PRACTICAL - 7 me need 240191919 pens bodgeres 6/9/24 write a program to implement from control at data link layer using SLIDING WINDOW PROTOCOL simulate the flow of frames from one node to Program should achieve at least below given auother requirement. You can make it is vidirectional BILLIAN JEBRESERFORM OF THE OLODOBOLLOLOD program waxein receiver is sending its data frames with acknowledgement (Piggybacking) Dared with redundant pro 0100100000110100000 create a sender program with following feautures 1) Input windowsize from the user. Journes 2) Input a text mersage from Aperuser o some 3) considée planacte per france 109 est estus 4) create à frame with following fields Elframen Entres destroised out position. P DATAJ 5) send the frame [print the output on screen and saire it in a file called sender_Buffee] 6) wait for the asknowledgement from the Receive [Induce delay in the program] 7) Reader a file called Receiver-Buffel 8) cheek Ack field for the Acknowledgement number. Proposition does desired 102-2000 X2 DOME JURE JURE JURE

Justific

after S

Viswai

Solicita

Mehta,

State of

the bull

continue

court de

delines

demolitic

It was)

ixe", Just

ONTINUET

PAGE 12

Justin

9) If the Acknowledgement number is as expected, send new set of frames accordingly. PROGIRAM: sender Py import time import randon det sender (window_size, text_nessage): 109 bocksom with 41 district frames = [] # prepare frames from the tent message for i, char in enumerate (text-message): frames append ([i,char]) #write frances to sender-Buffer. the with open ("sender-Buffer. +xt", "w") as file: for frame in frames file write (f" > frame [0]}, {frame [1]}m") Start = 0 while start < leu (frames): #Get the current window of frame to send windows frames [start: start + window size] print (f'sending frames: fwindows") # write the window to sender - Buffer. + nt with open ("sender Buffer tat", "w") as fêle: for frame in window file write (f" ? frame [o]), ? france [i]] In") time sleep(1)

OL

frames

ubires:

Cframeno

mand

ceives.

Read the acknowledgement from Receiver - Buy "frame_n with open ("Receiver-Buffer. +xt", "n") as file. ack-line-file readline(). strip() #check if ack line is valid MODELO F DEGINI if ack we! ack-ho= int (ack-line split(",")[0]) print(f"Ack received for frame sack-no?") self ack nos start word werend swaging # : (spotastizack wo + tronounies in sono, i sof (Ecolo 37) busque 29just raise value Error ("Empty acknowledgement Line") else: except (Valuet mol, Inder Error) : vage strou print (f' Invalid & enepty ack-line, resending frames starting from Estarty") pront (" the frames sent successfully") # Example Usage window size = int (input ("Enter window Size: ")) text_message=enput= ("Enter the tent message:") sender (window-seze, tent-neesage) Receiver Buffer py: import time det read-sender buffer (filenance: "sender-Buffer + nt"): with open (filename, 171) as f: frames = f. readlines()

Pass

```
wee-Buffy)
              # passe each line assuming the format is
           "frame_no", characters"
sfile.
              passed france = []
              for live in frames!
                ef line stript)
                 parts-line.strip().split(",")
                 frame-no=int(parts[0])
                 chae = parts [1]
                 parsed frames append (frame-no, char)
               except (Inden Error, Value Error):
                 print (f" skipping malfonsred line: { line;")
t Live")
             return parsed-frames
                                       getter - 73
         det write-receiver-buffer(ack-list, filenanne = Receiver-
pue
                                    Buffer + xt "):
             with open (filename, 'w') as f:
              for ack in ack-like:
                 f. write (f" sack?, Acklu")
       det receiver():
                       expected-frame_no=0
           while True:
              frames = read sender - buffer ()
              print (f' Received frames: ? frames)")
             ack list = CJ
             for france no, data in frances:
```

Morthcation, and even The St nead lly guidell R. Justific after S 1019124 Viswr 4 frame no == expected-frame_no. Solich Mehte State o Justi the bul print (f" France & france no) continu court de delines DULL demolitie received successfully") It was axe", Just CONTINUE » PAGE 12 ack-list append (expected-frame-ho) expected-frame-no+=1 print (f "Frame & frame no) out of order expecting frames & expected frame no ack-list append (expected-frame-no-1) write-receiver-buffer (ack-list) time sleep (2) # Main deiver if -- name _- = " -- main _- ". receiver() output: Enfer window size = 2 Eute the tent wersage : Hello Received frames: [(0, 'H'), (1, 'e')] Frame o received successfully frame, regained successfully Received frames: [10, 'H', (1, 'e')] France o out of order, expecting frame? Frame, out of order, expecting frame? Result: Thus, the program for sliding window protocol is successfully executed and the output is verified.

Packet

Procede

1. Cres

a. Che

sevous

3. As

4. R

the

#

#

TI

#

#

1

4