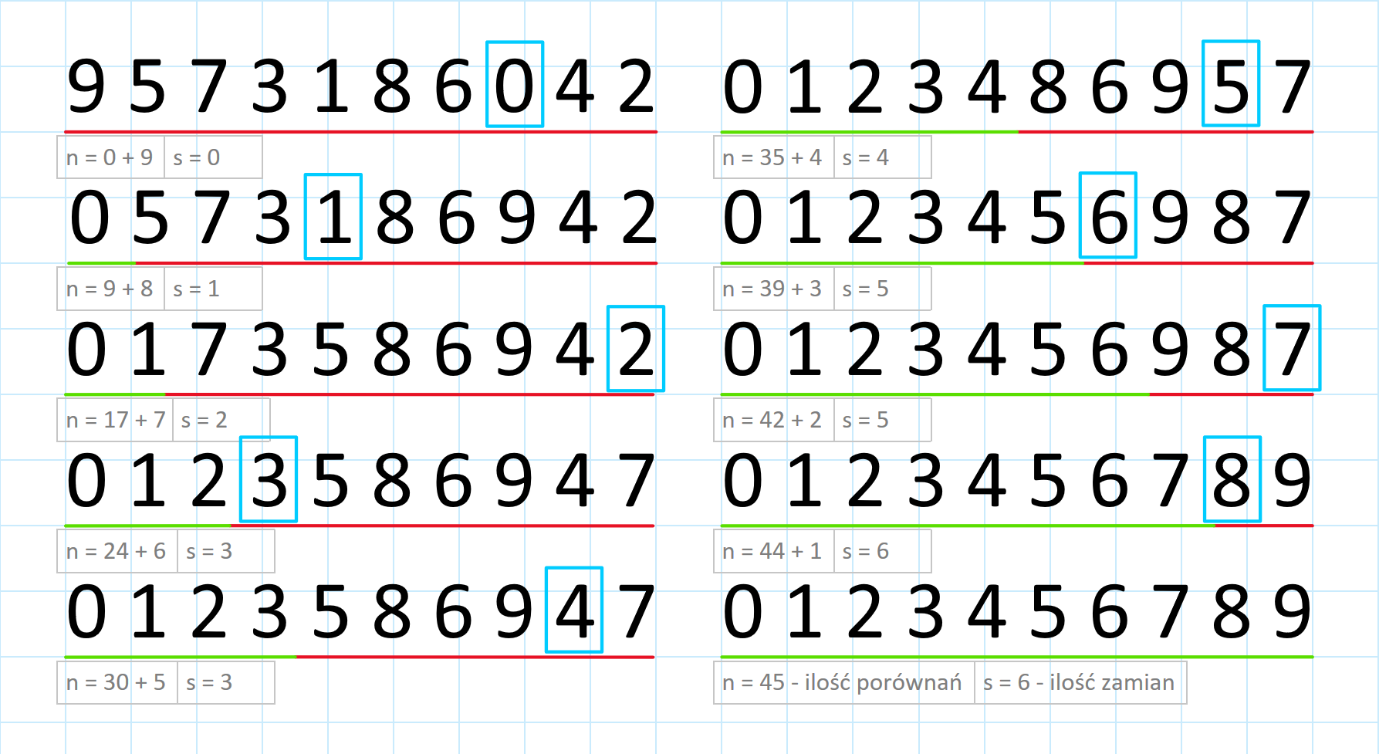
# Wstęp i opis algorytmów

Głównym zadaniem projektu było napisanie dwóch algorytmów sortowania, a mianowicie: sortowanie przez wybieranie oraz sortowanie kopcowe. Mniejszymi zadaniami były algorytmy odczytu danych z plików i generowania danych testowych o różnej złożoności sortowania dla każdego algorytmu.

## Sortowanie przez wybieranie

Sortowanie przez wybieranie to niestabilny porównujący algorytm sortowania na miejscu. Ma złożoność czasową , co oznacza, że jest nieefektywnym w przypadku dużych list. Taki algorytm charakteryzuje się prostotą i ma przewagę wydajnościową nad bardziej złożonymi algorytmami w pewnych sytuacjach, zwłaszcza gdy pamięć pomocnicza jest ograniczona.

Algorytm dzieli listę wejściową na dwie partycji: posortowaną i nieposortowaną. Początkowo posortowana część jest pusta, a nieposortowana partycja zajmuje całą listę wejściową. Algorytm jest wykonywany poprzez znalezienie najmniejszego elementu w niesortowanej części, zamianę go na pierwszy nieposortowany element i przesunięcie podziału listy.

Największą zaletą takiego algorytmu jest minimalna możliwa liczba permutacji elementów (w najgorszym przypadku ). Mimo to efektywność czasowa sortowania według wyboru jest kwadratowa, więc istnieje wiele algorytmów, które mają mniejszą złożoność czasową.

## Sortowanie przez kopcowanie

Jednym z takich algorytmów jest sortowanie przez kopcowanie. Podobnie do algorytmu sortowania przez wybieranie, kopcowy algorytm sortuje na miejscu metodą porównań, nie jest stabilnym i analogicznie dzieli dane wejściowe na posortowaną i nieposortowaną partycji. Sortowanie kopcowe wybiera największy element z części niesortowanej i wstawiając go do posortowanej części, iteracyjnie zmniejsza nieposortowaną partycję. W przeciwieństwie do sortowania przez wybieranie, sortowanie kopcowe nie skanuje liniowo cały nieposortowany obszar. Ten algorytm obsługuje nieposortowaną część w postaci kopca, aby szybko znaleźć największy element, który zawsze będzie znajdować się na górze. Po każdym usunięciu największego elementu w jego miejsce dodawany jest element z podstawy kopca, który następnie jest zastępowany przez jego potomków i przenoszony w nowe miejsce.

Sortowanie kopcowe ma pesymistyczną złożoność czasową , co jest nawet lepsze niż szybkie sortowanie (), chociaż w praktyce działa wolniej (oczekiwana złożoność obu algorytmów jest taka sama i wynosi ).

Sortowanie kopcowe zostało wynalezione przez J. Williamsa w 1964 roku.