

The background of the slide is white and decorated with numerous realistic water droplets of various sizes. Some droplets are large and prominent, while others are small and subtle. They are scattered across the slide, with a higher concentration in the top-left and bottom-right corners, creating a clean, fresh, and modern aesthetic.

# EFEKTYVESNI RŪŠIAVIMO ALGORITMAI – MERGE SORT, QUICK SORT

## (PRAKTIKA)

# PAMOKOS TIKSLAI

- SUPAŽINTI SU MERGE SORT IR QUICK SORT ALGORITMŲ VEIKIMO PRINCIP AIS.
- IŠNAGRINĖTI ALGORITMŲ PSEUDO KODUS
- PRAKTIŠKAI ĮGYVENDINTI ALGORITMUS PROGRAMUOJANT

# PRAĖJUSI PAMOKA

- KOKIUS RŪŠIAVIMO ALGORITMUS JAU MOKĖMĖS?
- KUO JIE SKYRĖSI VIENAS NUO KITO?
- KOKIA BUVO JŲ VYKDYMO LAIKO ANALIZĖ?

# MERGE SORT ALGORITMAS

TAI REKURSINIS ALGORITMAS, KURIS NAUDOJA „DIVIDE AND CONQUER“ STRATEGIJĄ.

VEIKIMO PRINCIPAS:

1. PADALINAME MASYVĄ Į DVI LYGIAS DALIS.
2. REKURSIŠKAI RŪŠIUOJAME KIEKVIENĄ DALĮ.
3. SULIEJAME DVI IŠRIKIUOTAS DALIS Į VIENĄ.

# MERGE SORT (PSEUDO KODAS)

```
1 MergeSort(arr)
2   if length of arr <= 1
3     return arr
4   mid = length of arr / 2
5   left = MergeSort(left half of arr)
6   right = MergeSort(right half of arr)
7   return Merge(left, right)
8
9 Merge(left, right)
10  result = []
11  while left is not empty and right is not empty
12    if left[0] <= right[0]
13      append left[0] to result
14      remove left[0] from left
15    else
16      append right[0] to result
17      remove right[0] from right
18  append remaining elements from left (if any) to result
19  append remaining elements from right (if any) to result
20  return result
```

# QUICK SORT ALGORITMAS

TAI REKURSINIS ALGORITMAS, KURIS TAIP PAT NAUDOJA „DIVIDE AND CONQUER“ STRATEGIJĄ.

VEIKIMO PRINCIPAS:

1. PASIRENKAMAS ATSITIKTINIS ELEMENTAS (PIVOT).
2. MASYVAS PADALIJAMAS Į DVI DALIS: MAŽESNI UŽ PIVOT IR DIDESNI UŽ PIVOT.
3. REKURSIŠKAI RŪŠIUOJAMOS ABI DALYS.

# QUICK SORT (PSEUDO KODAS)

```
1 QuickSort(arr)
2   if length of arr <= 1
3       return arr
4   pivot = choose a pivot (e.g., arr[0])
5   left = all elements in arr < pivot
6   right = all elements in arr > pivot
7   middle = all elements in arr == pivot
8   return QuickSort(left) + middle + QuickSort(right)
```

# PRAKTINĖ UŽDUOTIS

PARAŠYTI PASIRINKTA PROGRAMAVIMO KALBA(PYTHON, C++)

- MERGE SORT ALGORITMĄ
- QUICK SORT ALGORITMĄ



# APIBENDRINIMAS

KURIUO ATVEJU MERGE SORT AR QUICK SORT YRA EFEKTYVESNIS?

KURIE ALGORITMAI TINKA KOKIOMS SITUACIJOMS?

KOKIUS RŪŠIAVIMO ALGORITMUS PASIRINKSITE ATEITIES PROJEKTUOSE?