Design min stack with constnt time but O(N) space

```
import random


class MinStack:
    def __init__(self):
        self.stack = []
        self.minStack = []

    def size(self):
        return len(self.stack)

    def top(self):
        if self.size() != 0:
            return self.stack[-1]
        else:
            return 'Underflow'

    def pop(self):
        if self.size() == 0:
            return 'UnderFlow'
        else:
            temp = self.stack.pop()
            if temp == self.minStack[-1]:
                self.minStack.pop()
            return temp

    def push(self, val):
        self.stack.append(val)
        if len(self.minStack) == 0:
            self.minStack.append(val)
        if val < self.minStack[-1]:
            self.minStack.append(val)

    def getMin(self):
        return self.minStack[-1] if len(self.minStack) else 'Empty Min
stack'
```

```
stack = MinStack()

for i in range(5):
    temp = random.randint(-100, 100)
    print(temp,end = ' ')
    stack.push(temp)

print()

for i in range(5):
    print(stack.pop(), end=',')
    # print()
    print(stack.getMin(), end='| ')
```