	Test	Input

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		`apple orange apple banana apple orange`

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		`Hello world! Hello everyone. Welcome to the world of prog`

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		`One fish two fish Red fish blue fish`

Create a Python program to delete a specific line from a text file based on a given line number.

Description:

1. Input:
    - A text file with multiple lines.
    - A line number to delete.
2. Output:
    - The updated file with the specified line removed in file "output.txt".

Example:

- Input File Content:
  "Line one.
  Line two.
  Line three.
  Line four."
  2

  Updated line two.

Output:

Line one.
Line three.
Line four.

For example:

Test	Input	Result

`with open('output.txt', 'r') as file:`
`  text = file.read()`
`  print(text)`
	input1.txt
	2
		Line one.
		Line three.
		Line four.

```
fi=input()
n=int(input())
o='output.txt'
with open(fi,'r') as f:
  l=f.readlines()
l.remove(l[n-1])
with open(o,'w')as f:
   f.writelines(l)
```

	Test	Input	Expected	Got	

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input1.txt
		2
			Line one.
			Line three.
			Line four.
				Line one.
				Line three.
				Line four.
					✔

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input2.txt
		3
			Line A.
			Line B.
				Line A.
				Line B.
					✔

Passed all tests! ✔

Correct

Develop a Python program to copy the contents of one file to another file.

Description:

1. Input:
   - Source file and destination file names.
2. Output:
   - The content of the source file copied to the destination file.

For example:

Test	Input	Result

`with open('output1.txt', 'r') as file:`
`  text = file.read()`
`  print(text)`
	`input1.txt`
	`output1.txt`
		`This is the source file.`
		`It contains multiple lines of text.`
		`Here is another line.`

```
i=input()
o=input()
with open(i,'r')as f:
   with open(o,'a')as f1:
      f1.write(f.read())
```

	Test	Input	Expected	Got

✔	`with open('output1.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		`input1.txt`
		`output1.txt`
			`This is the source file.`
			`It contains multiple lines of text.`
			`Here is another line.`
				`This`
				`It c`
				`Here`

✔	`with open('output2.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		`input2.txt`
		`output2.txt`
			`Hello, world!`
			`Python programming is amazing.`
			`Let's copy this text to another file.`
				`Hell`
				`Pyth`
				`Let'`

✔	`with open('output3.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		`input3.txt`
		`output3.txt`
			`Single line.`	`Sing`

Passed all tests! ✔

Write a Python program to reverse the contents of a specific line in a text file based on a given line number.

Description:

1. Input:
   - A text file with multiple lines.
   - A line number to reverse.
2. Output:
   - The updated file with the specified line's contents reversed in file "output.txt".

Example:

- Input File Content:
  "Line one.
  Line two.
  Line three.

Line four."
3

Output:

Line one.
Line two.
eerht eniL.
Line four.

For example:

Test	Input	Result

`with open('output.txt', 'r') as file:`
`  text = file.read()`
`  print(text)`
	input1.txt
	3
		Line one.
		Line two.
		eerht eniL.
		Line four.

```
f1=input()
n=int(input())
with open(f1,'r')as f:
    l=f.readlines()
l[n-1]=l[n-1][::-1]
l[n-1]=l[n-1][2:]+'.\n'
with open('output.txt','w') as f:
    f.writelines(l)
```

	Test	Input	Expected	Got	

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input1.txt
		3
			Line one.
			Line two.
			eerht eniL.
			Line four.
				Line one.
				Line two.
				eerht eniL.
				Line four.
					✔

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input2.txt
		2
			Line A.
			B eniL.
			Line C.
				Line A.
				B eniL.
				Line C.
					✔

Passed all tests! ✔

Create a Python program to write to a specific line in a text file, replacing the existing content of that line.

Description:

1. Input:
    - A text file with multiple lines.
    - A line number to write to.
    - New content for the specified line.
2. Output:
    - The updated file with the specified line replaced by the new content in file "output.txt".

Example:

- Input File Content:
  "Line one.
  Line two.
  Line three.
  Line four."
  2

  Updated line two.

Output:

Line one.
Updated line two.
Line three.
Line four.

For example:

Test	Input	Result

`with open('output.txt', 'r') as file:`
`    text = file.read()`
`    print(text)`
	`input1.txt`
	`2`
	`Updated line two.`
		`Line one.`
		`Updated line two.`
		`Line three.`
		`Line four.`

```
i=input()
n=int(input())
```

```
s=input()
with open(i,'r')as f:
    l=f.readlines()
l[n-1]=s+'\n'
with open('output.txt','w')as f:
 f.writelines(l)
```

	Test	Input	Expected	Got

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input1.txt
		2
		Updated line two.
			Line one.
			Updated line two.
			Line three.
			Line four.
				Line one.
				Updated line two.
				Line three.
				Line four.

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input2.txt
		2
		Line B Updated.
			Line A.
			Line B Updated.
			Line C.
				Line A.
				Line B Updated.
				Line C.

Passed all tests! ✔

Develop a Python program to identify and print all palindrome words from a given text file.

Description:

1. Input:
    - A text file containing multiple words.
2. Output:
    - A list of palindrome words found in the file name as 'output.txt'.

For example:

Test	Input	Result

`with open('output.txt', 'r') as file:`
`  text = file.read()`
`  print(text)`
	input1.txt	madam
		arora
		malayalam

```
i=input()
with open(i,'r') as f:
```

```
    l=f.read()
l=l.split()
s=''
for i in l:
    if i==i[::-1]:
        s+=i+'\n'
with open('output.txt','w')as f:
    f.write(s)
```

	Test	Input	Expected	Got	

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input1.txt	madam
			arora
			malayalam
				madam
				arora
				malayalam
					✔

Passed all tests! ✔

Develop a Python program to read a text file and count the total number of words in the file.

Description:

1. Input:
    - A text file containing several lines of text.
    - File name you should get as input.
2. Output:
    - The total number of words in the file.

For example:

Input	Result
input2.txt	Total words: 14
input3.txt	Total words: 15

```
import re
i=input()
```

```
with open(i,'r')as f:
    l=f.readlines()
l1=[]
for i in l:
    x=i.split()
    l1.extend(x)
print('Total words:',len(l1))
```

	Input	Expected	Got	
✔	input1.txt	Total words: 6	Total words: 6	✔
✔	input2.txt	Total words: 14	Total words: 14	✔
✔	input3.txt	Total words: 15	Total words: 15	✔

Passed all tests! ✔

Develop a Python program to read a specific line from a text file based on a given line number.

Description:

1. Input:
    - A text file with multiple lines.
    - A line number to read.
2. Output:
    - The content of the specified line.

input1.txt:

Line one.
Line two.
Line three.
Line four.

For example:

Input	Result

input1.txt 3	Line three.

```
i=input()
n=int(input())
with open(i,'r')as f:
    l=f.readlines()
print(l[n-1])
```

	Input	Expected	Got	
✔	input1.txt 3	Line three.	Line three.	✔
✔	input2.txt 3	Line C.	Line C.	✔

Passed all tests! ✔

Write a Python program to append a new line at a specific position in a text file, shifting existing lines down.

Description:

1. Input:
    - A text file with multiple lines.
    - A line number to insert the new line at.
    - New content for the new line.
2. Output:
    - The updated file with the new line inserted at the specified position, shifting the existing lines down in file "output.txt".

Example:

- Input File Content:
  "Line one.
  Line two.
  Line three.

Line four."
3

Inserted line..

Output:

Line one.
Line two.
Inserted line.
Line three.
Line four.

For example:

Test	Input	Result

`with open('output.txt', 'r') as file:`
`  text = file.read()`
`  print(text)`
	input1.txt
	3
	Inserted line.
		Line one.
		Line two.
		Inserted line.
		Line three.
		Line four.

```
i=input()
n=int(input())
s=input()
s+='\n'
with open(i,'r')as f:
    l=f.readlines()
if n-1==len(l):
    l[-1]+='\n'
l.insert(n-1,s)
with open('output.txt','w')as f:
    f.writelines(l)
```

	Test	Input	Expected	Got	

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input1.txt
		3
		Inserted line.
			Line one.
			Line two.
			Inserted line.
			Line three.
			Line four.
				Line one.
				Line two.
				Inserted line.
				Line three.
				Line four.
					✔

✔	`with open('output.txt', 'r') as file:`
	`  text = file.read()`
	`  print(text)`
		input2.txt
		4
		Inserted line D.
			Line A.
			Line B.
			Line C.
			Inserted line D.
				Line A.
				Line B.
				Line C.
				Inserted line D.
					✔

Passed all tests! ✔

Create a Python program to find the longest word in a text file.

- Input:
  - A text file containing multiple lines of text.
- Output:
  - The longest word in the file.

For example:

Input	Result
input1.txt	Longest word: containing

```
i=input()
with open(i,'r') as f:
    l=f.read()
l=l.split()
s=max(l,key=len)
print('Longest word:',s)
```

	Input	Expected	Got
✔	input1.txt	Longest word: containing	Longest word: containing
✔	input2.txt	Longest word: thousand	Longest word: thousand
✔	input3.txt	Longest word: supercalifragilisticexpialidocious	Longest word: supercalifragilistic

Passed all tests! ✔