Name:	Saeeda Naaz
ID:	14332
Semester:	BSCS-5th

Final Term Assignment
Operating System Concepts

Time Allowed: 6 hours ***Marks: 50***

Note: Attempt all questions. Copying from Internet and one another is strictly prohibited. Such answers will be marked zero.

Q1. In deadlock prevention strategy do you think it is necessary to check that either safe state exists or not? Give reason to support your answer.

Answer:

- Yes, in deadlock prevention strategy it is necessary to check that either safe state exists or not because if a safe state doesn't exist, then the system lead to deadlock.
- In safe state the system allocates all resources to the processes on the request of process (up to their stated maximum) without entering a deadlock state.
- If the system goes in deadlock state it means that processes are blocked because each process hold resources and waiting for another resources holder by other process.
- The reason of safe state existence in deadlock prevention strategy is to ensure that process must acquire and release resources in a specific order.

Q2. Differentiate between Dynamic loading and Dynamic Linking with the help of examples.

Answer:

Dynamic linking and loading are utility programs used during execution of a program. Some major differences are given below:

	Dynamic Loading	**Dynamic Linking**
Definition	• Dynamic loading refers to loading a routine of a program when it is called by the program.	• Dynamic linking is the process of collecting and combining various program modules and data in executable file.

function	• Dynamic loading loads the executable file of program into memory. • Loading allocate spaces to executable modules in memory.	• Dynamic linking generates these executable files. • Linking combines all modules and link with built in libraries.
Uses	• It is useful in efficient memory usage, as many subroutines may not be called at all.	• It also helps in efficient memory usage, as If the library or file already present in memory then the stub replace itself.
Example	• the Apache Web Server's *.dso "dynamic shared object" are libraries which are loaded at runtime.	• An example is the shared object version of the standard C library, libc.so.

Q3. Which component of an operating system is best suited to ensure fair, secure, orderly, and efficient use of memory? Also identify some more tasks managed by that component.

Answer:

- Memory management system is most suitable component of an operating system that ensure fair, secure, orderly and efficient use of memory.
- The **tasks managed by memory management system** includes keeping track of where, when, and how much memory is allocated and free.
- It also keeps track of used and free memory spaces.
- Memory management system is responsible for the processes swapping and in/out from primary/main memory.
- So, the purpose of memory management system is to ensure fair, secure, orderly, and efficient use of main memory as it is limited and one of the most important resource of computer system.

Q4. Differentiate between Symmetric and A-Symmetric encryption with the help of example.

Answer:

The main differences between symmetric and A-symmetric encryption is given below:

S.No	**Symmetric encryption**	**A-symmetric encryption**

1.	• In symmetric encryption a single key is used for encryption and decryption.	• In A-symmetric encryption two separate keys is used one to encrypt and the other one to decrypt.
2.	• Symmetric Key is transfer to one party from another.	• A-symmetric Key Encryption is based on public and private key encryption technique.
3.	• Symmetric encryption process is very fast.	• A-Symmetric encryption process is very slow.
4.	• It is used when a large amount of data is required to transfer.	• It is used when less/small amount of data is required to transfer.
5.	• It only provides privacy.	• It provides privacy, authenticity and acceptance.
6.	• Examples: 3DES, DES and RC4 etc.	• Examples: Diffie-Hellman Gamal, and RSA etc.

Q5. Describe the difference between external and internal fragmentation. Why should they be avoided?

Answer:

The min differences between external and internal fragmentation is given below:

	Internal fragmentation	**External fragmentation**
Introduction	• It refers to extra/wasted allocated spaces of memory assigned to different processes. • It is due to fixed size memory blocks are allocated to process/program.	• External fragmentation refers to unused spaces that are formed between the blocks of memory which are not together to each other.
Occurrence	• It occurs when the allocated memory is larger than requested memory of process.	• It occurs due to formation of holes in memory that are too small as compared to request.
Reason	• Why it occurs when the main memory is divided into fixed-size blocks?	• Why it occurs when main memory is divided into variable size blocks?

Solution	• It is natural phenomena and it can be eliminated by dynamic allocation of memory.	• It can be eliminated by compaction, paging, and segmentation so that allocation will done in a non-contiguous manner.

<u>Why should they be avoided?</u>

- Due to fragmentation, the system fails in allocation of contiguous memory to a process but allocate in non-contiguous manner.
- fragmentation badly effect the data accessing speed of computer system.

Q6. List and describe the four memory allocation algorithms covered in lectures. Which two of the four are more commonly used in practice?

Answer:

Operating system uses the following memory allocation algorithms:

I. **Single-Partition allocation:**
Memory is divided in single partitions.
In this type of allocation, relocation register scheme is used to changing OS code, data and protect user process.
Relocator register contains values of smallest physical addresses whereas limit register contains range of logical address, that must be less than the limit register.

II. **Multiple-Partition allocation:**
Memory is divided into multiple/fixed-size partitions where each partition should contain only one process.
Process is assigned to free partition and when the process terminated, the partition become available for other processes.

- In dynamic allocation, picking a portion of memory from the free list can be done using first-fit, next-fit, best-fit and worst-fit. **First-fit** and **next-fit** are more commonly used in practice.

a) **First-fit:**
The information is placed in the principle section, it limits the measure of looking.

b) **Next-fit:**
The information is placed in last portion. It is like originally fit, looking from the earliest starting point each other.

Q7. Why is the context switch overhead of a user-level threading as compared to the overhead for processes? Explain.

Answer:

This replacement of context switch overhead of a user-level threading as compared to the overhead for processes is known as setting switch. In which time is complete overhead, due to it no valuable work is achieve during exchange. When it is cyclic, the setting of exchanging is hard as well as the using of processor therefore no user code will be excited hence no completion of reasonably gainful registering is occurred.