Vocab

Three MIPS steps: Fetch, Decode, Execute

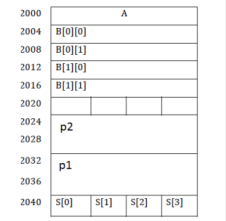
C Compiler is preprocessor and translator

Linker assembles all functions and make final exe

Loader loads program into memory

Memory Allocation

```
Int A = 200;
float B[2][2] = {{1.0,2.0}, {3.0,4.0}};
float *p2 = &(B[1][1]);
int *p1 = &A;
char s[ ] = "foo";
```



Swap

1. (7 pts) Write a MIPS code fragment that adds the value 0x4FA2C3 to register $1 and puts the result in $2. Modify only registers $1 and $2. (You may use hexadecimal immediate values.) *For maximum credit, include comments.*

Label	Instruction	Comment
	`lui $2, 0x4F`	`# load upper 16 bits`
	`ori $2, $2, 0xA2C3`	`# load rest of constant`
	`add $2, $2, $1`	`# add it to $1`

Values after

I x = 7, y = 7, and z = 2 I

if (z = x < y) { x += 3; y -= 1; }

else x = y++;

x = 7, y = 8, z = 0

Constant Type

29LU = unsigned long integer

"t" = string

-39.54Lf = long double

7hd = short integer

Values

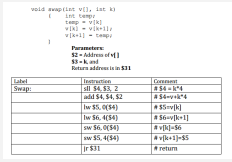
(p1 = 2000)(*p1 = 200)(&p1 = 2032)(p2 = 2016)

'*'p2 = 4)(&p2 = 2024)(&(s[1]) = 2041)*(s+3) = 0)

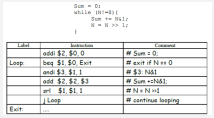
Loop Array C

int A[]; int i = 0; do {p = p*A[i]; i++;} while (A[i]);

Swap MIPS



Loop C to Mips



Sponsored by **Readable.com**

Measure your website readability!

https://readable.com