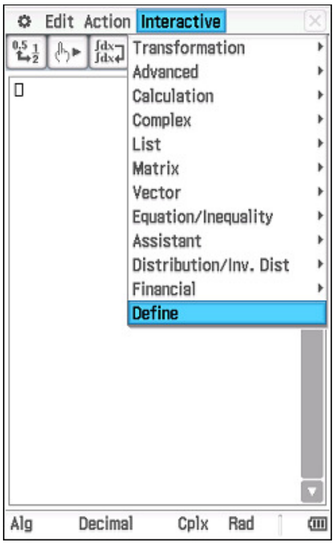
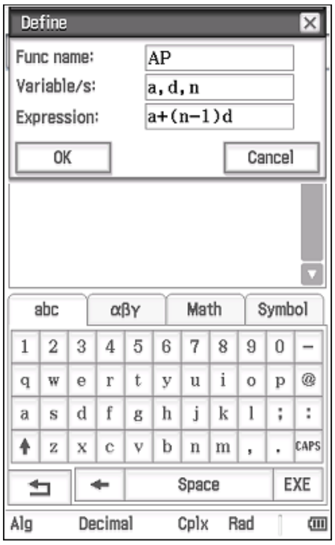
ClassPad II - How to add your own functions and programs

1. Define new functions in Main

This is useful for simple functions which use a formula to return a single value, such as the nth term of an AP. Open the Main application. Use the ***abc*** tab to enter the function name and don't use a name already reserved by ClassPad. For variables and expressions, be careful with syntax if you continue to use the ***abc*** tab.

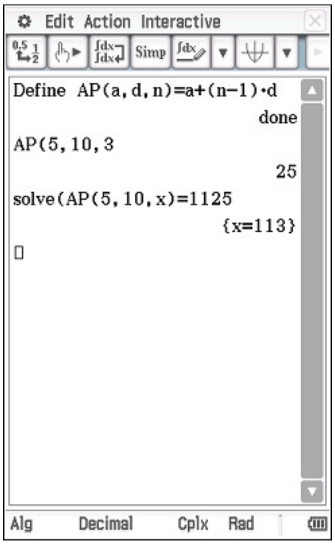



Define

Func name: AP

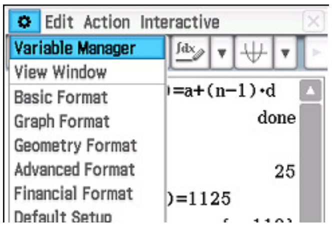
Variable/s: a, d, n

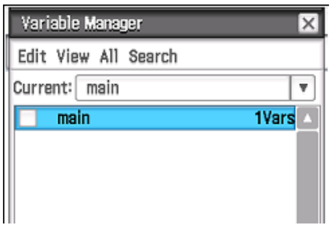
Expression: a+(n−1)d


```
Define AP(a,d,n)=a+(n−1)·d
                          done
AP(5,10,3)
                            25
solve(AP(5,10,x)=1125
                       {x=113}
```

> **Tip:**
> A user-defined function can only contain a single mathematical expression and must not contain any commands. You may want to consult the user manual: As an example, it is OK to use binomialCDf, but not binomialCD, as the former is a function but the latter is a command!

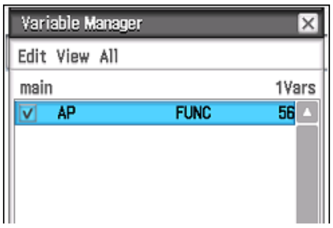
Once created in Main, a good idea is to shift functions into the library folder, so that the function can be accessed from within eActivities, as well as the rest of the calculator.





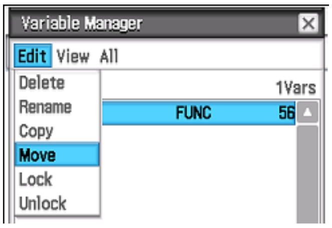
Current: main

main 1Vars



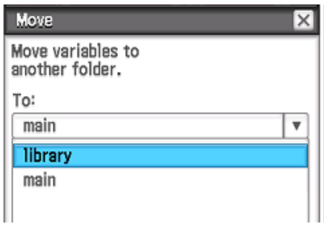
main 1Vars

☑ AP FUNC 56



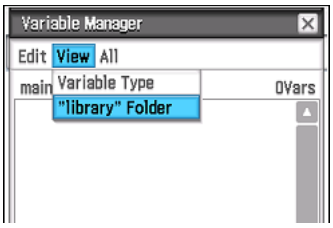
1Vars

FUNC 56

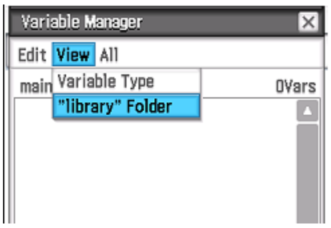

Move

Move variables to another folder.

To: main



main 0Vars



main 0Vars

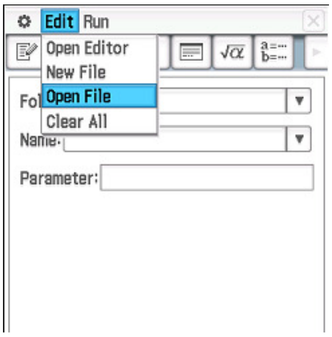
To delete a function, follow the above steps to move the function, but choose Delete rather than Move as shown in the fourth image.

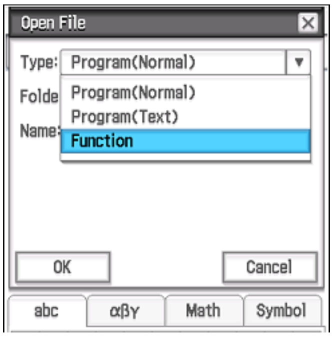
If a function doesn't behave as expected, try defining it again. If you use an existing user defined function name, ClassPad will simply overwrite the old function with the new one.

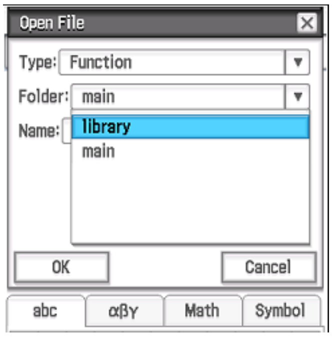
Watch videos 043, 044, 071 at www.classpad.com.au

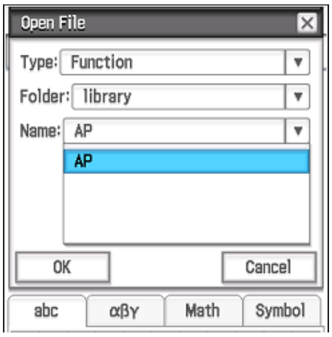
Alternatively, functions can be edited in the Program application.

From the main menu, open the Program application and tap Edit, Open File. Choose the Type, Folder and Name of function as illustrated below.

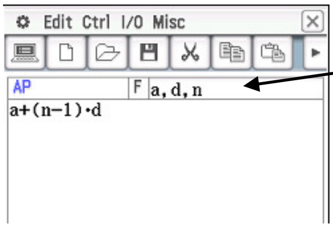








Then tap OK.



AP F a, d, n

a+(n−1)·d

Note that this line shows the function name (AP), the type (F = function) and the required variables in the order that they should be entered.

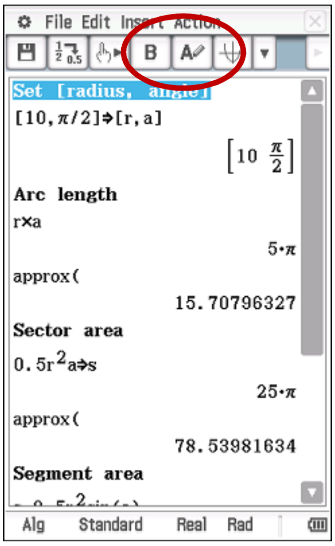
Make any corrections or edits to your function and then tap Edit, Save File.

2. Create open programs in eActivity

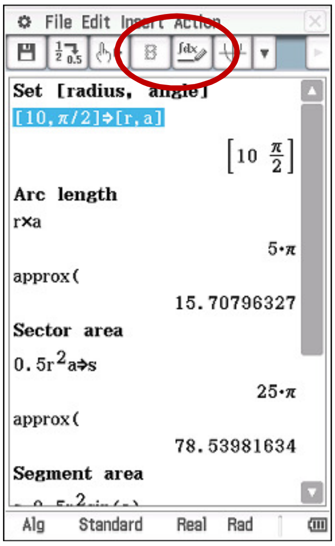
'Open' programs allow the user to see all the steps involved in producing the end result and allow for easy modification on the fly. Programs can be written either in the main body of an eActivity or in a Main strip within an eActivity. In the main body we can add line by line comments to explain our program, although using strips allows us to store more than one program within an eActivity (but no comments).

Open the eActivity application and start a new eActivity - tap File, New.

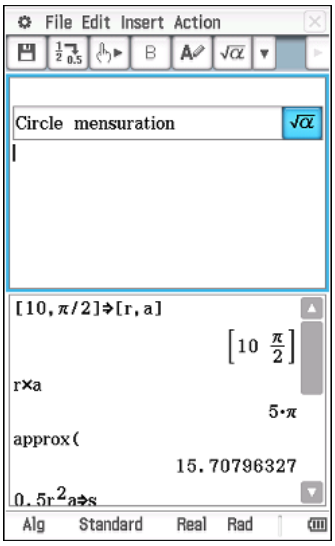
Watch video 632 at www.classpad.com.au



```
Set [radius, angle]
[10,π/2]⇒[r,a]
                    [10  π/2]
Arc length
r×a
                         5·π
approx(
              15.70796327
Sector area
0.5r²a⇒s
                        25·π
approx(
              78.53981634
Segment area
```

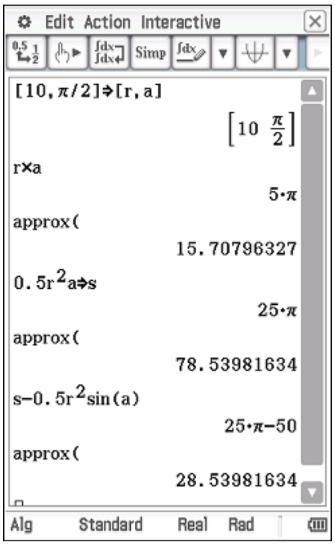


```
Set [radius, angle]
[10,π/2]⇒[r,a]
                    [10  π/2]
Arc length
r×a
                         5·π
approx(
              15.70796327
Sector area
0.5r²a⇒s
                        25·π
approx(
              78.53981634
Segment area
```



```
Circle mensuration

[10,π/2]⇒[r,a]
                    [10  π/2]
r×a
                         5·π
approx(
              15.70796327
0.5r²a⇒s
```



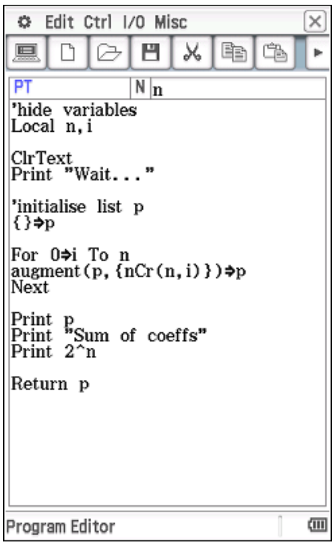
```
[10,π/2]⇒[r,a]
                    [10  π/2]
r×a
                         5·π
approx(
              15.70796327
0.5r²a⇒s
                        25·π
approx(
              78.53981634
s−0.5r²sin(a)
                     25·π−50
approx(
              28.53981634
```

Note how to store multiple variable values using just one line. Also note the difference between a text line, as selected in first image, and a calculation line, as selected in second image. Use ∫dx and A to toggle between the two types. Bold selected text using B.

A help sheet on creating eActivities is available from the www.charliewatson.com/classpad.

3. Create simple programs in Program.

For this example, we'll calculate the coefficients of the nth row of Pascal's Triangle - useful for binomial expansions, etc. Open the Program application and tap Edit, New File. Enter a name - PT - and tap OK.



```
'hide variables
Local n,i

ClrText
Print "Wait..."

'initialise list p
{}⇒p

For 0⇒i To n
augment(p,{nCr(n,i)})⇒p
Next

Print p
Print "Sum of coeffs"
Print 2^n

Return p
```

Notes

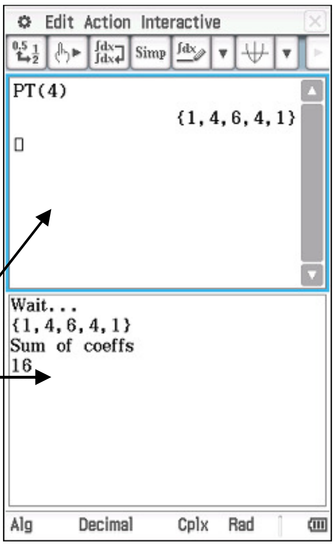
- This program has just one input variable, n
- Lines starting with an apostrophe are ignored
- Start a new line by pressing the EXE key

Local variables are used only within program and then cleared.

ClrText clears the result screen.

Print writes values and text to the result screen.

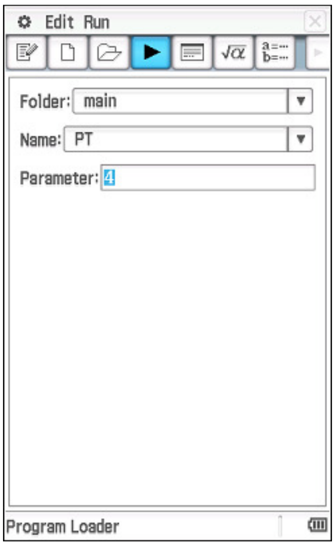
Return writes variables to the main screen, when program is run in main.



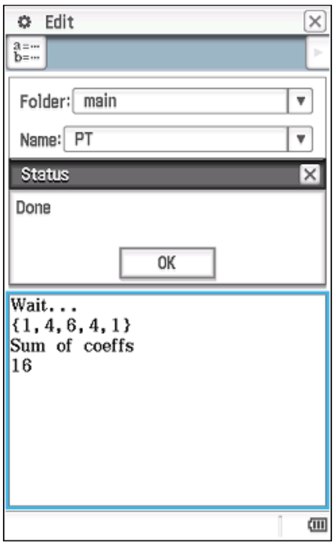
```
PT(4)
                {1,4,6,4,1}
```

```
Wait...
{1,4,6,4,1}
Sum of coeffs
16
```

Once your code has been entered, tap Edit, Save File and then tap 🖩 . Check that the name of the current program is PT, enter 4 as the parameter (n) and tap ▶ . If all goes well, the Status box appears with the message 'Done' and you can tap OK. Tap ☒ to return to the program application.



Folder: main
Name: PT
Parameter: 4



Folder: main
Name: PT
Status
Done

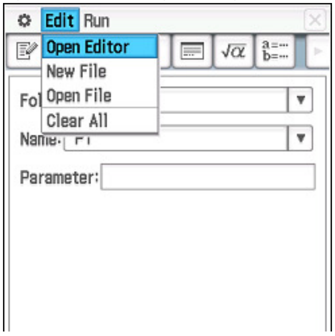
```
Wait...
{1,4,6,4,1}
Sum of coeffs
16
```

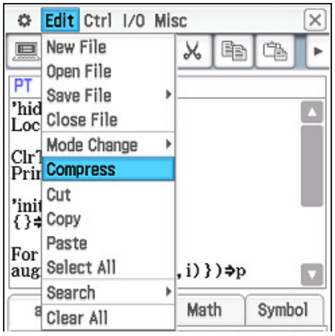
This program PT has no data input validation to prevent errors and the output is very basic.

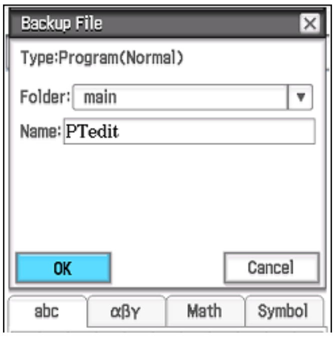
Consult the program chapter in the ClassPad manual for methods to handle more complex text output, including functions to handle strings.

Watch video 985 at
www.classpad.com.au

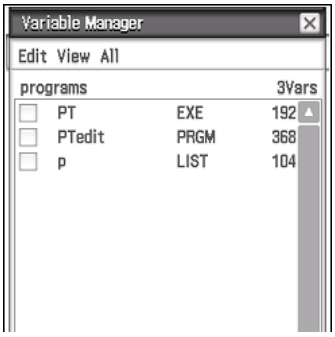
Tap Edit, Open Editor to edit your program. Choose Compress to reduce file size and prevent further editing (you will be prompted to save an editable backup).







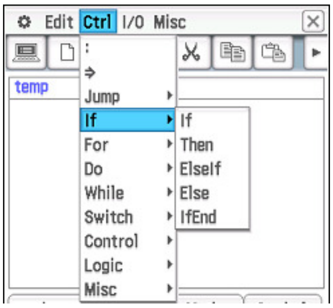
Type:Program(Normal)
Folder: main
Name: PTedit

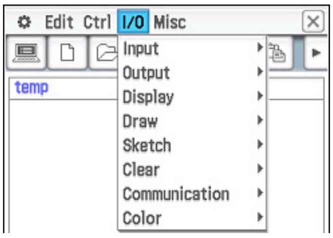


programs		3Vars
□ PT	EXE	192
□ PTedit	PRGM	368
□ p	LIST	104

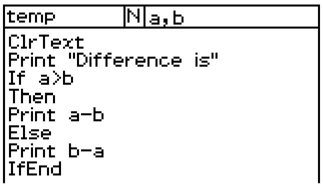
In the variable manager, within the main folder, you should find PT, PTedit and p amongst the variables. Delete, move and rename your program from here.

Additional programming tips - Ctrl and I/O Menus





If ... Then ... Else ... IfEnd
Use for branching and decision making.



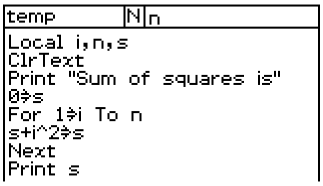
```
temp    N a,b
ClrText
Print "Difference is"
If a>b
Then
Print a-b
Else
Print b-a
IfEnd
```

Program takes two values and prints out the difference.

Else is optional.

Can also include ElseIf statement.

For ... Next
Use for repeating fixed number of times

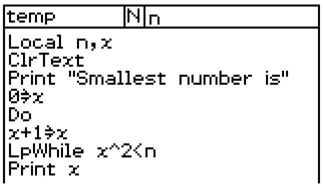


```
temp    N n
Local i,n,s
ClrText
Print "Sum of squares is"
0⇒s
For 1⇒i To n
s+i^2⇒s
Next
Print s
```

Program sums the squares of all integers from 1 to n.

Can add 'Step *k*' to increment *i* in steps of *k*.

Do ... LpWhile
Use for looping until a condition is met.

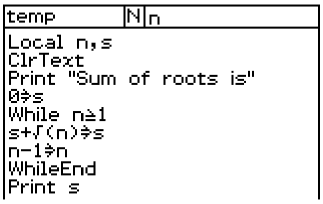


```
temp    N n
Local n,x
ClrText
Print "Smallest number is"
0⇒x
Do
x+1⇒x
LpWhile x^2<n
Print x
```

Programs finds the smallest number which has a square greater than n.

Beware of writing an endless loop! Press the Clear button to break out of such a program.

While ... WhileEnd
Use for looping, but checks for condition at start of loop.

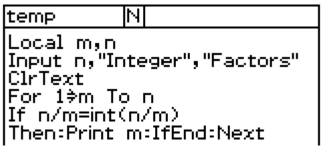


```
temp    N n
Local n,s
ClrText
Print "Sum of roots is"
0⇒s
While n≥1
s+√(n)⇒s
n-1⇒n
WhileEnd
Print s
```

Program sums the roots of integers from 1 to n.

Again, beware of endless loops.

Input
Use for obtaining a single number input from user.

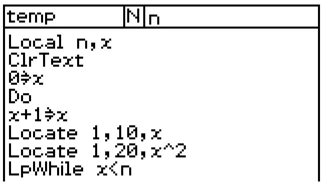


```
temp    N
Local m,n
Input n,"Integer","Factors"
ClrText
For 1⇒m To n
If n/m=int(n/m)
Then:Print m:IfEnd:Next
```

Program displays all factors of any integer.

Note use of colons to enter multiple commands on a single line.

Locate
Use for writing to specific location (x,y) on the screen.

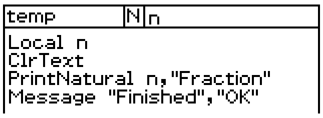


```
temp    N n
Local n,x
ClrText
0⇒x
Do
x+1⇒x
Locate 1,10,x
Locate 1,20,x^2
LpWhile x<n
```

This program prints the squares of numbers from 1 to n at the location (1, 20) on the screen.

x: 1 to 290
y: 1 to 290
Upper left corner is (1,1)

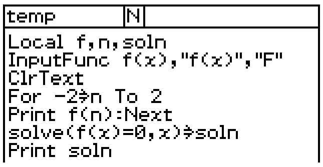
PrintNatural and **Message**
Pause and display result.



```
temp    N n
Local n
ClrText
PrintNatural n,"Fraction"
Message "Finished","OK"
```

In Main, check the setup is Standard rather than Decimal. Run the program by entering a decimal number. The program displays the decimal as a fraction and then the message "Finished".

InputFunc
Use for obtaining a function from the user.



```
temp    N
Local f,n,soln
InputFunc f(x),"f(x)","F"
ClrText
For -2⇒n To 2
Print f(n):Next
solve(f(x)=0,x)⇒soln
Print soln
```

Program displays value of function for x=-2 to 2 and then finds roots.