

BasList vs 1.0.2

Lists a BBC BASIC program file in Acorn or R.T. Russell (Z80) format

SYNTAX

```
BasList [/file=]filename ([from line] [to line]) | [line,line]) [Options]
([IF ...] | [IFX ...])
```

```
BasList [/file=]filename [/V] [/addnumbers] [/auto] [/align] [/indent]
[/nonumbers] [/nospaces] [/bare] [/pause] [/prettyprint]
```

```
BasList /? or /h - help
```

`/V` Allow BASIC V keywords. See Compatibility below.

`/addnumbers` Show line numbers (starting at 10, in tens) when none in the program (Z80 only)

`/align` Right-align line numbers

Without `/align` (line numbers displayed with one following space):

```
10 REM An example program
100 MODE 7
1000 DEFPROCmain
```

With `/align` (line numbers ranged right and one following space):

```
10 REM An example program
100 MODE 7
1000 DEFPROCmain
```

`/indent` Indent listing of loops (unless `Nospaces` specified).

The equivalent of LISTO7 in BASIC IV

`/nonumbers` Omits line numbers

`/nospaces` Omits all spaces after line numbers. Also cancels `/indent`.

```
10REM An example program
100MODE 7
1000DEFPROCmain
```

`/bare` Omits additional messages

`/prettyprint` Adds additional spaces and syntax colouring

EXAMPLES

```
BasList Program ,200 - List up to line 200
```

```
BasList Program 1000, - List from line 1000
```

BasList Program 200,1000 - List from line 200 to 1000
 BasList Program 200 1000 - List from line 200 to 1000
 BasList Program IF PRINTTAB - List only lines containing the specified text
 BasList Program IFX - As IF, respecting spaces and case

NOTES

- Options may be specified in any order, may start with / or - and can be abbreviated. Note that **all** matching options apply. For example, /p activates both /pause and /prettyprint.
- Parameters containing spaces must be enclosed by double quotes.
- The options /?, -?, /h and -h, or when BasList is followed by no parameters at all, display the Help text at the start of this article.
- Any IF or IFX clause must be at the end. Multiple matches may be entered and BasList will list any line containing at least one of them.
- If you are producing a listing to be *EXEC'd into a BBC Micro, use /bare /nospaces to ensure a clean listing.

FURTHER INFORMATION

1. Compatibility

BasList is compatible with a BBC BASIC program using Basic I to Basic IV (and the version of BASIC issued with the Master Compact, sometimes known as BASIC 6) as used in Acorn computers.

By using the /V switch, it will also recognise the extended keywords of BASIC V as issued with RiscOS.

For example, a program that uses the expression COLOUR 127 TINT 255 will list as such with /V, but without lists as COLOUR 127 LOADCOUNT 255.

It is also compatible with Z80 BASIC format or any BASIC version by R.T. Russell. The line structure is slightly different from Acorn's for better performance on Zilog Z80 and Intel 8086+ processors, such as BASIC for CP/M machines, Amstrad Colour Computers (CPC464, CPC664 and CPC6128), Tatung Einstein and Cambridge Z88, as well as BBC BASIC (86) for IBM compatibles. Later versions such as BBC BASIC for Windows also have a few tokens that differ from BASIC V's.

BasList determines the format automatically.

2. Filenames

If the filename includes spaces, it must be enclosed by double quote characters ("), e.g.

```
BasList "C:\Users\My Documents\Basic Programs\Example"
```

If the filename is numeric it will be mistaken for a line number unless preceded by /file=

3. Line Numbers

By default, BasList displays the entire program.

As with the BBC Micro, you may limit the lines listed using:

- `number,` - e.g. `100,` – specifies the ‘from line’ or lowest line number displayed
- `, number` – e.g. `,1000` – specifies the ‘to line’ or highest line number displayed

Note that the comma must be last or first character with no spaces (unless also enclosed in double quotes) e.g.

```
BasList Program 100, ,200
```

- A range – `100,200` – specifies range of lines displayed from lowest to highest. May not contain spaces, whether quoted or not.
- A range e.g. `BasList 100 200`. In this case, the first number is the ‘from line’, the second the ‘to line’ number.

If you inadvertently specify the numbers in reverse, so that the ‘to line’ is less than the ‘from line’, BasList will swap them rather than display no lines.

Theoretically, you can combine the three methods of specifying line numbers, e.g.

```
BasList Program 100, 200
```

```
BasList Program 100 ,200
```

However, more complex combinations risk confusion!

4. Options

Also called ‘Switches’, each option may be preceded by a slash (/) or hyphen. There must be a space before each switch. The following are equivalent:

```
/align
```

```
-align
```

They may be in upper, lower or mixed case.

The order in which they appear is not significant.

Options may be shortened as far as possible. They are evaluated in the order listed above.

Thus `/nospaces` cannot be shortened more than `/nos` if it is to be distinguished from `/nonumbers`. (`/n` activates both options, cancelling indentation.)

A non-existent option (preceded by / or -) will result in a message (unless preceded on the line by /bare) but will otherwise be ignored.

5. IF, IFX

Approximately equivalent to LIST IF in BASIC IV and above.

IF performs a case-insensitive match that is not restricted to whole words.

IFX (IF eXact) performs case-sensitive comparison and respects spaces.

You may use *either* IF *or* IFX but not both.

The switches are not case sensitive, i.e./ IF, /if, /If and /iF are equivalent.

IF/ IFX is followed by a list of conditions. For example,

```
BasList Program IF print colour red
```

will cause any line containing 'print' *or* 'colour' *or* 'red' to be listed.

IF

The comparison is not case sensitive. It also ignores spaces unless they are between quotes or in a REM statement.

```
IF PRINTTAB will show lines containing both PRINTTAB and PRINT TAB.
```

```
IF printer will show a line containing PRINT ERR, because PRINTERR contains 'printer' when case is ignored.
```

If you are attempting to match a literal space, enclose the search string in double quotes

```
IF "PRINT ERROR" will find that literal text where it appears in quotes or REMs in the program, but
```

```
IF "PRINT ERR" will only match PRINT ERR and not PRINTERR.
```

Detail: If an IF clause is specified, BasList makes an internal copy of each program line, removing all spaces *except* those in quoted strings or in REM statements. Each match string is checked against the copy to see if it is contained in the line using case-insensitive comparison. It lists the line if any of them match. This has the advantage that it will, for example, find both PRINT TAB(and PRINTTAB(, but has the disadvantage that there is no way to limit the match to whole words.

IFX

Case-sensitive comparisons are made against the *original* program line, respecting spaces.

```
BasList program IFX print
```

will list lines with 'printer', 'PROCrubandprint' but not PRINT.

Note: Using PrettyPrint with IFX may result in unexpected results, as colour changes are embedded in the line. This is not an issue with IF.

You cannot use keyword abbreviations. On a Master 128, LISTIF P. will find 'PRINT'; BasList works purely at the textual level.

Also consider saving the listing (see below), opening in a text editor and using Find.

6. Using the listings

To display the listing in 'paged mode', use the /pause option. The listing will stop at the bottom of the screen and wait for a keypress. Pressing Space will advance to the next screenful. Return will advance one line. Escape breaks off the listing. (Other than Escape, these keypresses have been chosen for compatibility with MORE, which has the disadvantage of suppressing PrettyPrint colours.)

To save the text to a file, add `>filename` to the end. E.g.

```
BasList program >file.txt
```

This will save the listing to a file named `file.txt` in the current directory, but you will not see it on screen. Enclose in double quotes if the file specification includes spaces.

To append several listings, use `>>filename`. Example:

```
BasList program 10,100 >file.txt
```

```
BasList program 1000,1200 >>file.txt
```

```
BasList program 3000,3100 >>file.txt
```

Note that the **first** command has **one** chevron, in order to create the text file, and subsequent ones have a **double** chevron to append to it.

Saving a text file in order to *EXEC into a BBC Micro or copy and paste into BBC BASIC for Windows (for example) is a good way to convert between the Acorn and R.T. Russell formats.

Version Information

1.0.0 First release

1.0.1 Bugs fixed:

- PrettyPrint added space after 'TO in the keyword TOP
- Now respects increased line number range for BASIC V

1.0.2 Enhancement:

- Removed the limitation that the filename must be the first parameter. It may now appear anywhere in the list of parameters (though IF or IFX must be last). The first parameter not preceded by / or – and not a line number will be treated as a file.
- Added /file= or /f= option for cases when filename is numeric and would otherwise be seen as a line number.

© Andrew Rowland 2022-2023