**Python programs (part 2)**

Q3. Convert IP address

|  |
| --- |
|  |

# importing the module

import ipaddress

# converting IPv4 address to int

addr1 = ipaddress.ip\_address('191.255.254.40')

addr2 = ipaddress.ip\_address('0.0.0.123')

print(int(addr1))

print(int(addr2))

# converting IPv6 address to int

addr3 = ipaddress.ip\_address('2001:db7:dc75:365:220a:7c84:d796:6401')

print(int(addr3))

|  |
| --- |
| Q4.  # Python program to check  # if a word is isogram or not  def is\_isogram(word):        # Convert the word or sentence in lower case letters.      clean\_word = word.lower()        # Make an empty list to append unique letters      letter\_list = []        for letter in clean\_word:            # If letter is an alphabet then only check          if letter.isalpha():              if letter in letter\_list:                  return False              letter\_list.append(letter)        return True    if \_\_name\_\_ == '\_\_main\_\_':      print(is\_isogram("Machine"))      print(is\_isogram("isogram"))      print(is\_isogram("GeeksforGeeks"))      print(is\_isogram("Alphabet "))  Q5.  s='hello'  new=[]  for i, val in enumerate(s[:]):      up=s[i].upper()      c=s[:i] + up + s[i+1:]      new.append(c)  print(new)  Q6.  # Python program to implement  # the above approach    def maxnumber(n, k):  # Function to return the  # largest number possible        for i in range(0, k):          # Generate the largest number          # after removal of the least K digits          # one by one          ans = 0          i = 1            while n // i > 0:          # Remove the least digit          # after every iteration              temp = (n//(i \* 10))\*i + (n % i)              i \*= 10          # Store the numbers formed after          # removing every digit once            # Compare and store the maximum              if temp > ans:                  ans = temp          n = ans        # Return the remaining number      # after K removals      return ans;      n = 6358  k = 1  print(maxnumber(n, k))  Q7.  # Python program to print the maximum number  # from the set of digits of a given number    # Function to print maximum number  def printMaximum(inum):        # Hashed array to store count of digits      count = [0 for x in range(10)]        # Connverting given number to string      string = str(num)        # Updating the count array      for i in range(len(string)):          count[int(string[i])] = count[int(string[i])] +  1        # Result stores final number      result = 0      multiplier = 1        # traversing the count array      # to calculate the maximum number        for i in range(10):          while count[i] > 0:              result = result + ( i \* multiplier )              count[i] = count[i] - 1              multiplier = multiplier \* 10        # return the result      return result    # Driver code  num = 38293367  print (printMaximum(num))    Q8  # Python3 code to find frequency of each word  # function for calculating the frequency  def freq(str):        # break the string into list of words      str\_list = str.split()        # gives set of unique words      unique\_words = set(str\_list)        for words in unique\_words :          print('Frequency of ', words , 'is :', str\_list.count(words))    # driver code  if \_\_name\_\_ == "\_\_main\_\_":        str ='apple mango apple orange orange apple guava mango mango'        # calling the freq function      freq(str)  Q9.  import matplotlib  print(matplotlib.colors.to\_hex([ 0.47, 0.0, 1.0 ]))  print(matplotlib.colors.to\_hex([ 0.7, 0.321, 0.3, 0.5 ], keep\_alpha=True))  print(matplotlib.colors.to\_rgb("#aabbcc"))  print(matplotlib.colors.to\_rgb("#ddee9f")) |