Introduction to Operating Systems

Question @ Answers

Question 1

Show a rough historical timeline for the different types of Microsoft operating systems and computer architectures.

Year	Microsoft OS	Architecture	Details
1985	MS Windows 1.0	80x286	Segmentation
1987	MS Windows 2.0	80x386	Segmentation
1990	MS Windows 3.0	80x386	Segmentation, GUI, Mouse
1992	MS Windows 3.1	80x486	Multitasking –first time to run more than one Program
1993	MS Windows NT 3.11	Pentium I	Paging
1995	MS Window 95		Pentium Pro, Internet commercially available
1996	MS Windows NT 4.0	80x486	
1998	MS Window 98	Pentium II	
2001	MS Window XP	Pentium 4	Internet Explorer 6
2007	MS Windows Vista	Intel Core	
2009	MS Windows 7	Intel Core	

Question 2

What is a Portable Operating System Interface (POSIX)?
Should an operating system be POSIX compliant?

Question 3

What are the two main functions of an operating system?

Question 4

What does the CPU do when there are no programs to run?

Question 5

What is the periodic time of a 1 GHz computer clock?

Time = 1 / Frequency
Time = 1/ 1*E9
Time = 1 nanosecond

Question 6

How much memory can be accessed using 16-bit address lines?

$2^{16} = 65,536$

Question 7

Show various levels of storage and their typical size, access time, and order of cost.

Question 8

Explain why code optimization is typically performed on assembly code?

Question 9

What is the purpose of the common language runtime (CLR)?

Question 10

What are some of the advantages and disadvantages using a Command Interpreter versus a Graphical User Interface?

Question 11

List some advantages for using a virtual machine.

Question 12

What are the states a process can enter?

Question 13

What is typically contained in the process control block (PCB)?

Question 14

Explain the difference between a context switch and swapping.

Question 15

The benefits of multithreaded programming can be broken down into four major categories:

1. Responsiveness
2. Resource sharing
3. Economy
4. Scalability

Provide a short explanation of each benefit.

Question 16

What is the principal advantage of multiprogramming?

Increased CPU utilization

Question 17

What is the principal disadvantage of multiprogramming?

Question 18

Explain the role of an operating system when a multicast delegate is invoked.

Question 19

What are some of the obstacles running a multi-threaded process in a dual core technology?

Question 20

Provide a short definition of Throughput.

Throughput = Total processing time / number of processes.
In reality, one hardly ever determines throughput using this equation.
Before a system is developed, a study is conducted of the expected work load. The work load most likely varies over time, which requires that a stress test is performed over at least 1 week time duration. Throughput is then defined in these terms.

Question 21

Provide a short definition of Turnaround Time.

Turnaround Time = Time a process is in the system

Question 22

What scheduling algorithm provides the shortest Turnaround Time?

Question 23

Is Turnaround Time inversely proportional to Throughput?

Question 24

Provide a short definition of Response Time.

Response Time = The time it took for the system to respond to the user. It is not the time required to produce an output.

Question 25

Provide the Gantt chart for the processes and associated burst times given below. Use the First-Come First-Serve (FCFS) scheduling algorithm.

Process	Arrival Time	Burst Rate
A	0.00	5
B	1.00	2
C	2.00	5
D	3.00	3

Process	A	A	A	A	A	B	B	C	C	C	C	C	D	D	D	
A	5	4	3	2	1	0	0	0	0	0	0	0	0	0	0	0
B	0	2	2	2	2	2	1	0	0	0	0	0	0	0	0	0
C	0	0	5	5	5	5	5	5	4	3	2	1	0	0	0	0
D	0	0	0	3	3	3	3	3	3	3	3	3	3	2	1	0
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Turnaround Time		
A	5 - 0	5
B	7 - 1	6
C	12 - 2	10
D	15 - 3	12
Avg Turnaround Time		**8.25**

Wait Time		
A	5 - 5	0
B	6 - 2	4
C	10 - 5	5
D	12 - 3	9
Avg Wait Time		**4.5**

Response Time		
A	0-0	0
B	5 - 1	4
C	7 - 2	5
D	12 - 3	9
Avg Resp Time		**4.5**

Process	Burst Time
P1	12
P2	4
P3	6
P4	7

Question 26

Provide the Gantt chart for the processes and associated burst times given above. Use the First-Come First-Serve (FCFS) scheduling algorithm.

Question 27

Given the data in problem 25, provide the average Wait Time, average Turnaround Time, and Response Time.

Question 28

Provide the Gantt chart using the Shortest-Job-First scheduling algorithm.

Question 29

Given the data in problem 27, provide the average Wait Time and average Turnaround Time.

Question 30

Process	Burst Time	Priority
P1	8	2
P2	3	1
P3	4	3
P4	7	4

Provide the Gantt chart using priority scheduling.

Question 31

Process	Burst Time
P1	10
P2	5
P3	22
P4	15
P5	9

Provide the Gantt chart using a Round-Robin scheduling algorithm and a time quantum 4.

Question 32

Given the data in Question 30, calculate the Turnaround time, Throughput, Waiting Time, and Response Time.

Question 33

Repeat question 30 and question 31 using a time quantum 3.

Question 34

Given data for Process A – D below provide the Gantt chart for a round – robin scheduling algorithm with a quantum = 3.

Process	Arrival Time	Burst Rate
A	0.00	5
B	1.00	2
C	2.00	5
D	3.00	3

Q = 3

Process	A	A	A	B	B	C	C	C	D	D	D	A	A	C	C	
A	5	4	3	2	2	2	2	2	2	2	2	2	1	0	0	0
B	0	2	2	2	1	0	0	0	0	0	0	0	0	0	0	0
C	0	0	5	5	5	5	4	3	2	2	2	2	2	2	1	0
D	0	0	0	3	3	3	3	3	3	2	1	0	0	0	0	0
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Question 35

Given the data in problem #33, what is the turn-around time, wait time, throughput and response time?

Turnaround Time = Time a process is in the system

Turnaround Time		
A	13 - 0	13
B	5 - 1	4
C	15 - 2	13
D	11 - 3	8
Avg Turnaround Time		**9.5**

4 Processes / T15 = **0.27 Processes per 1 Burst rate or 3.75 T per Process**

Wait Time = Turnaround Time - Burst Rate

Wait Time		
A	13 - 5	8
B	4 - 1	3
C	13 - 5	8
D	8 - 3	4
Avg Wait Time		**5.75**

Response Time = The time it took for the system to respond to the user. It is not the time requried to produce an output.

Response Time		
A	0-0	0
B	3 - 1	2
C	5 - 2	3
D	8 - 3	5
Avg Resp Time		**2.5**

Question 36

Given data for Process A – D above, provide the Gantt chart for a shortest remaining time scheduling algorithm.

Process	Arrival Time	Burst Rate
A	0.00	5
B	1.00	2
C	2.00	5
D	3.00	3

Process	A	B	B	D	D	D	A	A	A	A	C	C	C	C	C	
A	5	4	4	4	4	4	4	3	2	1	0	0	0	0	0	0
B	0	2	1	0	0	0	0	0	0	0	0	0	0	0	0	0
C	0	0	5	5	5	5	5	5	5	5	5	4	3	2	1	0
D	0	0	0	3	2	1	0	0	0	0	0	0	0	0	0	0
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Turnaround Time		
A	10 - 0	10
B	3 - 1	2
C	15 - 2	13
D	6 - 3	3
Avg Turnaround Time		**7**

Wait Time		
A	10 - 5	5
B	2 - 2	0
C	13 - 5	8
D	3 - 3	0
Avg Wait Time		**3.25**

Response Time		
A	0-0	0
B	1 - 1	0
C	10 - 2	8
D	3 - 3	0
Avg Resp Time		**2**

Question 37

Given data for Process A – D, provide the Gantt chart for a shortest job first time scheduling algorithm.

Process	Arrival Time	Burst Rate
A	0.00	5
B	1.00	2
C	2.00	5
D	3.00	3

Process	A	A	A	A	A	B	B	D	D	D	C	C	C	C	C	
A	5	4	3	2	1	0	0	0	0	0	0	0	0	0	0	0
B	0	2	2	2	2	2	1	0	0	0	0	0	0	0	0	0
C	0	0	5	5	5	5	5	5	5	5	5	4	3	2	1	0
D	0	0	0	3	3	3	3	3	3	1	0	0	0	0	0	0
Time	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

Turnaround Time		
A	5 - 0	5
B	7 - 1	6
C	15 - 2	13
D	10 - 3	7
Avg Turnaround Time		**7.75**

Wait Time		
A	5 - 5	0
B	6 - 2	4
C	13 - 5	8
D	7 - 3	4
Avg Wait Time		**4**

Response Time		
A	0-0	0
B	5 - 1	4
C	10 - 2	5
D	7 - 3	4
Avg Resp Time		**3.25**

Question 38

In a multi-level queuing scheduling OS, what mechanism or preventive steps must be taking to prevent a process from starving?

Question 39

What happens in a real-time operating system when the CPU utilization exceeds 69.3 percent?

Question 40

Suppose a real-time operating system has 100 time slices allocated per second for user processes and requires one millisecond per context-switch. Assume the OS consumes 35 percent of the CPU for internal processing.

What percent of CPU utilization is spent on context-switching?

Question 41

Indicate if the statements are true or false.

- An OS requires at least two different modes of operation, user mode and kernel mode.
- The OS is responsible for scheduling processes and threads on the CPUs.
- The OS is responsible for suspending and resuming processes.
- The OS in not responsible for creating or deleting both user and system processes.
- Subroutines and supervisor calls are instructions executable by user programs. Traps and interrupts are not; they are events triggered by hardware.
- Round Robin is a preemptive algorithm that selects the process that has been waiting the shortest time.
- Shortest Remaining Time (SRT) is a preemptive version of shortest job first.

Question 42

Indicate if the transition between process states is possible or not..

- Run → Ready
- Run → Blocked
- Run → Sapped-blocked
- Blocked → Run
- Run → Terminated

Question 43

Provide a short explanation for a process using ***cascading termination***.

Question 44

A process is waiting on memory resources. What happens to the process once the resources are assigned to it?

Question 45

It is possible for a programmer to assign a thread procedure a thread priority of "high". What considerations must be given before such thread priorities are assigned. Who should determine thread priorities?

Question 46

A process has multiple threads and signals the OS that it wants to shut down. One of the threads is blocking on accessing a message queue. What mechanism needs to be implemented to properly shut the thread down?

Question 47

What is a critical section?

The section of code that requires access to shared resources is called a critical section. To avoid race conditions, a mechanism is needed to synchronize the execution within critical sections.

Question 48

What three requirements must be observed to eliminate the critical section problem?

Mutual Exclusion
Progress
Bounded Waiting

Question 49

What is a semaphore?

Question 50

What is a spinlock?

Question 51

What are the necessary conditions for a system to become deadlocked?

Question 52

What approaches are possible to recover from a deadlock?

Automatic preemption
Automatic termination
Manual intervention

Question 53

What is a mutex?

Process	Resource Allocation
P1	R1
P1	R2
P1	R3
P1	R4
P2	R1
P2	R3
P3	R1
P3	R4

Resource	Qty
R1	5
R2	5
R3	2
R4	2
R5	3
R6	3
R7	4
R8	4

Question 54

Provide the resource allocation graph for the information above

Question 55

Refer to the information above.
Is the system deadlocked if process P3 requests resource R3 <Yes/No>?

Provide an explanation for your <Yes/No> answer.

Process	Resource Allocation
P1	R1
P1	R3
P2	R2
P2	R4

Resource	Qty
R1	2
R2	1
R3	2
R4	2

Question 56

Provide the resource allocation graph for the information above.

Question 57

Refer to the information above.
Is the system deadlocked if process P1 requests resource R2 <Yes/No>?

Provide an explanation for your <Yes/No> answer.

Question 58

Refer to the information above.
Suppose process P1 requests resource R2 and process P2 requests process R1.
Will this request result in a system deadlock?

Question 59

A particular smart device has 16,777,216 bytes of memory. Suppose the fixed partitions are of size 65,536 bytes. How many bits are required to access an entry in the partition table?

Question 60

Explain why paging does not suffer from external fragmentation.

Question 61

True or False

- The operating system is not involved in partitioning a program into segments.
- A process is divided into blocks called pages which are the same size as frames.
- When a dirty bit is set the Operating System raises an out-of-bounds memory exception.
- The LRU algorithm can suffer from Belady's anomaly.
- The working set model is based on data locality.

Question 62

What is *thrashing*?

A high paging activity is called thrashing. A process is thrashing when it spends more time paging than executing.

Question 63

Suppose a smart device has 30 KB of memory and the OS uses a first-fit memory allocation algorithm. Processes P1 through P10 are in the ready queue. Show the memory allocation for the processes, making the assumption that a process will finish executing in the order they were allocated to memory.

Process	Memory Requirement (KB)
P1	2
P2	3
P3	10
P4	4
P5	8
P6	12
P7	7
P8	3
P9	5
P10	4

Memory	Process
1	p1
2	p1
3	p2
4	p2
5	p2
6	p3
7	p3
8	p3
9	p3
10	p3
11	p3
12	p3
13	p3
14	p3
15	p3
16	p4
17	p4
18	p4
19	p4
20	p5
21	p5
22	p5
23	p5
24	p5
25	p5
26	p5
27	p5
28	p8
29	p8
30	p8

Memory	Process
1	p9
2	p9
3	p9
4	p9
5	p9
6	p3
7	p3
8	p3
9	p3
10	p3
11	p3
12	p3
13	p3
14	p3
15	p3
16	p4
17	p4
18	p4
19	p4
20	p5
21	p5
22	p5
23	p5
24	p5
25	p5
26	p5
27	p5
28	p8
29	p8
30	p8

Memory	Process
1	p9
2	p9
3	p9
4	p9
5	p9
6	p7
7	p7
8	p7
9	p7
10	p7
11	p7
12	p7
13	p10
14	p10
15	p10
16	p10
17	
18	
19	
20	p5
21	p5
22	p5
23	p5
24	p5
25	p5
26	p5
27	p5
28	p8
29	p8
30	p8

Memory	Process
1	p9
2	p9
3	p9
4	p9
5	p9
6	p7
7	p7
8	p7
9	p7
10	p7
11	p7
12	p7
13	p10
14	p10
15	p10
16	p10
17	p6
18	p6
19	p6
20	p6
21	p6
22	p6
23	p6
24	p6
25	p6
26	p6
27	p6
28	p6
29	
30	

Question 64

Suppose a smart device has 40 KB of memory and the OS uses a worst-fit memory allocation algorithm. Use the process information provided in question 63.

Question 65

What are the differences between equal frame allocation and proportional frame allocation?

Question 66

Given the following page requirements and a memory management using 4 frames per process, determine the number of page faults. Use the FIFO algorithm.

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frm1	7	7	7	7	7	3	3	3	3	3	3	3	3	3	2	2	2	2	2	2
Frm2		0	0	0	0	0	0	4	4	4	4	4	4	4	4	4	4	7	7	7
Frm3			1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0
Frm4				2	2	2	2	2	2	2	2	2	2	1	1	1	1	1	1	1
Pg Faults	*	*	*	*		*		*			*			*	*			*		

Page Faults = 10

Question 67

Given the following page requirements and a memory management using 5 frames per process, determine the number of page faults. Use the FIFO algorithm.

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frm1	7	7	7	7	7	7	7	4	4	4	4	4	4	4	4	4	4	4	4	4
Frm2		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	7	7	7
Frm3			1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0
Frm4				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	1
Frm5						3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
Pg Faults	*	*	*	*		*		*										*	*	*

Page Faults = 9

Question 68

Given the following page requirements and a memory management using 4 frames per process, determine the number of page faults. Use the Optimal Page replacement algorithm.

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frm1	7	7	7	7	7	3	3	3	3	3	3	3	3	1	1	1	1	1	1	1
Frm2		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Frm3			1	1	1	1	1	4	4	4	4	4	4	4	4	4	4	7	7	7
Frm4				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Pg Faults	*	*	*	*		*		*						*				*		

Page Faults = 8

Question 69

Given the following page requirements and a memory management using 4 frames per process, determine the number of page faults. Use the LRU replacement algorithm.

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frm1	7	7	7	7	7	3	3	3	3	3	3	3	3	3	3	3	3	7	7	7
Frm2		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Frm3			1	1	1	1	1	4	4	4	4	4	4	1	1	1	1	1	1	1
Frm4				2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
Pg Faults	*	*	*	*		*		*						*				*		

Page Faults = 8

Question 70

Given the following page requirements and a memory management using 4 frames per process, determine the number of page faults. Use the Second Chance algorithm. (Highly Error Prone)

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
Frm1	7	7	7	7	7	7	7	7	2	2	2	2	2	2	2	2	2	2	2	2
Frm2		0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
Frm3			1	1	1	3	3	3	3	3	3	3	3	3	3	0	0	0	0	0
Frm4				2	2	2	2	4	4	4	4	4	4	4	4	4	4	7	7	7
Pg Faults	*	*	*	*		*		*	*					*		*		*		

Page Faults = 10

Question 71

Given the following page requirements and a memory management using 3 frames per process, determine the number of page faults. Use the FIFO algorithm.

Pg	2	1	0	4	5	4	6	7	3	3	4	5	4	2	3	2	1	0	6	7	5	0	1	2

Question 72

Given the following page requirements and a memory management using 4 frames per process, determine the number of page faults. Use the FIFO algorithm.

Pg	2	1	0	4	5	4	6	7	3	3	4	5	4	2	3	2	1	0	6	7	5	0	1	2

Question 73

Given the following page requirements and a memory management using 5 frames per process, determine the number of page faults. Use the FIFO algorithm.

Pg	2	1	0	4	5	4	6	7	3	3	4	5	4	2	3	2	1	0	6	7	5	0	1	2

Question 74

Given the following page requirements and a memory management using 4 frames per process, determine the number of page faults. Use the LRU algorithm.

Pg	2	1	0	4	5	4	6	7	3	3	4	5	4	2	3	2	1	0	6	7	5	0	1	2

Question 75

A process was allocated 200 frames and the page fault rate was initially measured at 20 faults per 10,000 page requests. Explain what measures the Operating System should take when the page fault rate increases and exceeds the upper boundary.

Question 76

Give an example of access to variables within a program that result in temporal locality.

Question 77

Give an example of access to variables within a program that result in spatial locality.

Question 78

True or False

- A peak in the page fault rate occurs when a process begins demand-paging due to a new locality.
- The OS should reduce the number of frames allocated to a process when the page fault rate falls below the minimum threshold.
- The OS should reduce the number of frames allocated to a process when the page fault rate peaks above the maximum threshold.

Question 79

Suppose a working set uses 6 references.
What is the working set W (t, Δ) for $15^{th} < t \leq 16^{th}$ reference?

Page References: 0, 9, 0, 1, 8, 1, 8, 7, 8, 7, 1, 2, 8, 2, 7, 8, 2, 3, 8, 3

W (t, Δ) = {1, 2, 7, 8}

Question 80

Suppose a working set uses 7 references.
What is the working set W (t, Δ) for $14^{th} < t \leq 15^{th}$ reference?

Page References: 0, 9, 0, 1, 8, 1, 8, 7, 8, 7, 1, 2, 8, 2, 7, 8, 2, 3, 8, 3

W (t, Δ) = { … }

Question 81

Suppose a working set uses 5 references, and the working set is recalculated when a page fault occurs.
Given the page references determine the number of page faults. Assume 5 frames are available for the process.

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1

Note: If a page is loaded in memory and the page is not part of the working set, the OS does not unload the page unless the frame is needed for a different page. Pages not included in the working set are highlighted in yellow, below.

1. Page fault 1. WS = {7}
2. Page fault 2. WS = {0, 7}
3. Page fault 3. WS = {0, 1, 7}
4. Page fault 4. WS = {0, 1, 2, 7}
5. Page fault 5. WS = {0, 1, 2, 3}
6. Page fault 6. WS = {0, 2, 3, 4}
7. Page fault 7. WS = {0, 1, 2, 7}

Pg	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
WS	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
WS		7	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	1	1	1
WS			7	2	2	2	2	3	3	3	3	3	3	3	3	3	3	2	2	2
WS				7	7	3	3	4	4	4	4	4	4	4	4	4	4	7	7	7
Fr						7	7	1	1	1	1	1	1	1	1	1	1	3	3	3
Pg Faults	*	*	*	*		*		*										*		

Page Faults = 7

Question 82

Suppose a working set uses 5 references, and the working set is recalculated when a page fault occurs.
Given the page references determine the number of page faults. Assume 5 frames are available for the process.

Pg	2	1	0	4	5	4	6	7	3	3	4	5	4	2	3	2	1	0	6	7	5	0	1	2

Question 83

Suppose a working set uses 6 references, and the working set is recalculated when a page fault occurs. Contrast the performance with the 5 reference working set (see question 82).

Question 84

List the possible file operations.

- Create File
- Write to File
- Read from File
- Reposition within the File
- Delete File
- Truncate the File

Question 85

List the possible file attributes.

- File Name
- File Identifier
- File Type
- File Location
- File Size
- File Protection
- File Time, date, user identification

Question 86

Describe sequential file access.

Question 87

Describe direct file access.

Question 88

List three file allocation methods.

- Contiguous Allocation
- Linked Allocation
- Indexed Allocation

Question 89

Identify a problem associated with contiguous allocation.

Question 90

True or False

- Linked allocation is inefficient for direct-access files.
- There is no external fragmentation associated with linked allocation.
- Contiguous allocation suffers from external fragmentation.
- When the file allocation table becomes corrupted, the data contained in a file is loosed to the user.
- When the OS uses indexed allocation, every file must have an indexed block.
- The performance of indexed allocation depends on the index structure, on the size of the file, and on the position of the block desired.

Question 91

Contrast vertical and horizontal striping in a Storage Area Network (SAN.)

Question 92

Define the following security violations.

- Breach of confidentiality
- Breach of integrity
- Breach of availability
- Theft of service
- Denial of service

Question 93

What is a Trojan horse?

Question 94

List the categories a virus may belong to. Briefly explain how they work.

- File
- Boot
- Macro
- Source Code
- Polymorphic
- Encrypted
- Stealth
- Tunneling
- Multipartite
- Armored

Question 95

What is a Worm?

Question 96

What is Cryptology?

Cryptology is comprised of two categories, Cryptography and Cryptanalysis.
Cryptography is the science of secret writing with the goal of hiding the meaning of the message.
Cryptanalysis is the science of breaking cryptosystems.

Question 97

Why is **Cryptanalysis** an important science (art) in Computer Science?

Cryptanalysis is the only way to assure that a cryptosystem is secure. *Secure* implies that deciphering data will take a considerable long time.

Question 98

What is Letter Frequency Analysis?

Languages have properties that could be exploited to decipher a message.

- The frequency of every cypher text message may allow an AI system to decipher the message.
- The frequency of occurrence of cypher text can be extended to pairs or triples. Example: et, tt, ter.
- Word separators can lead to deciphering messages.

Letter	Frequency	Letter	Frequency
A	0.0817	N	0.0675
B	0.0150	O	0.0751
C	0.0278	P	0.0193
D	0.0425	Q	0.0010
E	0.1270	R	0.0599
F	0.0223	S	0.0633
G	0.0202	T	0.0906
H	0.0609	U	0.0276
I	0.0697	V	0.0098
J	0.0015	W	0.0236
K	0.0077	X	0.0015
L	0.0403	Y	0.0197
M	0.0241	Z	0.0007

Question 99

Decipher the following encrypted text.

iq ifcc vqqr fb rdq vfllcq na rdq cfjwhwz hr bnnb hcc hwwhbsqvqbre hwq vhlq

Question 100

What are the estimated times for successful *brute force* attacks on symmetric algorithms with different key length?

Key Length	**Security Estimation**
56 - 64 bits	a few hours
112 - 256 bits	several decades

Quantum computers can solve 112 – 128 bits relatively fast; however 256 bits would take decades even for a quantum computer. Most systems use intelligent attacks which can make computer systems extremely vulnerable.