

HƯỚNG DẪN DEBUG CHƯƠNG TRÌNH DÙNG GDB

Mục tiêu: Dùng gdb để debug 1 chương trình C trong Linux

Nội dung:

1. Các lệnh

STT	Lệnh debug (Linux)	Lệnh debug (VC++)
1	<code>gdb</code>	
2	<code>run</code>	Ctrl + F5
3	<code>Break</code>	F9
4	<code>Next</code>	F10
5	<code>Step</code>	F11
6	<code>Continue</code>	F5
7	<code>Print</code>	Xuất giá trị biến
8	<code>Delete</code>	Xoá break point
9	<code>Quit</code>	Shift + F5

Giả sử có chương trình test.c như sau:

```
1  #include <stdio.h>
2  void test()
3  {
4      printf("Debug dùng gdb\n");
5  }
6  int main()
7  {
8      int a[5];
9      int i;
10     test();
11     for(i=0; i<5; i++)
12     {
13         a[i] = i;
14         printf("%d\n", a[i]);
15     }
16     printf("Kết thúc chương trình test\n");
17     return 0;
18 }
```

Chương trình cần được biên dịch trước khi debug. Ví dụ biên dịch chương trình test.c thành file thực thi test và có thể debug bằng gdb được, ta dùng lệnh sau:

`gcc -g test.c -o test`

Chuyển sang mode gdb để debug file thực thi test bằng lệnh

`gdb <tên file thực thi>`

Ví dụ: `gdb test`

Giải sử chương trình kịch bản quá trình debug như sau

Vào chế độ debug chương trình	<code>root# gdb test</code>
Đặt breakpoint tại dòng code thứ 10, 13 và 16	<code>(gdb) break test.c : 10</code> <code>(gdb) break test.c : 13</code> <code>(gdb) break test.c : 16</code>
Bắt đầu debug	<code>(gdb) run</code>
Nhảy qua thủ tục <code>test()</code> , chỉ lấy kết quả trả về của <code>test()</code>	<code>(gdb) step</code>
Tiếp tục chạy debug đến breakpoint ở dòng 13	<code>(gdb) continue</code>
Tiếp tục đến lệnh tiếp theo	<code>(gdb) next</code>
Xem giá trị <code>i</code>	<code>(gdb) print i</code>
Xem giá trị <code>a[0]</code>	<code>(gdb) print a[0]</code>
...	...
Tiếp tục chạy	<code>(gdb) continue</code>
Đến breakpoint trên dòng 16, next để qua lệnh này, thông báo kết thúc chương trình	<code>(gdb) next</code>
Thoát khỏi chế độ debug	<code>(gdb) quit</code>

--Hết--