FULL NAME: EPITA - ING1 - CAAL

Computer Architecture and Assembly Language

Exam - January 13th, 2015

No additional material (lecture slides, notes, etc.) is allowed during the exam

The exam questions have only **one** correct answer and are all independent of each other

Execution flow

Q1 On RISC processors, what is a delay-slot?

- ☐ The instruction preceding a branch instruction in sequential order
- ☐ The instruction following a branch instruction in sequential order
- ☐ The instruction targeted by a branch instruction
- ☐ The return address after a branch-and-link instruction has been taken

Q2 On a SPARC processor, delay slots are:

- ☐ Never executed
- ☐ Executed only if the branch is taken
- ☐ Executed only if the branch is not taken
- ☐ Always executed

Q3 What kind of information does a stack frame hold?

- ☐ The local variables, input arguments, etc. of its corresponding function
- ☐ The static and read-only variables of its corresponding function
- ☐ All the necessary debugging information of its corresponding function
- ☐ The complete memory context of the process

Q4 What kind of event is an interrupt?

- ☐ Planned and Asynchronous
- ☐ Unplanned and Synchronous
- ☐ Unplanned and Asynchronous
- ☐ Planned and Synchronous

Memory

Q5 On a little-endian processor, what is the value of `A` after executing the following code?

```
uint32_t B = 0x23456789;
uint16_t A = ((char*)&B)[3];
```

- ☐ 0x23
- ☐ 0x4523
- ☐ 0x89
- ☐ 0x6789

Q6 What size is the following structure (i.e. `sizeof(struct foo)`)?

```
struct foo {
  uint32_t  a;
  uint32_t  x;
  int8_t    b;
  uint16_t  c;
  uint8_t   f;
};
```

- ☐ 15 bytes
- ☐ 12 bytes
- ☐ 16 bytes
- ☐ 20 bytes

Q7 What is an aligned memory address?

- ☐ An address multiple of the system bus size
- ☐ An address multiple of `sizeof(int)`
- ☐ An address multiple of the memory access size
- ☐ An address multiple of the destination register size

Q8 What type of addressing mode does the following instruction use?

```
ld [%i0 + 0x4], %l0
```

- ☐ Immediate
- ☐ Absolute
- ☐ Register indirect
- ☐ Complex

Q9 What is a processor memory cache?

- ☐ A transparent copy of a portion of the central memory
- ☐ A visible extension of the central memory
- ☐ A fast memory for accelerating disk access
- ☐ An extension of the register bank

x86-RISC

Q10 Which of the following statements about the 32-bit x86 architecture is false?

- ☐ It uses little-endian byte order
- ☐ It is based on two operand instructions
- ☐ Its instruction set follows the *Reduced Instruction Set Computing* paradigm
- ☐ The `AL` register is an alias for accessing the 8 less significant bits of `EAX`

Q11 What is the only common property of MIPS, SPARC, PPC and ARM architectures?

- ☐ Fixed-length instructions
- ☐ 32 general purpose registers
- ☐ Delay-slot
- ☐ Register window

Memory mapping

Q12 Using memory paging, which of the following would be sufficient to copy from a parent process when forking?

- ☐ The content of its entire memory space
- ☐ The memory pages it has used
- ☐ The memory pages it has modified
- ☐ Its page table

Q13 Is it possible to map the same physical memory page as read-only in a virtual memory space and as read-write in another virtual memory space?

- ☐ Yes
- ☐ No

Q14 Is it possible to map the same physical memory page in two different virtual memory spaces, at two different virtual memory addresses?

- ☐ Yes
- ☐ No

Q15 Is it possible to map the same physical memory page in a single virtual memory space, at two different virtual memory addresses?

- ☐ Yes
- ☐ No

Future systems

Q16 Which of the following statements about systems will probably be false in the future, if it is not already?

- ☐ They will follow a Uniform Memory Access model
- ☐ Networks will replace system buses
- ☐ One of their biggest challenges will be coherent shared-memory approaches
- ☐ General-Purpose GPU will assist processors for speeding up computation

Objects

Q17 Which tool generates the binary instruction sequence for the target processor architecture?

- ☐ Preprocessor
- ☐ Compiler
- ☐ Assembler
- ☐ Linker

Q18 Considering the following declaration of the variable `i`, in which section will `i` be contained after compilation?

```
static int i = 3;
```

- ☐ .bss
- ☐ .data
- ☐ .rodata
- ☐ .text

Q19 What does an object file contain?

- ☐ Only the binary code optimized for the target processor
- ☐ The entire assembly source code
- ☐ The binary code, the data and various directives
- ☐ Only the relocations and directives for the linking stage

Q20 We assume an object file `A.o` of size 0x45 and a variable `a` whose local address in `A.o` is 0x25; we also assume an object `B.o` of size 0x15 and a variable `b` whose local address in `B.o` is 0x5; which one of these statements is true after linking both object files with the following command?

```
$ ld -o AB B.o A.o
```

- ☐ The address of variable `a` is 0x25
- ☐ The address of variable `b` is 0x4A
- ☐ The address of variable `a` is 0x40
- ☐ The address of variable `a` is 0x3A

CPU optimizations

Q21 Considering the execution of the following function on a 32-bit processor, which value will it return if called with (x=-42)?

```
int mysterious(int x)
{
    int sign_word = x >> 31;
    return (x ^ sign_word) - sign_word;
}
```

- ☐ 42
- ☐ -42
- ☐ It is unpredictable

Processor

Q22 Within a processor, the unit performing operations between operands is usually called?

- ☐ The control unit
- ☐ The register bank unit
- ☐ The arithmetic and logical unit
- ☐ The processing and computational unit

Q23 The following instruction belongs to which category in Flynn's taxonomy?

```
add %g2, %g3, %g4
```

- ☐ SISD
- ☐ SIMD
- ☐ MIMD

Q24 Up to how many instructions can be processed at the same time in a pipelined processor?

- ☐ Only one
- ☐ As many as the number of processing steps
- ☐ As many as the number of instructions contained in a cache line
- ☐ As many as the number of existing opcodes

Q25 In a pipelined processor, what mainly determines the processor frequency?

- ☐ The depth of the pipeline
- ☐ The latency of the fastest processing step
- ☐ The latency of the slowest processing step
- ☐ The latency of the processor memory cache

Assembly language

Q26 What does the following SPARC-optimized function return if given (s1="epita", s2="ilovesparc")?

```
static inline
int cryptic_function(char *s1, char *s2)
{
    int res = 0;
    char c1, c2;
    asm (
        "1:                 \n"
        "ldub [%0], %3      \n"
        "ldub [%1], %4      \n"
        "tst  %3            \n"
        "bz   2f            \n"
        "inc  %0            \n"
        "tst  %4            \n"
        "bz   2f            \n"
        "inc  %1            \n"
        "ba   1b            \n"
        "inc  %2            \n"
        "2:                 \n"
        : "=&r" (s1), "=&r" (s2),
          "=&r" (res),
          "=&r" (c1), "=&r" (c2)
        ::);
    return res;
}
```

- ☐ 15
- ☐ 5
- ☐ 6
- ☐ 10

Q27 Assuming that `%g5` contains the value 0x13 and `%g3` the value 0x7, what is the value of `%g0` after executing the following instruction?

```
add %g5 , %g3 , %g0
```

- ☐ 0x20
- ☐ 0x1A
- ☐ 0x0
- ☐ The instruction causes an overflow exception

Q28 Among the following code sequences, which one is a correct sequence to return from a function?

- ☐ ret; nop; restore;
- ☐ ret; restore; nop
- ☐ restore; nop; ret;
- ☐ restore; ret; nop;

Q29 In SPARC assembly, if a function decides **not** to slide the register window (thus **not** to use the instruction `save`), then where will the function find its first argument?

- ☐ in the register `%i0`
- ☐ in the register `%o0`
- ☐ in the register `%l0`
- ☐ in the register `%g0`