

DISK INDEXING PROGRAM VERSION 2.0 FOR THE TRS-80 MODEL I AND MODEL III
Mumford Micro Systems - Box 400 Summerland, California 93067
Copyright (C) 1981 by Bryan Mumford - All rights reserved

INTRODUCTION

The purpose of this program is to organize, alphabetize, categorize, and computerize your program library. If you have very many disks you are probably well aware of the problem of remembering which disk contains which version of which program. This program will simplify that task by allowing you to create a master index of all your program titles. This index may be searched, sorted, printed on a line printer, and saved on disk for future reference and update. Programs may be added manually, from previously stored data files, or by reading disk directories directly. A 48K machine will hold up to 1500 programs. Alphabetizing is accomplished with a high-speed machine language Shell/Metzner sort, which will alphabetize 1,000 programs in less than 10 seconds. The program supports formatted output to the line printer in several formats including labels and lists. In addition, a purge command is included to facilitate rapid removal of obsolete programs from your disks. It is known to run under TRSDOS, NEWDOS+, NEWDOS/80 versions 1.0 and 2.0, VTOS, and DOSPLUS.

LOADING

There are two parts to this program. The bulk of the program is in Basic. The other part is the machine code module for the alphabetizing and disk read routines. Furthermore, there are two versions of the machine code module, one for 32K machines and one for 48K. If you received this program on disk you will find them called INDEX/BAS, INDEX32/CMD, and INDEX48/CMD. If you received this program on tape you will need to transfer these modules to disk before you can use them. To do this power-up your computer and enter Disk Basic. Type CMD"T" to disable interrupts before loading the tape. The first program on the tape is the Basic module. Load it with the CLOAD command. After it has loaded, LIST it to be sure it looks valid and SAVE it on disk as INDEX/BAS. Skip over the next program on the tape which is the same as the first (in case there is an error in the first one). The 3rd program on the tape is the machine language module for 32K systems. On the Model III you may load this program and the next one onto disk with the TAPE utility which came with TRSDOS. On the Model I use the following procedure. Load it with the SYSTEM command and the file name INDEX3. After it has loaded successfully DO NOT hit the ENTER key. Let the tape run past the 4th program (which is the same as the 3rd in case of errors). Load the 5th module by typing INDEX4 in response to the SYSTEM prompt and hitting ENTER. This is the machine code module for 48K systems, and there are two copies of it also. After this has loaded successfully and the SYSTEM prompt returns hit the BREAK key. Return to DOS with CMD"S". Now save the machine language modules (which have been stored in RAM) onto disk with the DUMP command. The syntax for this command varies with operating systems, but the RAM addresses are the same. They are as follows:

SYSTEM SIZE	MODULE NAME	START	END	ENTRY
-----	-----	-----	-----	-----
32K	INDEX32/CMD	BB00H	BFFFH	BFB2H
48K	INDEX48/CMD	FB00H	FFFFH	FFB2H

You are now ready to load and run the program. For demonstration purposes the loading procedure for a 48K machine will be described. Loading the 32K version just requires the INDEX32/CMD module instead of INDEX48/CMD. To begin, load the machine code module from DOS READY by typing INDEX48 and hitting the ENTER key. The module will load and DOS READY will be displayed again. Now enter Basic with 1 disk file buffer (NEWDOS uses 3 unless you tell it not to). The correct memory size will be set automatically by the program. Once the Basic READY prompt has been displayed load and run the Basic program by typing RUN INDEX/BAS. The program will do some initialization and the directory will be displayed.

THE DIRECTORY

There are 12 functions in the program. They are pretty much self-explanatory, but I'll describe the use of each one. The most important functions will be described in order of use rather than in strict numerical sequence. The functions in the directory are grouped by type rather than order of use. This may sound confusing but it will make more sense shortly. Also, the program remembers the last function requested, so to ask for the same function again (as when reading several directories in a row to build a list) it is only necessary to hit <ENTER>. As a final note, it may sometimes happen that an error (usually disk-related) will break the program. If you type RUN you will loose all variables. To return to the directory at any time without losing data, type GOTO 200.

In addition to the command summary, the directory displays the number of programs in the file and the theoretical maximum number allowed. The maximum is theoretical because it depends on how long your disk names are. The maximum assumes the average length of disk names is 2 characters. This may seem awfully short, but code names such as AA or BD will distinctly identify a very large number of disks (676 to be exact). If you want to use longer disk names that is your choice. The longer they are the more memory they take and the fewer programs your file will hold. A 48K machine will hold about 1300 programs with 5 character disk names. Another point to keep in mind is that as far as the computer is concerned, 10 is less than 2. That is, if you have a disk named A10 and another disk named A2, after alphabetizing the A10 disk will be first because the 1 in A10 has a lower ASCII value in the 2nd position than the 2 in A2 does. To keep numerical names in numerical sequence they must all have the same number of characters. In this case, use A02 and A10 to maintain numerical as well as alphabetical order. Onward...

FUNCTION 1 - READING DIRECTORIES

Since there are no programs in the file to begin with, the first operation will be to read a directory and build a file. To do so, type 1 and hit <ENTER>. You will then be asked for the drive number to get the directory from. Answer with a number from 0 to 3. If you hit <ENTER> without specifying a number the last number specified will be used. You will then be asked if you want the invisible programs to show. Answer with Y or N. If you hit <ENTER> without specifying Y or N your last response will be used. You will then be asked for a name for this disk. You may specify any name you like, but as mentioned earlier short code names are recommended. If you hit <ENTER> without specifying a name you will be asked again. A blank name is not allowed. The specified drive will then start up, the directory will be read, and control will return to the directory. Gee, that was fast....

FUNCTION 5 - ALPHABETIZE BY DISK

The directory will now show that there are more than zero programs in the file. Before we look at them, though, they should be alphabetized. Type 5 and hit <ENTER>. The word "SORTING" will appear for a split second and the directory will return. Gee, that was REALLY fast!

FUNCTION 6 - DISPLAY FILE BY DISK

Now we can look at the programs. Type 6 and hit <ENTER>. You will be told to hit C for continuous hardcopy or L for labels. These functions are for printers if you have one. If you just want to see the file on the video screen, hit <ENTER> without typing either C or L. (If you want to use a "funny" printer that requires a "custom driver" with this program, see the paragraph at the end of this documentation.) The file will then be displayed. The first line contains the disk name you specified. The second line starts with the number of free GRANS available on this disk. The next entry will be the "disk ID" which will be either the operating system or the name assigned during formatting. Following that are the program names in alphabetical order. Hit <ENTER> and you will return to the directory. If there are many disks in the file you may halt the display by hitting any key except D while the display is scrolling. Start the display up again by hitting another key. If you hit D control will immediately return to the directory.

But what if you specify C for continuous printing? You will be asked if you want to pause between pages for single sheets. Answer Y or N. The list will then be sent to the line printer just as you see it on the video screen. As the list approaches the end of a page the program will try to break pages between disks for improved readability. If you asked for pauses you will be prompted on the video screen to insert a new page and hit <ENTER>. Printing may be halted or aborted by hitting any key or the D key as explained in the preceding paragraph.

If you specify L for labels you will again be asked if you want to pause between pages. Then disk labels will be printed two to a page. The format of the label is such that it may be cut out and slipped in the jacket with the disk. It is assumed that your printer is 10 characters/inch and 6 lines/inch. The left edge of the label is the left edge of the printing. The right edge of the label is indicated by a colon to the right of the disk name. After printing the program names the printer will skip several lines and then print a short line. This indicates the bottom of the label which is approximately 5 1/4 inches square. It may be cut out with scissors or a paper cutter and kept with the disk. As described above, printing may be halted or aborted.

FUNCTION 11 - ALPHABETIZE FILE BY PROGRAM NAME

A list of your disks by disk name is useful, but an alphabetized list by program name is REALLY handy. From the directory, type 11 and hit <ENTER>. The word "SORTING" will appear briefly and the directory will again be displayed. The file is now alphabetized by program name.

FUNCTION 12 - DISPLAY FILE BY PROGRAM NAME

Type 12 and hit <ENTER>. You will be told to hit H for hardcopy. We'll get to that in a minute. Hit <ENTER> instead and the list will be displayed in alphabetical order. The first entries will be the amount of free space left on each disk. Following this will be the disk ID's bracketed with equal signs. The equal signs assure that the disk ID's will go to the start of the list. Each ID will reference the disk name you assigned to it. After this will come the program names themselves. While this list is scrolling it may be halted by hitting any key but D. You may start it up again by hitting another key. If you hit D you will be returned to the directory.

If you asked for hardcopy you will be asked if you want to pause after each page. The file will then be printed in three columns on a sheet, 56 lines/page, with a page number, with 10 blank lines between pages. Start printing near the left edge of the paper to be sure it will all fit, and if you didn't take my advice and use short disk names they will be abbreviated here to fit onto a normal page.

The rest of the functions will be described in numerical order.

FUNCTION 2 - LOAD A FILE FROM DISK

The file this is intended for is an index created earlier and stored with function number 8. You will be asked for the file name, and the file in question will then be loaded. You may load several files one after another and they will load end to end, the second one not erasing the first one. To erase a file before loading a new one, hit <BREAK> and type RUN.

FUNCTION 3 - ADD A PROGRAM TO THE FILE

This is for manually adding a program name. You may want to do this rather than deleting a whole disk and re-reading the disk directory. Specify the disk name, and when asked, the program name. Twelve periods will be printed on the screen to indicate the 12 character positions allowed for the program name, which will be added to the end of the list. To merge it into alphabetical sequence execute the appropriate alphabetizing function. If you want to keep a short description of each disk (other than the disk name) with that disk, you may add such a description here, preceded by an ASCII character lower than a number (such as a space, a dash, or an asterisk). These characters will assure that the description goes to the beginning of any list and also remind you that it is not a program name. Like program names, this description may only be 12 characters long. For example, if your disk names are numerical (3000, 3001, 3002, etc.), you might want to describe each disk as "GAMES", "BUSINESS", "BACKUP:3001", etc. In these examples, the descriptions would be added as if they were programs named "* GAMES", "-BUSINESS", or " BACKUP:3001".

FUNCTION 4 - SEARCH FILE FOR A DISK

You will be asked for a disk name. You must type this name exactly as it is stored in the file. You will then be told to type L if you want a label. The label generated here will be printed in the same format as was described earlier for function number 6. If you hit <ENTER> without typing L no label will be printed. The file will then be searched for the disk name. If the disk cannot be found you will be told so. If the disk is found all programs on that disk will be shown IF the list has been alphabetized by disk name (function number 5). This is because the routine assumes it has found all programs on a given disk as soon as it finds a program on another disk. The alternative is to search the entire file which would take longer than alphabetizing it appropriately beforehand.

FUNCTION 7 - PURGE DIRECTORIES

After you get your disk library in order you may find a lot of duplication. This function will facilitate removing unwanted files from disks. You will first be asked which drive number to look at. Specify a number between 0 and 3. The directory in this drive will then be read and the drive number, free GRANS, and contents will be displayed. Each program will be numbered and you will be asked for the number of the program you want to delete. Enter a number and that program will be removed from the disk. NOTICE THIS: This does not mean it will be deleted from some data list. It will actually be KILLED on the disk and be gone. Enter X instead of a number to exit this function and return to the directory.

FUNCTION 8 - SAVE A FILE ON DISK

This function will store the resident file on disk for future use. It may be read back in latter with function number 2.

FUNCTION 9 - DELETE A PROGRAM FROM THE FILE

You will first be asked for the disk on which the program is stored. If no such disk is in the file you will be told so. If the disk is found you will then be asked for the program name, which must be specified exactly as it appears in the file. You may delete a whole disk by entering "/" instead of a program name. If the program is found it will be deleted. If it is not found you will be told so. If you have asked for the whole disk to be removed, each program that is deleted from the file will be displayed. The entire list will be searched to accomplish this, after which the list will be alphabetized by disk as in function number 5.

FUNCTION 10 - SEARCH FILE FOR A PROGRAM

This function actually does a little more than it sounds like. It will first ask you for the characters you want to find rather than a program name. This is because it will search for any group of characters rather than a specific program name. If, for example, you specified "/BAS" as the characters in question, you would be shown all the programs in the file with the "/BAS" extension. All the command files (/CMD) may be found in a similar manner. If you have a series of programs called something like JUNK1, JUNK2/SRC, JUNKA, and JUNK99/DAT, they may all be found by specifying "JUNK" as the characters to search for. Hardcopy may be requested and the list may be halted or aborted as in other functions. If the characters cannot be found you will be told so.

USING "FUNNY" PRINTERS

If you have a serial printer, or any printer than needs a special driver in upper memory to run it you will have to generate a version of this driver that loads below and does not conflict with the memory size that INDEX uses. This program automatically sets memory size in line 130 for 32K systems and in line 140 for 48K systems. These lines will also contain a remark to indicate what that value is. To set a memory size manually, specify the value you want to use when you enter Basic. Then before you RUN the program, delete the two POKE instructions in line 130 or 140. You should use a memory size small enough to protect your driver. As an alternate method, run the program once in the normal way to create a data file. Then run the program again with your own memory size and printer driver, but do not use functions 1, 5, 7, 9, or 11 (which will crash without their machine code).