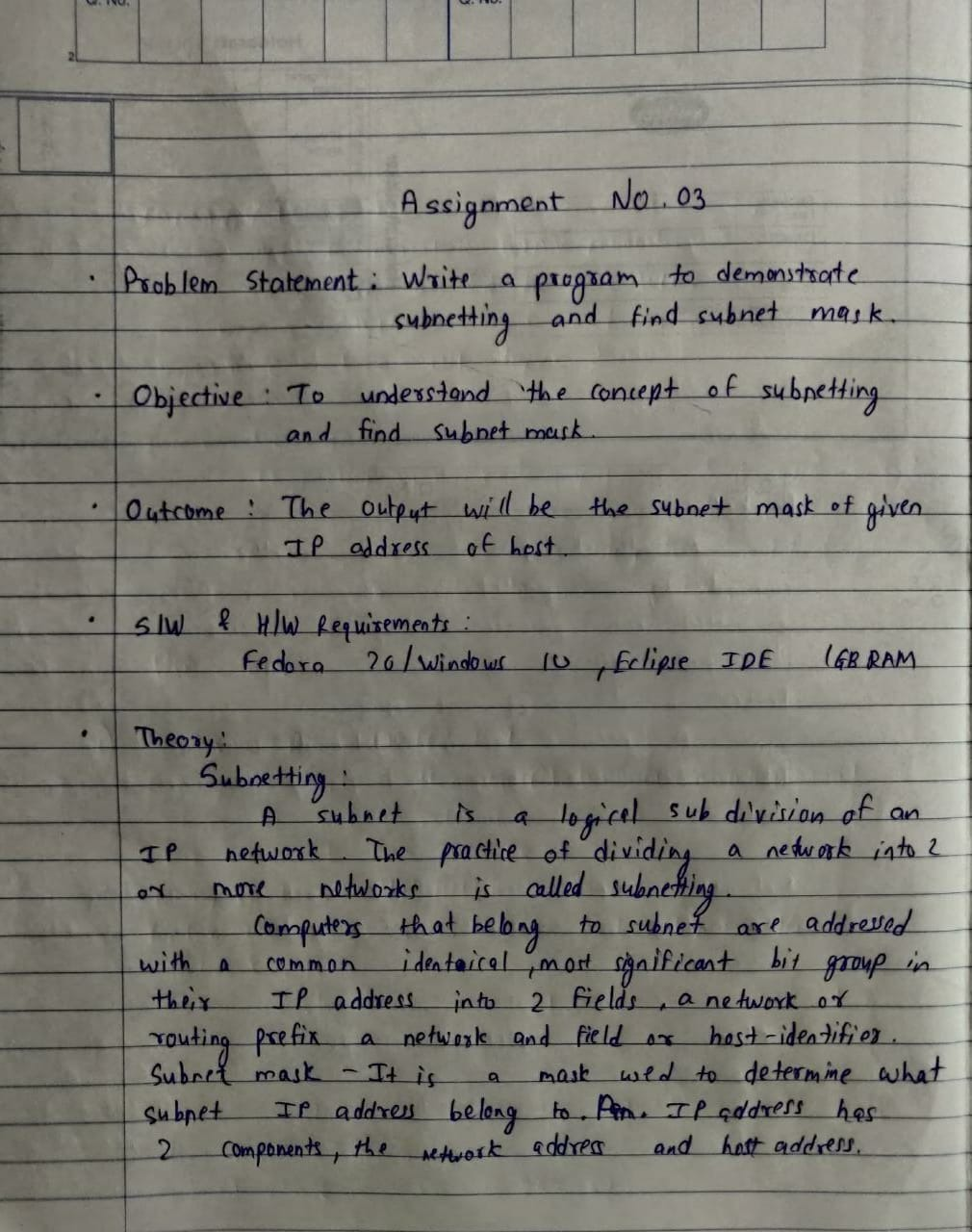
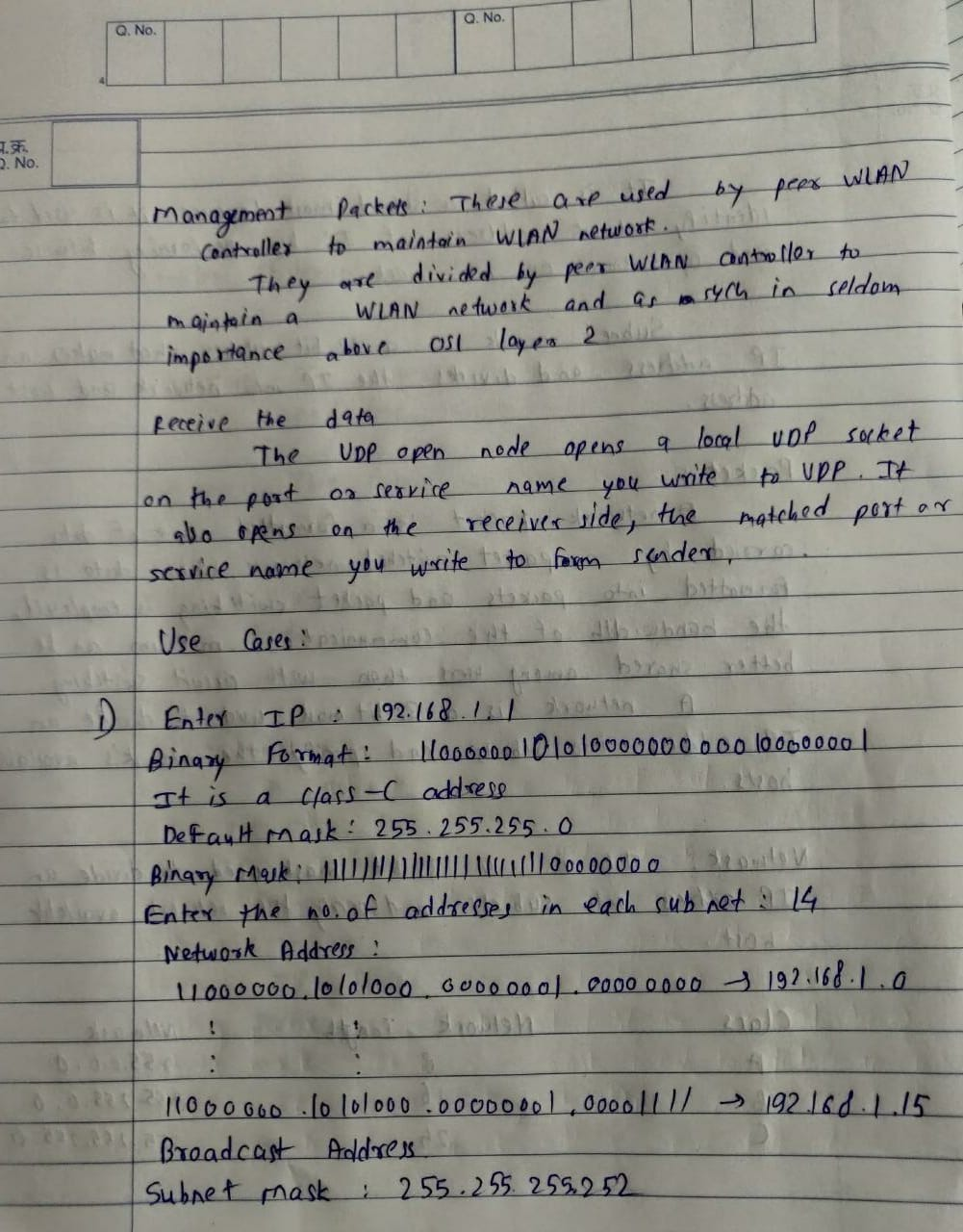
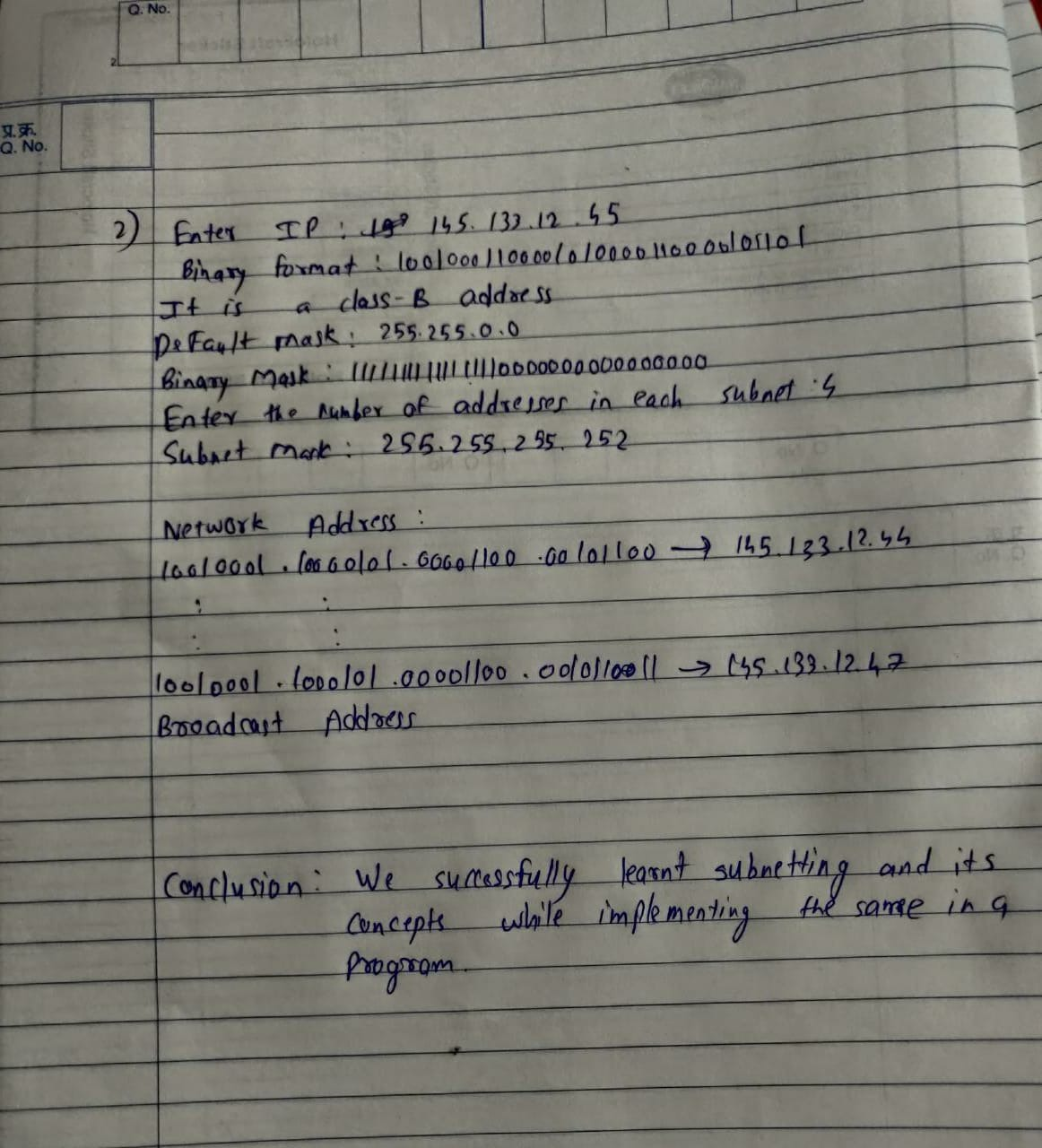
***CNL ASSIGNMENT 3 (Subnetting) (31139-Durvesh)***

***//Write-up***









***//SAMPLE CODE***

**from math import \***

**def IP2bin(ip):**

**ipsplits = ip.split('.')**

**binstr = ""**

**for num in ipsplits:**

**binstr += dec2binstr(num)**

**return binstr**

**def dec2binstr(dec):**

**binstr = bin(int(dec)).replace("0b","")**

**additionalbs = 8-len(binstr)**

**return "0"\*additionalbs+binstr**

**IP = input("Enter IP: ")**

**IPsplits = IP.split('.')**

**print("Binary Format: ",end="")**

**MainIP = IP2bin(IP)**

**print(MainIP)**

**classn = int(IPsplits[0])**

**if classn<=127 :**

**classip = "A"**

**elif classn<=191:**

**classip = "B"**

**elif classn<=223:**

**classip= "C"**

**elif classn<=239:**

**classip = "D"**

**else:**

**classip = "E"**

**print("It is a Class-"+classip+" address")**

**classiptonum = {"A":1,"B":2,"C":3,"D":4,"E":5}**

**classnum = classiptonum[classip]**

**DefaultMask = ("255."\*classnum+"0."\*(4-classnum))[:-1]**

**print("Default Mask: "+DefaultMask)**

**print("Binary Mask: ",end="")**

**DefaultMaskbin = IP2bin(DefaultMask)**

**print(DefaultMaskbin)**

**naddr = int(input("Enter the Number of addresses in each subnet: "))**

**host\_bits = ceil(log(naddr)/log(2))**

**additional\_bits = (8\*(4-classnum))-host\_bits**

**subnet\_mask = DefaultMaskbin[:classnum\*8]**

**subnet\_mask+="1"\*additional\_bits**

**subnet\_mask += "0"\*host\_bits**

**splits = [subnet\_mask[i:i+8] for i in range(0, len(subnet\_mask), 8)]**

**subnetmaskip = str(".".join(splits))**

**print("Subnet Mask: "+str(subnetmaskip))**

**decsubnet = ""**

**for sp in splits:**

**decsubnet+=str(int(sp,2))+"."**

**decsubnet = decsubnet[:-1]**

**print("Subnet Mask (in decimals): "+decsubnet)**

**print("Number of subnets: "+str(2\*\*additional\_bits))**

**print("Number of hosts in each subnet: "+str(2\*\*host\_bits-2))**

**## host bit number of LSBs should be 0 for network address and 0 for broadcast smthng**

**tempstr = MainIP[:-host\_bits]**

**NetworkAddress = tempstr + "0"\*host\_bits**

**broadAddress = tempstr + "1"\*host\_bits**

**netsplits = [NetworkAddress[i:i+8] for i in range(0, len(NetworkAddress), 8)]**

**netaddr = str(".".join(netsplits))**

**decnetaddr = ""**

**for sp in netsplits:**

**decnetaddr+=str(int(sp,2))+"."**

**decnetaddr = decnetaddr[:-1]**

**broadsplits = [broadAddress[i:i+8] for i in range(0, len(broadAddress), 8)]**

**broadaddr = str(".".join(broadsplits))**

**decbroadaddr = ""**

**for sp in broadsplits:**

**decbroadaddr+=str(int(sp,2))+"."**

**decbroadaddr = decbroadaddr[:-1]**

**print("Network Address:")**

**print(str(netaddr)+" -> "+decnetaddr)**

**print(": :")**

**print(": :")**

**print(str(broadaddr)+" -> "+decbroadaddr)**

**print("Broadcast Address")**

***//OUTPUTS***

