



COMP2300

- Home
- StudyAt
- Assessment
- Schedule
- Lecture Notes
- Tute / Labs
- Assignments
- Announcements
- Discussion
- Reading material
- Old Exams
- Links
- PeANUt@Home
- Getting Help

Quick Links

- ANU Home
- FEIT Home
- DCS Home
- Search



COMP2300

Tutorial / Laboratory 05 - PeANUt Experiments

Semester 1, 2009

Week 6 (30 March - 03 April)

Note that for this session, there is a submitable laboratory exercise which is due by 09 am Tuesday 07 April (week 7), which will contribute up to 1% of your assessment (in the Tute/Lab mark).

It is expected that you have looked over your lecture notes of the PeANUt module as well as read the relevant parts of the *Specification of the PeANUt Computer* before doing this session. It is also expected that you have completed the lab exercises of the previous session (at least part 3, **Experimenting with the PeANUt**, and part 5, **Your multiplication program**). You should also look at the [selected answers](#) now up on the web page.

Preparation Exercises

Complete the following questions on a separate sheet of paper, with your name and student number clearly written. Please ensure your writing is legible. Hand in to your tutor at the *beginning of your tutorial / laboratory session*.

1. Briefly describe the term *von-Neumann architecture*.
2. Suppose the five 16-bit words starting at byte 1000 of a big-endian machine contain the integers 1, 2, 3, 4, and 5. Further suppose that these five words are transmitted to a little-endian machine as a sequence of bytes in address order and stored at consecutive addresses starting at byte 30000. What are the values of the five 16-bit words now?
3. Write a sequence of PeANUt MLI instructions that would do the same computation as the C statement `x = 3 * y + 1;`. Assume `x` and `y` are at memory locations `a0` and `a1` respectively.

Tutorial Exercises

1. Write a sequence of PeANUt MLI instructions that would do the same computation as the C statement `if (x==0) y++;`. Assume `x` and `y` are at memory locations `a0` and `a1` respectively.
2. Write a PeANUt program that would read a character and outputs the same character (but in lower case, if the original character was in upper case). You may (or may not) find useful the following ASCII codes: 'A'=0101, 'Z'=0132, 'a'=0141, 'z'=0172.
Outline how you might extend this program to one that reverses the case of upper and lower case characters.
3. Analyse the claim that it is always possible to replace a procedure call with the body of a procedure, and therefore that the procedure is not necessary.
4. Allocate space in PeANUt assembly language for the following:

```
char c; int x;
```

5. What is value of **AC** after executing (if **mem[x] = 59, SP = 70, XR = 5**):

```
load #15 ; AC = 15
mul #4 ; AC = AC * 4
store !-10 ; mem[SP-10] = AC
dvd #3 ; AC = AC / 3
store *54 ; mem[XR+54] = AC
load x ; AC = mem[x]
sub *55 ; AC = AC - mem[XR+55]
add #40 ; AC = AC + 40
```

Note: all numbers are in decimal.

6. Explain why `load #530` is illegal, and what problems are encountered in the instruction sequence `load #500` followed by `mul #500`.

Laboratory Exercises