

CS2318

Midterm Solutions

10/27/16

C++ --C

ii) $\$w$ $\$Rd$, $((-\$Rs))$

$\$Rs \leftarrow \$Rs - 1 \text{ word offset}$

$\text{Mem}[\text{Mem}[\$Rs]] \leftarrow \Rd

ii) $\$w$ $\$Rd$, $((+\$Rs))$

$()$ pointer ~~$\pm \$Rd \pm$~~

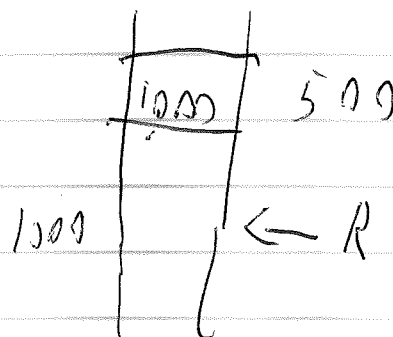
$(\pm \$R \pm)$ pre inc pre post inc
 dec dec

subi $\$Rs, \$Rs, 4$

$\$Rs = 504 \Rightarrow 500$

lw $\$at, 0(\$Rs)$

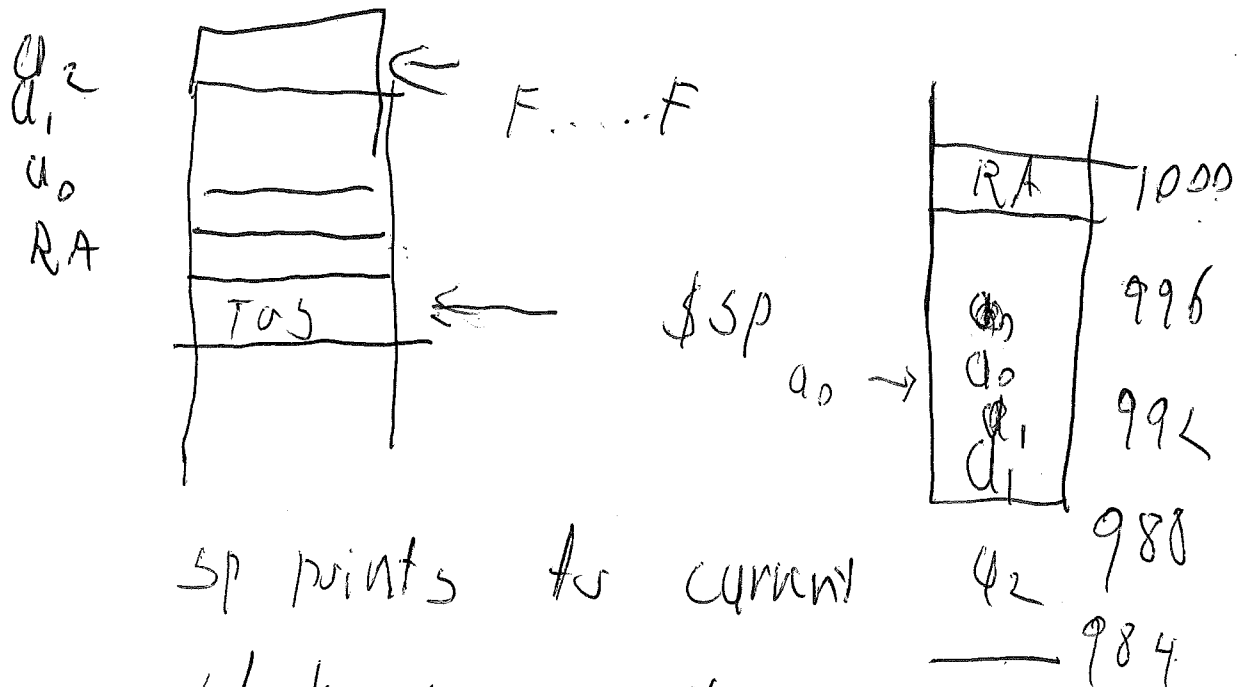
sw $\$Rd, 0(\$at)$



2)

C-function Target

pushing ^{\$a0}\$Ra, \$a1, \$a2 a0, a1, a2



~~addi \$sp, \$sp, 16~~ addi here

sw \$Ra, 4(\$sp)

sw \$a0, 8(\$sp)

sw \$a1, 12(\$sp)

sw \$a2, 16(\$sp)

addi \$sp, \$sp, 16 \rightarrow JAL Target

3

F - return

Before getting back

LW \$RA, 0(\$SP)

SW \$V0, 0(\$SP)

SW \$V1, 4(\$SP)

addi \$SP, \$SP, 4

$\begin{array}{c} \text{Y XXX} \dots \text{X} \quad \text{Hex} \\ \swarrow \quad \searrow \\ \text{D} \quad \text{F} \end{array}$

$\begin{array}{c} 8000 \text{ FFF} \text{ 4} \\ \text{Y XXX} \quad \text{X} \end{array}$

$\begin{array}{|c|} \hline \$RA \leftarrow SP \\ \hline \end{array}$

4

Q 3

yxxxxxxx

J label
Jol label

can ~~an~~ label must be

at an address that starts
with y

8000FFF4 J out

[8]xxxxxxx out:

crossing
0 segment

[9]xxxxxxx out?

X

5

8000FFF4 J out

9FFFFFF04 out:

La ~~and~~ sat, out

Jr sat

8[] J out1

out1

out:
[8]FFFFFFC J out_2 ✓

90000000 J out_2 : J out

9xxxxx... out

6 8C J JOL out \swarrow Filr

9C J out filr

near -
JOL to end of 8 out lib₁
to near beginning of 9 out lib₂
J to filr out lib₂

8xx JOL lib₁

lib₁ to lib₂

J filr

JR \$ Rq

7

Jal ~~V~~ ← \$RA

V ~~low~~

\$w \$5, 0(\$RA)

0

JR \$RA

Jal V

add \$5, \$6, \$V0 ← get back

↓ JAL R000

V: L: ~~[inst 1]~~, \$5 [ins]

\$w \$5, 0(\$RA)

OC / x x x x x

JAL

8

JOB function

8XXXXXX

↔ inst

← get loc from
function

function li \$5, 0x0C000000

JAL 00000000 ~~jump~~ / call

a function at the beginning
of segment

sw \$5, 0(\$RA)

JR \$RA

9

```
int j = 0;
int i = _____
while (i > 0) {
    i j = j + i
    i = i - 1
}
```

come through WI

Accum

}

used for termination

```
while (condition) {
    stmt
}
```

→

stmt that changes condition
state

}

used in recursive call

10

int \downarrow $\text{Accum}()$ E

\leftarrow check that L is legal

if ~~$(\text{L} \neq 1)$~~ ($\text{L} = 1$)

Return 0

~~return~~

~~return~~

~~Return accum(i)~~

~~Return accum(i)~~

$\text{L} = \text{L} - 1$

Return accum(L) + ($\text{L} + 1$)

Sequence n if $\begin{cases} n \text{ is even } n/2 \\ n \text{ is odd } \frac{3n+1}{2} \end{cases}$
Collatz

16 8 4 2 1 // 10 5 16 8 4 2 1