學號：405410117

姓名：郭紘安　　Email：andy8766kuo@gmail.com

實驗名稱：Building a Cross Debugger for ARM Linux

實驗目的：

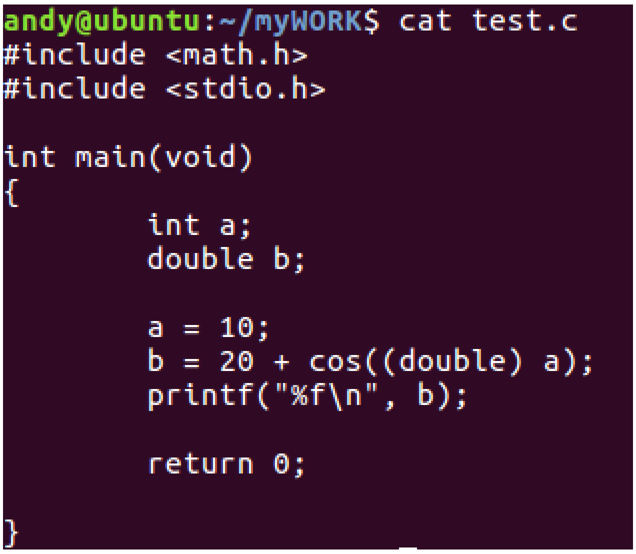
1. Introduction to GDB
2. Build a cross debugger
3. Testing and Practice: command line mode

實驗步驟：

- GDB

```
andy@ubuntu:~/myWORK$ tar -zxvfgdb-8.1.tar.gz
andy@ubuntu:~/myWORK$ export PATH="/home/andy/WORK/crossgcc2/bin:$PATH"
andy@ubuntu:~/myWORK$ mkdir build_gdb
andy@ubuntu:~/myWORK$ ../gdb-8.1/configure prefix=/home/andy/WORK/crossgcc2 --target=arm-linux-gnueabihf --enable-tui=yes
andy@ubuntu:~/myWORK$ make
andy@ubuntu:~/myWORK$ make install
```

create file test.c



```
andy@ubuntu:~/myWORK$ cat test.c
#include <math.h>
#include <stdio.h>

int main(void)
{
        int a;
        double b;

        a = 10;
        b = 20 + cos((double) a);
        printf("%f\n", b);

        return 0;

}
```

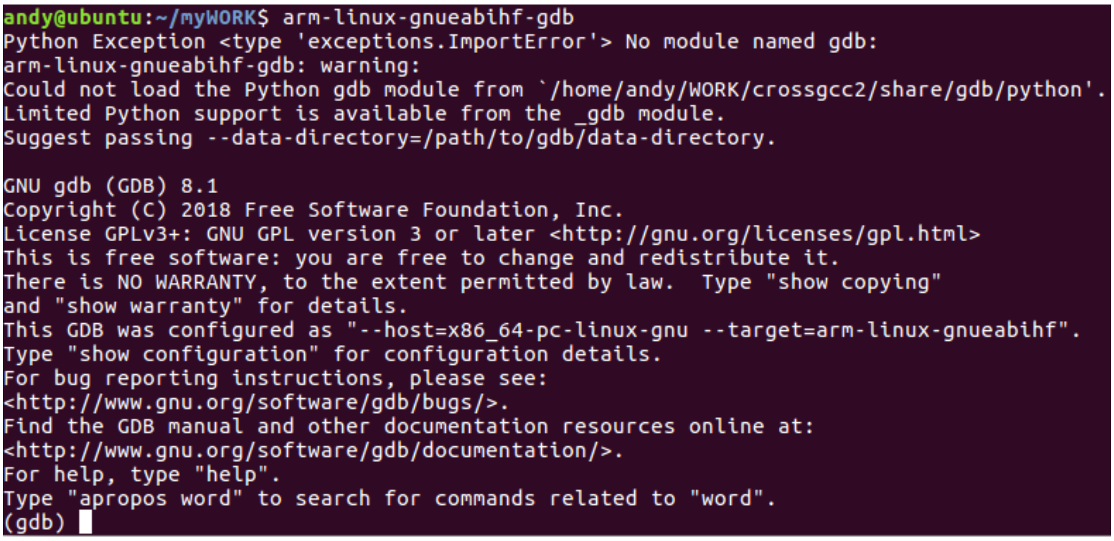
學號：405410117

姓名：郭紘安　　　　　　　　　　Email：andy8766kuo@gmail.com

andy@ubuntu:~/myWORK$ arm-linux-gnueabihf-gcc -static -g test.c -lm -o test.exe

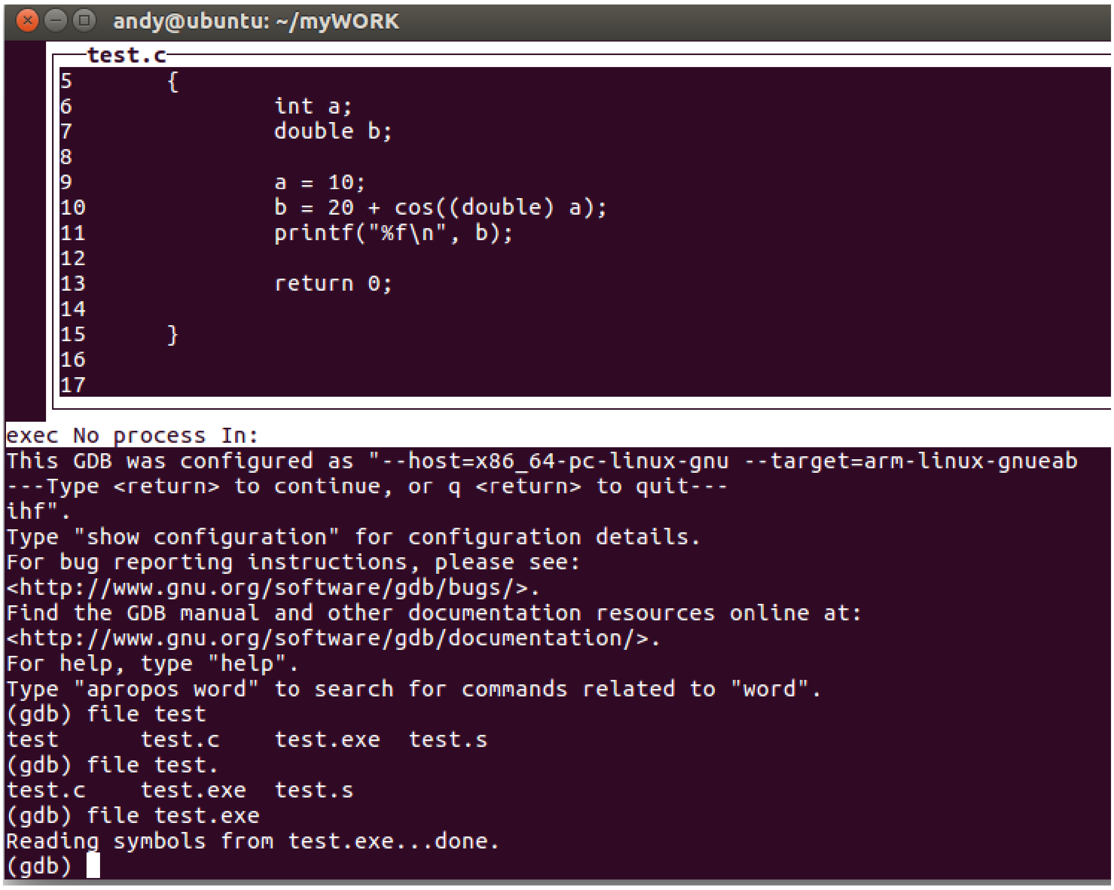
(Standard C Library supports the Base System math routines, as defined in Volume 1. The `cc` option `-lm` is used to search this library.)

andy@ubuntu:~/myWORK$ arm-linux-gnueabihf-gdb



```
andy@ubuntu:~/myWORK$ arm-linux-gnueabihf-gdb
Python Exception <type 'exceptions.ImportError'> No module named gdb:
arm-linux-gnueabihf-gdb: warning:
Could not load the Python gdb module from `/home/andy/WORK/crossgcc2/share/gdb/python'.
Limited Python support is available from the _gdb module.
Suggest passing --data-directory=/path/to/gdb/data-directory.

GNU gdb (GDB) 8.1
Copyright (C) 2018 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.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "--host=x86_64-pc-linux-gnu --target=arm-linux-gnueabihf".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb) 
```

andy@ubuntu:~/myWORK$ arm-linux-gnueabihf-gdb -tui



```
test.c
        {
                int a;
                double b;

                a = 10;
                b = 20 + cos((double) a);
                printf("%f\n", b);

                return 0;

        }


exec No process In:
This GDB was configured as "--host=x86_64-pc-linux-gnu --target=arm-linux-gnueab
---Type <return> to continue, or q <return> to quit---
ihf".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word".
(gdb) file test
test      test.c    test.exe   test.s
(gdb) file test.
test.c    test.exe   test.s
(gdb) file test.exe
Reading symbols from test.exe...done.
(gdb) 
```

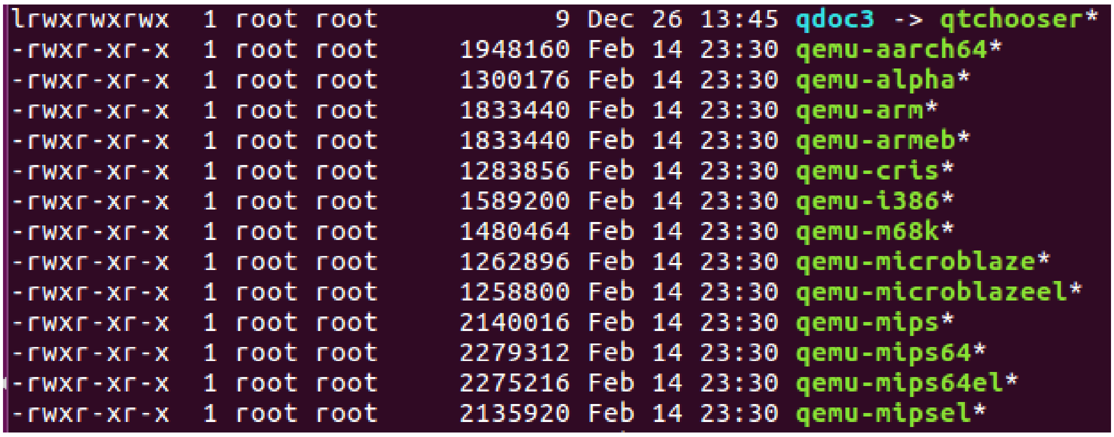
學號：405410117

姓名：郭紘安　　　　　　　　　Email：andy8766kuo@gmail.com

- Qemu

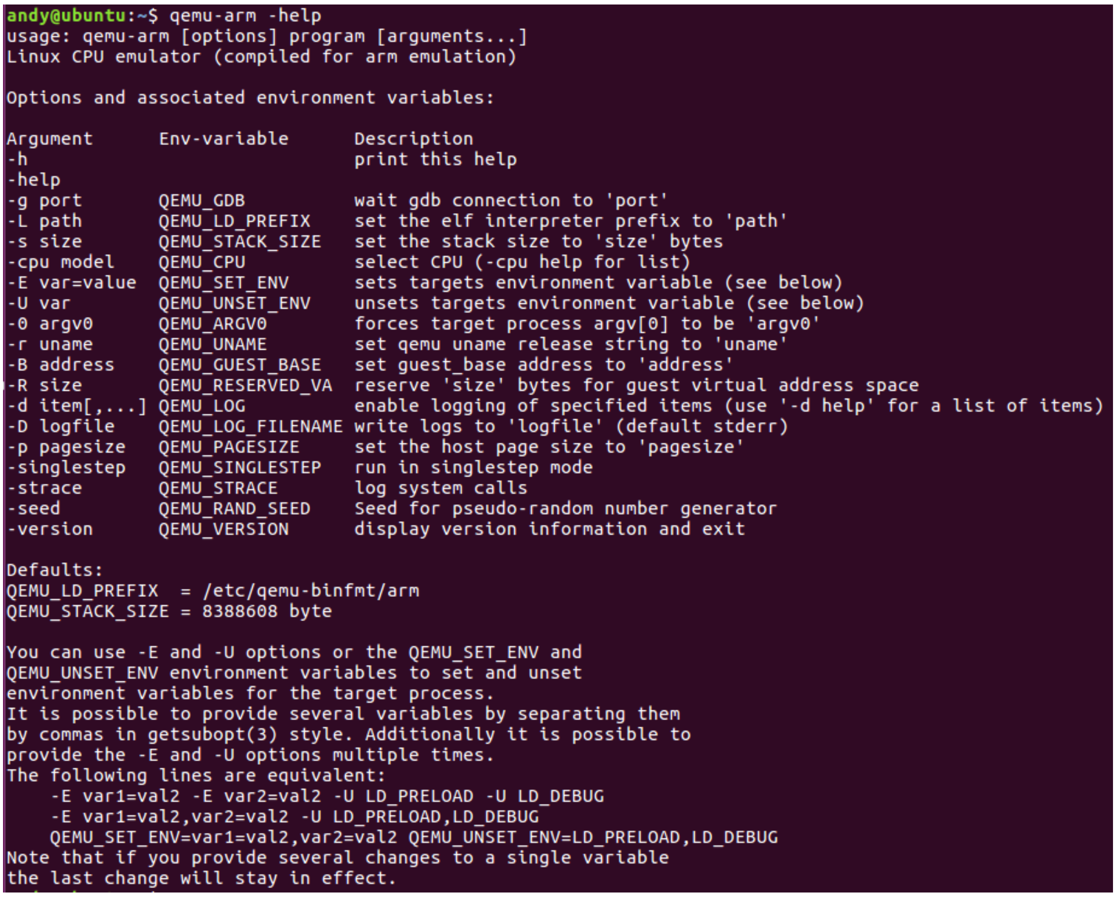
andy@ubuntu:~/myWORK$ apt-get install qemu-user

安裝成功，在 usr/bin 目錄下有執行檔 qemu-arm



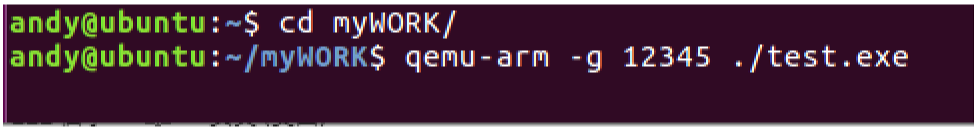
```
lrwxrwxrwx  1 root root          9 Dec 26 13:45 qdoc3 -> qtchooser*
-rwxr-xr-x  1 root root    1948160 Feb 14 23:30 qemu-aarch64*
-rwxr-xr-x  1 root root    1300176 Feb 14 23:30 qemu-alpha*
-rwxr-xr-x  1 root root    1833440 Feb 14 23:30 qemu-arm*
-rwxr-xr-x  1 root root    1833440 Feb 14 23:30 qemu-armeb*
-rwxr-xr-x  1 root root    1283856 Feb 14 23:30 qemu-cris*
-rwxr-xr-x  1 root root    1589200 Feb 14 23:30 qemu-i386*
-rwxr-xr-x  1 root root    1480464 Feb 14 23:30 qemu-m68k*
-rwxr-xr-x  1 root root    1262896 Feb 14 23:30 qemu-microblaze*
-rwxr-xr-x  1 root root    1258800 Feb 14 23:30 qemu-microblazeel*
-rwxr-xr-x  1 root root    2140016 Feb 14 23:30 qemu-mips*
-rwxr-xr-x  1 root root    2279312 Feb 14 23:30 qemu-mips64*
-rwxr-xr-x  1 root root    2275216 Feb 14 23:30 qemu-mips64el*
-rwxr-xr-x  1 root root    2135920 Feb 14 23:30 qemu-mipsel*
```

andy@ubuntu:~$ qemu-arm -help



```
andy@ubuntu:~$ qemu-arm -help
usage: qemu-arm [options] program [arguments...]
Linux CPU emulator (compiled for arm emulation)

Options and associated environment variables:

Argument      Env-variable      Description
-h                              print this help
-help
-g port       QEMU_GDB          wait gdb connection to 'port'
-L path       QEMU_LD_PREFIX    set the elf interpreter prefix to 'path'
-s size       QEMU_STACK_SIZE   set the stack size to 'size' bytes
-cpu model    QEMU_CPU          select CPU (-cpu help for list)
-E var=value  QEMU_SET_ENV      sets targets environment variable (see below)
-U var        QEMU_UNSET_ENV    unsets targets environment variable (see below)
-0 argv0      QEMU_ARGV0        forces target process argv[0] to be 'argv0'
-r uname      QEMU_UNAME        set qemu uname release string to 'uname'
-B address    QEMU_GUEST_BASE   set guest_base address to 'address'
-R size       QEMU_RESERVED_VA  reserve 'size' bytes for guest virtual address space
-d item[,...] QEMU_LOG          enable logging of specified items (use '-d help' for a list of items)
-D logfile    QEMU_LOG_FILENAME write logs to 'logfile' (default stderr)
-p pagesize   QEMU_PAGESIZE     set the host page size to 'pagesize'
-singlestep   QEMU_SINGLESTEP   run in singlestep mode
-strace       QEMU_STRACE       log system calls
-seed         QEMU_RAND_SEED    Seed for pseudo-random number generator
-version      QEMU_VERSION      display version information and exit

Defaults:
QEMU_LD_PREFIX  = /etc/qemu-binfmt/arm
QEMU_STACK_SIZE = 8388608 byte

You can use -E and -U options or the QEMU_SET_ENV and
QEMU_UNSET_ENV environment variables to set and unset
environment variables for the target process.
It is possible to provide several variables by separating them
by commas in getsubopt(3) style. Additionally it is possible to
provide the -E and -U options multiple times.
The following lines are equivalent:
    -E var1=val2 -E var2=val2 -U LD_PRELOAD -U LD_DEBUG
    -E var1=val2,var2=val2 -U LD_PRELOAD,LD_DEBUG
    QEMU_SET_ENV=var1=val2,var2=val2 QEMU_UNSET_ENV=LD_PRELOAD,LD_DEBUG
Note that if you provide several changes to a single variable
the last change will stay in effect.
```

andy@ubuntu:~/myWORK$ qemu-arm -g 12345 ./test.exe



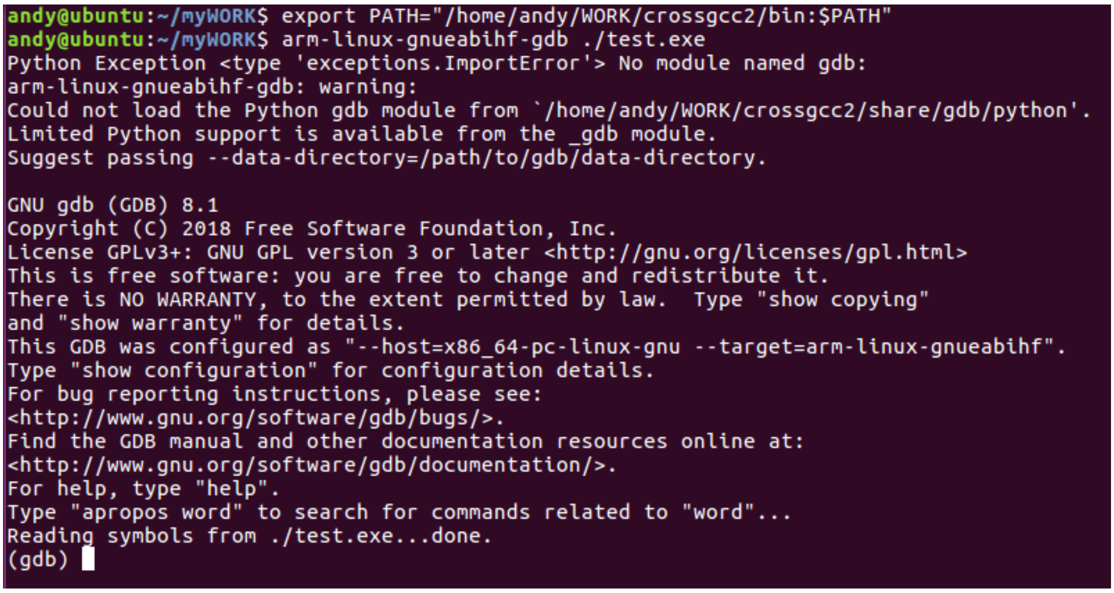
```
andy@ubuntu:~$ cd myWORK/
andy@ubuntu:~/myWORK$ qemu-arm -g 12345 ./test.exe
```

學號：405410117

姓名：郭紘安　　　　　　　　Email：andy8766kuo@gmail.com

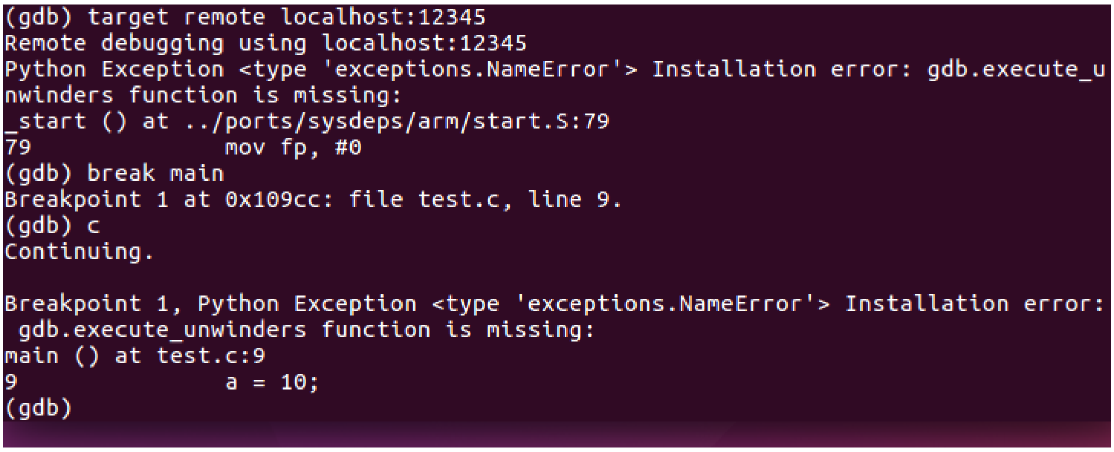
開啟另一個終端機視窗

```
andy@ubuntu:~/myWORK$ export
PATH="/home/andy/WORK/crossgcc2/bin:$PATH"
andy@ubuntu:~/myWORK$ arm-linux-gnueabihf-gdb ./test.exe
```



```
andy@ubuntu:~/myWORK$ export PATH="/home/andy/WORK/crossgcc2/bin:$PATH"
andy@ubuntu:~/myWORK$ arm-linux-gnueabihf-gdb ./test.exe
Python Exception <type 'exceptions.ImportError'> No module named gdb:
arm-linux-gnueabihf-gdb: warning:
Could not load the Python gdb module from `/home/andy/WORK/crossgcc2/share/gdb/python'.
Limited Python support is available from the _gdb module.
Suggest passing --data-directory=/path/to/gdb/data-directory.

GNU gdb (GDB) 8.1
Copyright (C) 2018 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.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "--host=x86_64-pc-linux-gnu --target=arm-linux-gnueabihf".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<http://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
<http://www.gnu.org/software/gdb/documentation/>.
For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./test.exe...done.
(gdb) 
```

```
(gdb) target remote localhost:12345
(gdb) break main
(gdb) c
```



```
(gdb) target remote localhost:12345
Remote debugging using localhost:12345
Python Exception <type 'exceptions.NameError'> Installation error: gdb.execute_u
nwinders function is missing:
_start () at ../ports/sysdeps/arm/start.S:79
79                  mov fp, #0
(gdb) break main
Breakpoint 1 at 0x109cc: file test.c, line 9.
(gdb) c
Continuing.

Breakpoint 1, Python Exception <type 'exceptions.NameError'> Installation error:
 gdb.execute_unwinders function is missing:
main () at test.c:9
9                   a = 10;
(gdb)
```

學號：405410117

姓名：郭紘安　　　　　　　　　　Email：andy8766kuo@gmail.com

問題與討論：

- **What is “GDB server”?**

  GDB server是GDB stub的一種具體實踐，gdbserver 是Linux上的實際的程式，跑在Target端，透過TCP 或 serial port跟host端的gdb通聯

- **What is “GDB stub”?**

  1. **GDB stub is an agent of the GDB host**
  2. **Monitor and control the debugged program**
  3. **Communicate with GDB host**
  4. **Be compiled and linked with the debugged**

心得：

這次的實驗讓我了解 GDB 的使用方法，並且也了解如何使用 QEMU 來 Remote Debugging，也了解了 GDB stub 和 GDB host 之間的關聯性。