

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	1	

COBGDB - The GnuCOBOL TUI DEBUGGER / ANIMATOR
<i>HOW TO USE COBGDB</i>



Table of Contents

1.Introduction.....	3
1.1.Installing the debugger COBGDB on Windows.....	4
1.2.Compile and Debug GnuCOBOL programs.....	5
1.3.Main Commands.....	6
2.Tutorial - Sample Debugging Session.....	7
2.1.Help Command.....	9
2.2.Run Command.....	10
2.3.Step Command.....	12
2.4.Go Command.....	14
2.5.Display Variables.....	15
2.6.Show Command.....	16
2.6.1.Edit subCommand.....	17
2.7.Variable Command.....	18
2.7.1.Enter subCommand.....	19
2.7.2.Edit subCommand.....	20
2.7.3.Return subCommand.....	21
2.8.Step Command.....	22
2.9.Focus Command.....	24
2.10.Pop-up Variable windows.....	26
2.11.File Command.....	27
2.12.Run Command.....	30
2.13.Window Size command.....	31
2.14.Quit Command.....	33
2.15.Attach Command.....	34
3.Other Commands.....	35
3.1.Debugging a pre-compiled Program.....	35
3.2.Debugging sub programs.....	36
3.3.COBGDB Version.....	38
4.Document Change Log.....	39

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	2	

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	3	

1. Introduction

COBGDB is a TUI (Text User Interface) application, programmed in C, designed to assist in animate and debugging GnuCOBOL programs using **GDB** (the GNU Debugger at <https://www.gnu.org/software/gdb/>).

The COBGDB project is hosted at <https://github.com/marcsosduma/cobgdb>

Very important: you don't need to know how to use the GDB product and its many commands (<https://www.sourceware.org/gdb/>).

COBGDB has its own user interface (described in this document) that is very simple to use and is responsible for interfacing the underlying GDB which is the real debug and animate engine but operates practically in a transparent way to the GnuCOBOL developer.

The COBGDB application is based on the extension for Visual Studio Code (VSCode) created by Oleg Kunitsyn, which can be found on GitHub: <https://github.com/OlegKunitsyn/gnucobol-debug>.

At <https://github.com/marcsosduma/cobgdb> in the Windows subdirectory, the executable program [cobgdb.exe](#) for this operating system is available and ready to use.

To compile COBGDB from C source code on Windows, you can use MinGW.

The Makefile is configured to generate the program `cobgdb.exe` for both Windows and Linux.

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	4	

1.1. Installing the debugger COBGDB on Windows


On Windows, just download **cobgdb.exe** from following folder:
<https://github.com/marcsosduma/cobgdb/tree/main/windows>.

As an example you can put **cobgdb.exe** into the "bin" folder of your GnuCOBOL installation (the same folder where the GnuCOBOL compiler **cobc.exe** is located)

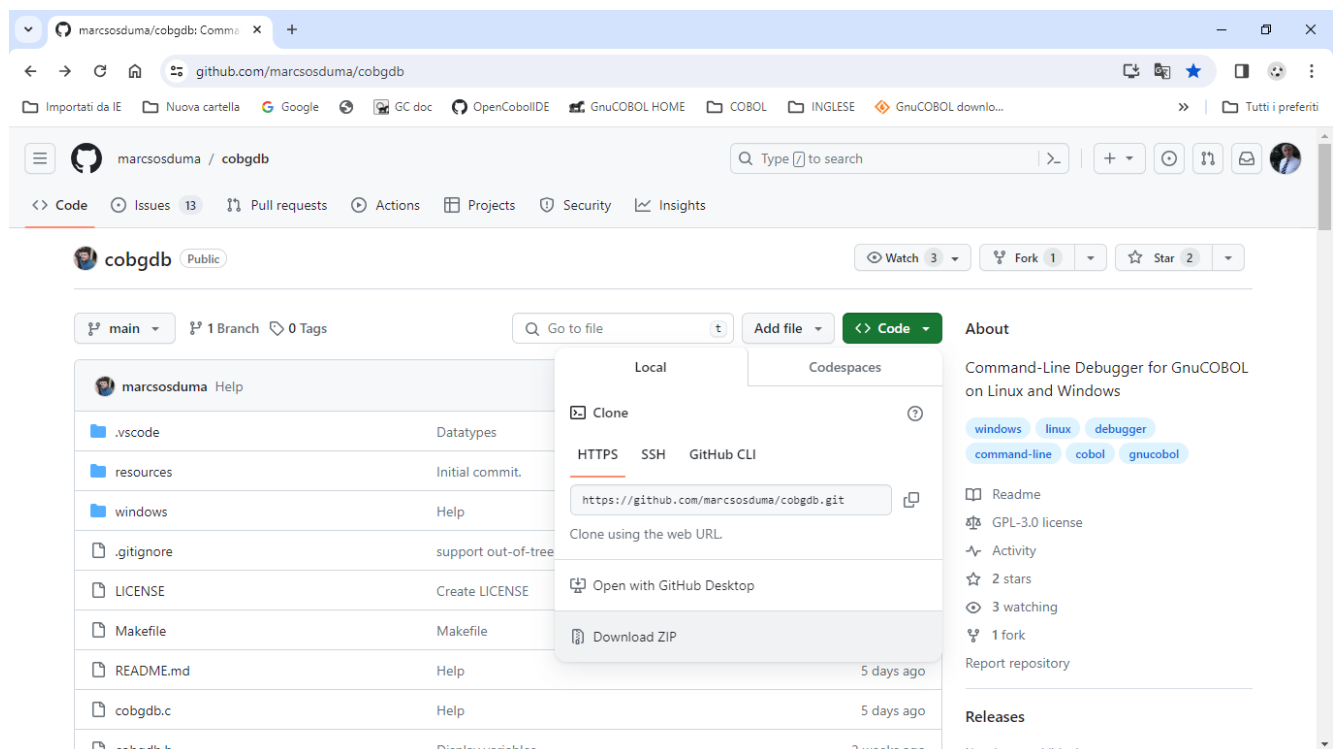
or

first install MinGW (Minimalist GNU for Windows).

Then execute the make ('mingw32-make' for Windows) command to compile the code from C source.

Note: if you have security problems downloading an .exe file then you can try download the entire repository with the following github button  --> **Download ZIP**

Then unzip the file and copy cobgdb.exe to the "bin" folder of your GnuCOBOL installation (the same folder where the GnuCOBOL compiler cobc.exe is located)



DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	5	

1.2. Compile and Debug GnuCOBOL programs

Compile and run a debugging session of the sample program using the following command:

```
cobgdb customer.cob -x -lpdcurses
```

Source code of customer.cob used also for following tutorial is at:

<https://github.com/marcosduma/cobgdb/tree/main/windows>

Note: '-lpdcurses' is an instance of an argument that can be indirectly passed to 'cobc' by 'cobgdb,' even if it is not used by 'cobgdb' itself.

or, other example for cobc parameters , use : **cobgdb customer.cob -x -Tcustomer.txt**
(-T creates a compilation list output into the **customer.txt** file)

COBGDB takes one or more programs with COB or CBL extension as parameters and runs the GnuCOBOL compiler with the following format:

```
cobc -g -fsource-location -ftraceall -v -free -O0 -x prog.cob prog2.cob ...
```

To debug multiple programs, use COBGDB with the following syntax :

```
cobgdb prog.cob subprog1.cob subprog2.cob ...
```

This will create a single prog.exe executable to debug.

To debug sub programs separately , see also "Debugging sub Programs" chapter in this document.

You can also run GDB/GDBSERVER remotely using the "A" (Attach) key.

COBGDB will prompt you to provide the server and port in the format server: port or the PID of the application.

Example:

- localhost: 5555
- 9112

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	6	

1.3. Main Commands

Cmd		Description
?	Help	Show the HELP window
R	Run	This is the first command to always use to start a debugging session. Start and Run the program until a breakpoint is encountered, also when, by default, the first Breakpoint is at first program statement.
B	Breakpoint	Set or Unset a Breakpoint at a specific line of the Procedure Division code.
C	Cursor	Runs the program until it reaches the selected line at Cursor location.
N	Next	Runs the program until the next line but does not enter a subroutine executed by CALL or PERFORM.
S	Step	Runs the program until the next line. If needed it goes into a subroutine executed by CALL or PERFORM.
G	Go	Continues the program execution until it encounters a stopping point: a breakpoint, the end of the program, or the return from a subroutine (PERFORM / CALL).
J	Jump	Ask for a line number and Runs the program until it reaches that line.
V	Variables	Displays a window with a list of all variables and their content for the running program. From this window you can also change the content of variables.
H	sHow	Displays a window with a list of variables and its content from the cursor selected line. From this window you can also edit / change the content of variables from the selected line. Right-click on a row is same as command "H".
D	Display	Automatic Display of variables of current and previous statement in execution during a debugging / animation session is settled to OFF or ON. At program start is OFF.
O	fOcus	When program encounters an ACCEPT verb, this command switch the focus from debugging / animation screen to the application screen to accept an user input and then automatic go back to the debugging / animation screen.
F	File	When COBGDB is executed with more than one program, allows selecting one of those source file to manage debugging commands.
A	Attach	Attach to GDBSERVER or to an Application PID.
W	Window	Switch between two window size: 24 rows x 80 cols or 34 rows x 132 cols.
CTRL - F	Find	Search for text in the source code
CTRL - L	Go to	Go to Line.
Q	Quit	Quits (ends) the debugging / animation session and the program (or programs).

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	7	

2. Tutorial - Sample Debugging Session

This tutorial is on a Windows 10 platform using following version of GnuCOBOL:

```
cobc (GnuCOBOL) 3.2.0
Copyright (C) 2023 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <https://gnu.org/licenses/gpl.html>
This is free software; see the source for copying conditions. There is NO
warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.
Written by Keisuke Nishida, Roger While, Ron Norman, Simon Sobisch, Edward Hart
Built Jul 28 2023 16:07:38
Packaged Jul 28 2023 16:58:47 UTC
C version (MinGW) "13.1.0"
```

Downloaded from <https://www.arnoldtrembley.com/GnuCOBOL.htm>

===== Version 3.2 =====

GnuCOBOL 3.2 (28Jul2023) MSYS2 64-bit [GC32M-BDB-x64.7z](#) -- MSYS2 64-bit GnuCOBOL 3.2

Final release **with full debugging support**. (95.4 Megabytes).

GnuCOBOL 3.2 (28Jul2023) MSYS2 32-bit [GC32M-BDB-x32.7z](#) -- MSYS2 32-bit GnuCOBOL 3.2

Final release **with full debugging support**. (96.1 Megabytes).

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	8	

After executing **cobgdb customer.cob -x -lpdcurses** the application automatically insert a Breakpoint at first executable program statement of PROCEDURE DIVISION (see the **B** symbol at left of line 103 in this sample) and displays following screen:

You can scroll the source code window with cursor keys UP and DOWN, PG UP and PG DOWN or with mouse wheel or with mouse left click on the right scroll bar. Use cursor RIGHT and cursor LEFT to scroll horizontally,

In the upper right window corner there is a "button bar" where you can find some buttons (symbols):




- > = Run command
- = Next command
- ↓ = Step command
- ↑ = Go command
- = Quit command
- <-> = Focus command
- D = Display command
- ? = Help command

when you hover over one of these symbol, you get the corresponding command description (like a tooltip) displayed at the bottom left of the screen.

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	9	

2.1. Help Command


Type ? (key) HELP command or left click with mouse on the  button:

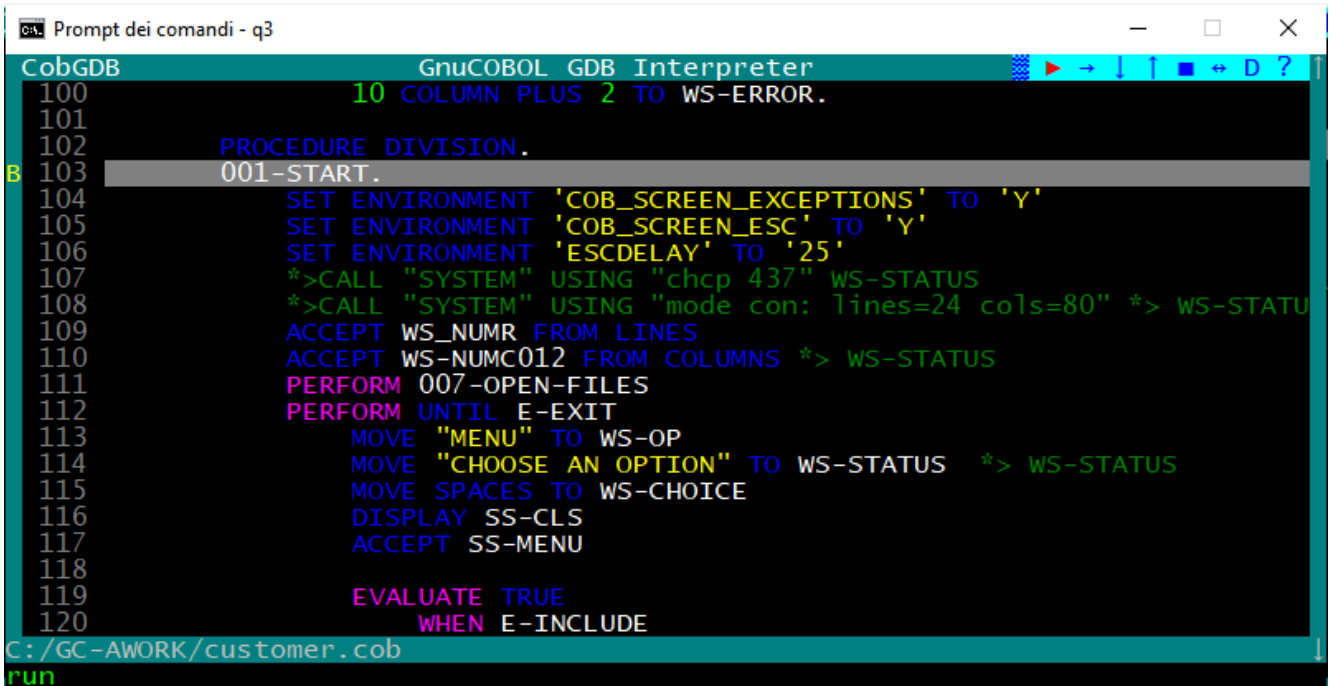
the HELP window is displayed

scroll the Help window with cursor keys UP and DOWN or mouse wheel.
Use ESC or Enter or left click to exit from this HELP window and return to debugging session.

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	10	

2.2. Run Command

To start executing the program and the debugging session from first program statement you always must use the "R" command (key) or left click with mouse on the  Run button



```

CobGDB GnuCOBOL GDB Interpreter
100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110          ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116              DISPLAY SS-CLS
117              ACCEPT SS-MENU
118
119              EVALUATE TRUE
120              WHEN E-INCLUDE
run
C:/GC-AWORK/customer.cob

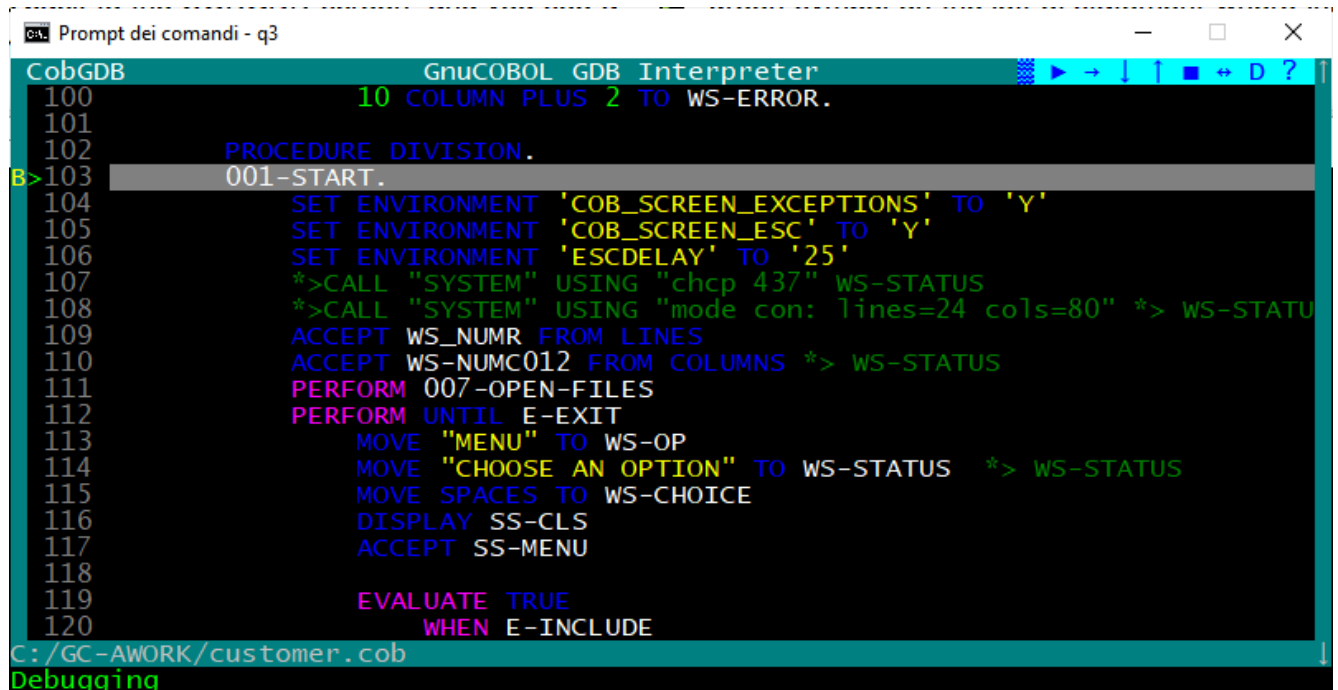
```

cobgdb opens the program terminal window (the application will run in this separate window.).



DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	11	

go back to the COBGDB screen, and you see a  green symbol on the left of statement where initial B (Breakpoint) is present (in our example is at line 103 **B>103**):



From that moment on, you can use all the commands (keys) or corresponding buttons to "animate" and debug the application, example: **"S"** (Step), **"N"** (Next), **"G"** (Go) and so on.


After pressing D button and during the source code animation, the debugger automatically shows some pop-up windows with variables content from the line in execution.

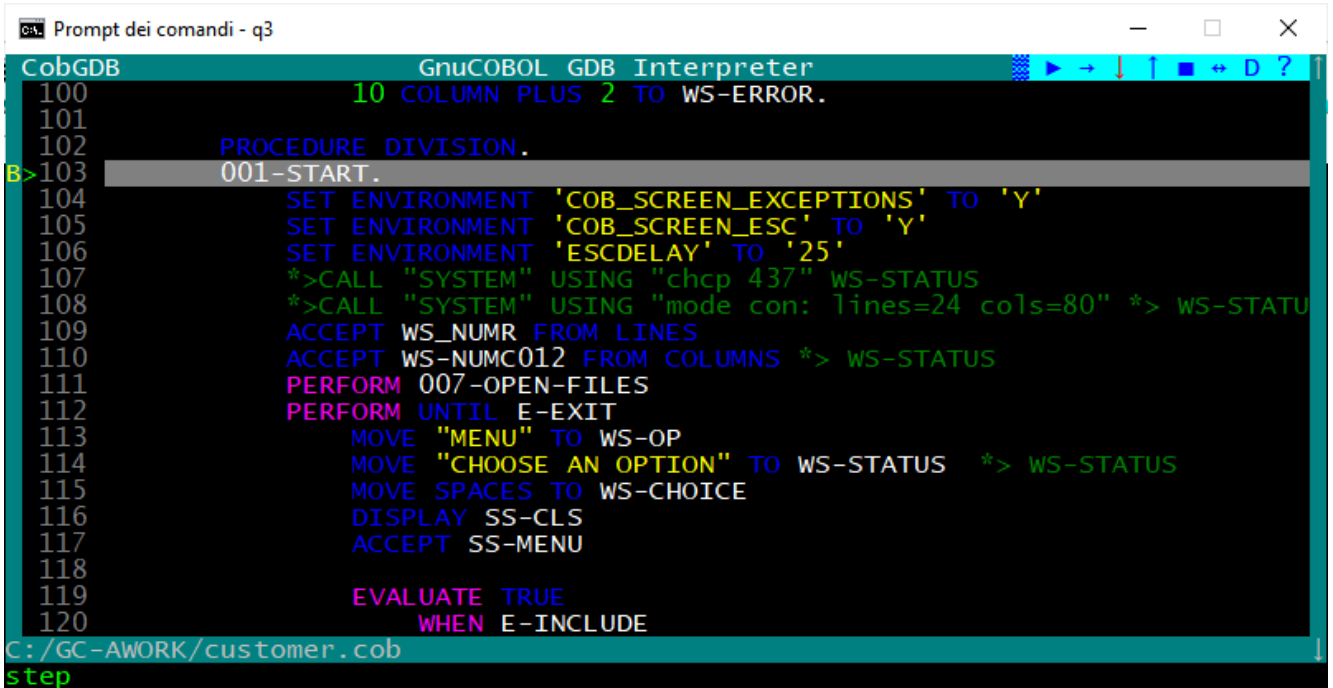
In order not to slow down the debugging / animation of the program unnecessarily, the automatic display of all variables of the statement in execution is disabled by default at program startup.


To activate the Automatic Display of all variables of the statement in execution use the "D" command.

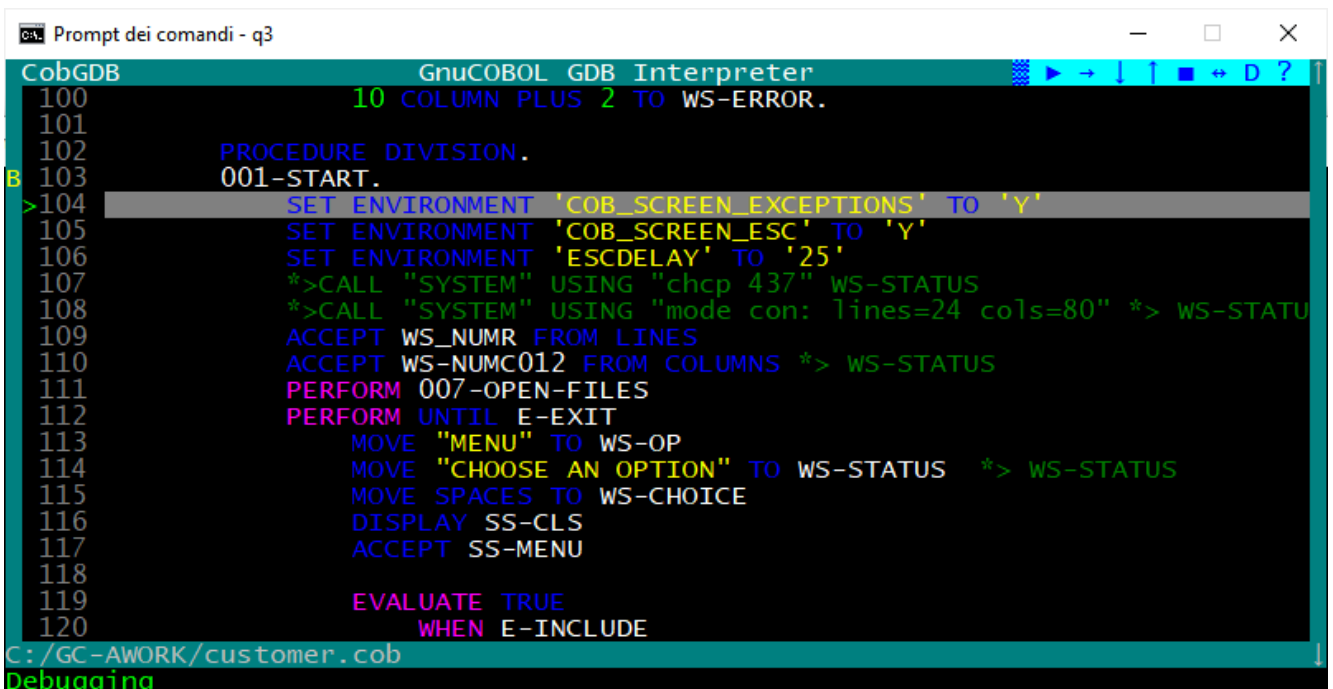
DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	12	

2.3. Step Command

Proceed with **S** (Step) command or left click with mouse the  button:.



the  green symbol now is on the following line 104:



Now you can proceed with S command or N command or as an example:

- File: "COBGDB-GnuCOBOL-DEBUGGER-V10-20251110.odt" -

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	13	

- Scroll with cursor down to select line 116 of Procedure Division and type "B" (to set a Breakpoint), (you also can simply click with mouse left button on the 116 row number)

- The application displays a "B" on the left of the line (type B again - or re-click - when you want to delete the Breakpoint, not do that at this moment)

The screenshot shows a window titled "CobGDB GnuCOBOL GDB Interpreter". The window contains a COBOL program with the following code:

```

100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
103      001-START.
104      >    SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110          ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116      B    DISPLAY SS-CLS
117          ACCEPT SS-MENU
118
119          EVALUATE TRUE
120              WHEN E-INCLUDE

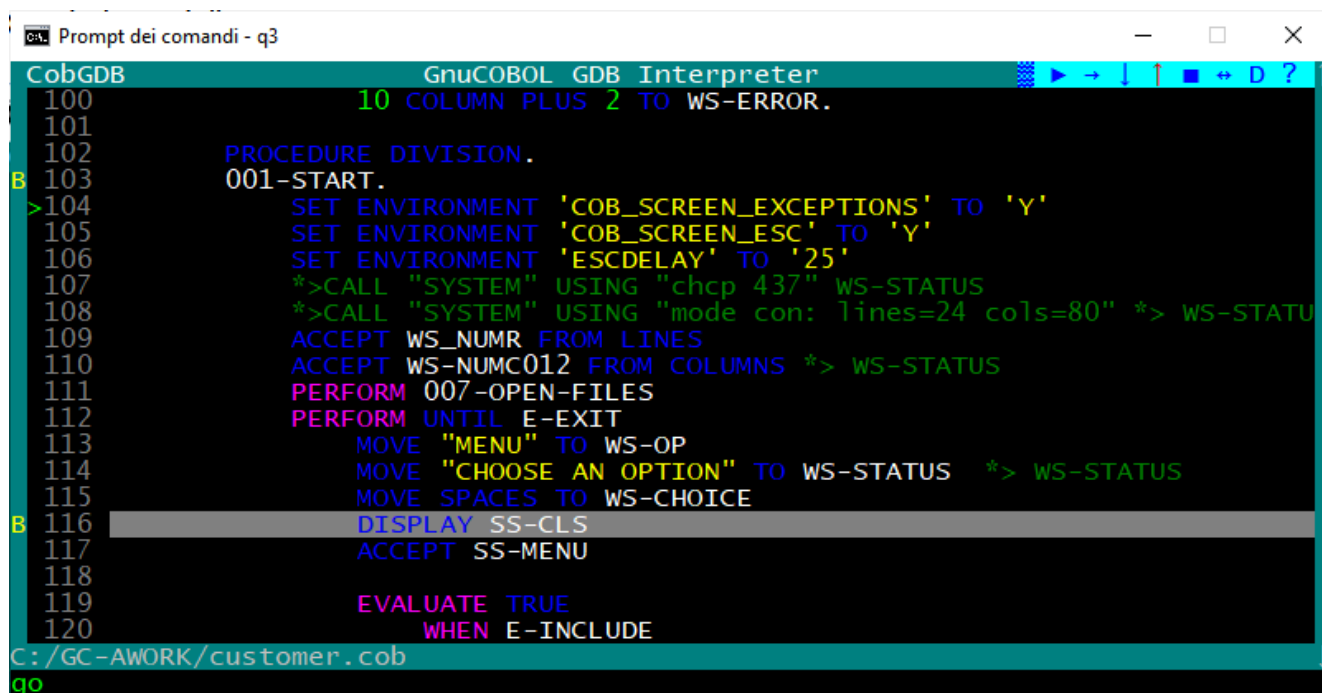
```

A yellow "B" is visible to the left of line 116, indicating a breakpoint. The window title bar shows "Prompt dei comandi - q3". The status bar at the bottom shows "C:/GC-AWORK/customer.cob" and "breakpoint".

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	14	

2.4. Go Command


Type **G** (Go) or left click with mouse the  button to execute the program until a B Breakpoint is detected:

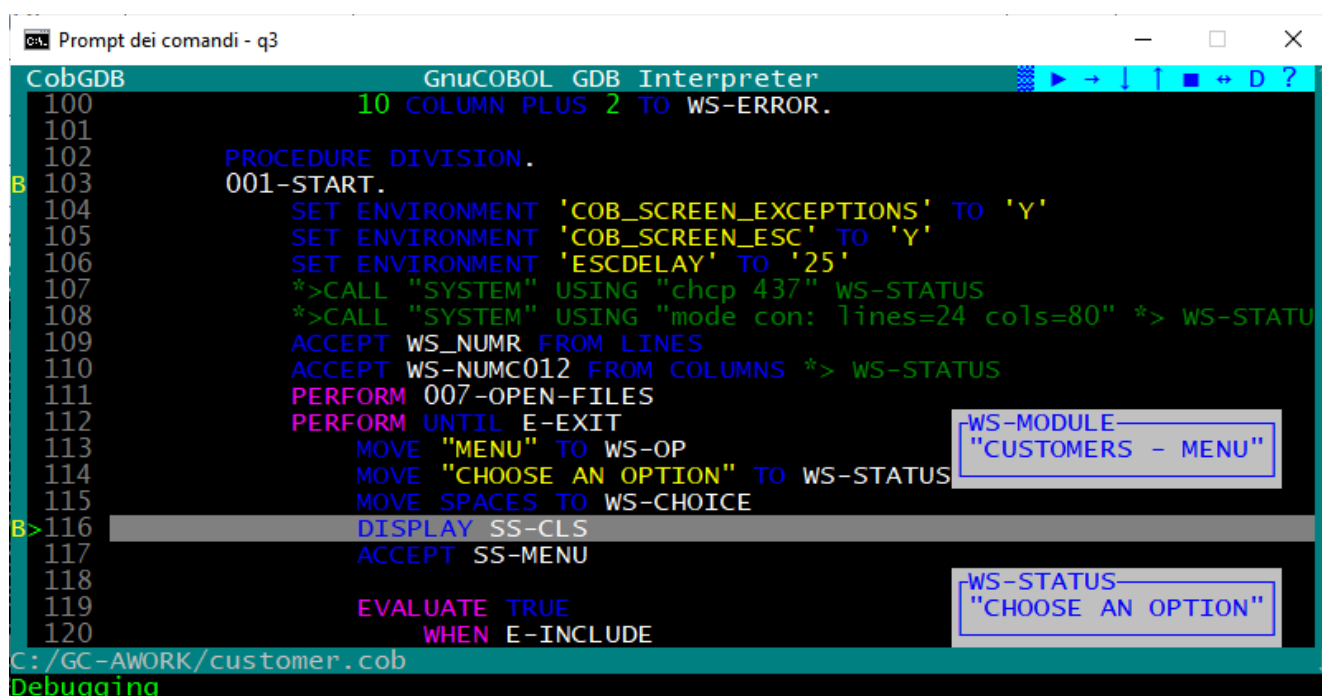


```

CobGDB                               GnuCOBOL GDB Interpreter
100                                10 COLUMN PLUS 2 TO WS-ERROR.
101
102                                PROCEDURE DIVISION.
103                                001-START.
>104                                SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105                                SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106                                SET ENVIRONMENT 'ESCDELAY' TO '25'
107                                *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108                                *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109                                ACCEPT WS_NUMR FROM LINES
110                                ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111                                PERFORM 007-OPEN-FILES
112                                PERFORM UNTIL E-EXIT
113                                    MOVE "MENU" TO WS-OP
114                                    MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115                                    MOVE SPACES TO WS-CHOICE
B 116                                DISPLAY SS-CLS
117                                ACCEPT SS-MENU
118
119                                EVALUATE TRUE
120                                    WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
go

```

the system reach the second breakpoint at line 116 and displays a green  symbol to the left of the line to be executed, see following screen:



```

CobGDB                               GnuCOBOL GDB Interpreter
100                                10 COLUMN PLUS 2 TO WS-ERROR.
101
102                                PROCEDURE DIVISION.
103                                001-START.
104                                SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105                                SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106                                SET ENVIRONMENT 'ESCDELAY' TO '25'
107                                *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108                                *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109                                ACCEPT WS_NUMR FROM LINES
110                                ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111                                PERFORM 007-OPEN-FILES
112                                PERFORM UNTIL E-EXIT
113                                    MOVE "MENU" TO WS-OP
114                                    MOVE "CHOOSE AN OPTION" TO WS-STATUS
115                                    MOVE SPACES TO WS-CHOICE
B>116                                DISPLAY SS-CLS
117                                ACCEPT SS-MENU
118
119                                EVALUATE TRUE
120                                    WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
Debugging

```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	15	

2.5. Display Variables

The debugger / animator can do an automatic display of all variables content from the statement executed and from the previous statement.

This can slow down the animation / execution and it may not be useful to have it activated everywhere.

Type **D** (Display Variables) command to enable or disable this display at your convenience.

Alternatively, click on the "D" symbol in the top right bar of the screen.

By moving the mouse over the "D" symbol in the bar you will see a message with the current status of this option displayed at the bottom left of screen: Display of Variables: ON or Display of Variables: OFF.

At the beginning of the program this option is always set to OFF to speed up the animation.

When you have reached the point of the program that interests you, you can click on "D", or type the command "D", to activate the automatic display of all the variables of the statement in execution (black pop-up windows) and of the previous statement (blue pop-up windows).

The screenshot shows the CobGDB GnuCOBOL GDB Interpreter window. The title bar reads "CobGDB GnuCOBOL GDB Interpreter". The window contains COBOL code with line numbers 100 to 120. The code includes a procedure division and a loop. Two blue pop-up windows are visible, displaying variable values: "WS-MODULE" with value "CUSTOMERS - MENU" and "WS-STATUS" with value "CHOOSE AN OPTION". The status bar at the bottom left indicates "display of variables: ON".

```

100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110          ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116          DISPLAY SS-CLS
117          ACCEPT SS-MENU
118
119          EVALUATE TRUE
120              WHEN E-INCLUDE
  
```

WS-MODULE
CUSTOMERS - MENU

WS-STATUS
CHOOSE AN OPTION

C:\GC-AWORK\customer.cob
display of variables: ON

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	16	

2.6. Show Command

typing the 'H' command (key) allows you to view the variables on the cursor selected (highlighted) line.

```

CobGDB GnuCOBOL GDB Interpreter
100 10 COLUMN PLUS 2 TO WS-ERROR.
101
102 PROCEDURE DIVISION.
103 001-START.
104 SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105 SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106 SET ENVIRONMENT 'ESCDELAY' TO '25'
107
108 -SS-CLS: " CUSTOMERS - MENU CHOOSE AN OPTION
109 -SS-CLS: " CUSTOMERS - MENU CHOOSE AN OPTIO
110 -Implicit FILLER: ""
111 -Implicit FILLER: ""
112 -Implicit FILLER: ""
113 -Implicit FILLER: "CUSTOMERS - MENU "
114 -SS-STATUS: "CHOOSE AN OPTION "
115 -Implicit FILLER: "CHOOSE AN OPTION "
116
117 ACCEPT SS-MENU
118
119 EVALUATE TRUE
120 WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
E - Edit variable

```

Display the content of variables also clicking right mouse button on a source line, example click right mouse button on line 114 will execute the H command on that line and give you:

```

CobGDB GnuCOBOL GDB Interpreter
100 10 COLUMN PLUS 2 TO WS-ERROR.
101
102 PROCEDURE DIVISION.
103 001-START.
104 SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105 SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106 SET ENVIRONMENT 'ESCDELAY' TO '25'
107
108 -WS-STATUS: CHOOSE AN OPTION "
109 -WS-STATUS: "CHOOSE AN OPTION "
110
111 PERFORM 007-OPEN-FILES
112 PERFORM UNTIL E-EXIT
113 MOVE "MENU" TO WS-OP
114 MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115 MOVE SPACES TO WS-CHOICE
116 DISPLAY SS-CLS
117 ACCEPT SS-MENU
118
119 EVALUATE TRUE
120 WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
E - Edit variable

```

Now at lower left line on screen is a message: E = Edit (change) the variable value.

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	17	

2.6.1.Edit subCommand

Scroll with cursor key UP and DOWN (or with mouse wheel). to select one of the variable.

The key **E** can be used to edit the content of the highlighted variable.

Change the value and type Enter to confirm changes or use ESC to exit without changes :

```

CobGDB      GnuCOBOL GDB Interpreter
100          10 COLUMN PLUS 2 TO WS-ERROR.
101
102          PROCEDURE DIVISION.
103      B 001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107
108          Edit Variable
109          WS-STATUS: xxxxxSE AN OPTION
110
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS * > WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116          B 116          DISPLAY SS-CLS
117          >117          ACCEPT SS-MENU
118
119              EVALUATE TRUE
120              WHEN E-INCLUDE
C:/GC-AWORK/customer.cob

```

Resulting:

```

CobGDB      GnuCOBOL GDB Interpreter
100          10 COLUMN PLUS 2 TO WS-ERROR.
101
102          PROCEDURE DIVISION.
103      B 001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107
108          Show Line Variables
109          -WS-STATUS: \"xxxxxSE AN OPTION\"
110
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS * > WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116          B 116          DISPLAY SS-CLS
117          >117          ACCEPT SS-MENU
118
119              EVALUATE TRUE
120              WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
E - Edit variable

```


DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	20	

2.7.2.Edit subCommand

Now you can select WS-MODULE subfield whit cursor DOWN or mouse wheel and type "E" (Edit)
COBGDB shows a Edit Variable window:

The screenshot shows a terminal window titled "Prompt dei comandi - q3". The main text area displays COBGDB commands and their outputs. A red rectangular box highlights the "Edit Variable" window, which is titled "WS-MOD Edit Variable". Inside this box, the text "WS-OP: WS-MODULE: CUSTOMERS - MENU" is visible, with a cursor positioned at the end of the text. The background text in the terminal includes various status and menu information, such as "RETURN-CODE: 0", "WS-MODULE: 'CUSTOMERS - MENU'", and a menu list: "1 - INCLUDE2 - CONSULT3 - UPDATE4 - DELETEx - EXITCHOICE: ' '".

Change "CUSTOMERS" to "TEST" and type Enter to change the value (use ESC to exit without changes) :

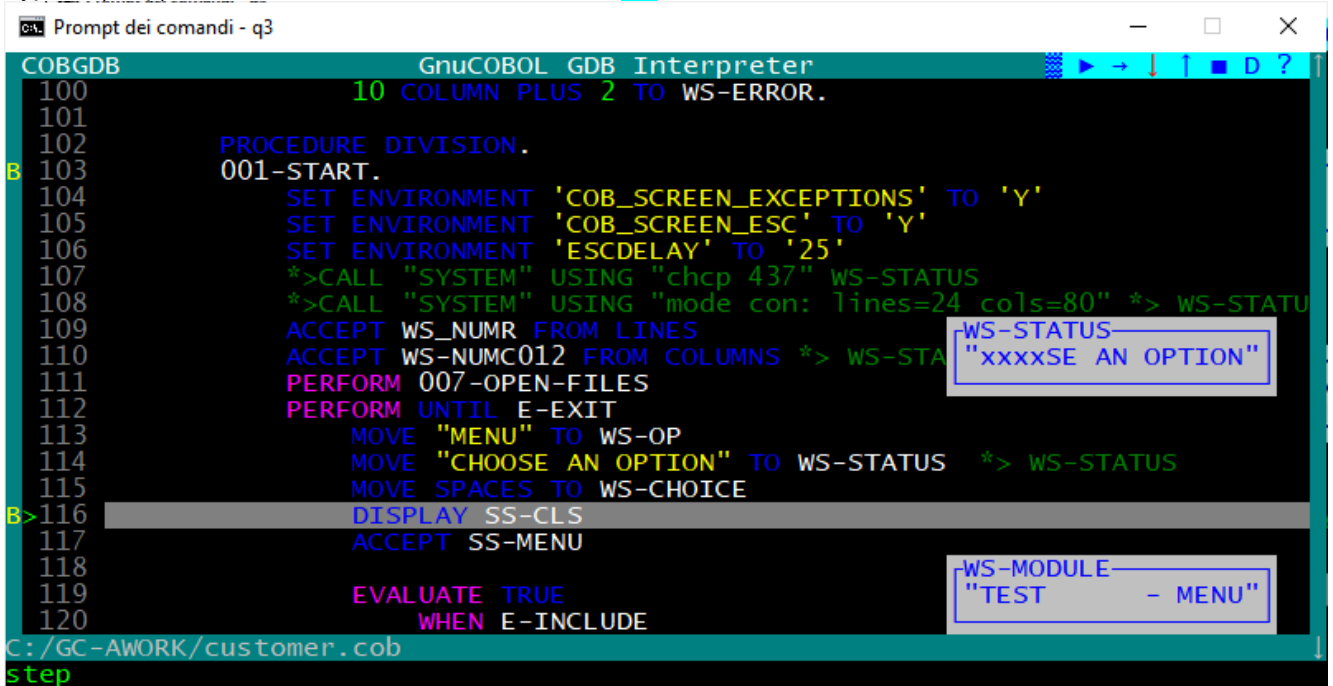
This screenshot is similar to the previous one, showing the same terminal window. The red rectangular box highlights the "Edit Variable" window, but now the text inside is "WS-OP: WS-MODULE: TEST - MENU". The cursor is still at the end of the text. The background text in the terminal remains the same as in the previous screenshot.

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	22	

2.8. Step Command

now you are back in the main debugging window:

Type **S** (Step) command or left click with mouse the  button to execute the DISPLAY statement at line 116



```

COBGDB GnuCOBOL GDB Interpreter
100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110          ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116      B>116      DISPLAY SS-CLS
117              ACCEPT SS-MENU
118
119              EVALUATE TRUE
120                  WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
step

```

in the other application window you can see the result of DISPLAY statement



```

C:\GC-AWORK\customer.exe
TEST - MENU

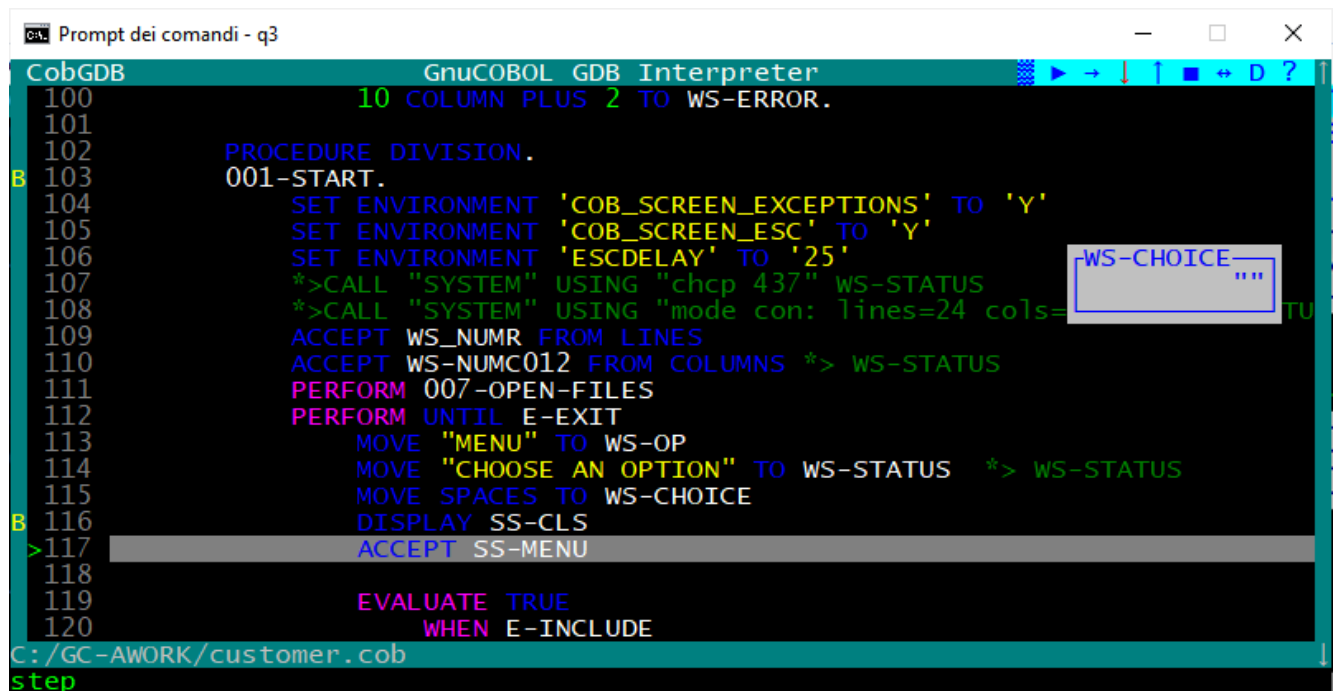
xxxxSE AN OPTION

```

go back to debugging window and now the ACCEPT statement will be executed with S command

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	23	

Type **S** (Step) command or leftclick with mouse the  button again to execute ACCEPT statement at line 117:



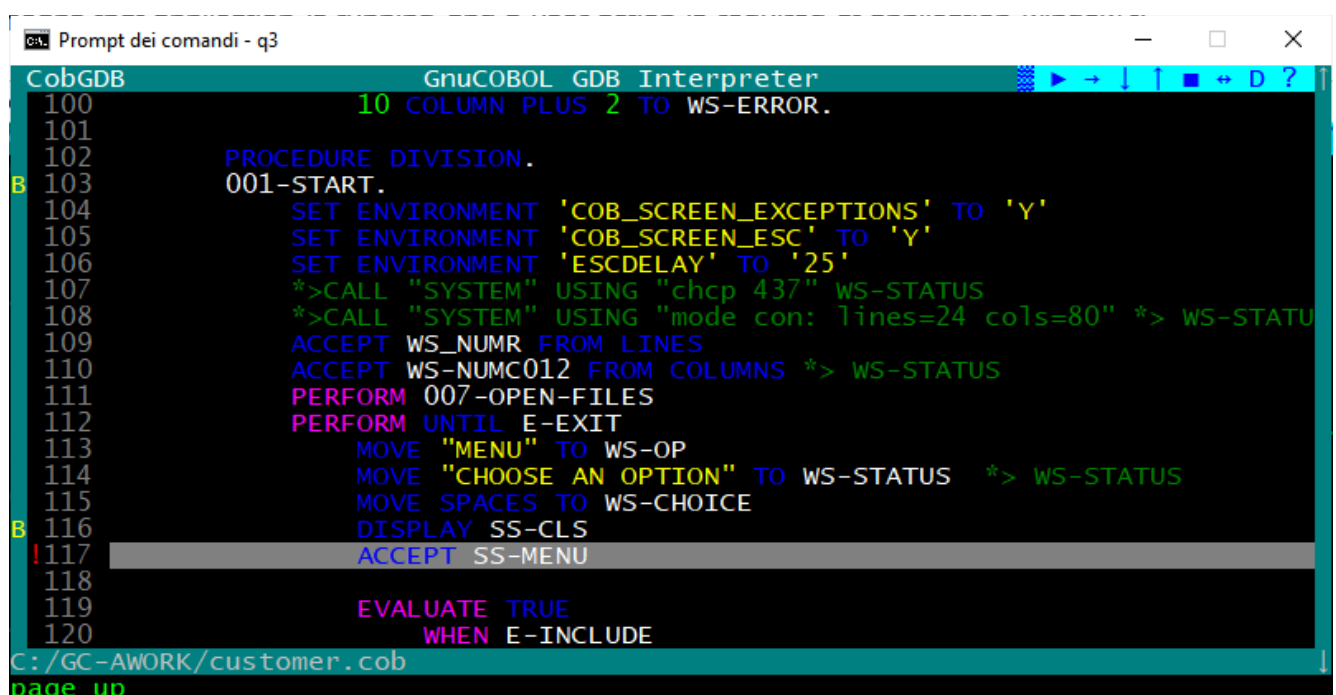
```

CobGDB GnuCOBOL GDB Interpreter
100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110          ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116              DISPLAY SS-CLS
117          ACCEPT SS-MENU
118
119          EVALUATE TRUE
120              WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
step

```

A red **!** exclamation mark appears on the line **!117**.

This means that the application is running and a user action is required at application window.



```

CobGDB GnuCOBOL GDB Interpreter
100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110          ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116              DISPLAY SS-CLS
117          ACCEPT SS-MENU
118
119          EVALUATE TRUE
120              WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
page up

```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	24	

2.9. Focus Command

The application is running in a separate window.

The 'O = Focus' command switches the focus from debugging / animation window to the application window. Instead of using the "O" command you can left-click on the <-> icon.

```

CobGDB GnuCOBOL GDB Interpreter
100 10 COLUMN PLUS 2 TO WS-ERROR.
101
102 PROCEDURE DIVISION.
103 001-START.
104 SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105 SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106 SET ENVIRONMENT 'ESCDELAY' TO '25'
107 *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108 *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109 ACCEPT WS_NUMR FROM LINES
110 ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111 PERFORM 007-OPEN-FILES
112 PERFORM UNTIL E-EXIT
113 MOVE "MENU" TO WS-OP
114 MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115 MOVE SPACES TO WS-CHOICE
116 DISPLAY SS-CLS
117 ACCEPT SS-MENU
118 EVALUATE TRUE
119 WHEN E-INCLUDE
120 WHEN E-INCLUDE
C:/GC-AWORK/customer.cob
switch to the debug output

```

The focus now is on the application window where a user action is needed.

In our sample, we type the "X" choice and Enter:

```

C:\GC-AWORK\customer.exe
TEST - MENU

1 - INCLUDE
2 - CONSULT
3 - UPDATE
4 - DELETE
X - EXIT

CHOICE: x
CHOOSE AN OPTION

```


DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	25	

Now the system automatic switch back to the debugger / animation window to continue debugging.
You can see that the ACCEPT statement has been executed, now you are on line **>119** :

```

CobGDB GnuCOBOL GDB Interpreter
101
102     PROCEDURE DIVISION.
103 001-START.
104     SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105     SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106     SET ENVIRONMENT 'ESCDELAY' TO '25'
107     *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108     *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" WS-STATUS
109     ACCEPT WS_NUMR FROM LINES
110     ACCEPT WS_NUMC012 FROM COLUMNS *> WS-STATUS
111     PERFORM 007-OPEN-FILES
112     PERFORM UNTIL E-EXIT
113         MOVE "MENU" TO WS-OP
114         MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
115         MOVE SPACES TO WS-CHOICE
116     DISPLAY SS-CLS
117     ACCEPT SS-MENU
118
>119     EVALUATE TRUE
120         WHEN E-INCLUDE
121             PERFORM 002-INCLUDE THRU 002-INCLUDE-END
C:/GC-AWORK/customer.cob
Debugging
  
```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	26	

2.10.Pop-up Variable windows

During a debugging session COBGDB shows variable content.

Blue frame and values : variables of executing cobol statement

Black frame and values : variables of last executed cobol statement.

Sample:

```

COBGDB GnuCOBOL GDB Interpreter
262 continue after 0.5 seconds
263
264 continue.
265 DisplayCoverEx. exit.
266
267 TetraminoDisplay.
268   perform varying wRow from 1 by 1 until wRow > 4
269   perform varying wCol from 1 by 1 until wCol > 4
270     compute wLinD = wBaseLin + wRow
271     compute wColD = wBaseCol + wCol
272     if wBlockEle (wRow wCol) = 'X'
B> 273       display wChar at line wLinD column wColD :BC0: blue :FC0: wCol
274     end-if
275   end-perform
276 end-perform
277 continue.
278 TetraminoDisplayEx. exit.
279
280 TetraminoSet
281   move spa
282   evaluate
C:/GC-AWORK/GC99TETRIS.COB
go

```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	27	

2.11.File Command

To show this command we use following sample:

```
cobgdb sample.cbl subsample.cbl subsubsample.cbl -x -lpdcurses
```

where sample.cbl is the main program; it calls

--> subsample.cbl; it calls

--> subsubsample.cbl

Source code is at <https://github.com/marcsosduma/cobgdb/tree/main/resources>.

This will create a single sample.exe executable.

This example shows that when you need to debug only subsample.cbl or only subsubsample.cbl you need to execute COBGDB with all three programs.

COBGDB sets the B breakpoint at first executable statement of first program "sample.cbl".

here use the R Run command to start the debugging session.

The screenshot shows a terminal window titled "Prompt dei comandi - q4" running the "COBGDB GnuCOBOL GDB Interpreter". The program code is displayed with line numbers 5 through 24. A breakpoint (B>) is set at line 20, which is highlighted. The code includes several data declarations (WS-NUMERIC, WS-SIGNED-DECIMAL, WS-UNSIGNED-DECIMAL, WS-ALPHABETIC, WS-ALPHANUMERIC, WS-GROUP), a procedure division, and a call to a subsample program. The status bar at the bottom indicates the current file is "C:/GC-AWORK/sample.cbl" and the state is "Debugging".

```

5      01 WS-NUMERIC PIC 9(2) VALUE 45.
6      01 WS-SIGNED-DECIMAL PIC S9(3)V9(2) VALUE -123.45.
7      01 WS-UNSIGNED-DECIMAL PIC 9(3)V9(2) VALUE 123.45.
8      01 WS-ALPHABETIC PIC A(6) VALUE 'ABCDEF'.
9      01 WS-ALPHANUMERIC PIC X(5) VALUE 'A121$'.
10     01 WS-GROUP.
11         05 WS-GROUP-NUMERIC PIC 9(2) VALUE 45.
12         05 WS-GROUP-SIGNED-DECIMAL PIC S9(3)V9(2) VALUE -123.45.
13         05 WS-GROUP-UNSIGNED-DECIMAL PIC 9(3)V9(2) VALUE 123.45.
14         05 WS-GROUP-ALPHABETIC PIC A(6) VALUE 'ABCDEF'.
15         05 WS-GROUP-ALPHANUMERIC PIC X(5) VALUE 'A121$'.
16     01 WS-CHECK PIC 9(2).
17         88 WS-CHECK-LITTLE VALUES ARE 50 THRU 99.
18         88 WS-CHECK-BIG VALUES ARE 00 THRU 49.
19     PROCEDURE DIVISION.
B> 20     DISPLAY "Hello";
21     CALL 'subsample' USING BY CONTENT WS-GROUP
22     END-CALL.
23     DISPLAY "world"
24     STOP RUN.
C:/GC-AWORK/sample.cbl
Debugging

```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	28	

Now you can type the **F File** command and you will have the "Source Files" window.
In this sample we select the second program in the list (subsample.cbl) and type Enter.

```

COBGDB GnuCOBOL GDB Interpreter
5 01 WS-NUMERIC PIC 9(2) VALUE 45.
6 01 WS-SIGNED-DECIMAL PIC S9(3)V9(2) VALUE -123.45.
7 01 WS-UNSIGNED-DECIMAL PIC 9(3)V9(2) VALUE 123.45.
8 01 WS-ALPHABETIC PIC A(6) VALUE 'ABCDEF'.
9 01 WS-ALPHANUMERIC PIC X(5) VALUE 'A121$'.
10
11 Source Files
12 C:/GC-AWORK/sample.cbl
13 C:/GC-AWORK/subsample.cbl
14 C:/GC-AWORK/subsubsample.cbl
15
16
17
18
19
20 B>
21
22 END-CALL.
23 DISPLAY "world"
24 STOP RUN.

C:/GC-AWORK/sample.cbl
Debugging

```

COBGDB shows the selected program source code where in this sample we type a B command at line 14.

```

COBGDB GnuCOBOL GDB Interpreter
1 IDENTIFICATION DIVISION.
2 PROGRAM-ID. subsample.
3 ENVIRONMENT DIVISION.
4 DATA DIVISION.
5 WORKING-STORAGE SECTION.
6 LINKAGE SECTION.
7 01 WS-GROUP.
8     05 WS-GROUP-NUMERIC PIC 9(2).
9     05 WS-GROUP-SIGNED-DECIMAL PIC S9(3)V9(2).
10    05 WS-GROUP-UNSIGNED-DECIMAL PIC 9(3)V9(2).
11    05 WS-GROUP-ALPHABETIC PIC A(6).
12    05 WS-GROUP-ALPHANUMERIC PIC X(5).
13 PROCEDURE DIVISION USING WS-GROUP.
14 B DISPLAY WS-GROUP-ALPHABETIC.
15 CALL 'subsubsample' USING BY CONTENT WS-GROUP
16 END-CALL.
17 END PROGRAM subsample.
18

C:/GC-AWORK/subsample.cbl
breakpoint

```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	29	

now we type the F command again, then select the "sample.cbl" program and press Enter

```

COBGDB GnuCOBOL GDB Interpreter
1 IDENTIFICATION DIVISION.
2 PROGRAM-ID. subsample.
3 ENVIRONMENT DIVISION.
4 DATA DIVISION.
5 WORKING-STORAGE SECTION.
6
7 Source Files
8 C:/GC-AWORK/subsample.cbl
9 C:/GC-AWORK/sample.cbl
10 C:/GC-AWORK/subsubsample.cbl
11
12
13
14
15
16
17
18
C:/GC-AWORK/subsample.cbl

```

now we are back to the sample.cbl program to continue the debugging session as we need.

```

COBGDB GnuCOBOL GDB Interpreter
5 01 WS-NUMERIC PIC 9(2) VALUE 45.
6 01 WS-SIGNED-DECIMAL PIC S9(3)V9(2) VALUE -123.45.
7 01 WS-UNSIGNED-DECIMAL PIC 9(3)V9(2) VALUE 123.45.
8 01 WS-ALPHABETIC PIC A(6) VALUE 'ABCDEF'.
9 01 WS-ALPHANUMERIC PIC X(5) VALUE 'A121$'.
10 01 WS-GROUP.
11     05 WS-GROUP-NUMERIC PIC 9(2) VALUE 45.
12     05 WS-GROUP-SIGNED-DECIMAL PIC S9(3)V9(2) VALUE -123.45.
13     05 WS-GROUP-UNSIGNED-DECIMAL PIC 9(3)V9(2) VALUE 123.45.
14     05 WS-GROUP-ALPHABETIC PIC A(6) VALUE 'ABCDEF'.
15     05 WS-GROUP-ALPHANUMERIC PIC X(5) VALUE 'A121$'.
16 01 WS-CHECK PIC 9(2).
17     88 WS-CHECK-LITTLE VALUES ARE 50 THRU 99.
18     88 WS-CHECK-BIG VALUES ARE 00 THRU 49.
19 PROCEDURE DIVISION.
20 DISPLAY "Hello".
21 CALL 'subsample' USING BY CONTENT WS-GROUP
22 END-CALL.
23 DISPLAY "world"
24 STOP RUN.
C:/GC-AWORK/sample.cbl
Debugging

```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	30	

2.12.Run Command

If you click the Run command during a debug session you will receive a confirmation request, because Yes will restart a new the debugging session from first Procedure Division executable statement:

```

COBGDB GnuCOBOL GDB Interpreter
100      10 COLUMN PLUS 2 TO WS-ERROR.
101
102      PROCEDURE DIVISION.
B 103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108
109      Message
110      Would you like to "Run" the program again ? (= Restart)
111      Yes No
112
113      PERFORM UNTIL E-EXIT
114          MOVE "MENU" TO WS-OP
115          MOVE "CHOOSE AN OPTION" TO WS-STATUS *> WS-STATUS
116          MOVE SPACES TO WS-CHOICE
117          DISPLAY SS-CLS
118          ACCEPT SS-MENU
119
120          EVALUATE TRUE
121              WHEN E-INCLUDE
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
2382
2383
2384
2385
2386
2387
2388
2389
2390
2391
2392
2393
2394
2395
2396
2397
2398
2399
2400
2401
2402
2403
2404
2405
2406
2407
2408
2409
2410
2411
2412
2413
2414
2415
2416
2417
2418
2419
2420
2421
2422
2423
2424
2425
2426
2427
2428
2429
2430
2431
2432
2433
2434
2435
2436
2437
2438
2439
2440
2441
2442
2443
2444
2445
2446
2447
2448
2449
2450
2451
2452
2453
2454
2455
2456
2457
2458
2459
2460
2461
2462
2463
2464
2465
2466
2467
2468
2469
2470
2471
2472
2473
2474
2475
2476
2477
2478
2479
2480
2481
2482
2483
2484
2485
2486
2487
2488
2489
2490
2491
2492
2493
2494
2495
2496
2497
2498
2499
2500
2501
2502
2503
2504
2505
2506
2507
2508
2509
2510
2511
2512
2513
2514
2515
2516
2517
2518
2519
2520
2521
2522
2523
2524
2525
2526
2527
2528
2529
2530
2531
2532
2533
2534
2535
2536
2537
2538
2539
2540
2541
2542
2543
2544
2545
2546
2547
2548
2549
2550
2551
2552
2553
2554
2555
2556
2557
2558
2559
2560
2561
2562
2563
2564
2565
2566
2567
2568
2569
2570
2571
2572
2573
2574
2575
2576
2577
2578
2579
2580
2581
2582
2583
2584
2585
2586
2587
2588
2589
2590
2591
2592
2593
2594
2595
2596
2597
2598
2599
2600
2601
2602
2603
2604
2605
2606
2607
2608
2609
2610
2611
2612
2613
2614
2615
2616
2617
2618
2619
2620
2621
2622
2623
2624
2625
2626
2627
2628
2629
2630
2631
2632
2633
2634
2635
2636
2637
2638
2639
2640
2641
2642
2643
2644
2645
2646
2647
264
```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	31	

2.13.Window Size command

When you start a debugging session, the screen size is 24 x 80 columns.

```

COBGDB GnuCOBOL GDB Interpreter
703 MOVE 1 TO U-CNT
704
705 *> EVALUATE WHITE
706 EVALUATE TRUE
707 WHEN PLAYER-TURN = 'W'
>708 PERFORM UNTIL INP-X-VAR-N = W-X-VAR(U-CNT)
709 AND INP-Y-VAR = W-Y-POS(U-CNT)
710 OR U-CNT >= 16
711
712 *> DEBUG
713 *> display INP-X-VAR-N at 2260
714 *> display W-X-VAR(U-CNT) at 2265
715 *> display INP-Y-VAR at
716 *> display W-Y-POS(U-CNT) at
717 *> DISPLAY U-CNT at
718
719 COORDINATES
720 "051508wKY041208wQY010308wRY082408wRY020608wHY072108wHY03090
721
722 ADD 1 TO U-CNT
723 END-PERFORM
724 EVALUATE TRUE
725
C:/GC-AWORK/GC99CHESS.COB
Debugging

```

Type **W** (Window Size) command to switch between two window size of the debugger : 24 x 80 or 34 x 132.

```

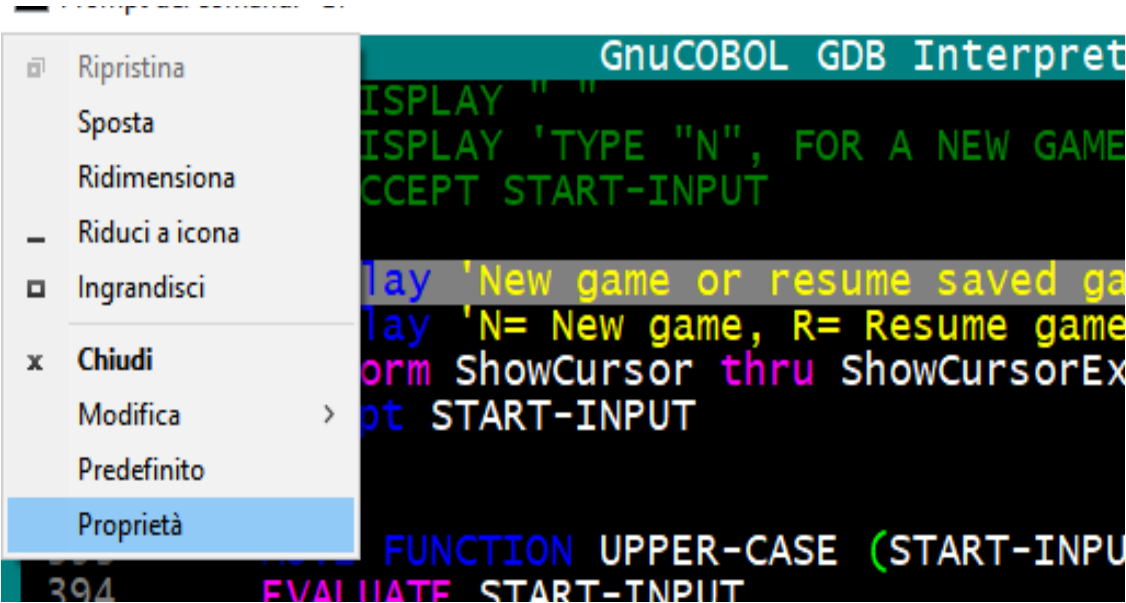
COBGDB GnuCOBOL GDB Interpreter
703 MOVE 1 TO U-CNT
704
705 *> EVALUATE WHITE
706 EVALUATE TRUE
707 WHEN PLAYER-TURN = 'W'
>708 PERFORM UNTIL INP-X-VAR-N = W-X-VAR(U-CNT)
709 AND INP-Y-VAR = W-Y-POS(U-CNT)
710 OR U-CNT >= 16
711
712 *> DEBUG
713 *> display INP-X-VAR-N at 2260
714 *> display W-X-VAR(U-CNT) at 2265
715 *> display INP-Y-VAR at 2260
716 *> display W-Y-POS(U-CNT) at 2265
717 *> DISPLAY U-CNT at 2460
718 *> accept omitted
719 *> END DEBUG
720
721 ADD 1 TO U-CNT
722 END-PERFORM
723 EVALUATE TRUE
724 WHEN W-PIECE(U-CNT) = '--'
725 SET NOT-VALID TO TRUE
726 *> DISPLAY "NO VALID PIECE HERE"
727 move "No valid piece here." to wMESSAGE
728 perform DisplayMessage thru DisplayMessageEx
729 PERFORM GA-ACCEPT-CORD
730 WHEN INP-Y-VAR = W-Y-POS(U-CNT)
731 *> DISPLAY "YOU HAVE SELECTED " W-PIECE(U-CNT) " IN POSITION " INP-X-VAR INP-Y-VAR
732 string "Selected " W-PIECE(U-CNT) " in pos. " INP-X-VAR INP-Y-VAR into wMESSAGE
733 perform DisplayMessage thru DisplayMessageEx
734
C:/GC-AWORK/GC99CHESS.COB
Debugging

```

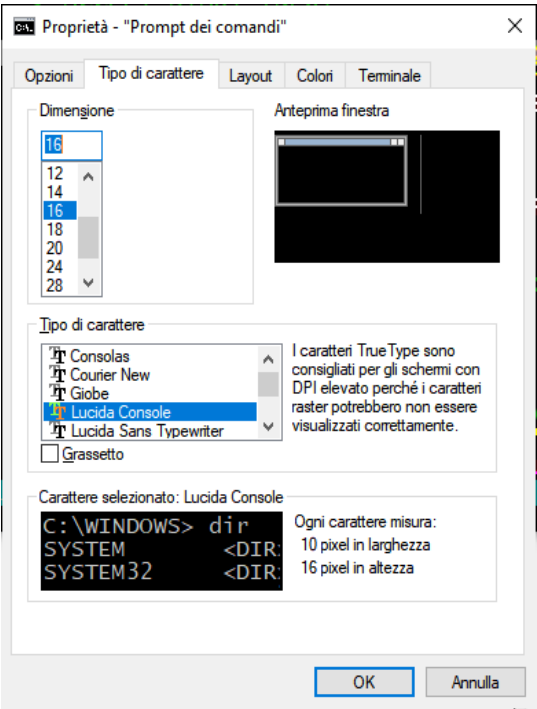
This can be very useful when you write GnuCOBOL code with the GnuCOBOL FREE FORMAT source option where each line of code can be longer than the classic 80 characters of GnuCOBOL FIXED FORMAT.

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	32	

Warning.
 Make sure the system font size is not too large otherwise the W command cannot display the 132 columns (your screen will "blink"). For a Windows environment use "Property" menu item to chek or change the Font size:



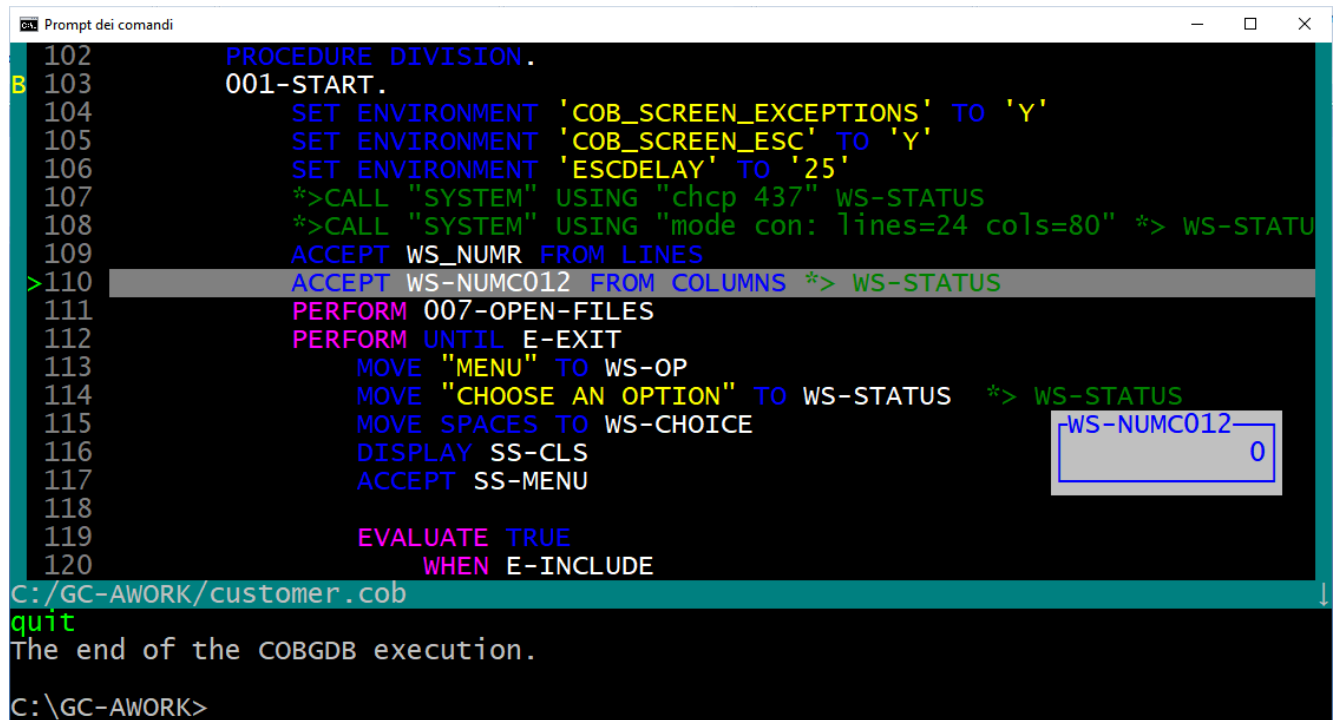
and then select a suitable font, for example Lucida Console of 16 might be adequate.



DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	33	

2.14.Quit Command

To close the debug session use the **Q** Quit command or left click with mouse the  button



```

102      PROCEDURE DIVISION.
103      001-START.
104          SET ENVIRONMENT 'COB_SCREEN_EXCEPTIONS' TO 'Y'
105          SET ENVIRONMENT 'COB_SCREEN_ESC' TO 'Y'
106          SET ENVIRONMENT 'ESCDELAY' TO '25'
107          *>CALL "SYSTEM" USING "chcp 437" WS-STATUS
108          *>CALL "SYSTEM" USING "mode con: lines=24 cols=80" *> WS-STATUS
109          ACCEPT WS_NUMR FROM LINES
110      >110  ACCEPT WS-NUMC012 FROM COLUMNS *> WS-STATUS
111          PERFORM 007-OPEN-FILES
112          PERFORM UNTIL E-EXIT
113              MOVE "MENU" TO WS-OP
114              MOVE "CHOOSE AN OPTION" TO WS-STATUS  *> WS-STATUS
115              MOVE SPACES TO WS-CHOICE
116              DISPLAY SS-CLS
117              ACCEPT SS-MENU
118
119              EVALUATE TRUE
120                  WHEN E-INCLUDE
  
```

C:/GC-AWORK/customer.cob
quit
The end of the COBGDB execution.
C:\GC-AWORK>

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	34	

2.15.Attach Command

Prerequisite: your programs are all compiled for debug (use the -g, a cobc compiler option), because you just compile everything for debug because that's no big overhead and you can attach and use system corefiles at any time. COBGDB needs the .cob, .c, .c.h, and .c.l.h files in order to perform debugging.

Scenario 1. Start a program from the command line.

You are "somewhere in the application" and find the current state to be strange or can reproduce the bug by something you do. Now you just open the debugger from a different terminal, **Attach** to the program, debug normally, then either kill or detach.

Scenario 2. The program has no terminal at all (like with [web]services, Java/C/HTML frontends, runs in the background, is started by other COBOL programs, ...) it is just "somehow started". Now you just open the debugger from a different terminal, **Attach** to the program, debug normally, then either kill or detach.

You can run GDB/GDBSERVER remotely using the A key. COBGDB will prompt you to provide the server and port in the format server:port or the PID of the application.

Example:

- `server:port = localhost:5555`
- `PID = 9112`

Command line:

- `cobgdb --connect localhost:5555 prog.cob`

```

CobGDB GnuCOBOL GDB Interpreter
327 *> *****
328 *>
329 *> *****
330 MAIN1 SECTION.
331 MAIN-LOOP.
332 perform InitialSettings thru InitialSettingsEx
333
334 PERFORM UNTIL wCRT-STATUS = K-ESCAPE
335     DISPLAY screen-menu
336     call 'TIMEDISP'
337     Attach - server:port or pid
338     Enter:
339
340     END-ACCEPT
341 end-perform
342
343 IF wCRT-STATUS = K-ESCAPE then
344     EXIT perform
345 END-IF
346 IF w-char2 NOT < 01 AND w-char2 NOT > 10 THEN
347     MOVE w-char2 TO w-Key
C:/GC-AWORK/MASTER.cob
Connecting

```

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	35	

3. Other Commands

3.1. Debugging a pre-compiled Program

You can use COBGDB to debug a previously generated executable file ex. **prog.exe**.
To do this, you must first compile the GnuCOBOL program **prog.cob** with these options:

```
cobc -g -fsource-location -ftraceall -v -O0 -x prog.cob prog2.cob ...
```

To start debugging without recompile the program, run cobgdb using the **--exe** directive as follows:

Windows:

```
cobgdb --exe prog.exe
```

Linux:

```
cobgdb --exe prog
```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	36	

3.2. Debugging sub programs

We use following sample programs `p0.cob` and `p1.cob` to show this feature:

```
>>SOURCE FORMAT IS FREE
IDENTIFICATION DIVISION.
PROGRAM-ID. p0.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 WS-NUMERIC                PIC 9(2)        VALUE 45.
01 WS-SIGNED-DECIMAL         PIC S9(3)V9(2)   VALUE -123.45.
01 WS-UNSIGNED-DECIMAL       PIC 9(3)V9(2)    VALUE 123.45.
01 WS-ALPHABETIC             PIC A(6)        VALUE 'ABCDEF'.
01 WS-ALPHANUMERIC           PIC X(5)        VALUE 'A121$'.
01 WS-GROUP.
   05 WS-GROUP-NUMERIC        PIC 9(2)        VALUE 45.
   05 WS-GROUP-SIGNED-DECIMAL PIC S9(3)V9(2)   VALUE -123.45.
   05 WS-GROUP-UNSIGNED-DECIMAL PIC 9(3)V9(2)    VALUE 123.45.
   05 WS-GROUP-ALPHABETIC     PIC A(6)        VALUE 'ABCDEF'.
   05 WS-GROUP-ALPHANUMERIC   PIC X(5)        VALUE 'A121$'.
   05 WS-MSG                  PIC X(70).
01 WS-CHECK PIC 9(2).
88 WS-CHECK-LITTLE VALUES ARE 50 THRU 99.
88 WS-CHECK-BIG    VALUES ARE 00 THRU 49.

PROCEDURE DIVISION.
  DISPLAY "PROGRAM P0"
  DISPLAY WS-GROUP-NUMERIC
  DISPLAY WS-GROUP-SIGNED-DECIMAL
  DISPLAY WS-GROUP-UNSIGNED-DECIMAL
  DISPLAY WS-GROUP-ALPHABETIC
  DISPLAY WS-GROUP-ALPHANUMERIC
  DISPLAY WS-MSG

  MOVE 'Program "P1" was called by "P0" ' TO WS-MSG
  CALL 'p1' USING BY CONTENT WS-GROUP END-CALL

  DISPLAY 'BACK TO "P0", received the message:'
  DISPLAY WS-GROUP-NUMERIC
  DISPLAY WS-GROUP-SIGNED-DECIMAL
  DISPLAY WS-GROUP-UNSIGNED-DECIMAL
  DISPLAY WS-GROUP-ALPHABETIC
  DISPLAY WS-GROUP-ALPHANUMERIC
  DISPLAY WS-MSG

  STOP RUN.
```

```
>>SOURCE FORMAT IS FREE
IDENTIFICATION DIVISION.
PROGRAM-ID. p1.
ENVIRONMENT DIVISION.
DATA DIVISION.
WORKING-STORAGE SECTION.
LINKAGE SECTION.
01 WS-GROUP.
   05 WS-GROUP-NUMERIC        PIC 9(2).
   05 WS-GROUP-SIGNED-DECIMAL PIC S9(3)V9(2).
   05 WS-GROUP-UNSIGNED-DECIMAL PIC 9(3)V9(2).
   05 WS-GROUP-ALPHABETIC     PIC A(6).
   05 WS-GROUP-ALPHANUMERIC   PIC X(5).
   05 WS-MSG                  PIC X(70).

PROCEDURE DIVISION USING WS-GROUP.
  DISPLAY "PROGRAM P1"
  DISPLAY "Message received: " WS-MSG
  DISPLAY 'BACK IN "P1" receiving the message:'
  DISPLAY WS-MSG
  GOBACK.
END PROGRAM p1.
```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	37	

It is possible to debug modules as long as you compile them in the same directory as your source code, using the following command:

```
cobc -g -fsource-location -ftraceall -v -00 -x p0.cob
cobc -g -fsource-location -ftraceall -v -00 -m p1.cob
```

In this example, p0.cob is the source code of the executable, and p1.cob is the source code of the module (.dll on Windows or .so on Linux). Debug Execution:

```
cobgdb --exe p0.exe
```

Example:

```

CobGDB GnuCOBOL GDB Interpreter
23  DISPLAY "PROGRAM P0"
24
25  DISPLAY WS-GROUP-NUMERIC
26  DISPLAY WS-GROUP-SIGNED-DECIMAL
27  DISPLAY WS-GROUP-UNSIGNED-DECIMAL
28  DISPLAY WS-GROUP-ALPHABETIC
29  DISPLAY WS-GROUP-ALPHANUMERIC
30  DISPLAY WS-MSG
31
32  MOVE 'Program "P1" was called by "P0"' TO WS-MSG
33  CALL 'p1' USING BY CONTENT WS-GROUP END-CALL
34
35  DISPLAY 'BACK TO "P0", received the message:'
36
37  DISPLAY WS-GROUP-NUMERIC
38  DISPLAY WS-GROUP-SIGNED-DECIMAL
39  DISPLAY WS-GROUP-UNSIGNED-DECIMAL
40  DISPLAY WS-GROUP-ALPHABETIC
41  DISPLAY WS-GROUP-ALPHANUMERIC
42  DISPLAY WS-MSG
43
D:/codigo_C/cobgdb2/p0.cob
Parsing file

```

Resulting in:

```

16  PROCEDURE DIVISION USING WS-GROUP.
>17  DISPLAY "PROGRAM P1"
18  DISPLAY "Message received: " WS-MSG
19  DISPLAY 'BACK IN "P1" receiving the message:'
20  DISPLAY WS-MSG
21  GOBACK.
D:/codigo_C/cobgdb2/p1.cob
Debugging

```

If p1.cob is already compiled as a module with:

```
cobc -g -fsource-location -ftraceall -v -00 -m p1.cob
```

the following command also works:

```
cobgdb p0.cob
```

DOCUMENT CODE	MODULE: xxxxxxxxxx	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	38	

When you are debugging separately compiled subprograms, as shown in the previous chapter, you can use the same **F File** command, which you should use when debugging subprograms that have all been compiled into the same executable. With **F File** command you can load the source code of a module and set a breakpoint.

3.3. COBGDB Version

Use command option: `cobgdb --version`

to display CobGDB version informations as follows:

```
CobGDB - GnuCobol GDB Interpreter - version 1.4.4
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
This CobGDB was configured as "MinGW32".
For bug reporting instructions, please see:
<https://github.com/marcsosduma/cobgdb>.
The end of the CobGDB execution.
```

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	39	

4. Document Change Log

CHANGE LOG
<p>Version 1 of 2023.12.12. First release</p> <p>. Version 2 of 2023.12.23. Step by Step sample of use is added Some minor changes</p> <p>. Version 3 of 2024.02.18. Restructured showing new cobgdb screens and features</p> <p>. Version 4 of 2024.04.01 and 20240403. Added EDIT subcommand at H Show Command when viewing the variable from a line of code Added cobgdb --version option</p> <p>. Version 5 of 2024.05.01. Added the W Window size command to change User interface screen size</p> <p>. Version 6 of 2024.05.05. Added the --exe option to debug precompiled program</p> <p>. Version 7 of 2025.04.24. Added the new D command and icon useful to enable or disable automatic display variables during debugging / animation</p> <p>. Version 8 of 2025.06.16. Added the new O (Focus) command and icon useful to switch the focus to application screen from debugging / animation screen</p> <p>. Version 9 of 2025.10.20. Added more documentation for the A (Attach) command.</p> <p>Version 10 of 2025.11.06. Added "Debugging sub programs" Chapter. Added F = File command also for separately compiled subprograms Added CTRL-F and CTR-L commands</p>

DOCUMENT CODE	MODULE: XXXXXXXXXX	USING COBGDB FOR GnuCOBOL	PAGE	GnuCOBOL
GC-901	GC-XXXXXX	Author: Eugenio Di Lorenzo	40	

Technical info

The screenshot shows a Windows desktop with a taskbar at the bottom. A command prompt window titled 'Prompt dei comandi - q3' is open, displaying the following text:

```

C:\GC-AWORK>q3
C:\GC-AWORK>cobgdb customer.cob -x -Tlista.txt -lpdcurses
Name: customer.exe
cobc -g -fsource-location -ftraceall -v -00 -x -x -Tlista.txt -lpdcurses C:/GC-AWORK/customer.cob
loading standard configuration file 'default.conf'
command line: cobc -g -fsource-location -ftraceall -v -00 -x -x -Tlista.txt -lpdcurses C:/GC-AWORK/customer.cob
preprocessing: C:/GC-AWORK/customer.cob -> customer.i
return status: 0
parsing: customer.i (C:/GC-AWORK/customer.cob)
return status: 0
translating: customer.i -> customer.c (C:/GC-AWORK/customer.cob)
executing: gcc -c -pipe -I/mingw32/include -I/mingw32/include -wno-unused -fsigned-char -wno-pointer-sign -ggdb3 -fasynchronous-unwind-tables -00 -o "C:\Users\DILO\AppData\Local\Temp\cob12088_0.o" "customer.c"
return status: 0
executing: gcc -w1,--export-all-symbols -w1,--enable-auto-import -w1,--enable-auto-image-base -o "customer.exe" "C:\Users\DILO\AppData\Local\Temp\cob12088_0.o" -L/mingw32/lib -lcob -l"pdcurses"

```

