Open the Program application.

We'll create a short program to list the values of $^{n}C_{0}$ to $^{n}C_{n}$ and call it **nc**.

Tap **Edit**, **New File**.

Enter a short name for the program, eg **nc** and tap **OK**.

We'll run the program in Main by typing **nc(**n**)** .

Enter n as a parameter in the box shown, so that Classpad expects a single parameter and assigns it to the variable n.

Precede a comment by an apostrophe as shown in the first line.

Blank lines are ignored.

The second line tells Classpad to keep the variables p, i and n 'local' - they are only visible to the program.

The third line clears the text window that we can use to display results.

The fourth line of code creates an empty list which is stored as the variable p.

The next three lines of code use a For... Next loop to augment the values nCr(0)... nCr(n) to the list p.
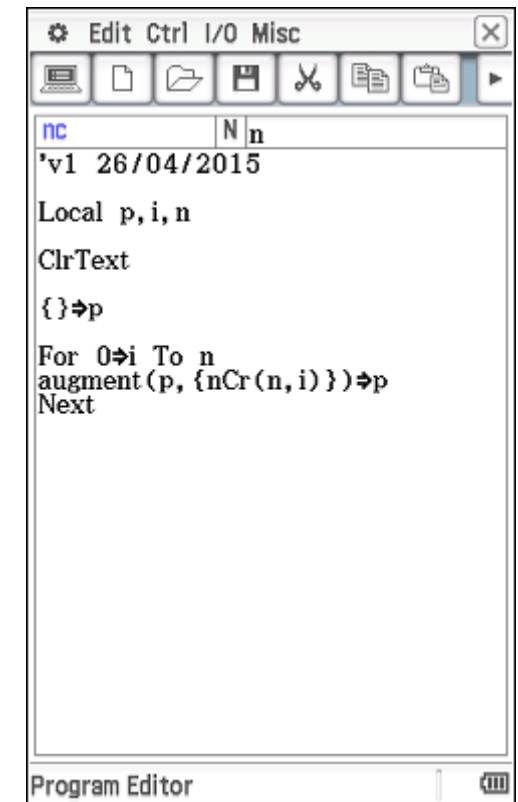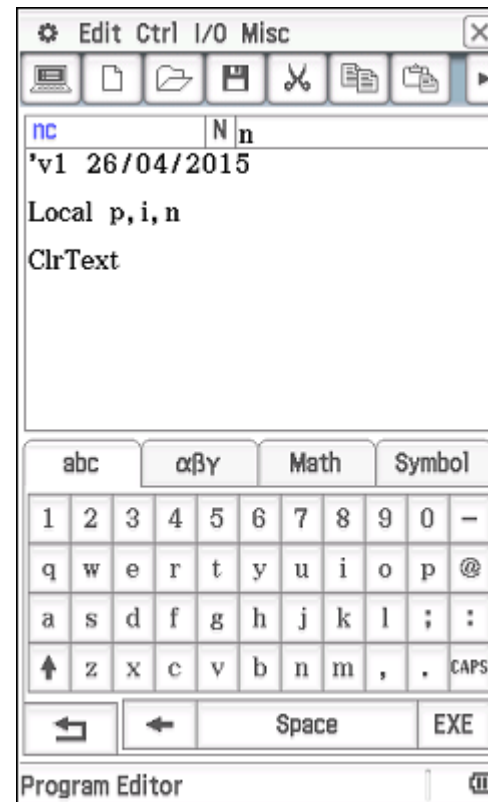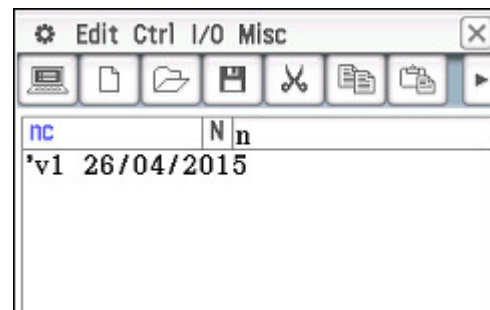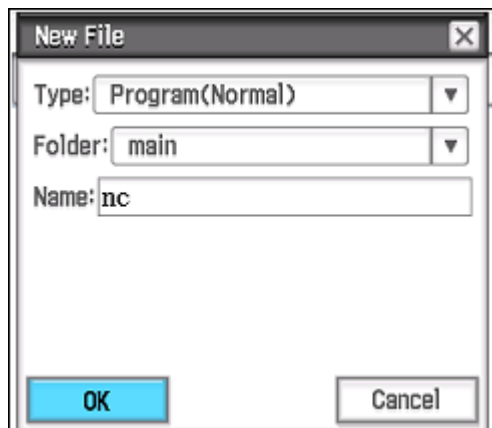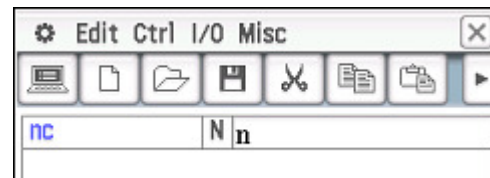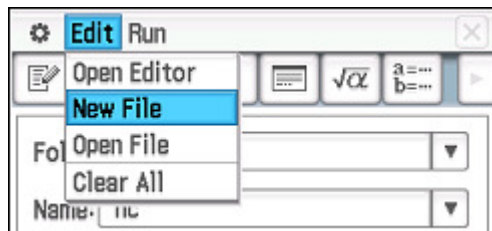
The three Print statements write a text message and then the values of *n* and *p* to the text window.

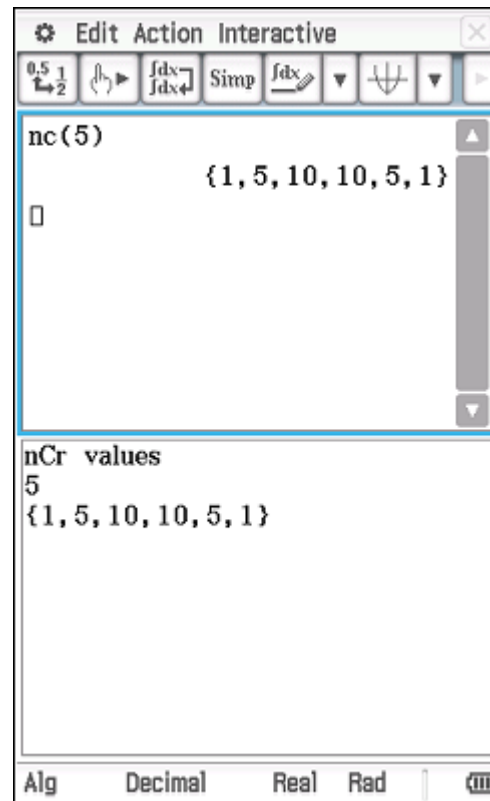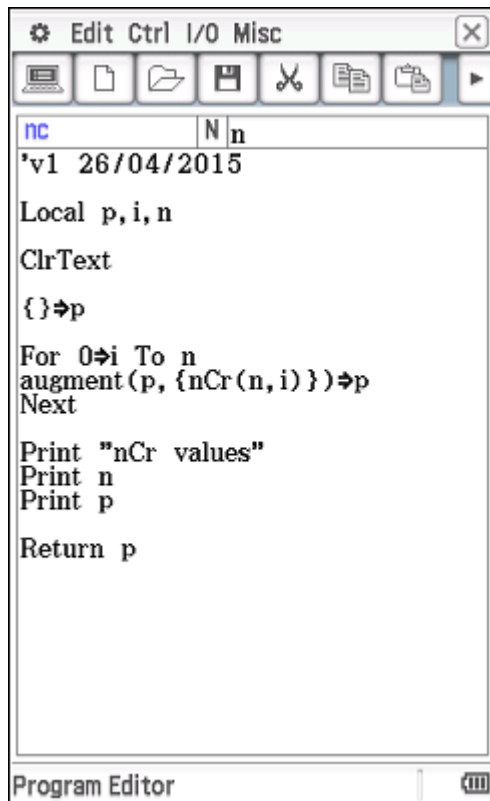The last line returns list *p* to the Main screen.

Tap **Edit**, **Save File**.

Open the **Main** app to test our program.

Enter **nc(***5***)** and tap **EXE**.

The six coefficients are returned in a list in the Main (top) window and the three print statements appear in the lower text box.

Tap Resize to close the text box.

Return to the program editor and re-open the **nc** program. For distribution, you may want to compress the program, which stops users from changing and possibly corrupting your code.

Tap **Edit**, **Compress**, enter a Backup File name, such as **ncb** and tap OK. Classpad now compresses and saves the original **nc** file, and also saves a copy of the original editable code as **ncb**.

The difference between the two files can be seen in the Variable Manger.

The compressed file, **nc**, takes up 160 bytes of memory and is an EXE type of file.

The copy of the original file, **ncb**, takes up 296 bytes of memory and is a PRGM type of file.