$

$n_k = 0$  ならば、最大公約数( $c$ )は、 $n_{k-1}$  となる。

### 例 題

〈例1〉

$$a = 238$$

$$b = 374$$

のとき

$$c = 34$$

〈例2〉

$$a = 23345$$

$$b = 9135$$

のとき

$$c = 1015$$

〈例3〉

$$a = 522952$$

$$b = 3208137866$$

のとき

$$c = 998$$

### 準備および操作

- ファイル名を登録します。
- プログラム実行時のモードを指定します。
- プログラムを計算機に書き込みます。
- 操作手順に従ってキーを押します。



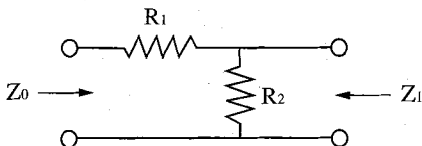
# 操作手順

手順	キー操作	表示	手順	キー操作	表示
1	F1 (COMP) F2 (DSP/CLR) F3 (Mc) EXE	Mc1 0	12		
2	FILE	Program [RUN] ■SOINSUU :CO KOUYAKUSUU :CO	13		
3	▼ EXE	A? 0	14		
4	238 EXE	B? 0	15		
5	374 EXE	C 34	16		
6	EXE	A? 102	17		
7	23345 EXE	B? 34	18		
8	9135 EXE	C 1015	19		
9	EXE	A? 4060	20		
10	522952 EXE	B? 1015	21		
11	3208137866 EXE	C 998	22		
メモリー内容	A	a、n <sub>0</sub>	H	O	V
	B	b、n <sub>1</sub>	I	P	W
	C	n <sub>k</sub>	J	Q	X
	D		K	R	Y
	E		L	S	Z
	F		M	T	
	G		N	U	

## 3. 最小損失整合

### 内容計算式等

$Z_0$ と $Z_1$ を最小損失でマッチングする $R_1$ 、 $R_2$ を求める。



$$R_1 = Z_0 \sqrt{1 - \frac{Z_1}{Z_0}}$$

$$R_2 = \frac{Z_1}{\sqrt{1 - \frac{Z_1}{Z_0}}}$$

$$\text{最小損失 } L_{\min} = 20 \cdot \log \left( \sqrt{\frac{Z_0}{Z_1}} + \sqrt{\frac{Z_0}{Z_1} - 1} \right) [\text{dB}]$$

### 例題

$$\left( \begin{array}{l} Z_0 = 500 \Omega \\ Z_1 = 200 \Omega \end{array} \right) \text{ のときの } \left( \begin{array}{l} R_1 \\ R_2 \\ L_{\min} \end{array} \right) \text{ を求めよ}$$

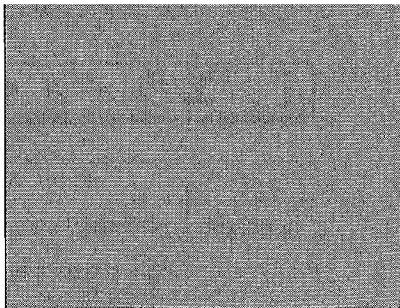
### 準備および操作

- ファイル名を登録します。
- プログラム実行時のモードを指定します。
- プログラムを計算機に書き込みます。
- 操作手順に従ってキーを押します。



# 操作手順

手順	キー操作	表示	手順	キー操作	表示
1	MODE (1) (COMP) EXE (2) (DSP/CLR) (3) (Mcl) EXE	Mcl 0	11		
2	FILE	Program [RUN] ■SOINSUU :CO KOUYAKUSUU :CO SONSHITU :CO	12		
3	▼▼ EXE	Z0? 0	13		
4	500 EXE	Z1? 0	14		
5	200 EXE	R1= 387.2983346	15		
6	EXE	R2= 258.1988897	16		
7	EXE	LMIN= 8.961393328	17		
8			18		
9			19		
10			20		
			21		
メモリー内容	A	$\sqrt{1-\frac{Z_0}{Z_1}}$	H	O	V
	B	$\frac{Z_0}{Z_1}$	I	P	W
	C		J	Q	X
	D		K	R	R <sub>1</sub> Y Z <sub>0</sub>
	E		L	S	R <sub>2</sub> Z Z <sub>1</sub>
	F		M	T	Lmin
	G		N	U	



## 卷末資料



# 1. エラーメッセージ一覧表

メッセージ	エラー内容	対策
Ma ERROR	<ul style="list-style-type: none"> <li>①計算結果が演算範囲を超えている。</li> <li>②関数桁容量の被演算数を超えた計算が行なわれた。</li> <li>③数学的な誤り（0による除算など）が行なわれた。</li> </ul>	①②③ 入力した数値を確認し、範囲内に直す。特にメモリーを使っている場合は、メモリー内の数値をチェックして正しくする。
Stk ERROR	<ul style="list-style-type: none"> <li>●数値用スタックおよび演算用スタックを超える計算式が実行された。</li> <li>●条件の数が15段を超えた条件ジャンプのプログラムを実行した。</li> </ul>	<ul style="list-style-type: none"> <li>●計算式を簡略化して数値用スタックは10段、命令用スタックは26段以内に収める。</li> <li>●計算式を2つ以上に分けて、スタック内に収める。</li> </ul>
Syn ERROR	<ul style="list-style-type: none"> <li>①計算式の書式に誤りがある。</li> <li>②プログラム内に書式上の誤りがある。</li> </ul>	① <input type="checkbox"/> または <input type="checkbox"/> を押してエラー箇所を表示させ、計算式を訂正する。 ② <input type="checkbox"/> または <input type="checkbox"/> を押してエラー箇所を表示させ、プログラムを訂正する。
Mem ERROR	増設していないメモリー（Z [5] など）を使った。	<input type="checkbox"/> <input type="checkbox"/> により、正しくメモリー数を増やす。
Arg ERROR	<ul style="list-style-type: none"> <li>①引き数を必要とする命令において引き数を間違えて入力した。</li> <li>②メモリーの残り容量を超えて、メモリーの増設を行なった。</li> </ul>	①引き数を正しい値に訂正する。 <ul style="list-style-type: none"> <li>●「Fix n」、「Sci n」の n を 0～9 の整数にする。</li> <li>●「Lbl n」、「Goto n」の n を 0～9 の整数または A～Z のアルファベット 1 文字にする。</li> <li>●「Defm n」の n を 0～残りバイト数以内の整数にする。</li> </ul> ② <ul style="list-style-type: none"> <li>●使用するメモリーの数を、現在設定されているメモリー数以内に収める。</li> <li>●記憶させるデータの内容を簡略化し、残り容量以内に収める。</li> <li>●不要なメモリー内のデータを削除する。</li> </ul>



メッセージ	エラー内容	対策
Ne ERROR	<p>「Prog」ファイル名」によるサブルーチンのネスティングが10レベル（10段）を超えている。</p>	<ul style="list-style-type: none"> <li>● サブルーチンからの戻りに、「Prog」ファイル名」を使っていないかをチェックし、不要な「Prog」ファイル名」を削除する。</li> <li>● サブルーチンのジャンプ先をたどり、元のプログラムエリアにジャンプしているかどうかをチェックし、正しく戻るように直す。</li> </ul>
Go ERROR	<p>①「Goto n」に対する「Lbl n」がない。          ②「Prog」ファイル名」により指定された場所にプログラムが記憶されていない。</p>	<p>①「Goto n」に対する「Lbl n」を正しく入れるか、不要であれば「Goto n」を削除する。          ②「Prog」ファイル名」により指定された場所にプログラムを記憶させるか、不要であれば「Prog」ファイル名」を削除する。</p>

## 2. 関数の入力範囲と精度

関数	入力範囲	内部演算桁数	精度	備考
$\sin x$ $\cos x$ $\tan x$	Deg $ x  < 9 \times 10^9$ Rad $ x  < 5 \times 10^7 \pi$ Gra $ x  < 1 \times 10^{10} \text{grad}$	15桁	原則として 10桁目±1	ただし $\tan x$ では、 $ x  \neq 90(2n+1)$ : Deg $ x  \neq \pi/2(2n+1)$ : Rad $ x  \neq 100(2n+1)$ : Gra
$\sin^{-1}x$ $\cos^{-1}x$	$ x  \leq 1$	〃	〃	
$\tan^{-1}x$	$ x  < 1 \times 10^{100}$			
$\sinh x$ $\cosh x$	$ x  \leq 230.2585092$	〃	〃	注意: $\sinh, \tanh$ では $x=0$ のとき特異点となります。この近傍では誤差が累積されて、精度が悪くなります。
$\tanh x$	$ x  < 1 \times 10^{100}$			
$\sinh^{-1}x$ $\cosh^{-1}x$ $\tanh^{-1}x$	$ x  < 5 \times 10^{99}$ $1 \leq x < 5 \times 10^{99}$ $ x  < 1$	〃	〃	
$\log x$ $\ln x$	$1 \times 10^{-99} \leq x < 1 \times 10^{100}$	〃	〃	
$10^x$ $e^x$	$-1 \times 10^{100} < x < 100$ $-1 \times 10^{100} < x \leq 230.2585092$	〃	〃	
$\sqrt{x}$ $x^2$	$0 \leq x < 1 \times 10^{100}$ $ x  < 1 \times 10^{50}$	〃	〃	
$x^{-1}(\frac{1}{x})$ $\sqrt[3]{x}$	$ x  < 1 \times 10^{100}, x \neq 0$ $ x  < 1 \times 10^{100}$	〃	〃	
$x!$	$0 \leq x \leq 69$ ( $x$ は整数)	〃	〃	
$nPr$ $nCr$	結果 $< 1 \times 10^{100}$ となる $n, r$ ( $n, r$ は、整数) $0 \leq r \leq n$ $n < 1 \times 10^{10}$	〃	〃	

関数	入力範囲	内部演算桁数	精度	備考
Pol(x, y)	$\sqrt{x^2+y^2} < 1 \times 10^{100}$	15桁	原則として 10桁目±1	
Rec(r, θ)	$ r  < 1 \times 10^{100}$ Deg $ \theta  < 9 \times 10^9$ ° Rad $ \theta  < 5 \times 10^9 \pi$ rad Gra $ \theta  < 1 \times 10^{10}$ grad			ただし、tanθでは、 $ \theta  \neq 90(2\pi+1)$ : Deg $ \theta  \neq \pi/2(2\pi+1)$ : Rad $ \theta  \neq 100(2\pi+1)$ : Gra
◦,,, ↔,,,	$ a , b, c < 1 \times 10^{100}$ $0 \leq b, c$  $ x  < 1 \times 10^{100}$ 60進数表示は $ x  \leq 27777777.777$	〃	〃	
$\wedge(x)$	$x > 0$ : $-1 \times 10^{100} < y \log x < 100$ $x = 0$ : $y > 0$ $x < 0$ : $y = n \cdot \frac{1}{2n+1}$ (nは整数) ただし、 $-1 \times 10^{100} < \frac{1}{y} \log  x  < 100$	〃	〃	
$\sqrt{y}$	$y > 0$ : $x \neq 0$ , $-1 \times 10^{100} < \frac{1}{x} \log y < 100$ $y = 0$ : $x > 0$ $y < 0$ : $x = 2n+1, \frac{1}{n}$ (n≠0, nは整数) ただし、 $-1 \times 10^{100} < \frac{1}{x} \log  y  < 100$	〃	〃	
$a^{b/c}$	(結果) 整数・分子・分母の合計が 10桁以内(ただし、区切り シンボルを含む)。 (入力) 整数・分子・分母の各項が $1 \times 10^{10}$ 未満の整数のとき 結果も分数となる。	〃	〃	



# 3. 仕 様

型 式：fx-4800P

## 演算部

基本計算：負数・指数・カッコを含む四則演算(加減乗除の優先順位判別機能つき)

組込関数：三角・逆三角関数(角度単位は度・ラジアン・グラード)、双曲・逆双曲線関数、対数・指数関数、べき乗、べき乗根、開平、2乗、逆数、立方根、階乗、負符号、指数部置数、 $\pi$ 、カッコ、内部数値丸め、乱数、角度単位指定、分数、10進 $\leftrightarrow$ 60進、座標変換、エンジニアリング、順列、組み合わせ、小数点以下設定、有効桁数設定

組込機能：指数表示範囲指定、デリート、インサート、アンサー機能、リプレイ、残り容量表示、イコール記号入力、マルチステートメント、出力命令入力

微分計算機能：中心差分による微分係数の算出

2次微分計算機能：2階数値微分公式による2次微分の算出

積分計算機能：シンプソン法による定積分の算出

$\Sigma$ (シグマ)計算機能：数列 $\{a_n\}$ の部分和の算出

複素数計算機能：複素数の加減乗除、逆数、平方根、自乗、絶対値/偏角計算、共役複素数の算出、実数部/虚数部の抽出

2進・8進・10進・16進計算機能：

2進・8進・10進・16進の四則計算、基数変換、負数計算(補数計算)、論理演算

統計計算機能：a. 標準偏差：データ数、平均、標準偏差(2種類)、総和、2乗和、 $t$ 検定  
b. 回帰計算：データ数、 $x$ の平均、 $y$ の平均、 $x$ の標準偏差(2種類)、 $y$ の標準偏差(2種類)、 $x$ の総和、 $y$ の総和、 $x$ の2乗和、 $y$ の2乗和、 $xy$ の積和、定数項、回帰係数、相関係数、 $x$ の推定値、 $y$ の推定値

数式記憶機能：数式の記憶/呼び出し/計算、テーブル機能、ソルブ機能、プログラムエリアへの数式の保存/呼び出し

メモリー：標準26メモリー(最大476メモリー)

計算範囲： $\pm 1 \times 10^{-99} \sim \pm 9.999999999 \times 10^{99}$ 、および0。内部演算は仮数部15桁を使用

小数点方式：浮動方式(工学浮動小数、指定小数)、

指数表示範囲切替可 Norm1モード： $10^{-2}>|x|, |x| \geq 10^0$

Norm2モード： $10^{-9}>|x|, |x| \geq 10^0$

四捨五入：有効桁数指定、または小数点以下桁数指定による四捨五入

### プログラム部

基本機能：プログラミング(ファイル名の登録、プログラムの書き込み・記憶、実行)、ファイル名/プログラムの訂正・消去、ファイル名によるプログラムの呼び出し

プログラム命令：変数入力命令「{ }」、変数確定命令「Fixm」、  
無条件ジャンプ「Goto」、「Lb1」  
条件ジャンプ「⇒」、「キ⇒」、「▲」、関係演算子「=、キ、>、<、≥、≤」  
カウントジャンプ「lsz」、「Dsz」  
サブルーチン「Prog」(ネスティング最高10レベル)  
ポーズ命令「Pause」

チェック機能：プログラムのチェック、デバッグ等

プログラム容量：最大4500バイト

### 共通部

表示方式および内容：液晶表示、16文字ドット4行表示、計算時・仮数部10桁+指数部2桁、2進・8進・16進数表示可、60進数表示可、分数表示可、複素数表示可

文字表示機能：関数命令表示、プログラム命令表示、アルファベット表示などを最大64文字表示

エラーチェック機能： $10^{100}$ 以上および計算不能、ジャンプ等不能をチェック、「○○○ ERROR」表示

電源：動作用……ボタン電池<CR2032>1個使用  
メモリー保護用……ボタン電池<CR2032>1個使用

消費電力：0.05W

電池寿命：動作用……約900時間(0表示状態で放置した場合)  
約1年間(電源OFFで放置した場合)

メモリー保護用…約2年間

オートパワーオフ機能：操作完了後約6分で自動電源OFF。

使用温度：0℃～40℃

大きさ：閉じたとき……幅81.5×奥行157×厚さ15mm  
開いたとき……幅165×奥行157×厚さ11mm

重さ：133g(電池込み)

## 保証・アフターサービスについて

### ● 保証書はよくお読みください。

保証期間は、お買上げ日から1年間です。

保証書(本書に刷り込まれています)は、必ず「お買上げ日・販売店名」などの記入をお確かめの上、販売店から受け取っていただき、内容をよくお読みの後、大切に保管してください。

### ● 修理を依頼されるときは

まず、もう一度本書にしたがって正しく操作していただき、直らないときには次の処置をしてください。

#### ● 保証期間中は

保証書の規定にしたがってお買上げの販売店またはカシオサービスセンターが修理させていただきます。製品に保証書を添えてご持参またはご送付ください。

#### ● 保証期間が過ぎているときは

お買上げの販売店またはカシオサービスセンターへご依頼ください。修理すれば使用できる製品については、ご希望により有料で修理いたします。

### ● あらかじめご了承くださいたいこと

「修理のとき一部代替部品を使わせていただくこと」や「修理が困難な場合は修理せず同等品と交換させていただくこと」があります。

また、特別注文された商品の修理では、ケースなどをカシオ純正部品と交換させていただくことがあります。

● 仕様が日本国内向けの製品は海外での修理受付ができません。修理品は日本まで移動の上、日本国内のカシオサービスセンターにご依頼ください。

### ● アフターサービスなどについて、おわかりにならないときは

お買上げの販売店または本書記載のカシオサービスセンターにお問い合わせください。

## カシオサービスセンター

■カシオ製品のアフターサービス業務は、カシオテクノ株式会社担当いたします。

### 北海道

札幌 ☎ 011-842-1231  
〒003-0805 札幌市白石区菊水五条1-19-5

### 北陸

金沢 ☎ 076-224-0061  
〒920-0064 金沢市南新保町ト52

### 東北

仙台 ☎ 022-256-8822  
〒983-0852 仙台市宮城野区榴岡5-1-35

盛岡 ☎ 019-646-3393  
〒020-0125 盛岡市上堂2-3-6

### 東海

静岡 ☎ 054-281-8085  
〒422-8056 静岡市津島町16-23

名古屋 ☎ 052-936-4601  
〒461-0004 名古屋市東区葵3-22-5

### 関東

水戸 ☎ 029-228-3155  
〒310-0803 水戸市城南3-10-17

宇都宮 ☎ 028-623-5588  
〒320-0053 宇都宮市戸祭町3009-8

高崎 ☎ 0273-22-9555  
〒370-0831 高崎市新町67-1

埼玉 ☎ 048-666-8567  
〒331-0043 大宮市大成町4-83

千葉 ☎ 043-243-1087  
〒260-0022 千葉市中央区神明町13-4

東京 ☎ 03-5294-7011  
〒101-0021 千代田区外神田2-2-18

秋葉原 ☎ 03-5820-9871  
〒101-0025 千代田区神田佐久間町2-15

多摩 ☎ 042-523-3990  
〒190-0012 立川市曙町1-22-17

横浜 ☎ 045-211-0811  
〒231-0006 横浜市中区南仲通3-26

### 近畿

京都 ☎ 075-351-1161  
〒600-8332 京都市下京区五条通り堀川東入ル

大阪 ☎ 06-352-7111  
〒530-0041 大阪市北区天神橋6-3-16

神戸 ☎ 078-392-2145  
〒650-0032 神戸市中央区伊藤町119

### 中国

岡山 ☎ 086-244-3404  
〒700-0926 岡山市西古松西町9-1

広島 ☎ 082-230-5900  
〒733-0001 広島市西区大芝2-14-10

### 四国

高松 ☎ 0878-62-5240  
〒760-0078 高松市今里町2-21

### 九州

福岡 ☎ 092-411-2939  
〒812-0007 福岡市博多区東比恵2-16-23

熊本 ☎ 096-367-0614  
〒862-0911 熊本市健軍1-38-7

鹿児島 ☎ 099-256-3573  
〒890-0065 鹿児島市郡元1-1-3

### 信越

信越 ☎ 025-287-1151  
〒950-0925 新潟市弁天橋通り3-9-12

※ 住所・電話番号などは変更になることがあります。あらかじめご了承ください。



# カシオ電卓保証書

持込修理

This warranty is valid only in Japan.

本書は、本書記載の内容により無料修理を行なうことをお約束するものです。お買上げの日から下記期間中に万一故障が発生した場合は、本書を提示の上、お買上げの販売店またはカシオサービスセンターに修理をご依頼ください。

★ご購入店様へ—

この保証書はお客様へのアフターサービスの実施と責任を明確にするものです。贈答品、記念品の場合も含めて必ず記入捺印してお客様にお渡しください。

機種名： <b>fx - 4800P</b>		保証期間：お買上げ日より本体1年間	
		お買上げ日： 年 月 日	
お客様	お名前 <span style="float: right;">様</span>		
	ご住所 〒		
	電話 — —		
販売店	住所・店名		

## 保証規定

- 取扱説明書の注意にしがった正常な使用状態で故障した場合には、お買上げの販売店が無料修理いたします。
  - 修理の必要が生じた場合は、製品と本証をお買上げの販売店または、カシオサービスセンターまでご持参またはご送付ください。
  - 修理品のご持参、お持ち帰りの交通費、またご送付される場合の送料および諸掛りはお客様のご負担となります。なお、ご送付の場合は適切な梱包の上、紛失防止のため受け渡しの確認できる手段(簡易書留や宅配など)をご利用ください。
  - 保証期間内でも次の場合は有料修理となります。
    - お買上げ後の輸送、移動時のお取扱いが不適当なため生じた故障・損傷
    - 誤用、乱用および取扱い不注意による故障・損傷
    - 不当な修理または改造による故障・損傷
    - 火災、地震、水害、その他の天災地変および異常電圧による故障・損傷
  - 消耗品(電池など)および付属品のお取替えの場合
  - 本書の提示がない場合および本書に(お買上げ日、お客様名、販売店名)の記入のない場合、あるいは字句を書き換えられた場合
  - 本書は日本国内においてのみ有効です。
  - 本書は再発行いたしませんので紛失しないよう大切に保管してください。
  - 修理内容などの記録は修理伝票にかえさせていただきます。
- ※ この保証書によってお客様の法律上の権利を制限するものではありませんので保証期間経過後の修理についてご不明の場合は、お買上げの販売店またはカシオサービスセンターにお問い合わせください。

カシオ計算機株式会社

〒151-8543 東京都渋谷区本町1-6-2

Phone 03-5334-4111

# CASIO

カシオ計算機株式会社

〒151-8543 東京都渋谷区本町1-6-2

SA9711-F Printed in Malaysia

A313025-66D M&M