

Algorytmy i struktury danych I – laboratorium, zajęcia 1

- 1. Napisać funkcję **RandomTable** według schematu blokowego z prezentacji. Rozmiar tablicy powinien być podawany jako argument funkcji. Funkcja ma tworzyć tablicę dynamiczną liczb typu int żądanego rozmiaru. Tablica po utworzeniu powinna być wypełniana liczbami pseudolosowymi typu int. Funkcja ma zwracać przez wartość wskaźnik do utworzonej tablicy, tak aby można tę tablicę wykorzystać.
- 2. Napisać funkcję typu void do wypisywania na ekran zawartości tablicy. Nazwa tablicy i jej rozmiar powinny być podawane jako argumenty funkcji.
- 3. Napisać funkcję **SequentialSearch** realizującą algorytm wyszukiwania sekwencyjnego według schematu blokowego z prezentacji. Wartość elementu do wyszukiwania, nazwa tablicy oraz rozmiar tablicy powinny być podawane jako argumenty funkcji.

Dodatek. Komentarz do prezentacji.

Będziemy rozważać problem **wyszukiwania elementu w tablicy** tab o długości length, zawierającej liczby (w dowolnej kolejności).

Najprostszy algorytm wyszukiwania to algorytm **wyszukiwania sekwencyjnego**.

Slajd 3:

Algorytm ten działa następująco: dokonujemy przebiegu tablicy od początku do końca (zmienna i na schemacie blokowym zmienia się od $i \leftarrow 0$ do $i \leftarrow \text{length}-1$ – zmienna ta jest indeksem tablicy). Dla danego indeksu i porównujemy szukany element (o wartości s) z elementem tablicy $\text{tab}[i]$. Jeżeli element tablicy o indeksie i jest równy szukanemu elementowi, to kończymy wyszukiwanie zwracając indeks i pod którym znaleźliśmy szukany element. Jeśli szukany element s nie jest równy elementowi $\text{tab}[i]$, przechodzimy dalej. Jeśli w ten sposób sprawdziliśmy wszystkie elementy tablicy i żaden z nich nie jest równy szukanemu elementowi, to szukanego elementu nie ma w tablicy (zwracamy wtedy -1).