رقم المقعد	الفصل	رقم الجلوس	اسم الطالب
50	**2**	**33115**	**صموئيل جورج عبد المسيح كيرلس**

Project Title (Ordered Delivery of Packets)

1 Overview

The Queue rearrange Packets in order and circular and Dropped Duplicated packets and out of window packets in Dropped array and Deliver Expected packets and when a sequence of expected ordered packets are received they are removed from queue.

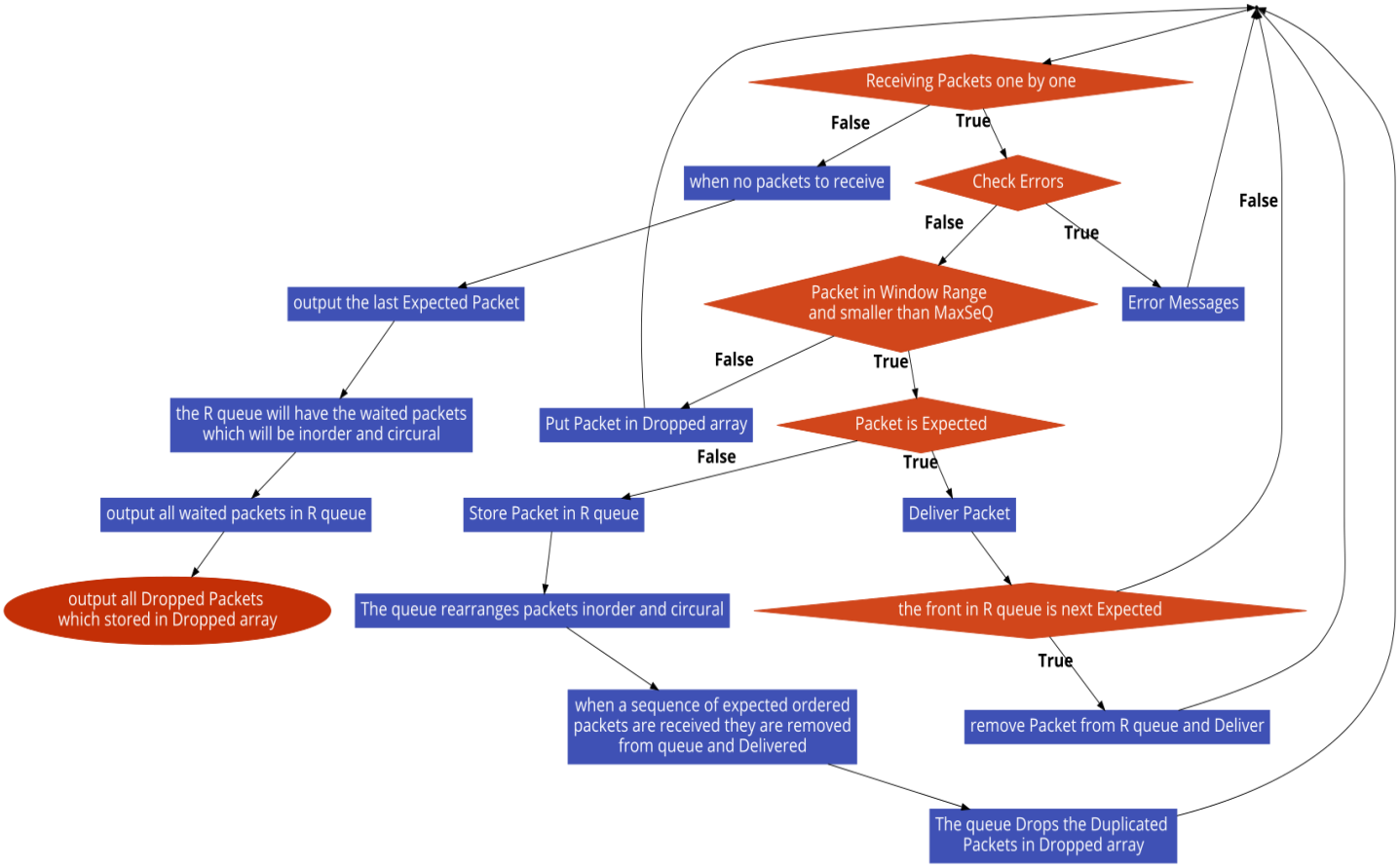
Packets SeqNumBits winSize initSeq pid 1 pid 2 pid 5 pid 4 pid 5 pid 6 pid 20

R Queue

Packets array[]	Pid 4	Pid 5	Pid 6	
Dropped array[]	Pid 5	Pid 20		

Output **R** pid1 pid2 **E** pid3 **W** pid4 pid5 pid6 **D** pid 5 pid 20

2 Main Flows



References

Format references as follows

[1] Lecture Slide Queues
[2] https://www.youtube.com/channel/UCRlDpEgLz9laAQ3lJPK99aA

2.1 Code Snippets

```
int main(int argc,char** argv)
{


    cout<<"R ";
    while(i <= argc-1)
    {
      if(!R.InWinSize(atoi(argv[i]),Expected,MaxSeq,winSize)) R.enqueue_Dropped(atoi(argv[i]));
// Drop packet if out of Window range
      else
      {
        if ( (atoi(argv[i]) < initSeq && atoi(argv[i]) < Expected) ||  !(atoi(argv[i]) < initSeq
&& atoi(argv[i]) < Expected) )   // Check if packet is Duplicated
        {
          if(atoi(argv[i]) == Expected)
          {
            cout<<atoi(argv[i])<<" ";                          // Deliver if packet Expected
            Expected = nextExpected(Expected,MaxSeq);
            while(!R.isEmpty())                                // Dequeue if front is Expected
            {
              if(R.front() == Expected)
              {
                cout<<R.dequeue()<<" ";
                Expected = nextExpected(Expected,MaxSeq);
              }
              else break;
            }
          }
          else R.enqueue(atoi(argv[i]),initSeq);
// Enqueue if packet waited for Expected
        }
        else  R.enqueue_Dropped(atoi(argv[i]));
// Drop if packet is Duplicated
      }
      i++;
    }
    cout<<"E "<<Expected<<" ";
    cout<<"W ";
    while(!R.isEmpty())   cout<<R.dequeue()<<" ";
    cout<<"D ";

    while(!R.Dropped_isEmpty())  cout<<R.dequeue_Dropped()<<" ";
  }

}
```