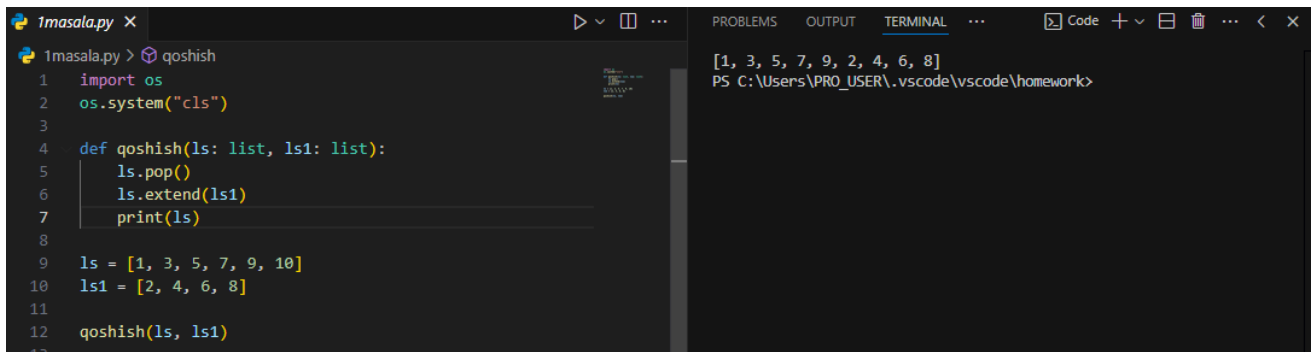


1 – masala



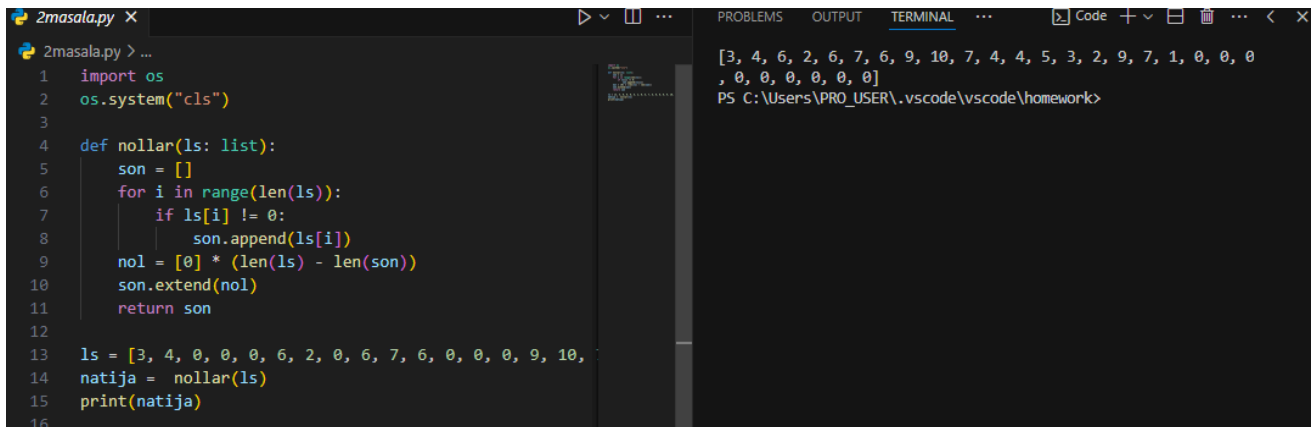
The screenshot shows a VS Code editor with a file named `1masala.py`. The code defines a function `qoshish` that takes two lists, `ls` and `ls1`, removes the last element of `ls`, and appends the elements of `ls1` to it. The function then prints the modified list. In the main code, `ls` is initialized as `[1, 3, 5, 7, 9, 10]` and `ls1` as `[2, 4, 6, 8]`. The function `qoshish` is called with these two lists. The terminal on the right shows the output: `[1, 3, 5, 7, 9, 2, 4, 6, 8]`.

```
1masala.py x
1masala.py > qoshish
1 import os
2 os.system("cls")
3
4 def qoshish(ls: list, ls1: list):
5     ls.pop()
6     ls.extend(ls1)
7     print(ls)
8
9 ls = [1, 3, 5, 7, 9, 10]
10 ls1 = [2, 4, 6, 8]
11
12 qoshish(ls, ls1)
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - - < x

[1, 3, 5, 7, 9, 2, 4, 6, 8]
PS C:\Users\PRO_USER\.vscode\vscode\homework>

2 – masala



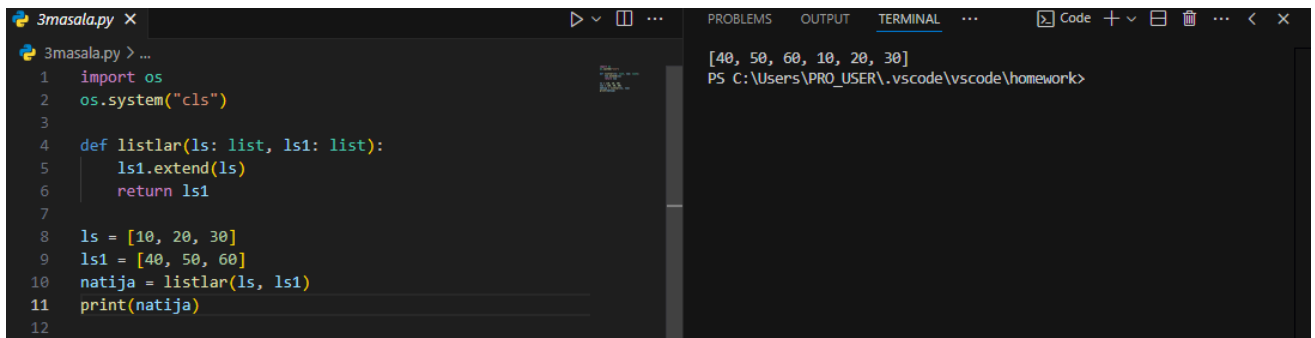
The screenshot shows a VS Code editor with a file named `2masala.py`. The code defines a function `nollar` that takes a list `ls` and returns a new list `son`. The function iterates through the elements of `ls`. If an element is not zero, it is appended to `son`. Additionally, a list `nol` of zeros is created, with a length equal to the difference between the original list's length and the new list's length. This list of zeros is then appended to `son`. In the main code, `ls` is initialized as `[3, 4, 0, 0, 0, 6, 2, 0, 6, 7, 6, 0, 0, 0, 9, 10]`, and the function `nollar` is called with `ls`. The result is stored in `natija` and printed. The terminal on the right shows the output: `[3, 4, 6, 2, 6, 7, 6, 9, 10, 7, 4, 4, 5, 3, 2, 9, 7, 1, 0, 0, 0, 0, 0, 0, 0]`.

```
2masala.py x
2masala.py > ...
1 import os
2 os.system("cls")
3
4 def nollar(ls: list):
5     son = []
6     for i in range(len(ls)):
7         if ls[i] != 0:
8             son.append(ls[i])
9     nol = [0] * (len(ls) - len(son))
10    son.extend(nol)
11    return son
12
13 ls = [3, 4, 0, 0, 0, 6, 2, 0, 6, 7, 6, 0, 0, 0, 9, 10]
14 natija = nollar(ls)
15 print(natija)
16
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - - < x

[3, 4, 6, 2, 6, 7, 6, 9, 10, 7, 4, 4, 5, 3, 2, 9, 7, 1, 0, 0, 0, 0, 0, 0]
PS C:\Users\PRO_USER\.vscode\vscode\homework>

3 – masala



The screenshot shows a VS Code editor with a file named `3masala.py`. The code defines a function `listlar` that takes two lists, `ls` and `ls1`, and extends `ls1` with the elements of `ls`. The function then returns `ls1`. In the main code, `ls` is initialized as `[10, 20, 30]` and `ls1` as `[40, 50, 60]`. The function `listlar` is called with these two lists, and the result is stored in `natija` and printed. The terminal on the right shows the output: `[40, 50, 60, 10, 20, 30]`.

```
3masala.py x
3masala.py > ...
1 import os
2 os.system("cls")
3
4 def listlar(ls: list, ls1: list):
5     ls1.extend(ls)
6     return ls1
7
8 ls = [10, 20, 30]
9 ls1 = [40, 50, 60]
10 natija = listlar(ls, ls1)
11 print(natija)
12
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - - < x

[40, 50, 60, 10, 20, 30]
PS C:\Users\PRO_USER\.vscode\vscode\homework>

4 – masala

```
4masala.py x
4masala.py > ...
1 import os
2 os.system("cls")
3
4 def dublikatlar(ls: list):
5     ls1 = []
6     [ls1.append(l) for l in ls if l not in ls1]
7     return ls1
8
9 ls = [[10, 20], [40], [30, 56, 25], [10, 20], [33], [40]]
10 print(ls, end="\n\n")
11 natija = dublikatlar(ls)
12 print(natija)
13
```

```
PROBLEMS OUTPUT TERMINAL ...
[[10, 20], [40], [30, 56, 25], [10, 20], [33], [40]]

[[10, 20], [40], [30, 56, 25], [33]]
PS C:\Users\PRO_USER\.vscode\vscode\homework>
```

5 – masala

```
5masala.py x
5masala.py > ...
1 import os
2 os.system("cls")
3 ls1 = []
4 ls = ['abcd', 'abc', 'bcd', 'bkie', 'cder', 'cdsw', 'sd']
5 for i in ls:
6     for j in range(len(i)):
7         if j == 0 and i[j] == 'a':
8             ls1.append(i)
9 print(ls1)
10 ls1.clear()
11 for i in ls:
12     for j in range(len(i)):
13         if j == 0 and i[j] == 'b':
14             ls1.append(i)
15 print(ls1)
16 ls1.clear()
17 for i in ls:
18     for j in range(len(i)):
19         if j == 0 and i[j] == 'c':
20             ls1.append(i)
21 print(ls1)
22 ls1.clear()
23 for i in ls:
24     for j in range(len(i)):
25         if j == 0 and i[j] == 'd':
26             ls1.append(i)
27 print(ls1)
```

```
PROBLEMS OUTPUT TERMINAL ...
['abcd', 'abc', 'acjd']
['bcd', 'bkie']
['cder', 'cdsw']
['dagfa']
PS C:\Users\PRO_USER\.vscode\vscode\homework>
```

6 – masala

```
6masala.py X
6masala.py > ...
1 import os
2 os.system("cls")
3
4 def sortlash(ls: list, ls1: list):
5     ls2 = []
6     kichik = min(len(ls), len(ls1))
7     for i in range(kichik):
8         ls2.append(ls[i])
9         ls2.append(ls1[i])
10    if len(ls) > kichik:
11        ls2.extend(ls[kichik:])
12    elif len(ls1) > kichik:
13        ls2.extend(ls1[kichik:])
14    return ls2
15
16 ls = [1, 2, 3, 4, 5]
17 ls1 = [11, 22, 33, 44]
18 natija = sortlash(ls, ls1)
19 print(natija)
20
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - - < X

```
[1, 11, 2, 22, 3, 33, 4, 44, 5]
PS C:\Users\PRO_USER\.vscode\vscode\homework>
```

7 – masala

```
7masala.py X
7masala.py > ...
1 import os
2 os.system("cls")
3
4 def oxshashlar(a: list, b: list):
5     a = set(a)
6     b = set(b)
7     new = set(a.intersection(b))
8     a = list(a)
9     b = list(b)
10    new = list(new)
11    return new
12
13 a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
14 b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
15 new = list(oxshashlar(a, b))
16 print(new)
17
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - - < X

```
[1, 2, 3, 5, 8, 13]
PS C:\Users\PRO_USER\.vscode\vscode\homework>
```

8 – masala

```
8masala.py X
8masala.py > ...
1 import os
2 os.system("cls")
3
4 def farqlilar(a: list, b: list):
5     a = set(a)
6     b = set(b)
7     new = set(a.difference(b))
8     b.difference_update(a)
9     b = list(b)
10    new = list(new)
11    new.extend(b)
12    new.sort()
13    return new
14
15 a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
16 b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]
17 new = list(farqlilar(a, b))
18 print(new)
19
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - - < X

```
[4, 6, 7, 9, 10, 11, 12, 21, 34, 55, 89]
PS C:\Users\PRO_USER\.vscode\vscode\homework>
```

9 – masala

```
9masala.py x
9masala.py > ...
1 import os
2 os.system("cls")
3
4 def sonlar(ls: list):
5     for j in ls:
6         if ls.count(j) == 1:
7             ls.remove(j)
8     print(ls)
9
10 ls = [int(input()) for i in range(int(input("N: ")))]
11 os.system("cls")
12 print(ls, end="\n\n")
13 sonlar(ls)
14
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - -

[1, 2, 3, 4, 5, 3, 2, 1, 6, 4]

[1, 2, 3, 4, 3, 2, 1, 4]

PS C:\Users\PRO_USER\.vscode\vscode\homework>

10 – masala

```
10masala.py x
10masala.py > ...
1 import os
2 os.system("cls")
3
4 def sonlar(ls: list):
5     ls = set(ls)
6     ls = list(ls)
7     ls.sort()
8     print(ls)
9
10 ls = [int(input()) for i in range(int(input("N: ")))]
11 os.system("cls")
12 print(ls, end="\n\n")
13 sonlar(ls)
14
```

PROBLEMS OUTPUT TERMINAL ... Code + - - - - -

[1, 2, 3, 6, 43, 1, 2, 21, 5, 3]

[1, 2, 3, 5, 6, 21, 43]

PS C:\Users\PRO_USER\.vscode\vscode\homework>