1. (7+7+6=20 pts)

(1) What is the relationship between cloud computing and distributed computing?

* 1. Cloud computing is used to capture this vision of computing as a utility. A cloud is defined as a set of Internet-based application, storage and computing services sufficient to support most users’ needs. It allows very simple computer or desktop to access the potential wide range resources and services.
  2. Distributed computing is the use of distributed systems to solve single large problems by distributing tasks to single computers in the distributing systems.
  3. Cloud computing is the use of network hosted servers to do several tasks like storage, process and management of data. Here we will give an in-depth analysis of the two.
  4. Distributed computing is the technique that divides a single task into multiple tasks and distributes them over several computers. These computers can communicate and coordinate the activities by exchanging messages through the network.
  5. Cloud computing delivers hosted services over the internet. While distributed computing is the computing technique that allows multiple computers to communicate and coordinate work by passing messages through the network to achieve a common task.
  6. Distributed computing can simply be defined as sharing of tasks by different computers which may be in different parts of the globe. The distributed system which is used here must be in networked computers so that communication and coordination of the tasks is handled smoothly. The main goal of distributed computing is to connect the users with the resources thereby maximizing the performance in a cost-effective way.
  7. Distributed computing is when multiple autonomous machines communicate through a central network to perform a common goal.
  8. Cloud Computing is more about computing as a service, that is given to a computer over a network.
  9. Yet another simplified compare:

A screenshot of a cell phone

Description automatically generated

* 1. Cloud computing provide a platform, infrastructure for application, file storage and other resources and hosting to customers on a pay-as-you-go basis. Customers do not own the physical infrastructure and no need to depend on the storage capacity of their local computer network rather they rent the usage from a third-party provider.

* 1. Distributed computing is actually a distributed system which solves a large task by breaking it into several tasks where each task is computed in the individual computer of this distributed system and the user feels as his task is being performed on a single coherent system. All these computers are connected with each other to complete a task by using their own memory and pass messages synchronously or asynchronously between them. Distributed computing is mostly used to solve the complex computational problems that cannot be resolved by a single computer within a reasonable time

(2)  In your understanding, what are the most important characteristics of cloud computing

1. Provides a platform, infrastructure for application, file storage and other resources and hosting to customers on a pay-as-you-go basis.
2. In another words, it delivers hosted services over the internet.
3. Customers do not own the physical infrastructure and no need to depend on the storage capacity of their local computer network rather they rent the usage from a third-party provider.
4. Cloud computing significantly reduced the cost of computation, content storage, and application hosting.
5. Capture computing as a utility.
6. It is defined as a set of Internet-based application, storage and computing services sufficient to support most users’ needs.
7. It allows very simple computer or desktop to access the potential wide range resources and services.

(3)  Provide two example cloud computing applications.

1. Amazon Web Services (AWS) provides on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered pay-as-you-go basis.
2. In aggregate, these cloud computing web services provide a set of primitive abstract technical infrastructure and distributed computing building blocks and tools.
3. One of these services is Amazon Elastic Compute Cloud (EC2), which allows users to have at their disposal a virtual cluster of computers, available all the time, through the Internet.
4. AWS's version of virtual computers emulates most of the attributes of a real computer, including hardware central processing units (CPUs) and graphics processing units (GPUs) for processing; local/RAM memory; hard-disk/SSD storage; a choice of operating systems; networking; and pre-loaded application software such as web servers, databases, and customer relationship management (CRM).

Another example: Financial trading markets.

1. The financial industry need for real-time access to a wide range of information sources (for example, current share prices and trends, economic and political developments). The industry employs automated monitoring and trading applications
2. Note that the emphasis in such systems is on the communication and processing of items of interest, known as events in distributed systems, with the need also to deliver events reliably and in a timely manner to potentially very large numbers of clients who have a stated interest in such information items.
3. This requires a very different style of underlying architecture from the styles mentioned above (for example client-server), and such systems typically employ what are known as distributed event-based systems. We present an illustration of a typical use of such systems below and return to this important topic in more depth.
4. This style of technology is increasingly being used in other areas of financial systems including the monitoring of trading activity to manage risk (in particular, tracking exposure), to ensure compliance with regulations and to monitor for patterns of activity that might indicate fraudulent transactions. In such systems, events are typically intercepted and passed through what is equivalent to a compliance and risk firewall before being processed
5. Yet a figure quoted from book “DISTRIBUTED SYSTEMS Concepts and Design” 5th edition.

A close up of a logo

Description automatically generated

**A close up of a necklace

Description automatically generated**

Sub funcion

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

count := count − 1;

if count != 0 then quit;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Main function

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

S1;

count\_A := 2; // prepare to fork L1

fork L1; // fork S3 !!!

s2;

s4;

goto L2; // join A

count\_B := 2; // prepare to fork L2

fork L3; // fork S5 !!!!

S6;

goto L4; // join B before S7

S7;

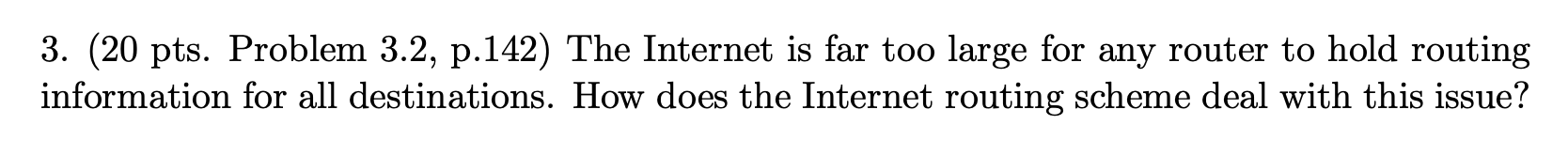
L1: S3;

L2: join count\_A;

L3: S5;

L4: join count\_B;

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



* + 1. If a router does not find the network id portion of a destination address in its routing table,
    2. it sends the packet to a default address an adjacent gateway or router that is designated as responsible for routing packets for which there is no routing information available.
    3. Each router’s default address carries such packets towards a router than has more complete routing information, until one is encountered that has a specific entry for the relevant network id.

4

**(1) Describe the dinning philosophers problem. Is this problem also an example of the critical section problem? Please briefly explain.**

1. There are K philosophers seated around a circular table with one chopstick between each pair of philosophers. There is only one chopstick between each philosopher.
2. A philosopher may eat if he can pick up the two chopsticks adjacent to him.
3. One chopstick may be picked up by any one of its adjacent followers but not both.
4. When the K philosopher ALL pickup their left hand side chopstick at the same time, the table becomes a deadlock
5. The dining philosopher is a classic synchronization problem as it demonstrates a large class of concurrency control problems.
6. Yet another image demonstrating the scenario:

A picture containing clock

Description automatically generated

1. And yes, it is an example of critical section problem.
2. It includes the need to allocate limited resources among multiple processes to avoid deadlock situation.
3. However, when the table is very large, it may not necessary be a critical section problem.

**(2) Provide a solution to the dinning philosophers problem using only semaphores.**

A semaphore object maintains a private integer which can only be accessed by the operations P and V.

These are declared as synchronized which means the procedures(methods) execute indivisibly on the semaphore’s value when they are invoked by different threads. And semaphore is a tool based on process synchronization.

It defines two functions as wait() and post().

Here wait() is used when access to critical section and post() is used when leaving the critical section.

Lets initialize the semaphore as s,

1. Pseudo code:

wait(s):

s.value=s.value-1;

if (s.value≤0)

{

add the process to s.list of process

block

}

post(s):

s.value=s.value+1;

if(s.value<0)

{

remove a process P from s.list of processes

wake up(p)

}

value=1: means there is no process in waiting queue and the critical section is empty.

value=0: means there are no processes in waiting queue and the critical section is occupied by a process.

value=-2: means there are 2 processes in waiting queue.

1. Real Java code

A screenshot of a computer screen

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screen shot of a computer

Description automatically generated