QUESTION 1

1. When called by a user program the mode bit would switch from user mode to kernal mode

- ○ C functions
- ◉ System calls
- ○ Application programming interface (API)
- ○ Machine code

QUESTION 2

1. It enables application software to be ported easily, through recompilation, to other systems that support the same API.

- ◉ Application Programming Interface (API)
- ○ Application binary interface (ABI)
- ○ Instruction set architecuture (ISA)
- ○ None of the above

QUESTION 3

1. Command Line Interface (CLI), Graphical User Interface (GUI), Batch are example of

- ☐ A. Protection and security
- ☑ B. User Interface
- ☐ C. Error detection
- ☐ D. Accounting

QUESTION 4

1. Microkernal system structure This method structures the operating system by removing all nonessential components from the kernel and
implementing them as system programs and user-level programs.

- ○ Leading to harder to extend microkernal and easier to port the OS to new architecture and zero overhead.
- ○ Leading to easier to extend microkernal and easier to port the OS to new architecture and zero overhead.
- ○ Leading to easier to extend microkernal and easier to port the OS to new architecture and zero overhead.
- ◉ Leading to easier to extend microkernal and easier to port the OS to new architecture and additional overhead due to communication between user space and kernal space.

QUESTION 5

1. Specifying and designing an operating system is a highly creative task. One important principle to seperate are Policy and Mechanism

- (●) ***Policy: What*** will be done?
  Mechanism**:** ***How*** to do it?
- ( ) Policy: How it will be done ?

  Methodolgoy: How it will be done in more details.
- ( ) ***Policy and Mechanism are supposed to be the same, seperating them is a waste of time***
- ( ) ***Policy: When it*** will be done?
  Mechanism**:** How to do it?

QUESTION 6

1. MS DOS was designed on an architecture that support

- ( ) Dual mode. It support multiprogramming and multitasking.
- ( ) Single mode. It supports multiprogramming and multitasking.
- (●) Single mode. It does not support multiprogramming and multitasking.
- ( ) Dual mode. It does not support multiprogramming and multitasking.

QUESTION 7

1. In Layer Approach, system is divided into a number of layers (levels), each built on top of lower layers.

- ( ) One major advantage is that layers are need to be carefully designed and They tend to be more efficient than other types.
- ( ) One major disadvantage is that layers are need to be carefully designed and They tend to be more efficient than other types.
- (●) One major disadvantage is that layers are need to be carefully designed and They tend to be less efficient than other types.
- ( ) One major advantage is that layers are need to be carefully designed and They tend to be less efficient than other types.

QUESTION 8

1. In uniprogramming you have

- (●) A. One active program at a time, and mostly with low CPU utilization.
- ( ) B. many active programs at a time, and mostly with low CPU utilization.
- ( ) C. One active program at a time, and mostly with high CPU utilization.
- ( ) D. many active programs at a time, and mostly with high CPU utilization.

QUESTION 9

1. Modules is one of the best current methodology for operating system design, communication can be performed

- ○ Modules can communicate through air, water, and solid materials.
- ○ All communication has to go through the kernal.
- ◉ Each module can communicate with other module through known interfaces
- ○ Modules are independent entity, there are absulotely no need for communication between modules

QUESTION 10

1. System call parameter passing can be performed by

- ○ System calls has no parameter, it knows the intention of the caller automatically.
- ◉ 1. Pass Parameter in registers, or
  2. Registers stored in block and address is stored in registers
  3. uses stack (program push parameters, and stack popped off by OS)
- ○ 1. Pass Parameter in registers, or
  2. Registers stored in block and address is stored in registers
  3. uses hash function
- ○ It can only done by using stack (program push parameters, and stack popped off by OS) because registers are meant for something else.