CS4023 **Week03 Tutorial**

The following questions are taken from the exercises at the end of Chapter 1 of SGG (ed. 9) and outside them. Students will need group discussions and personal research to get at some answers.

1. Write a C program that requests the user to input two integers and saves those two integers in two variables. Then it calls a function `mult` that computes the product of the two numbers. `mult` can **not** explicity return a result, yet you have to extract the answer out of that function somehow. Note: you are **not** allowed to use global variables. Hint: read about *pointers* in C. Before `mult` computes the product, it also calls another function `void case_example(int x)` that uses a `switch` statement with three `case` statements: if x==1, it prints 'x=1'; if x==2, it prints 'x=2'; otherwise, it prints 'out of range'.

   You must define the functions *after* first defining the `main` function. If you follow that sequence the compiler would require you to also write *function prototypes*; read about that.

2. Consider the various definitions of operating system. Next, consider whether the operating system should include applications such as Web browsers and mail programs. Argue both that it should and that it should not, and support your answers. (**Q1.6**)

3. What is the policy of providing *protection* in a computer System?

4. What are two *mechanisms* for providing protection in a computer system?

5. Why does address translation not suffice to provide protection against accessing memory of other programs?

6. How does the distinction between kernel mode and user mode act as a rudimentary form of protection (security) system? (**Q1.7**)

7. Which of the following instructions should be privileged?

   (a) Set value of timer
   (b) Read the clock
   (c) Clear memory
   (d) Issue a trap instruction
   (e) Turn off interrupts
   (f) Modify entries in device-status table
   (g) Switch from user to kernel mode
   (h) Access I/O device

8. In what way is an operating system like a government?

(a) It seldom functions correctly.

(b) It creates an environment within which other programs can do useful work.

(c) It performs most useful functions by itself.

(d) It is always concerned primarily with the individual's needs.

9. Which of the following would lead you to believe that a given system is an SMP-type system?

   (a) Each processor is assigned a specific task.

   (b) There is a bossworker relationship between the processors.

   (c) Each processor performs all tasks within the operating system.

   (d) None of the above

10. A ____ can be used to prevent a user program from never returning control to the operating system.

    (a) portal

    (b) program counter

    (c) firewall

    (d) timer

11. Embedded computers typically run on a ____ operating system.

    (a) real-time

    (b) Windows XP

    (c) network

    (d) clustered

12. What statement concerning privileged instructions is considered **false**?

    (a) They may cause harm to the system.

    (b) They can only be executed in kernel mode.

    (c) They cannot be attempted from user mode.

    (d) They are used to manage interrupts.

13. Describe the differences between physical, virtual, and logical memory.

14. What is the purpose of interrupts? What are the differences between a trap and an interrupt? Can traps be generated intentionally by a user program? If so, for what purpose?

15. Direct memory access is used for high-speed I/O devices in order to avoid increasing the CPUs execution load.

(a) How does the CPU interface with the device to coordinate the transfer?

(b) How does the CPU know when the memory operations are complete?

(c) The CPU is allowed to execute other programs while the DMA controller is transferring data. Does this require mechanisms to address memory coherence problems? Explain.

16. Some computer systems do not provide a privileged mode of operation in hardware. Is it possible to construct a secure operating system for these computer systems? Give arguments both that it is and that it is not possible. Hint: think of interpreted computer languages. Hint2: Will you allow non-compiler generated object code?