

# Mappeinnlevering

## Emne 5 Algoritmiske Metoder



TID: 2025-08-23 til 2025-09-05 12:00

Hjelpemidler: Alle

Vedlegg: «Emne 5 Mappe 1 2024.zip»

Innlevering: Zip fil av hele konsollapplikasjon prosjektet.

## Oppgave 1 Lineært Søk i C# (12.5 %)

**Oppgavebeskrivelse:** Du skal implementere en metode som utfører lineært søk i et array av heltall. Metoden skal finne og returnere indeksen til det første forekomsten av et gitt målelement. Hvis målelementet ikke finnes i arrayet, skal metoden returnere -1.

### Kravspesifikasjon:

1. Implementer en metode `int LinearSearch(int[] array, int target)` i C#.
2. Metoden skal returnere indeksen til det første elementet som matcher target.
3. Hvis target ikke finnes i array, skal metoden returnere -1.
4. Test metoden din med følgende data:
  - Array: [56, 34, 78, 12, 90, 34, 12]
  - target: 34 (skal returnere 1)
  - target: 90 (skal returnere 4)
  - target: 100 (skal returnere -1)

## Oppgave 2 Binært Søk i C# (12.5%)

**Oppgavebeskrivelse:** Implementer en metode som utfører binært søk på et array av heltall. Merk at arrayet kan være usortert, så du må først sortere det før du utfører binært søk. Metoden skal returnere indeksen til målelementet hvis det finnes, eller -1 hvis elementet ikke finnes.

### Kravspesifikasjon:

1. Implementer en metode `void SortArray(int[] array)` som sorterer arrayet i stigende rekkefølge.
2. Implementer en iterativ metode `int BinarySearch(int[] array, int target)` i C# som utfører binært søk.
3. Før du utfører binært søk, må arrayet sorteres ved hjelp av `SortArray`-metoden.
4. Test metodene dine med følgende data:
  - Usortert Array: [72, 35, 22, 89, 59, 10, 68, 47]
  - target: 47 (skal returnere 3 etter sortering)
  - target: 89 (skal returnere 7 etter sortering)
  - target: 100 (skal returnere -1 etter sortering)

## Oppgave 3 i C# (25%)

### - Oppgavebeskrivelse:

Du skal undersøke og sammenligne ytelsen til lineært søk og binært søk på store datasett. Oppgaven krever både implementasjon, måling, og refleksjon.

### - Kravspesifikasjon:

- Implementer både LinearSearch og BinarySearch i C# (kan gjenbruke fra Oppgave 1 og 2).
- Lag et program som:
  - Genererer lister med tilfeldige tall : 1 000 elementer.
  - Søker etter både eksisterende og ikke-eksisterende elementer.
  - Måler kjøretid (bruk Stopwatch i C#).

### - Sammenlign resultatene:

- Hvor raskt er lineært søk vs. binært søk ved ulike størrelser?
- Hvordan endrer forskjellen seg når data ikke er sortert (må sorteres først for binært søk)?
- Hva skjer når elementet ikke finnes i arrayet?

### - Leveranse

- Når er lineært søk bedre enn binært søk?
- Hvordan påvirker sorteringstiden den totale ytelsen?
- Hvorfor er binært søk mer brukt i praksis, selv om det har en «oppstarts-kostnad»?
- Finn minst ett reelt eksempel fra industri eller teknologi (databaser, søkemotorer, operativsystemer, etc.) hvor en av metodene brukes. Kildehenvisning kreves.

### Spørsmål til refleksjon

- Hva er forskjellen på teoretisk kompleksitet ( $O(n)$ ,  $O(\log n)$ ) og faktisk ytelse på maskinen din?
- Hvorfor kan det være avvik mellom teori og praksis?
- Hvordan kan datastrukturvalg (Array vs. List vs. Linked List) endre effektiviteten?