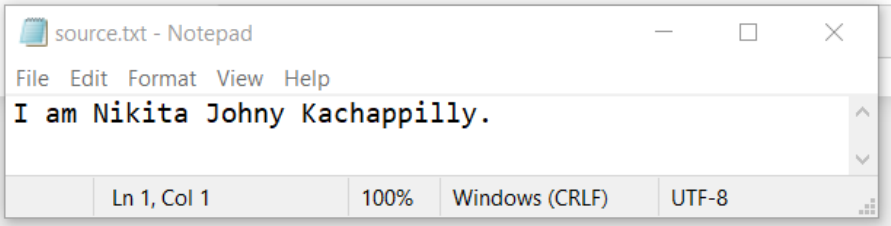
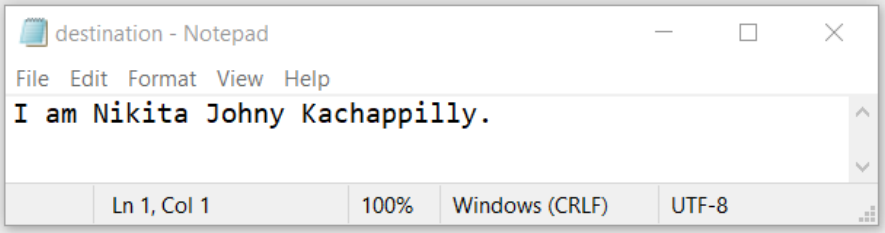
Files

1.Copy one file to another

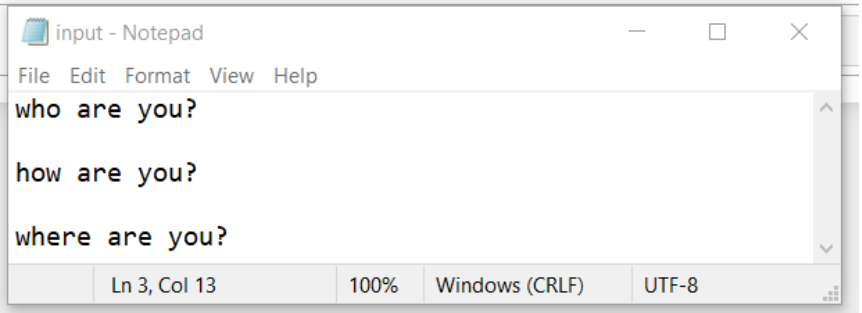
```
In [13]: f = open('source.txt','r')
         content = f.read()

         f1 = open('destination.txt','w')
         f1.write(content)
         f.close()
         f1.close()
```

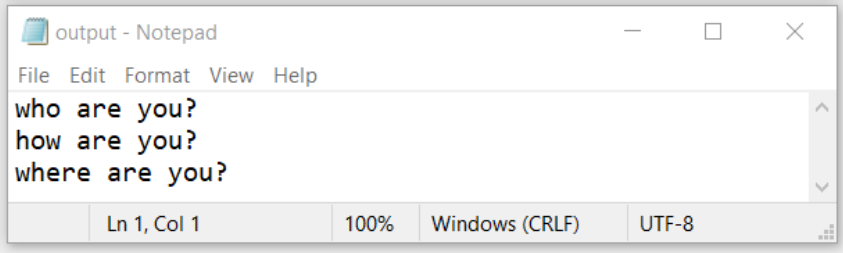




2.Copyone file to another after removing blank lines

```
In [24]: f1=open('input.txt','r')
         f2=open('output.txt','w')
         for line in f1:
             if line.strip():
                 f2.write(line)
```

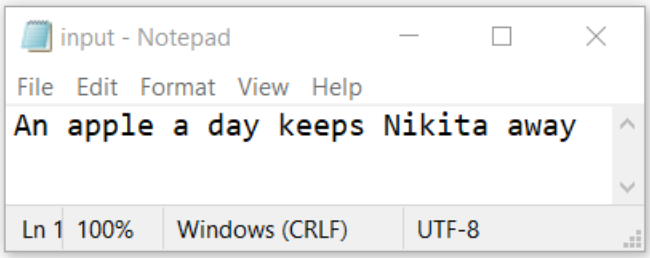




3.Print the words and length of words in sorted order of words from a file.

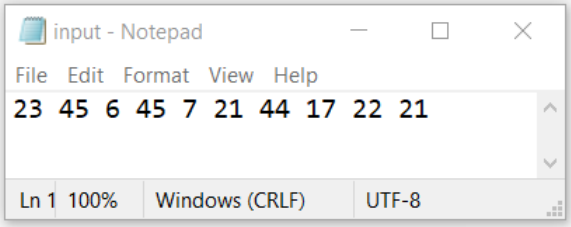
```
In [26]: file = open('input.txt', 'r')
         words = file.read().split()
         word_dict = {}

         for word in words:
             word_dict[word] = len(word)

         sorted_dict = dict(sorted(word_dict.items()))

         for word, length in sorted_dict.items():
             print(word, length)
```

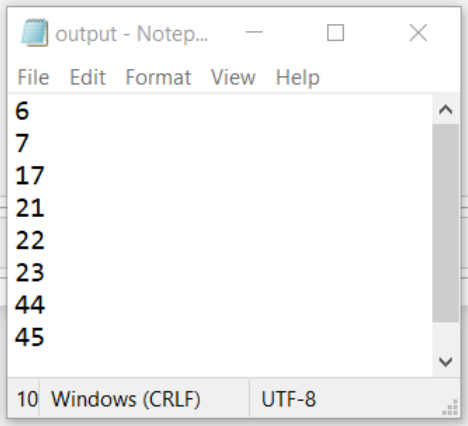
```
An 2
Nikita 6
a 1
apple 5
away 4
day 3
keeps 5
```



4. Read numbers stored in one file and store the sorted numbers in another file after deleting duplicates.

```
In [29]: f = open('input.txt', 'r')
         numbers = f.read().split()

         numbers = [int(x) for x in numbers]

         numbers = list(set(numbers))
         numbers.sort()

         f1 = open('output.txt', 'w')

         for num in numbers:
             f1.write(str(num) + '\n')
         f.close()
         f1.close()
```

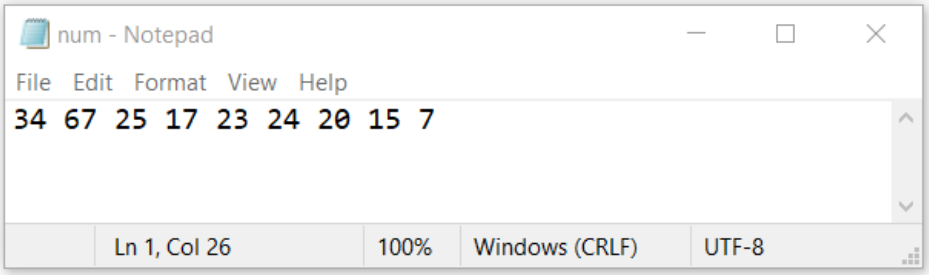


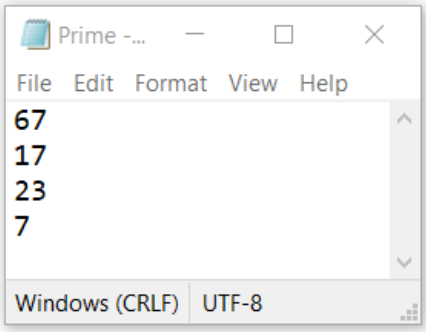


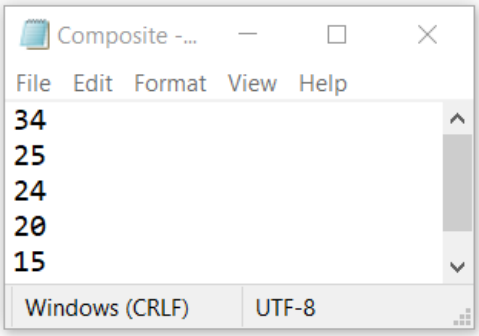

```
In [ ]:
```

5.List of numbers are stored in a file(num.dat). Create two file Prime (containing prime numbers) and composite(containing composite numbers) from the numbers stored in num.dat

In [32]:
```
import os
import math

f = open('num.txt', 'r')
numbers = [int(n) for n in f.read().split()]

prime_file = open('Prime.txt', 'w')
composite_file = open('Composite.txt', 'w')

for n in numbers:
    if n < 2:
        continue
    is_prime = True
    for i in range(2, int(math.sqrt(n))+1):
        if n % i == 0:
            is_prime = False
            break
    if is_prime:
        prime_file.write(str(n) + '\n')
    else:
        composite_file.write(str(n) + '\n')

prime_file.close()
composite_file.close()
```



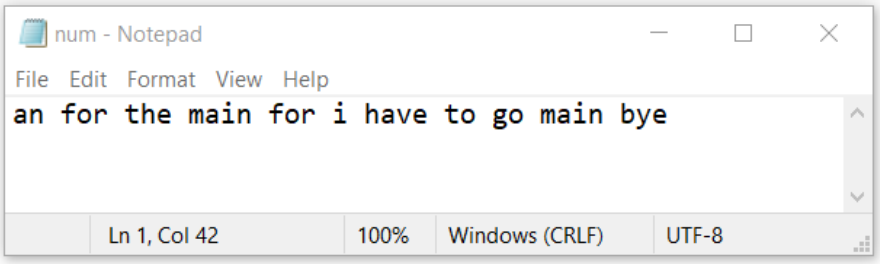




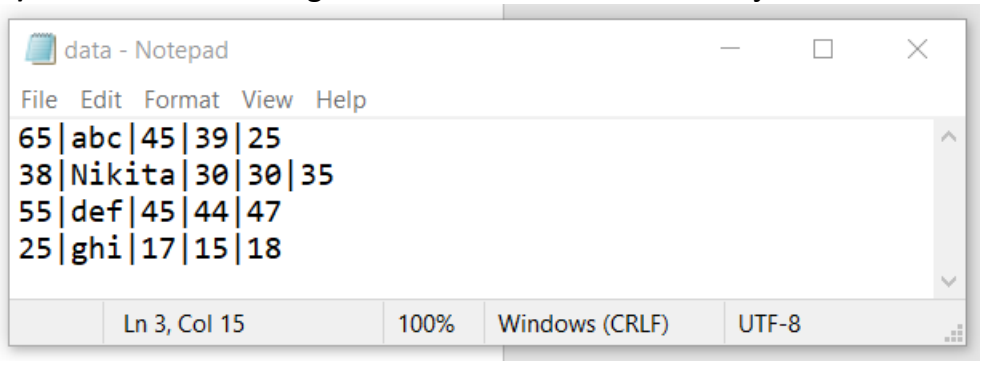
6.Find the words having largest frequency from the given file.

```
In [35]: file = open('num.txt', 'r')
         data = file.read()
         words = data.split()

         freq = {}
         for word in words:
             if word in freq:
                 freq[word] += 1
             else:
                 freq[word] = 1

         max_freq = max(freq.values())

         for word in freq:
             if freq[word] == max_freq:
                 print(word)
```

```
for
main
```



7.Create a data file in the following format(rno,name.m1,m2,m3 marks out of 50)
1|aswith|30|40|50
Write scripts for the following
a)Print the rno,name and Tot marks in the descending order of total marks (Rank list)
b)Print the list of passed students (marks >=25 for all three subjects)
c)Print the pass percentage in each subject
d)List of students having 80% or more in all the three subjects



In [5]:

```
# Open the file for reading
with open('data.txt', 'r') as f:
    data = f.readlines()

# Remove newline characters and split the fields
data = [line.strip().split('|') for line in data]

# Convert the marks to integers
for i in range(len(data)):
    data[i][2] = int(data[i][2])
    data[i][3] = int(data[i][3])
    data[i][4] = int(data[i][4])

# Compute the total marks for each student
for i in range(len(data)):
    data[i].append(data[i][2] + data[i][3] + data[i][4])

# Sort the data in descending order of total marks
data.sort(key=lambda x: x[5], reverse=True)

# Rank list
print("Rank List:")
print("Rno\tName\tTotal Marks")
for i in range(len(data)):
    print(f"{data[i][0]}\t{data[i][1]}\t{data[i][5]}")

# Passed students
passed = [x for x in data if x[2] >= 25 and x[3] >= 25 and x[4] >= 25]
print("\nList of Passed Students:")
print("Rno\tName\tTotal Marks")
for i in range(len(passed)):
    print(f"{passed[i][0]}\t{passed[i][1]}\t{passed[i][5]}")
```

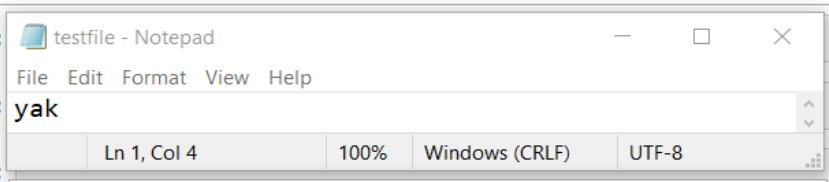
```
# Pass percentage in each subject
m1_passed = [x for x in data if x[2] >= 25]
m2_passed = [x for x in data if x[3] >= 25]
m3_passed = [x for x in data if x[4] >= 25]
print(f"\nPass Percentage in M1: {len(m1_passed)/len(data)*100:.2f}%")
print(f"Pass Percentage in M2: {len(m2_passed)/len(data)*100:.2f}%")
print(f"Pass Percentage in M3: {len(m3_passed)/len(data)*100:.2f}%")

# Students with 80% or more in all three subjects
high_scorers = [x for x in data if x[2] >= 40 and x[3] >= 40 and x[4] >= 40]
print("\nStudents with 80% or more in all three subjects:")
print("Rno\tName\tM1\tM2\tM3\tTotal Marks")
for i in range(len(high_scorers)):
    print(f"{high_scorers[i][0]}\t{high_scorers[i][1]}\t{high_scorers[i][2]}\t{
```

```
Rank List:
Rno     Name    Total Marks
55      def     136
65      abc     109
38      Nikita  95
25      ghi     50

List of Passed Students:
Rno     Name    Total Marks
55      def     136
65      abc     109
38      Nikita  95

Pass Percentage in M1: 75.00%
Pass Percentage in M2: 75.00%
Pass Percentage in M3: 75.00%

Students with 80% or more in all three subjects:
Rno     Name    M1      M2      M3      Total Marks
55      def     45      44      47      136
```

8. Write a Python program to create a text file. Read the contents of the file, encrypt every character in the file with a distance of 3 and write it to a new file.#Eg:yak is encrypted as bdn

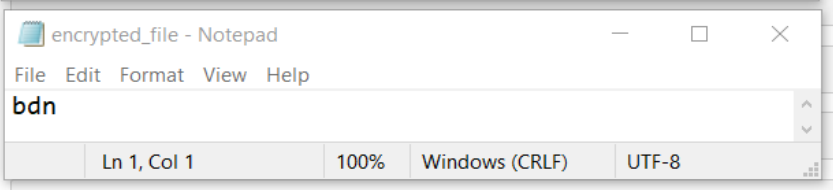
In [67]:
```
def encrypt(file_name):

    file = open(file_name, 'r')
    contents = file.read()

    encrypted_contents = ''

    for char in contents:
        if char.islower():
            encrypted_contents += chr((ord(char) - 97 + 3) % 26 + 97)

        elif char.isupper():
            encrypted_contents += chr((ord(char) - 65 + 3) % 26 + 65)

        else:
            encrypted_contents += char

    file = open('encrypted_file.txt', 'w')
    file.write(encrypted_contents)

encrypt('testfile.txt')
file.close()
```

In [ ]:



In [ ]:

In [ ]:

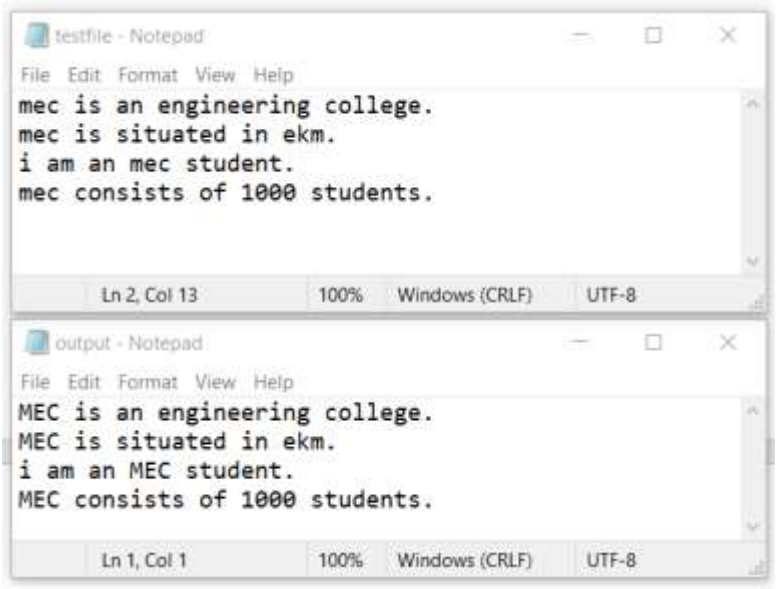


In [ ]:

In [ ]:

9.A text file contains the keyword mec replace all occurrence of the word mec with MEC

```
In [70]: file = open("testfile.txt", "r")
         data = file.read()

         data = data.replace("mec", "MEC")

         file1 = open("output.txt", "w")
         file1.write(data)

         file.close()
         file1.close()
```



10.Copy an image file (jpeg) into another file

```
In [72]: infile = open('infile.jpg', 'rb')
         outfile = open('copy.jpg', 'wb')
         outfile.write(infile.read())
         infile.close()
         outfile.close()
```