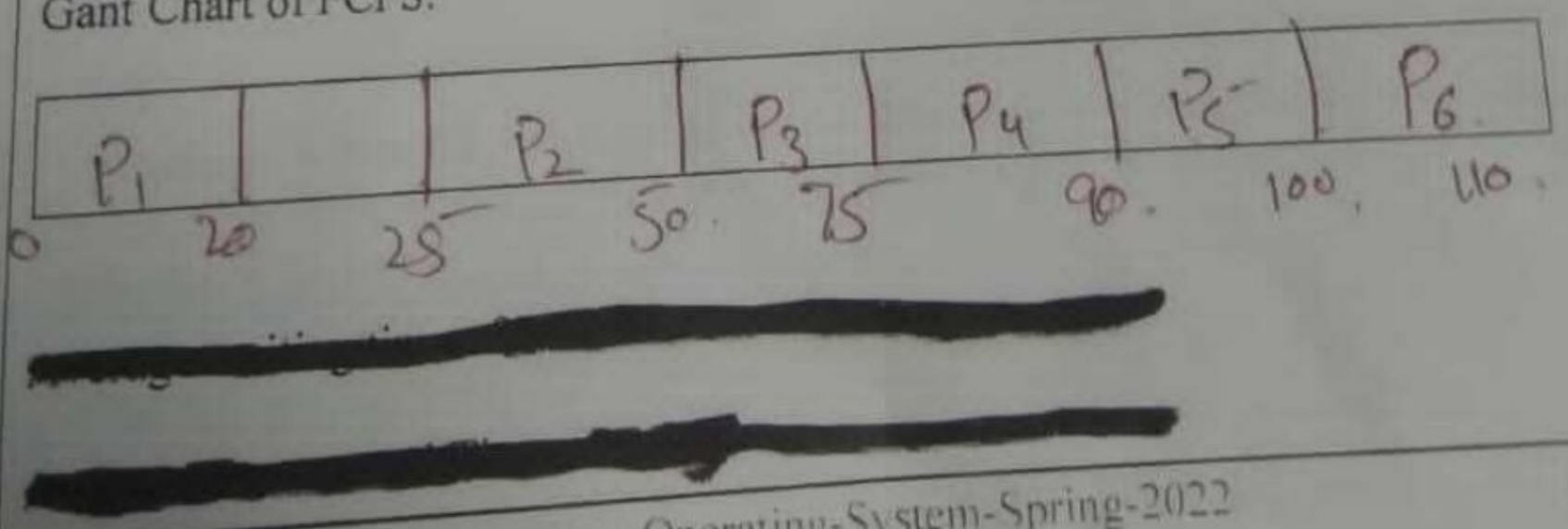
Question # 1 Complete the following code that creates 5 child processes and waits until all of the processes return? [Marks:10]

```
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
#include <stdlib.h>
#include <sys/wait.h>
int main(void)
{
        int n=5, i=0, status;
        pid_t pid[5];
        printf("Creating %d children\n", n);
        for(i=0 ; i<n ; i++)
        {
                ---Pid[i]---=---fork(--);
                if (pid < 0) /* If an error occurred */
                {
                        --Printf(--"error--Creating--child.");
                        ----break;----------
                }
                else if(pid == 0)
                ----------------break;--------
        }
        if(i>0 and pi)
        {
                        For (int j=0 ; j < i ; j++)
                                {
                                  ----wait(&status);
                                }
        }
        return 0;
}
```

Question # 2 Calculate the average waiting time and average turnaround time of the processes using FCFS, SJF and SRTF for the following set of processes. Draw Gantt chart clearly. [Marks:5*3=15]

Process	*Burst*	*Arrival Time*
P1	*20*	*0*
P2	*25*	*25*
P3	*25*	*30*
P4	*15*	*40*
P5	*10*	*50*
P6	*10*	*100*

Gant Chart of FCFS:



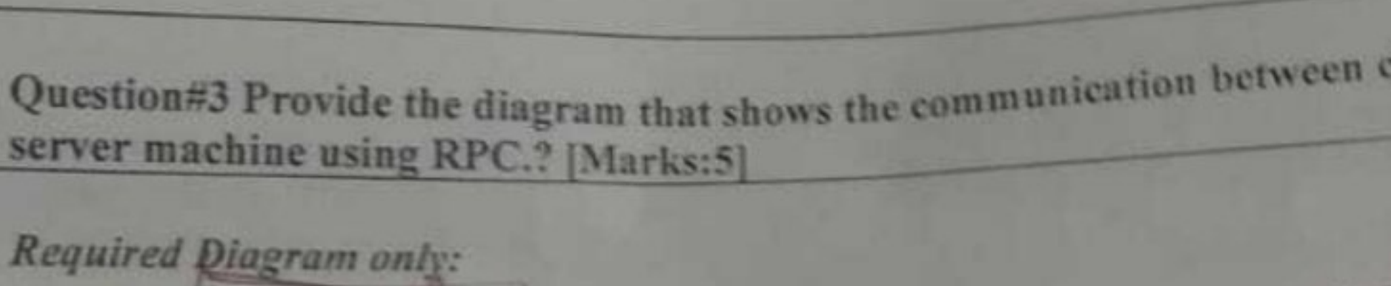
complete the following code that creates a ... ts until

0178

Average waiting time of FCFS: 0+20+25+45+50+60+75 / 6 = 15.83

Average Turnaround Time of FCFS: 20+25+45+50+60+75+100 => 31.67

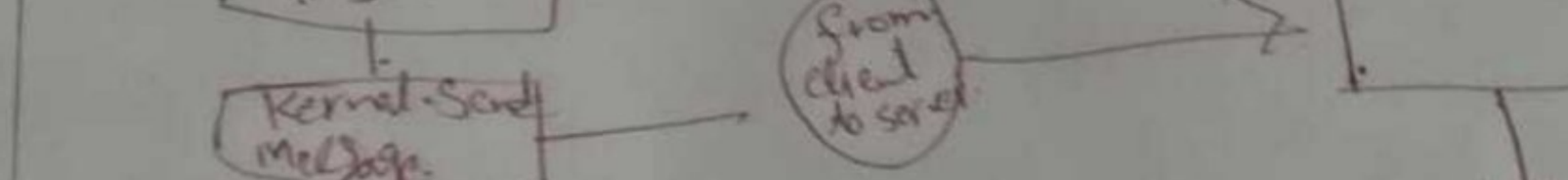
Gant Chart of SJF:



Average waiting time of SJF: 0+0+55+20+75+10 ..... 12.5

Average Turnaround Time of SJF: 20+25+70+20+10 25.83

Gant Chart of SRTF:



Average waiting time of SRTF: 0+0+55+20+0+10 => 12.5

Average Turnaround Time of SRTF: 25.83

Question#3 Provide the diagram that shows the communication between client and server machine using RPC.? [Marks:5]

Required Diagram only:

reference Slides.

Book chapter end.

Question# 4 What is a Zombie process? What are the benefits of having a zombie and what are the drawbacks? How can a zombie process be avoided in OS? [Marks:4+3+3=10]

What is... Completed it's Process but still remain in PCB. coz of Parent Process not call wait Signal.

Question # 1 Complete the following code that creates 5 child processes and waits until all of the processes return? [Marks:10] 0037

National University of Computer and Emerging Sciences
FAST School of Management Chiniot-Faisalabad Campus

0178

Benefits: Active Process id that not assigned any other Process

Avoidance of Zombie Processes: Parent Process force to intake wait system call.

Question # 5: Specify Three ways of Implicit Threading and explain one with example. [Marks:3+2=5]

1. Thread pool
2. OpenMP
3. Pooling

Name of any one: Anyone

Explanation: ref slides (Detail Mentioned).

Question # 6: Explain what is named pipes and ordinary pipes with the help of comparison. Provide the procedural steps/ code of communication between two processes using ordinary pipes? [Marks:2+2+6=10]

Named Pipes: to extent duplex relationship

Ordinary Pipes: Parent child relationship ← Parent

Set the file descriptor Pdfs[1]. Pdfs[0]

Procedural Steps:

Step 1: Create Process.

- set for child Pdfs[1] Pdfs[0]
- close one end of alternate end of
- Parent child end.

Step N:

Question # 7: Take an example *"Hello Ali I am Alina"* and dry Run the code of Producer Consumer Problem by First writing the code/Algorithm? [Marks: 2.5+2.5+2.5+2.5=10]

Producer is Alina:	Consumer is Ali:

while(true){ item next Produced;
while(((in+1)%BufferSize)==out); do nothing
	int next consumed.
	while(true)
	{
	while(in==out); do nothing
	out.read

0178

0037

buffer[in] = next_Product.
in = (in+1) % Buffer_Size.

next_consumer = buffer[out]
out = (out+1) % Buffer_Size
out = (out+1) % Buffer_Size

Dry RUN:

Iteration 1 Producer:	Hello
Iteration 2 Producer:	~~Hello~~ Ali
Iteration 3 Producer:	~~Hello~~ ~~Ali~~ I
Iteration 4 Producer:	Hello Ali I am
Iteration 5 Producer:	~~Hello Ali I am~~ Alina.

Iteration 1 Consumer:	Hello.
Iteration 2 Consumer:	Ali.
Iteration 3 Consumer:	I
Iteration 4 Consumer:	am
Iteration 5 Consumer:	alina.

Question # 8: Differentiate medium-term scheduler and short-term scheduler draw the block diagram how it works? [Marks:2+3+2+3=10]

Medium Term Scheduler: lower degree of Multiprogramming Partially Swap out the Process to meet the bounds.

Block Diagram:

ref: Slides. Book

Short Term Scheduler: To bring new Processes.

Block Diagram:

ref. slides Book

~~The Greatest gift is not afraid to Question~~
~Ruby Dee~