Pooja Baba
#002677117
09.15.2022

CSC 8980
Distributed Systems Fall 2022
Homework #1

1. (20pts) Nowadays, we frequently use a Global Positioning System (GPS) Receiver to determine our location and to find the shortest way to a specific destination. GPS technology uses satellites that continuously send certain signals to earth. How does the receiver utilize these signals to determine its dancer's current location? *ANS: Using **four** satellites, it is possible to determine the current location of anyone. The receiver does so by measuring the time it takes for a signal to arrive at its location from at least four satellites*
   To what precision can the GPS determine a location? *ANS: GPS-enabled smartphones are typically accurate to within a 4.9 m (16 ft.) radius under open sky*
   Is it possible to determine the hiker's altitude?*(ANS: Yes)* (How about the hiker's attitude? ☺)
   Research the workings of a GPS system and find answers to the questions above. Specifically, determine the following:
   a. how many satellites are participating in GPS system? Why?
      ***ANS**: As of 25 June 2022,*
      ***78** GPS satellites have been built:*
      - *31 are launched and operational,*
      - *3 are unhealthy or in reserve,*
      - *41 are retired,*
      - *2 were lost during launch,*
      - *1 prototype was never launched.*

      ***WHY**? The constellation requires a minimum of **24 operational** satellites, and allows for up to 32; typically, 31 are operational at any one time. (Ref: https://en.wikipedia.org/wiki/List_of_GPS_satellites)*
   b. how many satellites are needed to determine a location? Why?
      ***ANS**: A GPS receiver needs **four** satellites to work out its position in three dimensions.*
      ***WHY**? Three to determine a position on the Earth, and one to adjust for the error in the receiver's clock. The first satellite locates you somewhere on a sphere. The second satellite narrows your location to a circle created by the intersection of the two satellite spheres. The third satellite reduces the choice to two possible points. Finally, the forth satellite helps calculate a timing and location correction and selects one of the remaining two points as your position.*
      (Ref: https://gis.stackexchange.com/questions/12866/why-does-gps-positioning-require-four-satellites)
   c. how many satellites are required to determine one's altitude? Explain!
      ***ANS**: Total **4** satellites are required to determine one's altitude. An atomic clock synchronized to GPS is required in order to compute ranges from these three signals. However, by taking a measurement from a fourth satellite, the receiver avoids the need for an atomic clock. Thus, the receiver uses **four satellites** to compute latitude, longitude, altitude, and time. (Ref: https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/techops/navservices/gnss/gps/howitworks#:~:text=An%20atomic%20clock%20synchronized%20to,longitude%2C%20altitude%2C%20and%20time.)*
   d. what synchronization is required between satellites and receivers? Explain!
      ***ANS**: In addition to longitude, latitude, and altitude, the Global Positioning System (GPS) provides a critical fourth dimension – time. Each GPS satellite contains multiple atomic clocks that contribute very precise time data to the GPS signals. GPS receivers decode these signals, effectively synchronizing each receiver to the atomic clocks. This enables users to determine the time to within 100 billionths of a second, without the cost of owning and operating atomic clocks. (Ref: https://www.gps.gov/applications/timing/)*
   e. what is WAAS?
      ***ANS**: It is an abbreviation to Wide Area Augmentation System. It is an air navigation aid developed by the Federal Aviation Administration to augment the GPS with the goal of improving its accuracy, integrity and availability. Essentially, it is intended to enable aircraft to rely on GPS for all phases of flight including precision approaches to any airport within its coverage area. (Ref: https://en.wikipedia.org/wiki/Wide_Area_Augmentation_System)*
   f. why is a general-purpose GPS receiver less precise than a military-grade receiver?

Pooja Baba
#002677117
09.15.2022

***ANS**: The user range error (URE) of the GPS signals in space is actually the same for the civilian and military GPS services. However, most of today's civilian devices use only one GPS frequency, while military receivers use two. Using two GPS frequencies improves accuracy by correcting signal distortions caused by Earth's atmosphere. Dual-frequency GPS equipment is commercially available for civilian use, but its cost and size has limited it to professional applications. (Ref: https://www.gps.gov/systems/gps/performance/accuracy/)*

2. (20pts) Consider a system of 9 processes, **P** = {p1, …, p9} Associated with the system are 6 memory cells, **M** = {M1, .., M6}

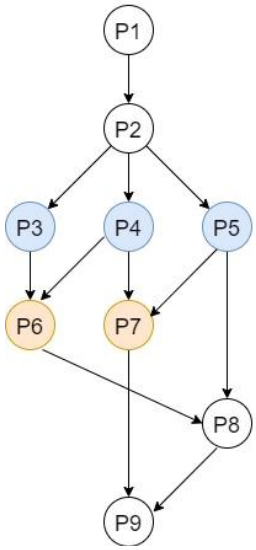
The domain and range for each process is given in the following table:

Process pi	Domain D(pi)	Range R(pi)
p1	M1, M2	M3
p2	M1	M5
p3	M3, M4	M1
p4	M3, M4	M5
p5	M3	M4
p6	M4	M4
p7	M5	M5
p8	M3, M4	M2
p9	M5, M6	M6

In addition, you are given the following precedence relation:
➔ = {(1,2),(1,6),(2,3),(2,4),(2,5),(3,6),(3,8),(4,6),(4,7),(5,7),(5,8),(6,8),(6,9),(7,9),(8,9)}

a. Construct the Precedence Graph (not containing redundant/transitive edges) from the given precedence relation.
ANS: Below is the precedence graph for the above-given precedence relation. Edges ***(1,6), (3,8)*** & ***(6,9)*** have been eliminated since they were redundant edges.

Precedence Graph



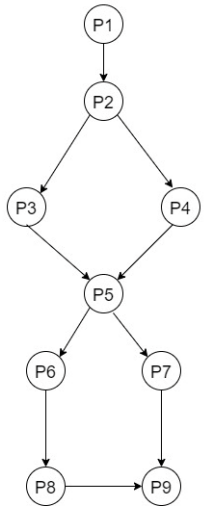
b. As per the given precedence relation, which processes can be executed concurrently?
ANS: As shown in the precedence graph for question a) –
- Processes ***P3, P4*** & ***P5*** can be executed concurrently at level 3
- Processes ***P6, P7*** can be executed concurrently at level 4

Pooja Baba
#002677117
09.15.2022

c. Make the necessary modifications to guarantee that the system is determinate for all interpretations fp.

ANS:

Determinate System

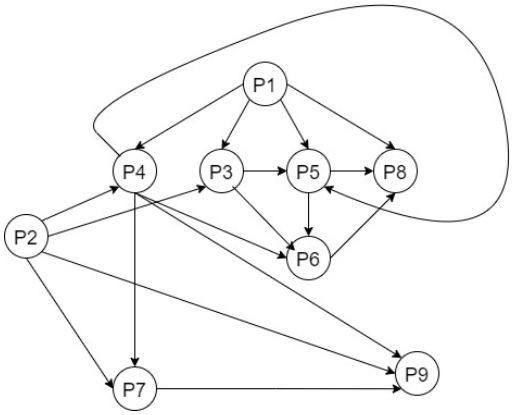


{(1,2), (2,3), (2,4), (3,5), (4,5), (5,6), (5,7), (6,8), (7,9)}

d. Modify the precedence relation above to construct a maximally parallel system.

ANS:

Maximally Parallel System



{(1,3), (1,4), (1,5), (1,8), (2,3), (2,4), (2,7), (2,9), (3,5), (3,6), (4,5), (4,6), (4,7), (4,9), (5,6), (5,8), (6,8), (7,9)}

Pooja Baba
#002677117
09.15.2022

3. (25pts) Outline a formal proof of the following theorem: *"A mutually non-interfering system of processes is determinate"*

PROOF:

- A system of processes is said to be determinate if, given the same input, the same results are produced regardless of the relative speeds of executions of the processes or the legal overlaps in execution. This indicates that the all the memory locations will have the same result despite the sequence in which the processes are being executed.
- A system of processes is said to be mutually non-interfering –
  - if any two processes meet the above Bernstein conditions –

    a)$D(Pi) \cap R(Pj) = \emptyset$

    b)$D(Pj) \cap R(Pi) = \emptyset$

    c)$R(Pi) \cap R(Pj) = \emptyset$

    OR

    $(D(Pi) \cap R(Pj)) \cup (D(Pj) \cap R(Pi)) \cup (R(Pi) \cap R(Pj)) = \emptyset$
  - Or one is the predecessor of the other.

In the case of mutually non-interfering systems, the processes will follow Bernstein's conditions and not lead to any conflicts as per the definition of the determinate system. During the execution of the processes in the mutually non-interfering system, the results produced will be the same regardless of the sequence of the processes in execution.

Hence considering the above definitions and the conditions, we can say that a mutually non-interfering system of processes is determinate.

4. (40pts) Write a simple sequence-number system through which three concurrent processes, P1, P2, and P3, **each** obtain unique integers in the range [1, 500]. Use the fork() call to create P1, P2, and P3. Given a file, F, containing a single number, each process performs the following steps:

a. Open F.
b. Read the sequence number N from the file.
c. Close F.
d. Output N and the process' PID (either on screen or test file).
e. Increment N by 1
f. Open F.
g. Write N to F.
h. Flush F.
i. Close F.

Describe the behavior of the three processes and explore the reason for this behavior. Provide evidence for your conclusion in the form of test-output. You must clearly document your code. Explain WHY the sequence number file must be **located on the local disk.**
HINT: On Linux, the \tmp directory is located on the local file system. What do you observe when the sequence number file resides in your home directory?

CODE:

```
import os
import time

file_name = "int.txt"

# method to read content from file - Opening text file in read mode
def check_file_content():
```

Pooja Baba
#002677117
09.15.2022

```
    file1 = open(file_name, 'r')
    time.sleep(.1)
    file_value = int(file1.read())
    file1.flush()
    file1.close()
    return file_value

# method to increment counter and write back to file - Opening text file in write mode
def read_write_file():
    file_content = check_file_content()
    file_content+=1
    print('{0} printing {1} to file.'.format(os.getpid(), file_content))
    file1 = open(file_name, 'w')
    file1.write(file_content.__str__())
    file1.flush()
    file1.close()

# main method to be executed
def parent_child():
    # loop to create 3 processes P1, P2, P3
    for process in range(1,4):
        # fork call to create a child process
        pid = os.fork()
        if pid==0:
            # execute the file read and increment count till the number in the file is < 500
            while check_file_content()<500:
                print("created process #{0}: {1} ".format(process, os.getpid()))
                read_write_file()
    os._exit(0)

# method call
parent_child()
```

INPUT FILE:
Initial content: 1
Final content: 500

OUTPUT:

```
created process #1: 23355
23355 printing 4 to file.
created process #1: 23355
23355 printing 7 to file.
created process #1: 23355
23355 printing 10 to file.
…
created process #2: 23356
23356 printing 3 to file.
created process #2: 23356
```

Pooja Baba
#002677117
09.15.2022

```
23356 printing 6 to file.
created process #2: 23356
23356 printing 9 to file.
created process #2: 23356
23356 printing 13 to file.
created process #2: 23356
23356 printing 16 to file.
…
created process #3: 23357
23357 printing 2 to file.
created process #3: 23357
23357 printing 5 to file.
created process #3: 23357
23357 printing 8 to file.
created process #3: 23357
23357 printing 11 to file.
created process #3: 23357
23357 printing 14 to file.
created process #3: 23357
23357 printing 17 to file.
```

OBSERVATION:

1. The code file and the text file were placed in the home directory during the execution of the above code.
2. Using the fork command, the three processes were created.
3. Every process reads the content of the file by opening the file in reading mode and performs the operation of incrementing the value read from the file. After reading the content from the file, the file is closed by the process.
4. Every process later, as per the algorithm, increments the value by one and reopens the file in write mode. Writes the incremented value to the file. flushes the file and closes it.
5. During this time, while the flush operation is being carried out, the internal buffer of the file is being cleared. Here when another process is trying to read the content from the file, the content isn't available for the process. Hence the process breaks.
6. In the case of the above code being run, this happened for process#1 and process#2. Finally process#3 continues for the rest of the time, i.e. the time till the range is satisfied.