



## CET Utilities: Bcopy, Bcreate, Blist, Bpretty, Brenum W32 App Builder Version

Major enhancements have been made to all of the CET BASIC utility programs. A new Bcopy program has also been added to provide you with much more functionality than is available with the COPY command that comes with the operating system.

The utilities are available as both 16 and 32-bit executables. They have been uploaded as UTIL16.ZIP to NETDOS or as UTIL32.ZIP to the WIN32 library on the BBS in Colorado.

Both versions are identical in name, syntax and operation. If you have copies of the CET BASIC product for DOS and Windows on the same system, please adjust your PATH so that the desired directory is specified first to avoid possible problems with trying to use the incorrect version.

On-line help is available for all the utilities by simply entering the command without any arguments. This information is stored in **butil.msg** in the same directory with the executables. The message file is provided in an ASCII text format so that it may be easily translated.

### The Bcopy Command

This utility may be used to copy or merge all of the records or a specific group of records from one file to another file. The destination file may be of the same or a different type.

The command line syntax is as follows:

**Bcopy [-f *fromkey*] [-t *tokey*] [-c *counter*] [-r] [-m] [-s] [-d *reclen*] [-i *reclen keylen*]<from><to>**

where:

#### ***from***

Specifies the name of the source file in either an MS DOS or THEOS format. For example, **cp\data\custname** and **cp.data.custname** are considered the same.

#### ***to***

Specifies the name of the target file in either an MS DOS or THEOS format.

#### **-f *fromkey***

Copies starting from the record with the *tokey*, a key in an indexed file or a record number in a direct file. The default is to start at the beginning.

#### **-t *tokey***

Copies the *from* file up to (but not including) the record specified by the *tokey*. The default is to stop at the end of the file.

#### **-c *counter***

Specifies the number of records to copy.

#### **-r**

Replaces the *to* file, if one exists. The default is to query the user if the existing file should be replaced.

#### **-m**

Merges records from the source *from* file into the destination *to* file. When this option is used and the *to* file does not exist, the program asks the user to specify which action to take "Create/Quit (C/Q)" and press Enter to continue.

Since this option will not change the parameters of an existing file, it is possible to copy records to a file with a different key and/or record length.

**-s**

Creates a sequential output file.

**-d *reclen***

Creates a direct output file.

**-i *reclen keylen***

Creates an indexed output file.

Operation Notes:

1. When the *-f (fromkey)* and the *-t (tokey)* or *-c (counter)* options are used in the same command line, copying starts from the record specified with *fromkey* and ends when the counter value is exceeded or when the record specified with *tokey* is reached, whichever comes first.
2. When the *-r (replace)* or *-m (merge)* options are not specified in the command line and the *to* file already exists, the program asks the user to specify which action to take:  
Replace/Merge/Quit (R/M/Q):  
Press desired key followed with Enter.
3. When both *-m* and *-r* options are defined, *-m* will be ignored.
4. When the destination file does not exist and none of the *-s*, *-d* or *-i* options are defined, the file is created with same type, record length and key length as the source file.  
When more than one option is used to specify the *from* file type, the last option specified in the command line will be used.
5. When the *-m (merge)* option is used and the key or record length in the *from* file differs from that in the *to* file, Bcopy will use the specifications in the *to* file and truncate or expand the length(s) accordingly.

## The Bcreate Command

The Bcreate utility may be used to create indexed and direct files or to clear the contents of an existing file. If you plan on creating files from within your application, we recommend that you use the Bcreate function to avoid having to exit the BASIC environment to perform the operation.

Note that this utility is an enhanced version of the CREATE command. If you like, you may rename this command to CREATE for compatibility purposes.

Bcreate recognizes a variety of command options. The minimum abbreviations have been underlined. Note that preceding the options with a left parenthesis is optional.

The syntax is:

**Bcreate <filename> [ ( ] [options]**

where:

***filename***

Specifies the name of the file in either an MS DOS or THEOS format. For example, **cp\data\custname** and **cp.data.custname** are considered the same.

**Nodesize value**

Specifies the node size to use when creating an indexed file. The default is 512 which is the node size used in the DOS product. Although W/32 programs can read files with the default node size, we recommend that you use "n 1024" to be consistent with files created with the W/32 Bcreate function.

**indexed**

Creates an indexed file which is actually two files: a **.idx** file for the record's keys and a **.dat** file for the data.

**direct**

Creates a direct (relative access) file.

### **reclen *n***

Allocates a record length of *n*.

### **keylen *n***

Allocates a key length of *n*. This option is ignored for direct files.

### **clear**

Clears the contents of an existing file. The key and record length will remain the same unless any of the *keylen*, *reclen* and/or *nodesize* options are specified in the command line.

Note that the file must be closed in order for this option to function successfully.

All CET BASIC files are dynamically allocated so any THEOS *filesize* option will be ignored. If the command is used to create a THEOS keyed file, the program will create an indexed file instead.

The following examples will create the indexed files **data\custname.idx** and **data\custname.dat** with a key length of 11 and a record length of 350:

```
Bcreate data\custname indexed reclen 350 keylen 11
```

```
Bcreate custname.data indexed reclen 350 keylen 11
```

## **The Blist Command**

Blist may be used to display the contents of indexed, direct or sequential files. The contents of indexed files will be displayed in alphabetical order by key. Direct files will be listed in ascending numerical order. A banner will be displayed to identify the name and type of file being listed.

The syntax of the command is as follows. Entering the command without any arguments will display a help screen.

```
Blist [-h] [-f fromkey] [-t tokey] [-c counter] [-l length] [-p page] [-m margin]  
[field-numbers] [-n] [-u] <filename>
```

where:

### ***filename***

Specifies the name of the file in a DOS or THEOS format. For example, either of the following names may be used to list **cp\data\custname**:

```
.\cp\data\custname (or cp\data\custname) cp.data.custname
```

### **-h**

Displays the contents of the file in a hex dump format.

### **-f *fromkey***

Lists starting from the record with the *fromkey*, a key in an indexed file or a record number in a direct file. The default is to start at the beginning.

### **-t *tokey***

Lists the file up to (but not including) the record specified by the *tokey*. The default is to stop at the end of the file.

### **-c *counter***

Specifies the number of records to list.

### **-l *length***

Specifies the maximum length of the print line. Characters that do not fit on the line will wrap to the next line. The default line length is 79.

### **-p *page***

Specifies the output page length. The default is 24 rows. Use "-p 0" to turn off pagination.

### **-m *margin***

Specifies the number of characters to indent the second and all subsequent lines of a multi-line record. The default is 5.

### **field-numbers**

Specifies the numbers of the fields to be displayed from the formatted file. The default is to display all the fields in the record.

The first data field starts with 1. The keys in an indexed file and the record numbers in a direct file are always displayed.

#### **-n**

Turns on field *number* mode and displays the field number, field type and length of field before the contents of each data field. The field type will be either S for a string, F for a float or I for an integer. For example:

123-45-6789: (1S17)D.G.Widgets, Inc.,(2S11)Don Gillett,(3S17)...

#### **-u**

Specifies that the files are in UX-BASIC format.

## **The Bpretty Command**

The Bpretty utility may be used to rearrange the spacing and indentation of CET BASIC source files to reflect the program structure.

If you have used this utility before, please note that there is a difference in syntax and operation. In the current version, Bpretty no longer accepts input from the keyboard. The name of the source file **must** be entered on the command line. The specified file will be rewritten in place after storing the original code in a file with a **.bak** extension. By default, the backup file will have the same name as the source file. (File names must be in a DOS format.)

The syntax is as follows. Note that the Bpretty options must be entered before the file names, otherwise the option will be interpreted as the `<source.bak>` and no backup file will be created.

**Bpretty [-t] [-i] <source.b> [ < source.bak> ]**

where:

#### **-t**

Replaces spaces with TAB characters wherever possible.

#### **-i**

Inserts text from #include files before performing any other operation.

Note that the current version does **not** support the -n option which had allowed you to change the indentation from the default four spaces. Please let us know if you would like this feature to be re-implemented.

## **The Brenum Command**

Brenum may be used to renumber or unnumber BASIC source programs with up to 999,999 lines. Filenames must be in a DOS format.

If you have used this utility before, please note that there is a difference in syntax and operation. Brenum is like Bpretty in that it no longer accepts input from the keyboard. The name of the source file **must** be entered on the command line. The specified file will be rewritten in place after storing the original code in a file with a **.bak** extension. By default, the backup file will have the same name as the source file.

The syntax is:

**Brenum [-a] [-n *initial*] [-i | + *increment*] [-b *begin*] [-e *end*] [-u] <source.b> [< source.bak>]**

where:

#### **-a**

Causes all lines within the given range to be renumbered. When omitted, the default is to renumber only the lines with numbers.

Note that the lines of code contained in any "#include" files will not be renumbered, even when the -a option is used.

**-n initial**

Starts renumbering with the number *initial*. The default is 10.

**-i increment**

Increments the new line numbers by *increment*. The default is 10.

**+ increment**

Works like the -i option to specify how the numbers are to be incremented.

**-b begin**

Begins renumbering at the *begin* line. When omitted, the default is to begin with the first line of the program.

**-e end**

Stops renumbering at the *end* line. When omitted, the default is to end with the last line of the program.

**-u**

Removes all line numbers from the *input* file.

Note that renumbering portions of a file may cause those lines to be moved to preserve line number order.

The flags may be arranged in any order provided that the parameters associated with the flags are entered in corresponding order. For example, the following command will renumber all lines in **source.b** after storing the original contents in a file with the default name **source.bak**.

```
Brenum -a source.b source.bak
```

The following examples will all produce the same results: **source.b** will be renumbered from line 200 to line 300. The new line numbers will begin with 600 and increase in increments of 5. The file **xsource.bak** will contain the original source code.

```
Brenum source.b xsource.bak -nibe 600 5 200 300  
Brenum source.b -b 200 -e 300 -n 600 -i 5 xsource.bak  
Brenum -n600 +5 -be 200 300 source.b xsource.bak
```

## The Dcheck Command

The W/32 product currently uses D-ISAM, an indexed file manager from Byte Designs. D-ISAM files are compatible with C-ISAM, the file structure from Informix which is used in the other CET BASIC products.

Dcheck is a D-ISAM utility designed to check and fix indexed file inconsistencies. (Bcheck is available under DOS to provide the same capabilities with the C-ISAM files.) This information is provided to supplement the information provided in earlier editions of the W/32 Application Builder User's Guide.

When using this utility, please note that

- Dcheck only accepts DOS filenames.
- Dcheck may be used to view the file's structure, list the contents of the keys and the data file order and check file consistency without actually modifying the file.
- Dcheck is capable of building a full index from a skeleton and a data file. Deleted records are not removed so the size of the file remains the same.

The command line syntax is as follows. Note that the first parameter (optional) is a dash; followed by one or more of the listed letters. The second parameter is the full path name of the file to be checked. Enter the command without any arguments for on-line help.

```
Dcheck [-] [i] [I] [n] [y] [q] [ b] [h] [x] [o] <isamfile>
```

where:

- i**  
Checks the index only. No data field checking is performed.
- l**  
Lists the contents of the indexes to the screen.
- n**  
Does not try to rebuild the index.
- y**  
Tries to rebuild the index if it is imperfect.
- q**  
Runs in quiet mode without displaying anything on the screen unnecessarily.
- b**  
Rebuilds the index from the data, even if the index is OK. To build an index for data from another source:
  1. Organize the data into a file with each record separated by a newline character (ASCII 10).
  2. Create an indexed file with a different name, with the desired key. You need not add the data.
  3. Move your data file to the **.dat** file for the newly created indexed file with the command:  
    dcheck -b isamfile.This will cause your data file to be fully indexed as described when you built the ISAM file.
- h**  
Only prints out the file structure, with no checking. This option does not require exclusive access to the file.
- x**  
Outputs the contents of each record as hexadecimal.
- o**  
Prints a list of numbers which refer to the record number in the order that they are indexed. This feature may be used with the -q option to create a filter for quick printing.  
If an error is encountered, or if the -b option is used, Dcheck disposes of the indexes, then rebuilds them from the data. The data is never destroyed in the repair process.