Assembly: instruction formats; load/store

COSC 208, Introduction to Computer Systems, 2021-10-06

Announcements

- Exam1 Q5

Outline

- Assembly
- Operands
- Load/store

Assembly

```c
#include <stdio.h>
int deref(int *p) {
    int v = *p;
    return v;
}
int main() {
    int x = 2;
    int *y = &x;
    int z = deref(y);
    printf("deref(y) = %d\n", z);
    return 0;
}
```

```
0000000000400584 <deref>:

    400584: d10043ff    sub sp, sp, #0x10

    400588: f90007e0    str x0, [sp, #8]

    40058c: f94007e8    ldr x8, [sp, #8]

    400590: b9400109    ldr w9, [x8]

    400594: b90007e9    str w9, [sp, #4]

    400598: b94007e0    ldr w0, [sp, #4]

    40059c: 910043ff    add sp, sp, #0x10

    4005a0: d65f03c0    ret
```

- Q1: *What do each of the columns contain?*
- Mapping between assembly and C code

Load/store

- Q2: *What is the C code equivalent for* `str x0, [x1]`*, treating registers as if they were variable names?*

```

```

- Q3: *What is the C code equivalent for* `ldr x2, [x3]`*, treating registers as if they were variable names?*

```

```

- Q4: *Write the C code equivalent for each line of assembly, treating registers as if they were variable names. For example, the C code equivalent for* `sub sp, sp, #0x20` *is* `sp = sp - 0x20`

```
0000000000400584 <deref>:
    400584: d10043ff    sub sp, sp, #0x10
    400588: f90007e0    str x0, [sp, #8]
    40058c: f94007e8    ldr x8, [sp, #8]
    400590: b9400109    ldr w9, [x8]
    400594: b90007e9    str w9, [sp, #4]
    400598: b94007e0    ldr w0, [sp, #4]
    40059c: 910043ff    add sp, sp, #0x10
    4005a0: d65f03c0    ret
```