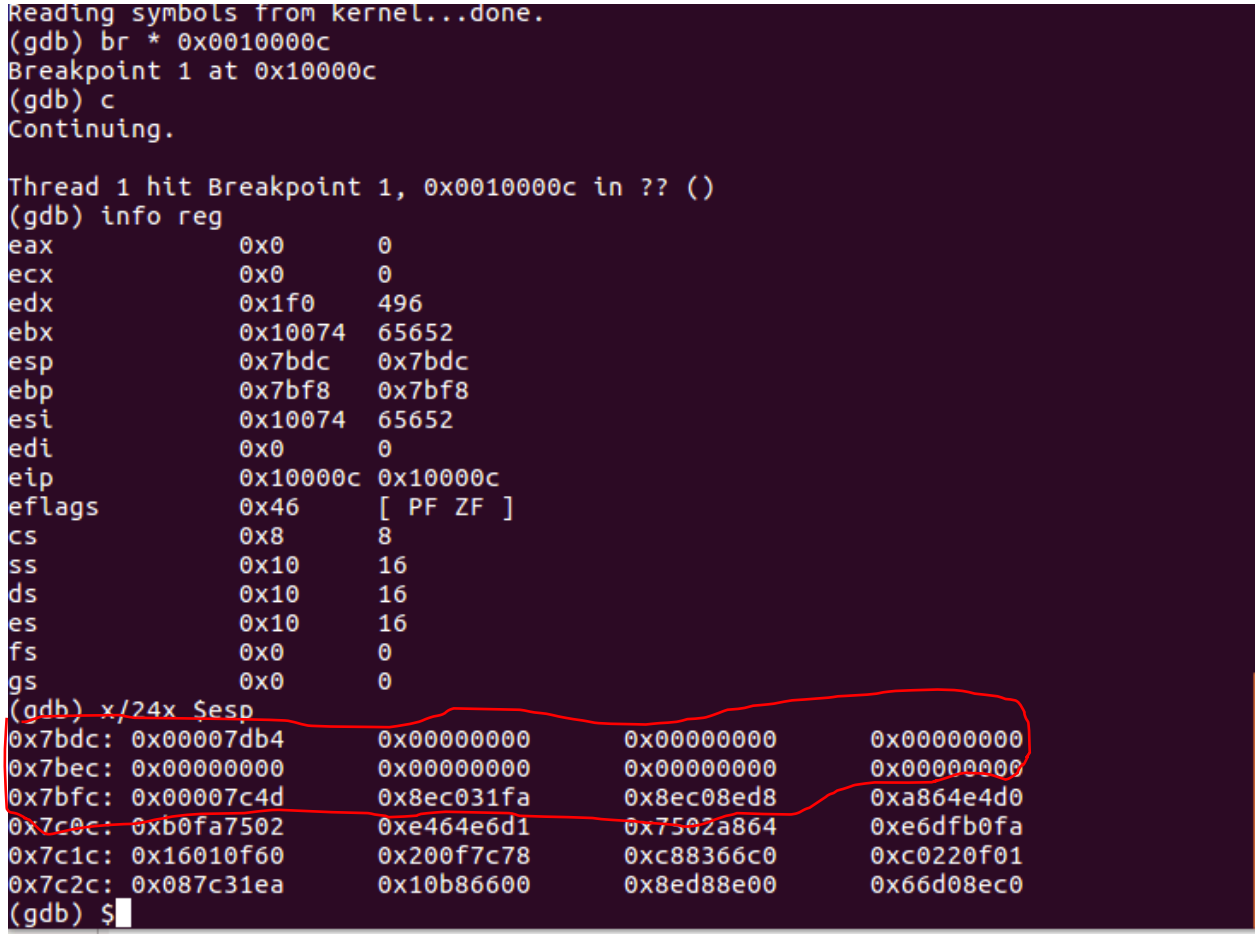
CS238P Operation System
Assignment 2
Name: Yuchun Liu (71562305)

Question 1:

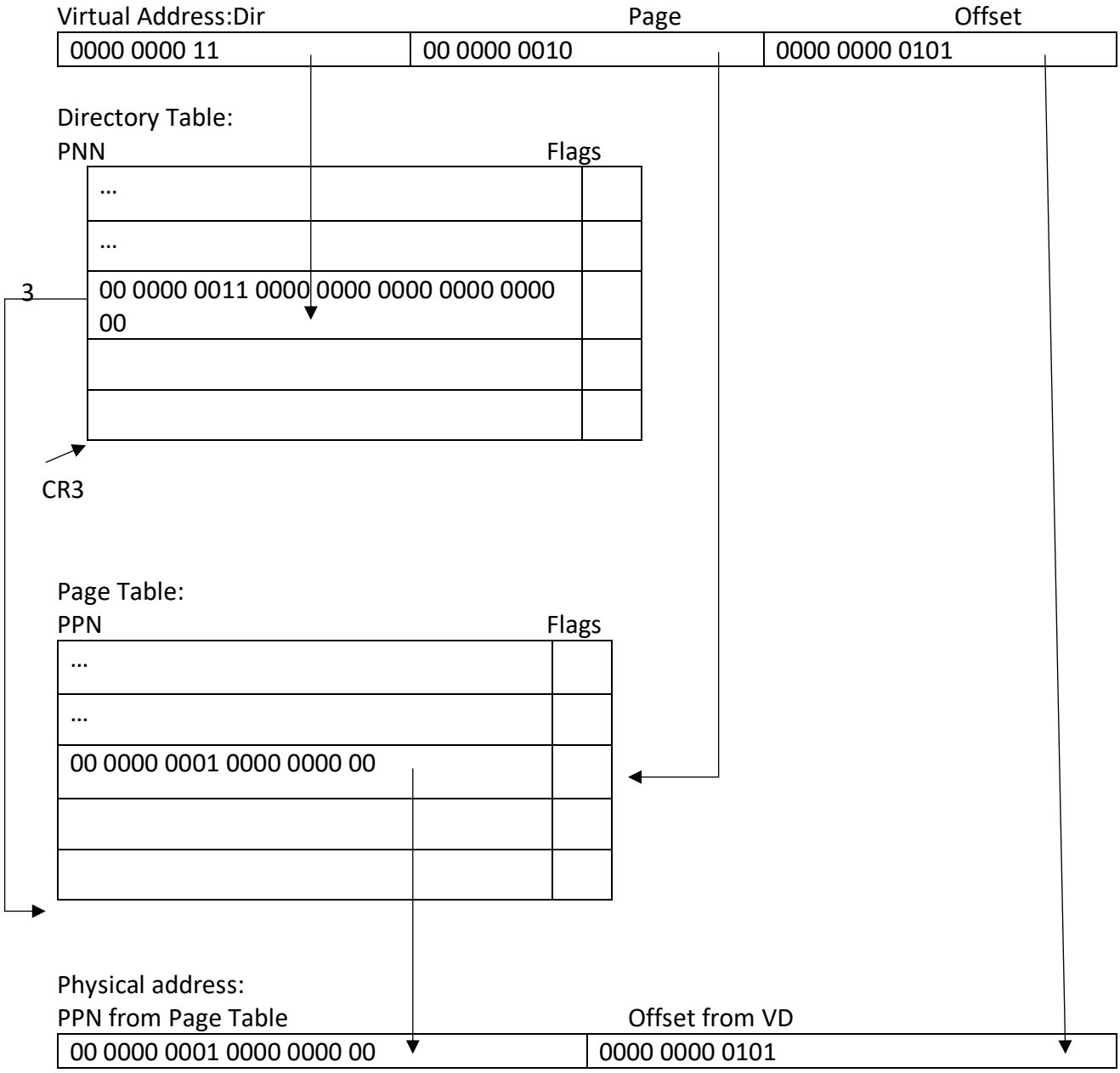


The element on the stack is labelled in red.
When the system is booted, it uses the segmentation. The logical address is the physical address and $esp, which points to the top of stack is set at address 0x7bdc. The stack will push towards the lower address.

When the bootmain.c calls the bootmain(). The calling instruction is pushed stack which indicates that the instruction is at address 0x00007c4d. Then the program saves part of memory for local variables and the address pointed by the $eph is push on the stack.

Finally, bootloader about finish its job, it will make another function call and push that function address on to the stack as well. So, we have total of 11 elements on our stack.

Question 2:


Virtual Address:Dir
Page
Offset
0000 0000 11
00 0000 0010
0000 0000 0101
Directory Table:
PNN
Flags
…
…
3
00 0000 0011 0000 0000 0000 0000 0000 00
CR3
Page Table:
PPN
Flags
…
…
00 0000 0001 0000 0000 00
Physical address:
PPN from Page Table
Offset from VD
00 0000 0001 0000 0000 00
0000 0000 0101

Question 3.

```
*** Now run 'gdb'.
qemu-system-i386 -nographic -drive file=fs.img,index=1,media=disk,format=raw -dr
ive file=xv6.img,index=0,media=disk,format=raw -smp 2 -m 512  -S -gdb tcp::26001
QEMU 2.3.0 monitor - type 'help' for more information
(qemu) info pg
VPN range     Entry         Flags        Physical page
[80000-803ff]  PDE[200]     ----A--UWP
  [80000-800ff]  PTE[000-0ff] --------WP 00000-000ff
  [80100-80101]  PTE[100-101] ---------P 00100-00101
  [80102-80102]  PTE[102]     ----A----P 00102
  [80103-80105]  PTE[103-105] ---------P 00103-00105
  [80106-80106]  PTE[106]     ----A----P 00106
  [80107-80107]  PTE[107]     ---------P 00107
  [80108-8010a]  PTE[108-10a] --------WP 00108-0010a
  [8010b-8010b]  PTE[10b]     ----A---WP 0010b
  [8010c-803ff]  PTE[10c-3ff] --------WP 0010c-003ff
[80400-8dfff]  PDE[201-237] -------UWP
  [80400-8dfff]  PTE[000-3ff] --------WP 00400-0dfff
[fe000-fffff]  PDE[3f8-3ff] -------UWP
  [fe000-fffff]  PTE[000-3ff] --------WP fe000-fffff
(qemu)
```

In order to understand the state of the page table, we have to look into the code. In the code, one function is particular important – kvmalloc(): this function is based on another function which is setupkvm: set up the kernel memory. This function distributed the virtual memory is such way:
0 – KERNBASE(KB): user memory (text, data, stack, heap)
KB –KB + EXTMEM: I/O devices
KB + EXTMEM – KB + STOP : w/o write permission (kernel data)
KB + STOP – KB + PHYSTOP : kernel data
DEVSAPCE

Therefore, we can see:
VPN range
[80000 – 800ff]: for BIOS

[80100 – 80101]: kernel text
[80102 – 80102]: kernel text
[80103 – 80105]: kernel text
[80106 – 80106]: kernel text
[80107 – 80107]: kernel text
[80108 – 8010a]: kernel data
[8010b – 8010b]: kernel data
[8010c – 803ff]: kernel data
[80400 – 8dfff]: I/O devices
[fe000 – fffff]: other devices