

Задание 1

Пользователь вводит с клавиатуры набор чисел. Полученные числа необходимо сохранить в список (тип списка нужно выбрать в зависимости от поставленной ниже задачи). После чего нужно показать меню, в котором предложить пользователю набор пунктов: 1. Добавить новое число в список (если такое число существует в списке, нужно вывести сообщение пользователю об этом, без добавления числа). 2. Удалить все вхождения числа из списка (пользователь вводит с клавиатуры число для удаления) 3. Показать содержимое списка (в зависимости от выбора пользователя список нужно показать с начала или с конца) 4. Проверить есть ли значение в списке 5. Заменить значение в списке (пользователь определяет заменить ли только первое вхождение или все вхождения) В зависимости от выбора пользователя выполняется действие, после чего меню отображается снова.

```
class stack:
    def __init__(self,numb):
        self.numb=numb
    def add_numb(self):
        a=int(input("Enter new number: "))
        for i in self.numb:
            if i != a:
                self.numb.append(a)
                return self.numb
            else:
                return "Эта цифра уже есть"
    def dell_numb(self):
        a = int(input("Enter number to delete: "))
        self.numb.remove(a)
        return self.numb
    def replace(self):
        print(self.numb)
        re = int(input("Enter place number to replace: "))
        ne = int(input("Enter new wrod: "))
        self.numb[re - 1] = ne
        return self.numb
    def find(self):
        a = int(input("Enter number: "))
        for i in self.numb:
            if i == a:
                return "Эта цифра есть"
    def show(self):
        v=input("1.(show from start)\n2.(show from end)\n")
        if v=="1":
            return self.numb
        if v=="2":
            return self.numb[::-1]
n=[int(input("Enter number: "))for i in range(3)]
Andereystack=stack(numb=n)
while True:
    v=input("1.(add number)\n2.(delete number)\n3.(replace numbers)\n4.(find)\n5.(show all)\n")
```

```

if v=="1":
    print(Andereystack.add_numb())
if v=="2":
    print(Andereystack.dell_numb())
if v=="3":
    print(Andereystack.replace())
if v=="4":
    print(Andereystack.find())
if v=="5":
    print(Andereystack.show())

```

```

Enter new number: 10
[3, 4, 5, 10]
1.(add number)
2.(delete number)
3.(replace numbers)
4.(find)
5.(show all)
3
[3, 4, 5, 10]
Enter place number to replace: 4
Enter new wrod: 44
[3, 4, 5, 44]
1.(add number)
2.(delete number)
3.(replace numbers)
4.(find)
5.(show all)
5
1.(show from start)
2.(show from end)
2
[44, 5, 4, 3]
1.(add number)
2.(delete number)
3.(replace numbers)
4.(find)
5.(show all)

```

Задание 2

Реализуйте класс стека для работы со строками (стек строк). Стек должен иметь фиксированный размер. Реализуйте набор операций для работы со стеком: ■ помещение строки в стек; ■ выталкивание строки из стека; ■ подсчет количества строк в стеке; ■ проверку пустой ли стек; ■ проверку полный ли стек; ■ очистку стека; ■ получение значения без выталкивания верхней строки из стека. При старте приложения нужно отобразить меню с помощью, которого пользователь может выбрать необходимую операцию.

```

class steck:
    def __init__(self,numb):
        self.numb=numb
    def add_numb(self):

```

```

        a=input("Enter new number: ")
        self.numb.append(a)
        return self.numb
def dell_num(self):
    a = input("Enter word to delete: ")
    self.numb.remove(a)
    return self.numb
def replace(self):
    return len(self.numb)
def empty(self):
    if len(self.numb) ==0:
        return "Steck is empty"
    else:
        return "Steck is not empty"
def full(self):
    if len(self.numb) >0:
        return "Steck is full"
    else:
        return "Steck is not full"
def clear(self):
    return self.numb.clear()
def show(self):
    return self.numb
n=[(input("Enter words: "))for i in range(3)]
Andereystack=steck(numb=n)
while True:
    v=input("1.(add word)\n2.(delete word)\n3.(count word)\n4.(empty)\n5.(full)\n6.(clear)\n7.(show)\n")
    if v=="1":
        print(Andereystack.add_num())
    if v=="2":
        print(Andereystack.dell_num())
    if v=="3":
        print(Andereystack.replace())
    if v=="4":
        print(Andereystack.empty())
    if v=="5":
        print(Andereystack.full())
    if v=="6":
        print(Andereystack.clear())
    if v=="7":
        print(Andereystack.show())

```

```
1
Enter new number: robert
['hello', 'world', 'bober', 'robert']
1.(add word)
2.(delete word)
3.(count word)
4.(empty)
5.(full)
6.(clear)
7.(show)
```

```
2
Enter word to delete: robert
['hello', 'world', 'bober']
1.(add word)
2.(delete word)|
3.(count word)
4.(empty)
5.(full)
6.(clear)
7.(show)
```

```
4
Steck is not empty
```

```
6.(clear)
7.(show)
```

```
6
None
1.(add word)
2.(delete word)
3.(count word)
4.(empty)
5.(full)
6.(clear)
7.(show)
```

```
4
Steck is empty
1.(add word)
2.(delete word)
3.(count word)
4.(empty)
5.(full)
6.(clear)
7.(show)
```

