

CS343-Assignment - 0

Vemulapati sai lakshmi swarupa

Roll Number : 200123070

Assignment0A -

1.

```
// Simple inline assembly example
//
#include <stdio.h>
int
main(int argc, char **argv)
{
    int x = 1;
    printf("Hello x = %d\n", x);

    // in-line assembly code to increment
    // the value of x by 1
    __asm__ ( "addl %%ebx, %%eax;"
             : "=a" (x)
             : "a" (x), "b" (1) );

    printf("Hello x = %d after increment\n", x);

    if(x == 2){
        printf("OK\n");
    }
    else{
        printf("ERROR\n");
    }
}
```

We wrote this line of code to increment x by 1

2. I have used “si” instruction in gdb which executes a single machine instruction.

```
0x0000ffff in ?? ()
+ symbol-file kernel
warning: A handler for the OS ABI "GNU/Linux" is not built into this configuration
of GDB. Attempting to continue with the default i8086 settings.

(gdb) si
[f000:e05b] 0xfe05b: cmpw $0xffc8,%cs:(%esi)
0x0000e05b in ?? ()
(gdb) si
[f000:e062] 0xfe062: jne 0xd241d416
0x0000e062 in ?? ()
(gdb) si
[f000:e066] 0xfe066: xor %edx,%edx
0x0000e066 in ?? ()
(gdb) si
[f000:e068] 0xfe068: mov %edx,%ss
0x0000e068 in ?? ()
(gdb) si
[f000:e06a] 0xfe06a: mov $0x7000,%sp
0x0000e06a in ?? ()
(gdb) si
[f000:e070] 0xfe070: mov $0x2d4e,%dx
0x0000e070 in ?? ()
(gdb) si
[f000:e076] 0xfe076: jmp 0x5575ff02
0x0000e076 in ?? ()
(gdb) si
[f000:ff00] 0xffff00: cli
0x0000ff00 in ?? ()
(gdb) si
[f000:ff01] 0xffff01: cld
0x0000ff01 in ?? ()
(gdb) si
[f000:ff02] 0xffff02: mov %ax,%cx
0x0000ff02 in ?? ()
(gdb) si
[f000:ff05] 0xffff05: mov $0x8f,%ax
0x0000ff05 in ?? ()
(gdb) si
[f000:ff0b] 0xffff0b: out %al,$0x70
0x0000ff0b in ?? ()
(gdb) si
[f000:ff0d] 0xffff0d: in $0x71,%al
0x0000ff0d in ?? ()
```

Assignment-0B:

Code is submitted as a separate file.

We have made changes in the following files:

```
int
sys_draw(void)
{
    void* buf;
    uint size;

    argptr(0, (void*)&buf, sizeof(buf));
    argptr(1, (void*)&size, sizeof(size));

    char text[] = "db      db d88888b db      db      .d88b.
                  88      88'      88      88      .8P  Y8.
                  88000088 8800000 88      88      88      88
                  88~~~~88 88~~~~~ 88      88      88      88
                  88      88.      88b000. 88b000. `8b  d8'
                  YP      YP Y88888P Y88888P Y88888P `Y88P' ";

    if(sizeof(text)>size)
        return -1;

    strncpy((char *)buf, text, size);
}
```

Forwarded

```
int
sys_draw(void)
{
    void* buf;
    uint size;
```

1) Makefile

2) syscall.c (added [SYS_draw] sys_draw &&
extern int sys_draw(void);)

3) syscall.h (added#define SYS_draw 22)

4) sysfile.c (added main function call)

5) user.h (added prototype of function int
draw(void* ,uint);)

6) usys.S (addedSYSCALL(draw))

7) Drawtest.c (added caller function)

- i. Run make clean , make, make qemu and then type Draw test in terminal. Output is displayed below

```
"db      db d88888b db      db      .d88b.
88      88 88'      88      88      .8P  Y8.
88ooo88 88ooooo 88      88      88      88
88~~~88 88~~~~~ 88      88      88      88
88      88 88.      88booo. 88booo. `8b  d8'
YP      YP Y88888P Y88888P Y88888P `Y88P' ";
```