

1. Napisz funkcję *rever(l)*, która zwraca listę z elementami w odwrotnej kolejności.
2. Napisz funkcję *is_pali(l)*, która sprawdza czy lista jest palindromem.
3. Napisz funkcję *all_true(l)*, która sprawdza czy wszystkie elementy listy są równe True.
4. Napisz funkcję *hm_true(l)*, która zliczy liczbę elementów listy równych True.
5. Napisz funkcję *cumsum(l)*, która policzy sumy częściowe.
6. Napisz funkcję *is_sorted(l)*, która sprawdzi czy lista jest posortowana rosnąco.
7. Napisz funkcję *min_max(l)*, która zwróci krotkę (mi, ma), odpowiednio największy i najmniejszy element listy l.
8. Napisz funkcję *min_max_arg(l)*, która zwróci krotkę (imi, ima), zawierającą indeks największego i najmniejszego elementu. Jedna pętla for dozwolona.
9. Niech l będzie listą elementów całkowitych o wartościach od 0 do k - 1 dla pewnego k. Napisz funkcję *bs_mode(l, k)*, która znajdzie modę w tej liście. Nie sortuj listy.
Podpowiedź: Wykorzystaj listę o długości k do zliczania wystąpień elementów.
10. Niech l i k będą jak wyżej, zaimplementuj algorytm sortowania kubełkowego.
11. Napisz funkcję *dir_mode(l)*, która wyznaczy modę z elementów listy l, wykorzystaj zbiór.
12. Napisz funkcję *are_unique(l)*, która sprawdzi czy wszystkie elementy w l są unikalne, wykorzystaj zbiór (set).
13. Niech l będzie posortowaną listą liczb całkowitych. Napisz funkcję *bin_search(l, x)*, która sprawdzi czy liczba x występuje w liście.