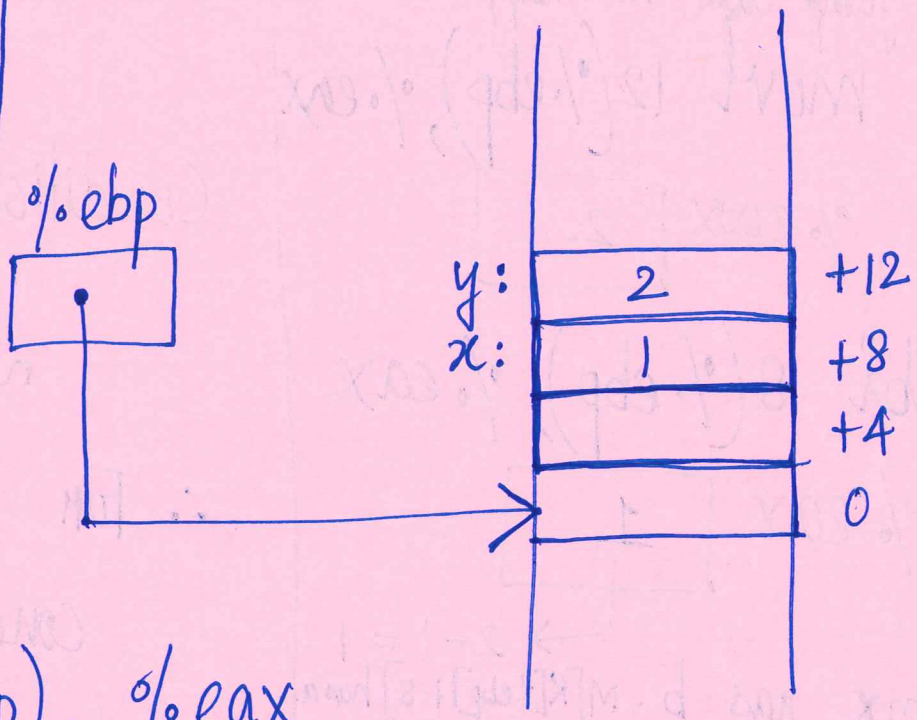
Feb 26, 2016

Lecture 16 - Jumps, conditionals, and loops

absdiff.c

```
if (a < b)
    return b-a;
else return a-b;
```



absdiff:

movl 8(%ebp), %eax

%eax | 1 |

movl 12(%ebp), %edx

%edx | 2 |

cmpl %edx, %eax

update flags based on a-b

%eax | 1 | %edx | 2 |

ZF	0	SF	1	OF	0

jge .L2

→ jump does not happen here.

movl 12(%ebp), %eax

%eax | 2 |

subl 8(%ebp), %eax

%eax | 1 | → 2-1 = 1

%eax has b. M[R[%ebp]+8] has a.

∴ b-a is stored in %eax.

jmp .L3

Condition for jl:

$SF \wedge OF$

Condition for jge:

$\sim (SF \wedge OF)$

∴ For our example,

cond = $\sim(1 \wedge 0)$

$= \sim 1 = 0$

(∵ only 1 bit)

.L2

movl 8(%ebp), %eax

subl 12(%ebp), %eax

%eax | 1 |

→ would be executed if $a > b$.

.L3

popl %ebp

ret

<main>:

8048410: e8 d9 ff ff ff call 80483ec <absdiff>

8048413:

little endian.

<absdiff>:

80483ec: 55 push %ebp

To calculate the address of the function absdiff:

%eip = 0x08048413 (Address in PC or next address after call).

relative jump offset = 0xFFFFFFD9

0x080483EC

Actually, the absdiff function is at a smaller address than main.

0xFFFFFFD9 is a negative number in 2's complement form.

Procedures

procedure call - ① passing data
eg. procedure parameters, return values.
② transfering control.

Stack Frame