

Funkcja `shell_sort(l)` ma na celu posortowanie listy liczb przy użyciu algorytmu Shella. Na początku przypisujemy długość listy do zmiennej `n` i inicjujemy flagę `zamiana` jako `False` oraz zmienną `delta` równą `n`. Funkcja sprawdza, czy lista nie jest posortowana, wykorzystując pętlę `while`, która trwa dopóki nie przestaniemy dokonywać zamian elementów.

Jeżeli `delta` jest większe lub równe 1, dzielimy ją przez 2, co zmniejsza wartość kroku, z jakim porównujemy elementy w liście. Następnie wchodzimy w kolejną pętlę, gdzie ustawiamy `zamiana` na `False` i zaczynamy iterować po liście od początku. Dla każdej pary elementów oddzielonych o aktualną wartość `delta`, sprawdzamy, czy element po lewej stronie jest większy od prawego. Jeśli tak, następuje zamiana tych dwóch elementów i flaga `zamiana` zmienia się na `True`, co świadczy o dokonanej zamianie.

Po przejściu całej listy dla danego kroku, jeżeli nie dokonano żadnej zamiany, wychodzimy z wewnętrznej pętli `while`. Proces ten powtarza się, zmniejszając `delta` do momentu, gdy osiągnie wartość mniejszą niż 1. Gdy `delta` jest mniejsze od 1, ustawiamy `zamiana` na `True`, co kończy główną pętlę i pozwala na zwrócenie posortowanej listy.

Na końcu definiujemy listę `l` z elementami, które chcemy posortować, wywołujemy funkcję `shell_sort(l)` i przypisujemy wynik do zmiennej `sorted_list`. Na koniec wypisujemy posortowaną listę, otrzymując w ten sposób uporządkowaną sekwencję liczb.

Schemat blokowy:

