

# DOS MENU MANAGER

Version 1.4

User Manual

Copyright (C) 1992, Geoff Friesen B.SC.  
All rights reserved.

## **Table of Contents**

Introduction	
What MNU Is.....	3
MNU v1.4 Upgrade Notes.....	3
The Shareware Concept.....	3
Installation	
System Requirements.....	4
Files List.....	4
Floppy Disk Installation.....	4
Fixed Disk Installation.....	4
The COMSPEC Variable.....	5
MNU.COM Versus MNU.EXE.....	5
Main Screen	
Title.....	6
Listbox.....	6
Context-Sensitive Help.....	6
Terminating.....	6
Entering A Program	
Program Components.....	7
Using a Data-Entry Form.....	7
An Example.....	8
Running A Program	
Saving Changes Prior to Running.....	9
How MNU Runs a Program.....	9
Program Maintenance	
Editting.....	10
Removing.....	10
Configuration	
Screen Blanking.....	11
Title.....	11
VGA 50-line Support.....	11
Screen Colors.....	11
Extras	
Alphabetizing.....	12
Clock/Calendar.....	12

Appendices	
Problems.....	13
MNU.COM and MNU.EXE Messages.....	14

## **Introduction**

Menu systems shield new users from the intricacies of the **DOS** command-line. Well-designed systems can even benefit veteran users. **MNU** is my version of a menu system which I call the **DOS MeNU Manager**. Version **1.4** is the latest version. It offers the following features.

***DBASE III+ compatible database with room for 400 program records***  
***context-sensitive help***  
***support for VGA 50-line display***  
***screen blanking***  
***support for color and monochrome monitors***  
***less than 2.5 kilobytes of memory taken away from running programs***  
***clock/calendar***  
***configurable title***  
***alphabetizing***  
***ability to run batch files***  
***intelligent listbox (remembers the most recent insert/delete position)***

Version **1.4** does away with the report generation facility found in version **1.3**. The help system has been improved. The TAB and SHIFT TAB keys can be used to switch the focus from one field to another in data entry forms. Changes to screen colors, screen size, and the title from the **configuration** data entry form are instantly reflected on the screen.

**MNU** is a shareware program. If you like this program and intend to use it then I encourage you to register your copy. Registered users will receive a professional laser printed copy of this manual. The registration fee is only **25** dollars. Payment may be sent to:

**Geoff Friesen  
Box 94  
Dauphin, MB  
CANADA R7N 2T9**

### **Installation**

**MNU** requires **MS-DOS** or **PC-DOS** version **3.30** or higher. A **VGA** adaptor is desirable but not necessary. All relevant files have been archived into **MNU.ZIP**.

***MNU.COM***  
***MNU.EXE***  
***MNU.WRI***

**MNU.COM** is a small machine-language program that runs **MNU.EXE** and programs selected by **MNU.EXE**. **MNU.EXE** manages the database and allows users to select programs. **MNU.WRI** contains documentation in the **Microsoft Windows 3.1 Write** format and organized for the **Hewlett-Packard Laserjet III** printer.

**MNU** can be installed on either floppy-based or fixed disk-based systems. You have less room to work with on a floppy-based system but planning can help make the most of your space limitations.

#### ***FLOPPY DISK INSTALLATION***

This installation example assumes two floppy drives, a bootable **DOS** disk (with the **FORMAT** program) in drive A:, an unformatted disk in drive B:, another disk containing **MNU.ZIP** and **PKUNZIP.EXE**, and drive A: is current.

1. Format the disk in drive B: (**FORMAT B: /S**)
2. Remove the disk from drive A: and insert the MNU disk
3. Unarchive MNU (**PKUNZIP -X MNU**)
4. Copy MNU to B: (**COPY MNU.\* B:**)
5. Erase MNU.ZIP from B: (**ERASE B:MNU.ZIP**)
6. Create an AUTOEXEC.BAT file on B:

```
COPY CON B:AUTOEXEC.BAT  
MNU  
^Z
```

7. Insert MNU disk into A: and reboot

### ***FIXED DISK INSTALLATION***

This installation assumes that MNU.ZIP is located on a floppy disk in drive A:, drive C: (the fixed disk) and the root directory (\) are current, and that PKUNZIP.EXE can be located via the **PATH** variable.

1. Make a subdirectory called MNU below the root (**MD MNU**)
2. Change to MNU (**CD MNU**)
3. Unarchive MNU.ZIP (**PKUNZIP -X A:MNU**)
4. Return to the root directory (**CD \**)
5. Insert the following lines into AUTOEXEC.BAT

```
CD \MNU  
MNU
```

6. Remove disk from A: and reboot

The **SET** command is used to display the contents of the environment (a list of strings of the form **variable=value**). One of the variables is called **COMSPEC** and is used by a variety of programs to locate COMMAND.COM. COMMAND.COM does not need to be located in the root directory but can be nested several directories deep. **COMSPEC** contains the directory path of COMMAND.COM. **MNU** uses this environment variable when running batch files. Batch files cannot be run directly like executable files because they only contain information meaningful to COMMAND.COM. They are run indirectly by running another copy of COMMAND.COM and passing the name of the batch file as an argument. If **COMSPEC** cannot be found then **MNU** will assume

that COMMAND.COM is located in the root directory of the current drive.

MNU.COM runs MNU.EXE. MNU.EXE displays an error message if you try and run it separately. MNU.EXE passes certain information regarding the program to be run back to MNU.COM where this information is used to run the program. The reason for this strange relationship is that MNU.COM only requires about **2.5K** of conventional memory while MNU.EXE requires **80K+**. Programs run from MNU.COM receive more memory than if they were run directly from MNU.EXE.

It is important to change to the **MNU** directory before running MNU.COM. If the **MNU** directory is part of your **PATH** and you attempt to run MNU.COM without changing to **MNU** then MNU.COM will run but display an error message indicating that it cannot locate MNU.EXE.

### **Main Screen**

The main screen that appears when MNU.EXE runs contains several pieces of information: a title, a listbox, and two function-key messages.

Centered on the top line is the title. The default is **MNU v1.4** but can be changed to anything desired. The title is selected via the **configuration** data entry form.

A **listbox** is displayed in the center of the screen and is used to display program descriptions. A **tally** of the current number of programs stored in the database is displayed centered below the listbox. If any descriptions are displayed then one of these is highlighted by a lightbar. The **lightbar** identifies the current record and can be moved from one description to another by pressing the UP ARROW, DOWN

ARROW, TAB, SHIFT TAB, PAGE UP, PAGE DOWN, HOME, or END keys. If there are more than **19 (44 in VGA 50-line mode)** descriptions in the database then the listbox will scroll through the descriptions.

The **F1-Help** function-key message is displayed in the lower-left corner. When pressed, a **help screen** is displayed. This help is **context-sensitive** because the information displayed is only relevant to the current context of the program (e.g., a data entry form, the listbox, etc). Part of the information is a list of keys that can be pressed while you remain in this context. You can exit from the help screen by pressing any key. Help is not available from within a message box or the clock/calendar.

The **F10-Exit** function-key message is displayed in the lower-right corner. It indicates that **MNU** can be terminated by pressing the F10 function key. However, F10 has no effect if you press it from within a help screen, data entry form, message box, or clock/calendar. Press any key to exit the help screen or clock/calendar, RETURN or ESCAPE to exit the data entry form, or the indicated keys to exit the message box before pressing F10. If any changes have been made to the internal memory image of the database then you will be prompted to save changes before you can exit. Press N or n (for no) if you do not want to save changes. Press Y or y (for yes) to save changes.

### **Entering a Program**

A program consists of several pieces of information: description, drive, directory, filename, and arguments list. The **description** identifies the program that is being stored and is displayed in the listbox. It gives you a way of identifying cryptic filenames and associated argument

lists. The **drive** identifies the disk drive where this file is located. The **directory** identifies the directory on that drive where the file exists. The **filename** must be specified with either a .COM, .EXE, or .BAT extension. No other extension is valid. Not entering an extension results in an error message. The **arguments list** can contain anything that would normally be passed to the program at the DOS command-line.

The INS key is used to insert a new program into the database. It brings up a data entry form in which you fill out the program description, drive and directory where the program is located, program name, and optional arguments list. When entering a directory, you do not need to enter an initial \ since **MNU** assumes all directories to be relative to the root directory. When INS is pressed, an error message indicating that the database is full will appear if the database has already reached the limit of **400** program records.

If the topmost description is highlighted then you will be asked to insert the new program with description above or below. The only choices are to press the A or a and B or b keys. If the topmost description is not highlighted then the description is always inserted below the highlighted description.

When you insert a new program, edit an existing program, or edit the configuration, a **data entry form** appears. A data entry form is a window that appears at the center of the screen and contains **text** and **fields**. The text identifies the field to its right. The field displays information and allows information to be entered at that spot in the data entry form. Unless information is being entered for the first time, the field will display the most recently entered information. You may keep this information by pressing RETURN, the UP ARROW key, the DOWN ARROW key, the TAB key, or the SHIFT TAB key. When RETURN is pressed on the last field, the data entry form exits with all changes saved. If ESCAPE is pressed then the data entry form exits with all changes cancelled.

There are three kinds of data entry fields: character, numeric, and list. A **character** field contains numbers, letters, and special symbols. An example is the description field. A **numeric** field contains numbers and sometimes a decimal point. A **list** field contains one of several predefined character strings. You toggle between these strings by pressing either the spacebar key or ALT and the spacebar keys. An example is the **Rows:** field on the **configuration** data entry form.



For character and numeric data entry fields, the cursor is positioned over the leftmost character when data entry begins. The cursor is a **blinking underline** to indicate that insert mode is in effect or a **blinking block** to indicate overwrite mode. **Insert mode** means that characters are inserted between other characters whereas **overwrite mode** means that characters overwrite other characters. The LEFT ARROW, RIGHT ARROW, HOME, and END keys move the cursor to other character positions within the field. The INS key toggles between insert and overwrite modes. The BACKSPACE key erases the character to the left of the cursor while the DEL key erases the character under the cursor. If you decide that you prefer the original value after making changes, press ESC to cancel these changes. You can then press RETURN, the UP ARROW/DOWN ARROW keys, or the TAB/SHIFT TAB keys to exit from the field. On the **configuration** data entry form, pressing RETURN on some fields will cause the screen to be redrawn. You will not exit that field.

The following example illustrates what you might enter in a **program** data entry form.

**Desc:** *Word Perfect*

**Drive:** C

**Dir:** WP

**File:** WP.EXE

**Arg:**

## Running A Program

After you have added new information and/or made changes, you will need to save this information to the database. Press the S or s key to save changes. As already mentioned, if you press F10 to exit **MNU** and changes have not been saved then you will be prompted to save these changes. When you press RETURN to run the program associated with the highlighted description, you will be prompted to save any changes.

When RETURN is pressed, MNU.EXE passes information regarding that program's drive, directory, filename, and argument list to MNU.COM. If a batch file is to be executed then MNU.EXE selects COMMAND.COM as the filename and the batch file as part of COMMAND.COM's argument list. MNU.COM saves the current drive and directory before attempting to run the program. If that program is prematurely aborted through a critical error or a Ctrl-Break then you will be left in that program's drive/directory. You will need to switch back to the **MNU** directory and type **MNU** to rerun the **MNU** program. Otherwise, MNU.COM returns to the original **MNU** directory and reruns MNU.EXE.

### **Program Maintenance**

If you make a mistake while entering a program record and discover this error after the information has been inserted into the database then you can correct the error by pressing the E or e key. This key allows you to edit the current program record.

The DEL key is used to delete the current program record. Press Y or y (for yes) in response to the **ARE YOU SURE?** message prompt if you wish to delete this record.

Remember that changes are not permanent until you respond with Y or y at the **SAVE CHANGES?** message prompt.

## Configuration

Part of the fun in working with **MNU** is the ability to customize it to suit your needs. The C or c key brings up a data entry form allowing you to view and modify the configuration.

**Screen blanking** is used to prevent monitor burnin. You can enter a value from **0** to **60** inclusive. A zero disables screen blanking. Any other value represents the number of minutes that must elapse after the most recent key press before the screen will blank.

You can specify a **title** that will appear centered on the top screen line. This title is displayed as soon as you leave the **configuration** data entry form.

If you have a **VGA** adaptor then you can set the number of screen lines to **50**. Simply press the space bar until **50** appears. This field is skipped if you do not have a **VGA** adaptor. The screen switches to **25** or **50** lines as soon as you press RETURN.

Various screen colors can also be changed. Press the spacebar to select YES to the right of the appropriate field and RETURN to bring up

a **color change chart**. Once you have selected an appropriate color, press RETURN. The change is reflected immediately.

### **Extras**

The A or a key allows you to alphabetize program records by description. You must press Y or y in response to the **ARE YOU SURE?** message prompt before alphabetizing takes place. Once alphabetizing has occurred, you will be prompted to save changes when attempting to leave **MNU**.

The D or D / T or t keys bring up a clock/calendar showing the current date and time. Press any key to exit the clock/calendar. Help is not available and you cannot exit MNU from within the clock/calendar.

## **Appendix A: Problems**

**PC-GLOBE 4.0** will not run under **MNU**. **PC-GLOBE** repeatedly issues interrupt **5** instructions (print screens) instead. The problem lies with **PC-GLOBE**. I tried running **PC-GLOBE** under **DEBUG** (any **DOS** program should be able to run under **DEBUG**) but the same problem occurred.

Some programs such as **GEOS** may display an error message when returning to **MNU.COM** if the **COMSPEC** variable is missing. However, this should not disrupt the **MNU** program.

## **APPENDIX B: MNU.COM and MNU.EXE Messages**

Most MNU.EXE messages appear in a message box. A **message box** is a window identifying the message and indicating what keys to press in response.

**mnu.com: invalid DOS version number - needs 3.30+**

MNU.COM will not operate under any version of **DOS** less than **3.30**.

The reason is that MNU.EXE requires **DOS 3.30** or higher and MNU.COM must run MNU.EXE.

**mnu.com: unable to resize memory**

MNU.COM must be able to reduce its memory space to just under **2.5** kilobytes. If it cannot do this then it displays this message and exits. This message will probably never be displayed.

**mnu.com: unable to run mnu.exe**

MNU.EXE may not be present in the current directory or there may be insufficient memory to run MNU.EXE.

**mnu.com: unable to save current directory**

MNU.COM could not save the current directory, the directory containing MNU.EXE. MNU.COM must be able to save this directory so that it can return to it and run MNU.EXE once the current program finishes running.

**mnu.com: unable to change directory**

MNU.COM could not switch to the program's directory.

**mnu.com: unable to run program - press any key**

MNU.COM could not find the program that it is supposed to run. It is waiting for you to press a key before it continues.

**mnu.com: unable to restore original directory**

MNU.COM could not switch back to the saved directory.

**mnu.exe: invalid DOS version number - needs 3.30+**

MNU.EXE will not operate under any version of **DOS** less than 3.30.

**mnu.exe: mnu.com expected as parent**

You are attempting to run MNU.EXE instead of MNU.COM. MNU.COM must be in the same directory as MNU.EXE.



### **mnu.exe: insufficient memory**

There is not sufficient memory for MNU.EXE to allocate its workspace. If you have shelled out of several programs or installed too many **device drivers** and **TSRs** then you will have to either unshell and exit these programs or remove some of the **drivers/TSRs**. Resetting the computer may also help.

### **mnu.exe: unable to load configuration**

The **MNU** configuration file MNU.CFG was found but **MNU** could not load it into memory. The default configuration will be used. Press the C or c key to examine this configuration.

### **ARE YOU SURE?**

This message is displayed to give you a final chance to change your mind before carrying out a deletion or alphabetizing operation. You respond by pressing Y or y (for yes) or N or n (for no).

### **DATABASE LIMIT REACHED**

The database file can accommodate a maximum of **400** program records. If you press the INS key to insert a new record and the limit has been reached then this message is displayed.

### **FILE CORRUPT - UNABLE TO READ**

**MNU** expects MNU.DBF to follow the **DBASE III+** file format. This message appears when **MNU** discovers that the file is not in this format.

### **INSERT ABOVE OR BELOW**

When inserting a program record into memory and the description for the first record appears highlighted in the listbox, this message is displayed. With the exception of the very first record, records are always inserted after the current record. With the first record, there is the option of inserting ahead of this record.

## **SAVE CHANGES?**

When you press F10 to exit, you will see this message if any changes have been made to the internal memory image of the database. Press Y or y (for yes) to save these changes. Press N or n (for no) if you do not want these changes saved.

## **UNABLE TO CREATE CONFIGURATION FILE**

When you change the configuration, **MNU** needs to save the new configuration to a configuration file called MNU.CFG. This message will appear if **MNU** is unable to create this file.

## **UNABLE TO CREATE FILE**

Before program records can be saved to MNU.DBF, it must be created. **MNU** could not create this file for some reason (insufficient room on the disk is a common occurrence).

## **UNABLE TO READ FROM FILE**

This message is displayed when **MNU** attempts to load MNU.DBF information but fails.

## **UNABLE TO SAVE CONFIGURATION**

When you modify the configuration, **MNU** attempts to save it immediately to MNU.CFG. If **MNU** is unable to save the configuration then this message will be displayed.

## **UNABLE TO WRITE TO FILE**

**MNU** was unable to write the **DBASE** file header, program records, or file terminator to the database file.

## **CRITICAL ERRORS**

A **critical error** is an error that **DOS** cannot resolve. You must tell **DOS** what course of action to take. For example, attempting to save changes to a floppy disk in A: when no disk is present will generate a critical error. In response, the critical error handler activates. **MNU** displays a window identifying the nature of the error and allowing you to select ABORT or RETRY options. If you find the disk and close the

drive handle then you could select RETRY to attempt the operation again. ABORT cancels the operation. Another error message will appear. You can press F1 while the critical error window is on the screen for more information.