



بسم الله الرحمن الرحيم

مقدمه:

دسترسی به محیط برنامه نویسی ماشین حساب های کاسیو و ایجاد فایل جدید برنامه نویسی در مدل‌های مختلف کمی با هم متفاوت می باشد به عنوان مثال برای دستگاه Casio fx-5800P توسط کلید [MODE] و سپس [5] قابل دسترسی است. جهت سایر مدل‌ها مانند الجبرا نیز محیط برنامه نویسی (PRGM) به راحتی از منوی اصلی، قسمت شماره ۸ قابل انتخاب می باشد.

در مدل ۵۸۰۰ بعد از ورود به محیط برنامه نویسی با فشردن کلید [1] و اختصاص نام مناسب به برنامه، کلید [EXE] را فشار دهید و آنگاه کلید [1] را جهت انتخاب Mode مناسب اعمال کنید. همچنین در مدل‌های الجبرا با استفاده از کلید [F3] می توان به سادگی فایل جدید ایجاد نمود.

در مدل کلاسپد محیط برنامه نویسی از منوی اصلی و انتخاب بخش Program قابل دسترس است. لازم به ذکر است برخی دستورات در مدل کلاسپد تفاوت‌هایی دارند که در این آموزش به فراخور نیاز توضیح داده می شوند.

طبقه بندی دستورات:

دستورات در ماشین حسابهای کاسیو به ۳ گروه اصلی زیر تقسیم می شوند:

دستورات ویژه برنامه نویسی (PROG)

دستورات بخش آماری (STAT)

دستورات بخش ریاضی (MATH)

دستورات فوق بعلاوه ۴ عمل اصلی و علامات، ارکان برنامه نویسی جهت برنامه های محاسباتی عمران می باشند. تمام دستورات فوق بعد از ورود به محیط برنامه نویسی ماشین حساب بوسیله کلیدهای تابعی و صفحه کلید دستگاه براحتی قابل دسترسی می باشند. همچنین انواع توابع معمول ریاضی که در محیط اصلی ماشین حساب قابل استفاده هستند در محیط برنامه نویسی نیز قابل کاربرد می باشند مانند توابع مثلثاتی، نمایی و لگاریتمی.

معرفی دستورات و علائم برنامه نویسی:

دستور ←

در انتهای هر خط از برنامه درج دستور ← جهت اجرای آن خط از برنامه ضروری است. این دستور با فشردن کلید [EXE] درج می گردد. پس از درج علامت این دستور، اشاره گر به طور خودکار به ابتدای خط بعدی رفته و آماده نوشتن خط بعدی برنامه می گردد.

مثال:

← 2×2

در مدل کلاسپد جهت اجرای هر خط از برنامه، نیاز به درج هیچ علامت خاصی نمی باشد و تنها فشردن کلید [EXE] برای رفتن به خط بعدی کفایت می کند.

دستور ▲

این دستور دقیقاً عملکردی مشابه دستور ← دارد و گاهی به جای آن استفاده می شود با این تفاوت که اگر از این دستور به جای دستور ← استفاده کنید برنامه در پایان اجرای آن خط با ایجاد یک وقفه منتظر می ماند تا شما کلید [EXE] را جهت ادامه اجرای برنامه اعمال کنید. هنگامی که قصد داشته باشیم یک خروجی یا متن را در صفحه نمایش دستگاه، نمایش دهیم این دستور کاربرد دارد.

مثال:

← 2+2

▲ 3×3

نتیجه اجرای برنامه فقط نمایش عدد ۹ خواهد بود، حال دستگاه منتظر است تا شما جهت ادامه یا اتمام اجرای برنامه کلید [EXE] را اعمال کنید.

در مدل کلاسپد برای نمایش نتیجه یک محاسبه همراه با وقفه باید از دستور PrintNatural و بدون وقفه از دستور Print استفاده گردد:

Print 2+2

PrintNatural 3×3,"Title"

PrintNatural 3×3

علامت " "

هر متن، علامت یا دستوری که داخل دو گیومه قرار گیرد فاقد هرگونه جنبه اجرایی خواهد بود و فقط در صفحه نمایش دستگاه نشان داده خواهد شد.

مثال:

▲ "SHORT COLUMN"

▲ "RO > RO MAX"

با توجه به اینکه هدف ما از کاربرد این علامت نمایش متن مورد نظر در صفحه نمایش می باشد بعد از این علامت حتما از دستور ▲ به جای دستور ← استفاده می نمایم در غیر این صورت برنامه سریعاً اجرا و پس از اجرای دستورات خطهای بعدی، متن غیر قابل مشاهده می گردد.

در مدل کلاسپد برای نمایش یک رشته متنی همراه با وقفه از دستور Message و بدون وقفه از دستور Print همراه با علامت " " استفاده می گردد:

Print "2×2=5!"

Message "Text","Title"

Message "Text"

دستور →

دستور فوق وظیفه تخصیص یک عدد را به یک متغیر دارد. ماشین حسابهای کاسیو ۲۶ متغیر که همان حروف الفبا هستند را به علاوه متغیرهای کمکی (بسته به مدل ماشین حساب) جهت محاسبات در اختیار شما می گذارند. به طور پیش فرض در همه مدلها (به جز کلاسپد) عدد تخصیص یافته به هر متغیر مقدار صفر می باشد.

مثال:

$$10 \rightarrow A \leftarrow$$

$$2(5) \rightarrow A \leftarrow$$

$$30 \div 3 \rightarrow A \leftarrow$$

نتیجه: هر یک از دستورات فوق به طور جداگانه عدد ۱۰ را داخل متغیر A قرار می دهد. تا زمانی که عدد دیگری به این متغیر تخصیص نیابد مقدار عددی متغیر A جهت استفاده های بعدی همان عدد ۱۰ خواهد ماند.

نکته: در ماشین حساب الجبرا و کلاسپد معمولاً علامت تقسیم به صورت / در صفحه نمایش درج می شود. همچنین در مدل کلاسپد می بایستی در صورت لزوم بین متغیرها و نیز قبل از پرانتزها علامت ضرب درج گردد.

مثال بیشتر:

$$3^{(3)} \rightarrow A \leftarrow$$

$$A \rightarrow B \leftarrow$$

نتیجه: مقدار متغیر A را به متغیر B اختصاص خواهد داد. تا زمانی که مقدار جدیدی به متغیر B اختصاص ندهیم مقدار آن عدد ۲۷ خواهد بود.

مثال دیگر:

$$2^{(3)} \rightarrow A \leftarrow$$

$$2(1+3) \rightarrow B \leftarrow$$

$$AB \rightarrow C \blacktriangle$$

نتیجه: مقدار A را در مقدار B ضرب می نماید و حاصل آن (۶۴) را به متغیر C تخصیص داده و مقدار متغیر C را در صفحه نمایش دستگاه نشان می دهد.

مثال مشابه:

$$2^{(3)} \rightarrow A \leftarrow$$

$$\leftarrow B \rightarrow 2(1+3)$$

$$\leftarrow B \rightarrow AB$$

$$\leftarrow Z \rightarrow B-2A$$

نتیجه: مقدار A را در مقدار B ضرب می نماید و حاصل آن (۶۴) را به متغیر B تخصیص می دهد. سپس عدد 48 را به متغیر Z تخصیص می دهد.

دستور ?

این دستور (?) برای گرفتن ورودی جهت تخصیص آن به یک متغیر کاربرد دارد.

مثال:

$$\leftarrow A \rightarrow "FY="?$$

نتیجه: دستور فوق عبارت FY=? را در صفحه نمایش نشان خواهد داد و منتظر خواهد ماند تا کاربر عددی را وارد نموده و سپس کلید [EXE] را جهت ورود آن عدد فشار دهد. عدد ورودی در متغیر A ذخیره خواهد شد.

مثال:

$$\leftarrow F \rightarrow "FC= (MPA)?"$$

$$\leftarrow A \rightarrow 0.85-0.0015F$$

$$\leftarrow B \rightarrow 0.97-0.0025F$$

$$\blacktriangle X \rightarrow AB$$

نتیجه: دستور فوق، عبارت FC= (MPA)? را در صفحه نمایش نشان خواهد داد در صورتی که مقدار ۲۵ را به متغیر F تخصیص دهیم مقادیر متغیرهای A و B به ترتیب اعداد ۰.۸۱۲۵ و ۰.۹۰۷۵ خواهند بود. در مرحله آخر مقدار A در B ضرب خواهد شد و مقدار ۰.۷۳ به متغیر X تخصیص خواهد یافت و در نهایت مقدار X در صفحه نمایش دستگاه نشان داده خواهد شد.

در مدل کلاسپد برای تخصیص ورودی ها از دستور Input استفاده می گردد:

Input A,"Text","Title"

Input A,"Text"

Input A

دستور :

علامت دو نقطه (:) جهت نمایش خروجی های برنامه کاربرد بسیاری دارد.

مثال :

 $400 \rightarrow Y \leftarrow$

"FY=":Y ▲

نتیجه: عدد 400 را به متغیر Y تخصیص خواهد داد و عبارت $FY=400$ را در صفحه نمایش نشان می دهد.

مثال دیگر:

 $400 \rightarrow Y \leftarrow$

"FY=":2(Y-250) ▲

نتیجه: عدد 400 را به متغیر Y تخصیص داده و عبارت $FY=300$ را در صفحه نمایش نشان خواهد داد.

مثال :

 $25 \rightarrow F \leftarrow$ $400 \rightarrow Y \leftarrow$

"FC×FY=":FY ▲

نتیجه: عبارت $FY \times FC = 10000$ را در صفحه نمایش دستگاه نشان خواهد داد.

نکته: پیشنهاد می گردد جهت نظم در برنامه نویسی حتما ابتدا محاسبات انجام گیرد و سپس نتیجه محاسبات در یک متغیر ذخیره گردد و خروجی نمایش داده شود یعنی دو مثال بالا بهتر است به صورت زیر بازنویسی گردند: (البته روش فوق نیز زمانی که بخواهیم در کاربرد متغیرها صرفه جویی نماییم گاهی مفید واقع می گردد به همین علت در اینجا جهت اهداف آموزشی ذکر گردید.)

بازنویسی برنامه اول:

 $400 \rightarrow Y \leftarrow$ $2(Y-250) \rightarrow F \leftarrow$

"FY=":F: ▲

نتیجه: خروجی برنامه هیچ تغییری نسبت به مثال مربوطه ندارد.

بازنویسی برنامه دوم:

25 → F ←

400 → Y ←

FY → W ←

"FC×FY=": W ▲

نتیجه: خروجی برنامه هیچ تغییری نسبت به مثال مربوطه ندارد.

در مدل کلاسپد نیز دستور فوق برای درج چند دستور در یک خط کاربرد دارد اما جهت نمایش عدد تخصیص یافته به یک متغیر همراه با وقفه باید از دستور PrintNatural و بدون وقفه از دستور Print استفاده گردد:

Print A

PrintNatural B,"Title"

PrintNatural B

دستور Cls یا ClrText

دستور فوق، خروجی صفحه نمایش دستگاه را کاملا پاک می نماید. از این دستور برای بهتر اجرا شدن برنامه ها و اخذ خروجی های آراسته تر استفاده می شود. دستور Cls متعلق به مدل ۵۸۰۰ و دستور ClrText مربوط به مدل های پیشرفته تر مانند الجبرا و کلاسپد می باشد.

مثال:

"AV/S=" ▲

"LONG COLUMN" ▲

Cls ←

دستور If

این دستور پرکاربرد شامل ۴ بخش If ~ Then ~ Else ~ IfEnd می باشد. شرط یا شروط فورا پس از کلمه If درج می گردد. همچنین می توان در صورت عدم نیاز، قسمت Else را استفاده نکرد.

ساختار کلی دستور:

If..... ←

Then ←

دستوراتی که در صورت برقراری شرط اجرا می شوند

Else ←

دستوراتی که در صورت عدم برقراری شرط اجرا می شوند

IfEnd ←

ساختار کلی دستور بدون Else :

If..... ←

Then ←

دستوراتی که در صورت برقراری شرط اجرا می شوند

IfEnd ←

استفاده از این دستور بدون Else بدان معناست که اگر شرط برقرار نبود نیازی به اجرای دستور خاصی نیست.

مثال :

If A=B ←

Then ←

"A=B" ▲

IfEnd ←

نتیجه: در صورتی که A=B باشد عبارت A=B نمایش داده خواهد شد.

مثال دیگر:

If R<M ←

Then ←

M→R ←

"RO MIN USED!" ▲

IfEnd ←

نتیجه: در صورتی که مقدار R کوچکتر از M باشد مقدار عددی M به متغیر R تخصیص داده خواهد شد و عبارت RO MIN USED! نمایش داده می شود.

نکته: دستور فوق مثلاً می تواند جهت اعمال کنترلهای فولاد حداقل و حداکثر مفید باشد.

مثال دیگر:

```

If R>B ←
Then ←
1+2→M ←
"M=":M ▲
Else ←
2+3→M ←
"M=":M ▲
IfEnd ←

```

نتیجه: در صورتی که مقدار R بزرگتر از B باشد عبارت M=3 نمایش داده خواهد شد و در صورتی که مقدار R کوچکتر از B یا مساوی B باشد آنگاه عبارت M=5 نمایش داده خواهد شد.

ترفند: می توان از یک دستور شرطی مانند If در یک دستور If دیگر استفاده نمود این کار جهت برنامه های عمرانی بسیار رایج می باشد.

مثال:

```

If R>B ←
Then ←
If A=B ←
Then ←
"A=B" ▲
Else ←
"A≠B" ▲
IfEnd ←

```

← M → 1+2

▲ M: "M="

← Else

← M → 2+3

▲ M: "M="

← IfEnd

نتیجه: در صورتی که مقدار R بزرگتر از B باشد و A مساوی B باشد عبارت $A=B$ و سپس عبارت $M=3$ نمایش داده می شود و اگر مقدار R بزرگتر از B باشد و A مساوی B نباشد عبارت $A \neq B$ و سپس عبارت $M=3$ نمایش داده خواهد شد و در صورتی که مقدار R کوچکتر یا مساوی B باشد آنگاه عبارت $M=5$ نمایش داده می شود.

نکته: از دستور If به صورت تو در تو می توان بدون هرگونه محدودیتی استفاده نمود البته این امر باعث پیچیده شدن برنامه خواهد شد که در ادامه آموزش راه حلی جهت رفع این مسئله ارائه می گردد.

ترفند: در مثالهای بالا بعد از عبارت If یک شرط قرار گرفته است مثلا $A=B$ یا هر شرط تکی دیگر. به این گونه If ها دستور If تک شرطی می گویند. دستور If می تواند به صورت تک شرطی، دو شرطی و یا چند شرطی به کار رود. برای روشن شدن موضوع به مثالهای زیر توجه نمایید:

← If $R \geq B$

معنای شرط فوق: اگر R بزرگتر یا مساوی B باشد.

← If $R < B$ And $R > M$

معنای شرط فوق: اگر R کوچکتر از B باشد و به طور همزمان R بزرگتر از M باشد.

نکته: دستور دو شرطی فوق مثلا می تواند جهت تشخیص نوع مقطع با فولاد فشاری به کار رود.

← If $R < B$ And $R > M$ And $A \neq B$

معنای شرط فوق: اگر R کوچکتر از B باشد و به طور همزمان R بزرگتر از M باشد و باز هم به طور همزمان مساوی B نباشد.

← If $A < B$ Or $C > D$

معنای شرط فوق: اگر A کوچکتر از B باشد و یا C بزرگتر از D باشد. (برقراری یکی از شروط برای برقراری کلی شرط کفایت).

نکته: عملگرهای منطقی And و Or مثل If دستور می باشند و (به جز کلاسپد) نباید تایپ گردند! این عملگرها به همراه دستورات مقایسه ای (\leq \geq $<$ $=$ $>$) در بخش دستورات ویژه برنامه نویسی قابل دسترسی می باشند.

یک مثال کاربردی با استفاده از دستور If :

```
"N="?" → N ←
```

```
If N=0 ←
```

```
Then ←
```

```
"AS="?" → A ←
```

```
Else ←
```

```
"DB= (mm)"? → X ←
```

```
0.25NπX^(2) → A ←
```

```
IfEnd ←
```

شرح مثال: در صورتی که مقدار صفر را به N اختصاص دهید برنامه مقدار AS را مستقیماً از شما دریافت خواهد کرد و در صورتی که به N عددی صحیح و بزرگتر از صفر تخصیص دهید با توجه به اینکه N تعداد میلگردها و DB قطر میلگرد می باشد مقدار AS توسط برنامه محاسبه خواهد شد.

مثال کاربردی دیگر:

```
"K="?" → K ←
```

```
If K ≤ 1 ←
```

```
Then ←
```

```
"BRACED" ▲
```

```
Else ←
```

```
"UNBRACED" ▲
```

```
IfEnd ←
```

نتیجه: در صورتی که مقدار K کوچکتر یا مساوی ۱ باشد قاب مهاربندی شده و در غیر این صورت قاب دارای حرکت جانبی می باشد.

در مدل کلاسپد با استفاده از جزء Elseif در دستور IF می توان حجم برنامه نویسی برای ایجاد شروط تو در تو را کاهش داد و برنامه ها را ساده تر و منظم تر نوشت.

دستور Stop

دستور فوق در هر جایی از برنامه به کار رود به اجرای برنامه فوراً خاتمه خواهد داد. این دستور بیشتر اوقات در داخل دستور If کاربرد دارد به صورتی که اگر حالت خاصی اتفاق افتاد پیغام خاصی نمایش داده شود و اجرای برنامه پایان یابد.

مثال:

```

If X≠Y ←
Then ←
1 → Z
Else ←
Stop ←
IfEnd ←

```

نتیجه: در صورتی که $X=Y$ باشد روند اجرای برنامه متوقف می گردد!

دستور Goto

این دستور ساده و پرکاربرد در برنامه های طولانی که قسمتهای مشابه هم دارند بسیار مفید می باشد و ضمن عدم نیاز به دوباره نویسی دستورات به خصوص دستورات ورودی و محاسبه ای از به وجود آمدن تعداد زیادی If تو در تو جلوگیری می کند. این دستور ۲ جزئی به صورت:

Goto n ~ Lbl n

می باشد که بعد از هر قسمت بجای حرف n یک عدد از صفر تا ۹ یا یک حرف از A تا Z قابل استفاده است.

مثال:

```

Goto 1 ←
Lbl 1 ←

```

یعنی هنگامی که روند اجرای برنامه به دستور Goto 1 برسد دستورات بعدی پس از آن را اجرا نخواهد کرد و مستقیماً به دستور Lbl 1 پرش خواهد کرد و دستورات پس از Lbl 1 اجرا خواهند شد. می توان به کرات و حتی به صورت تو در تو از این دستور در یک برنامه استفاده کرد. یکی از کاربردهای مفید این دستور استفاده از آن

داخل دستور If می باشد که در نهایت باعث عدم دوباره نویسی قطعه کدهای مشابه و سادگی برنامه های طولانی و پیچیده می شود.

مثال:

← دستور ۱

← دستور ۲

← If A=0

← Then

← دستور ۳

← Goto 1

← IfEnd

← دستور ۴

← دستور ۵

← دستور ۶

← Lbl 1

← دستور ۷

نتیجه: در برنامه فوق در صورتی که مقدار A مساوی صفر باشد فقط دستورات شماره ۱ و ۲ و ۳ و ۷ اجرا خواهند شد و در صورتی که A عددی غیر از صفر باشد همه دستورات غیر از ۳ اجرا می گردند.

مثال دیگر:

← دستور ۱

← دستور ۲

← If A=0

← Then

← دستور ۳

← Goto 1

← Else

← Goto 2

← IfEnd

← دستور ۴

← دستور ۵

← Lbl 2

← دستور ۶

← Lbl 1

← دستور ۷

نتیجه: در برنامه فوق در صورتی که مقدار A مساوی صفر باشد فقط دستورات شماره ۱ و ۲ و ۳ و ۷ اجرا خواهند شد و در صورتی که A عددی غیر از صفر باشد فقط دستورات شماره ۱ و ۲ و ۶ و ۷ اجرا خواهند شد.

ترفند: در مثال های فوق می توان دستور Lbl را قبل از دستور Goto قرار داد که منجر به ایجاد یک حلقه یا دستور بازگشتی می گردد. قابل ذکر است که این گونه حلقه ها برای کنترل مکرر مقاطع فولادی با مشخصات متفاوت (ایجاد چرخه های سعی و خطا) در برنامه های طراحی کاربرد فراوانی دارند.

دستور Prog

گاهی اوقات امکان فراخوانی یک برنامه به وسیله برنامه دیگر می تواند مفید واقع گردد مثلا اگر در سازه های بتن آرمه، برنامه نوع عملکرد مقطع T شکل را مستطیلی تشخیص دهد می توانید برنامه مربوط به آنالیز مقاطع مستطیلی را فراخوانی کنید. همچنین برنامه هایی که دارای قطعه های مشابه و تکراری هستند توسط این دستور ساده سازی و کم حجم تر می شوند بدین صورت که یک بار نوشته شده و در قسمت های مختلف از برنامه های مختلف استفاده شوند.

مثال:

← Prog "RECTANG"

نتیجه: برنامه ای با نام RECTANG فراخوانی و اجرا خواهد شد، سپس ادامه برنامه اصلی اجرا می گردد.

در مدل کلاسیک از دستور (RECTANG) برای فراخوانی برنامه فوق استفاده کنید.

دستور Return

این دستور کم کاربرد فقط در برنامه های فراخوانی شده کاربرد دارد و عملکرد آن بدین صورت می باشد که روند اجرای برنامه فراخوانی شده به همان برنامه اصلی که این برنامه را فراخوانی نموده است باز می گردد.

نکته: به طور طبیعی پس از اتمام اجرای برنامه فراخوانی شده این بازگشت انجام خواهد شد.

دستورات کنترل ساختار: (تکرار)

دستورات تکرار شامل 2 دستور اصلی زیر می باشند:

Do ~ LpWhile

While ~ WhileEnd

این دستورات انجام یکسری دستورات مشروط را تا برقراری شرطی معین تکرار می کنند و عملکردی کاملا شبیه به هم دارند با این تفاوت که در دستور اول، دستورات مشروط برای بار اول اجرا و سپس در صورت برقراری شرط، تکرار می شوند ولی در دستور دوم اگر شرط برقرار نباشد دستورات مشروط حتی برای بار اول هم اجرا نخواهند شد!

مثال کاربردی:

"N="?" → N ←

If N=0 ←

Then ←

D-M → N ←

IfEnd ←

0 → Z ←

0 → V ←

Do ←

Cls ←

Z+1 → Z ←

"A="?" → A ←

"B="?" → B ←

AB → Q ←

V+Q → V ←

LpWhile N>Z ←

"V=":V ▲

شرح: در ۵ خط اول برنامه، به وسیله عدد ورودی یا طی یک پروسه محاسباتی دلخواه مانند (D-M) حداکثر تعداد تکرارها (N) مشخص می گردد. از متغیر Z به عنوان شمارنده استفاده کردیم بنابراین در ابتدای پروسه به متغیر شمارنده مقدار صفر را تخصیص می دهیم. متغیر V پذیرنده عددیست که مجموع جوابهای محاسبات متوالی (AB) را در بر می گیرد که قبل از شروع محاسبات باید صفر گردد. در هر بار اجرای حلقه به شمارنده یکی اضافه گردیده و متغیر محاسباتی جدید (Q) به مقدار قبلی V افزوده می شود و تا زمانی که مقدار عددی شمارنده (Z) از مقدار عددی متغیر کنترل (N) کوچکتر باشد پروسه محاسباتی برای محاسبه متغیر V ادامه خواهد یافت.

یک مثال ریاضی برای درک بهتر کاربری دستورات تکرار:

"M="?" → M ←

"N="?" → N ←

0 → V ←

While N ≥ M ←

M^(2) → Q ←

V+Q → V ←

M+1 → M ←

WhileEnd ←

"V=":V ▲

نتیجه: تا زمانی که شرط (N ≥ M) برقرار باشد تمامی دستورات بین While و WhileEnd تکرار خواهند شد. برنامه فوق مجموع مربعات اعداد طبیعی از M تا N را محاسبه می کند، اگر مقادیر M و N را به ترتیب ۶ و ۹ وارد کنید حاصل نهایی مقدار ۲۳۰ خواهد بود.

نکته: در این مثال گام پرش دستور شمارنده (M+1) مقدار یک می باشد، همچنین در این مثال دستور شمارنده نمی تواند قبل از دستور محاسباتی (M^(2)) قرار گیرد!

دستور Break

دستور فوق برای قطع روند تکرار در دستورات کنترل ساختار به کار می رود. فرق این دستور با دستور Stop این است که فقط روند تکرار حلقه را متوقف کرده و ادامه برنامه را پی می گیرد در حالی که دستور Stop به طور کلی برنامه را متوقف می کند. این دستور عموماً به همراه دستور If داخل حلقه های تکرار به کار می رود.

دستورات آماری:

دستورات آماری در برنامه های عمرانی بیشتر جهت یافتن مقایر ماکسیمم و مینیمم کاربرد دارند، هر چند جهت یافتن مقادیر ماکسیمم و مینیمم داده ها می توان از دستور If به سادگی استفاده نمود اما استفاده از دستورات آماری به منظم تر و کوتاه تر شدن برنامه ها کمک می نماید. استفاده از این دستورات را که شامل دستورات مهم MaxX و MinX می باشد با چند مثال ساده توضیح خواهیم داد، همچنین لازم به یادآوری است که دستورات و علائم به کار رفته در این بخش و همچنین سایر دستورات ساده آماری مانند میانگین و غیره از قسمت دستورات آماری محیط برنامه نویسی ماشین حساب قابل درج می باشند.

مثال:

$$\leftarrow \text{List X} \rightarrow \{1,2\}$$

$$\leftarrow A \rightarrow 0.002 \text{MaxX}$$

$$\blacktriangle A: \text{RO MIN} = "$$

نتیجه: دستور فوق اعداد ۱ و ۲ را به ترتیب داخل ستون جدول Xهای بخش محاسبات آماری قرار داده و خط دوم دستور، مقدار دوهزارم ماکسیمم اعداد داخل جدول را به متغیر A تخصیص می دهد. نتیجه کلی برنامه، نمایش عبارت $\text{RO MIN} = 0.004$ می باشد.

مثال دیگر:

$$\leftarrow A \rightarrow 0.025$$

$$\leftarrow B \rightarrow 0.012$$

$$\leftarrow \text{List X} \rightarrow \{A, 2B, 0.03\}$$

$$\leftarrow C \rightarrow \text{MinX}$$

$$\blacktriangle C: \text{RO MAX} = "$$

نتیجه: خروجی نهایی این برنامه نمایش عبارت $\text{RO MAX} = 0.024$ است.

در مدل‌های جدیدتر مانند الجبرا و کلاسپد به جای دستور آماری List X از دستور مشابه List 1 و به جای دستورات MaxX و MinX به ترتیب از دستورات Max(List 1) و Min(List 1) استفاده می‌گردد.

دستورات ریاضی:

طیف وسیعی از دستورات ریاضی در محیط برنامه نویسی قابل استفاده می‌باشند که برای نمونه ۲ مورد ساده از آنها که کاربرد بیشتری دارند را جهت نمونه معرفی می‌کنیم، سایر دستورات رایج ریاضی نیز به سهولت قابل استفاده می‌باشند.

دستور Abs

این تابع ریاضی، قدر مطلق یک عدد را باز می‌گرداند و کاربرد آن بیشتر جهت ورودی‌های برنامه می‌باشد.

مثالها:

$$\text{Abs}(A) \rightarrow B \leftarrow$$

نتیجه: مقدار متغیر B در هر صورت مثبت می‌باشد.

$$-\text{Abs}(Z) \rightarrow Z \leftarrow$$

نتیجه: مقدار متغیر Z در هر صورت منفی می‌گردد.

دستور Int

این تابع ریاضی در برنامه‌های عمرانی بسیار پرکاربرد و مفید می‌باشد، تابع Int جز صحیح (غیر اعشار) یک عدد را بر می‌گرداند.

مثال کاربردی:

$$\text{"AS="} \rightarrow A \leftarrow$$

$$\text{"DB= (MM)" } \rightarrow X \leftarrow$$

$$(4A) \div (\pi X^{\wedge}(2)) \rightarrow N \leftarrow$$

$$\text{Int}(N) \rightarrow N \leftarrow$$

$$N+1 \rightarrow N \leftarrow$$

$$"N=":N \blacktriangle$$

شرح: برنامه فوق مقدار A یا سطح مقطع میلگردهای مورد نیاز و قطر میلگرد مورد استفاده را از کاربر می گیرد و سپس تعداد میلگردهای مورد نیاز را محاسبه و به سمت بالا رند کرده و به کاربر نمایش می دهد.

یک مثال کاربردی دیگر در مورد رند کردن فواصل:

$$180 \rightarrow S \leftarrow$$

$$"ROUND FACTOR = (MM)"? \rightarrow R \leftarrow$$

$$\text{If } R > 0 \leftarrow$$

$$\text{Then } \leftarrow$$

$$S \div R \rightarrow S \leftarrow$$

$$\text{Int}(S) \rightarrow S \leftarrow$$

$$SR \rightarrow S \leftarrow$$

$$\text{IfEnd } \leftarrow$$

$$"S=":S \blacktriangle$$

نتیجه: در مثال فوق اگر مقدار R را صفر بدهید از رند شدن مقدار متغیر S که مثلا می تواند فواصل محاسبه شده بین خاموتهای عرضی باشد صرفنظر می گردد. اما هر عددی به جز صفر مقدار S را با مضرب R به سمت پایین گرد خواهد نمود. اگر فاصله محاسباتی بین خاموتها مثلا ۱۸۰ باشد و به R عدد ۵۰ را تخصیص دهیم آنگاه فاصله بین خاموتها مقدار ۱۵۰ نمایش داده خواهد شد.

خطای اولویت محاسبات:

در عبارات محاسباتی جهت جلوگیری از بروز خطا حتما باید در موارد نیاز از پرانتز استفاده نمایید.

مثال:

$$3 \times 4 - 3 \blacktriangle$$

$$3(4-3) \blacktriangle$$

حاصل عبارت اول ۹ و حاصل عبارت دوم ۳ می باشد!

یک برنامه ساده:

"A="?→A←

"B="?→B←

"C="?→C←

 $B^2-4AC \rightarrow D \leftarrow$ $(-B+D^{0.5}) \div (2A) \rightarrow X \leftarrow$

"X=":X▲

شرح: برنامه فوق، ریشه مثبت و بزرگتر یک معادله درجه دوم با ضرایب A و B و C را نمایش می دهد (البته به شرط مثبت بودن مقدار A) این برنامه مثلا برای محاسبه ارتفاع بلوک تنش فشاری و نهایتا محاسبه تار خنثی در آنالیز مقاطع خمشی در حالتی که یکی از فولادها جاری نمی شود کاربرد دارد.

نکته: توجه کنید که اگر پرانتز برای عبارت 2A به کار برده نشود نتیجه محاسبات صحیح نخواهد بود، در صورت عدم کاربرد این پرانتز، عبارت اول، ابتدا بر ۲ تقسیم و سپس ضربدر A خواهد شد که منتهی به محاسبه جوابی غلط می گردد.

دستور Locate

این دستور یک رشته متنی یا عدد و یا مقدار یک متغیر را در یک مکان دلخواه در صفحه نمایش دستگاه نشان می دهد. فرم کلی دستور به صورت Locate X,Y,Z می باشد که پارامترهای X و Y مختصات صفحه نمایش و پارامتر Z رشته ای متنی، عبارت و یا عددی می باشد که می خواهیم در صفحه نمایش دستگاه نشان داده شود.

مثال:

5→A : 3→B : 1→C←

Locate A,B,C←

Locate 5,3,1←

Locate 5,3,"1"←

نتیجه: در هر صورت متن یا عبارت "1" در مختصات X=5 و Y=3 صفحه نمایش دستگاه نشان داده خواهد شد.

تهیه و تنظیم توسط:

سعید سیدعلیخانی