

Data-analytiikka – Tietorakenteet ja tietojen tallennus

Arvioitu oppijan taitotaso: aloittelija tai keskitaso

Tietorakenteet

Määritelmä

Tietorakenne määrittää, miten tietoja tallennetaan ja järjestetään.

Miksi?

Tehokkuus: Hyvin suunnitellut tietorakenteet mahdollistavat big datan tehokkaan hallinnan ja tehokkaiden algoritmien toteuttamisen. (Amplitude 2024; GeeksForGeeks 2024a.)

Organisaatio: Data on järjestetty helposti hallittavalla ja helppokäyttöisellä tavalla. (Amplitude 2024; GeeksForGeeks 2024a.)

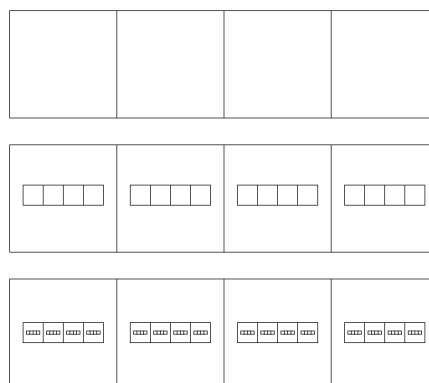
Abstraktio: Abstraktio mahdollistaa tietojen helpon lisäämisen, poistamisen ja päivittämisen ilman, että käyttäjä keskittyy yksityiskohtiin. (Amplitude 2024.)

Uudelleenkäytettävyys: Samaa tietorakennetta voidaan käyttää uudelleen toisessa sovelluksessa tai projektissa, mikä vähentää tuotantoon tarvittavaa aikaa. (Amplitude 2024.)

Optimointi: Optimaalisen tietorakenteen valitseminen sovellukselle vähentää monimutkaisuutta ja parantaa ohjelmiston tai palvelun nopeutta. (Amplitude 2024.)

Tyypit

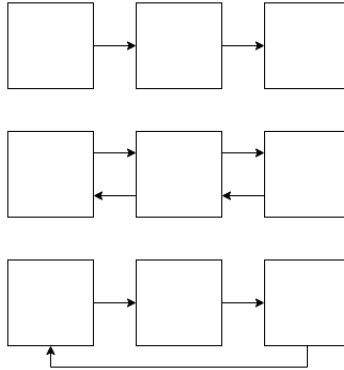
Valikoima (Array) = "taulukko" tai luettelo datakohteista (Amplitude 2024; GeeksForGeeks 2024a). Alla olevassa kuvassa on kolme esimerkkitaulukkoa. Ylhäältä alas: 1D-taulukko (yksinkertainen luettelo), 2D-taulukko (luetteloluettelo), 3D-taulukko (luetteloluetteloitten luettelo).



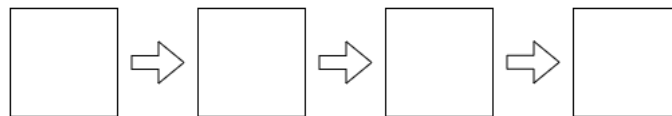
Linkitetty luettelo (Linked list) = Luettelo, jossa kukin kohde on yhdistetty joko edelliseen tai seuraavaan kohteeseen tai molempiin (Amplitude 2024; Bible & Moser 2023;



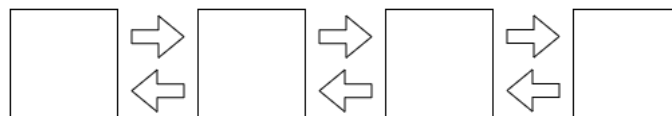
GeeksForGeeks 2024a). Alla olevassa kuvassa on kolme esimerkkiä linkitetystä luettelosta. Ylhäältä alas: yksittäin linkitetty luettelo (jokainen kohde on yhdistetty seuraavaan), kaksinkertaisesti linkitetty luettelo (jokainen kohde on yhdistetty edelliseen ja seuraavaan), pyöreä (yksittäin) linkitetty luettelo (jokainen kohde on yhdistetty seuraavaan ja viimeinen kohde on linkitetty ensimmäiseen).



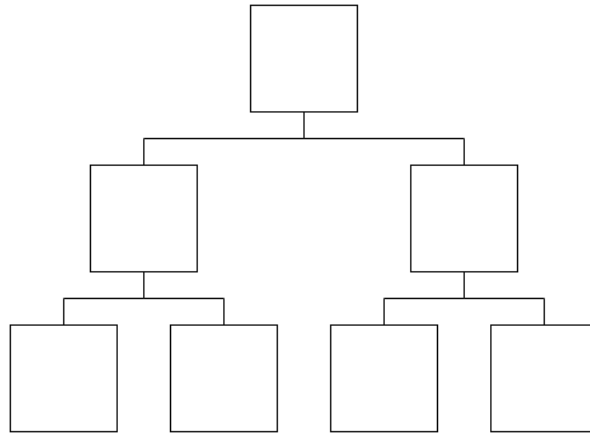
Jono (Queue) = Yksiuotteinen tietorakenne, jossa kohteita voidaan lisätä toiseen päähän ja poistaa toisesta FIFO- tai LIFO-periaatteen mukaisesti (Amplitude 2024; Bible & Moser 2023; GeeksForGeeks 2024a; GeeksForGeeks 2024b). Alla olevassa kuvassa on esimerkki jonosta, jossa nuolet osoittavat suunnan, johon nimikkeet lisätään ja poistetaan rakenteesta.



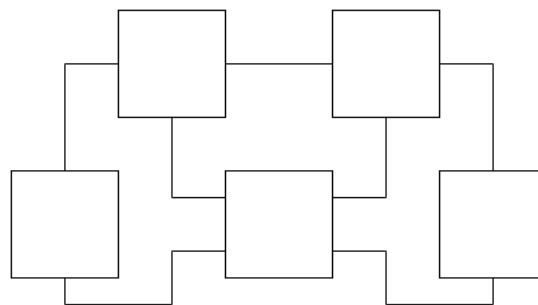
Pino (Stack) = Yksiuotteinen tietorakenne, johon kohteita voidaan lisätä ja poistaa FILO tai LIFO (first-in/last-out tai last-in/first-out) -järjestyksessä (Amplitude 2024; Bible & Moser 2023; GeeksForGeeks 2024a; GeeksForGeeks 2024b). Alla olevassa kuvassa on esimerkki pinosta, jossa nuolet osoittavat suunnan, johon kohteita lisätään ja poistetaan rakenteesta.



Puu (Tree) = Hierarkkinen rakenne, jossa kukin kohde voidaan yhdistää useisiin kohteisiin; esimerkiksi binääripuussa kukin kohde on yhdistetty (enintään) kahteen muuhun kohteeseen (Amplitude 2024; Bible & Moser 2023; GeeksForGeeks 2024a). Alla oleva kuva on esimerkki puusta.



Kaavio (Graph) = toisiinsa liittyvien kohteiden rakenne (Amplitude 2024; Bible & Moser 2023; GeeksForGeeks 2024a). Seuraava kuva on esimerkki kaaviosta.



Sovelluksia

Harjoitus: Yhdistä seuraavat tietorakenteet ja niiden ihanteelliset käyttötapaukset.

Array	Sosiaalisen verkoston edustaminen
Linkitetty luettelo	Tietokantojen indeksointi
Jono	Toimintojen kumoaminen
Pino	Asiakaspalvelulinjamalli
Puu	Soittolista, jota voidaan muuttaa dynaamisesti
Kuvaaja	Opiskelijoiden arvosanat
	6,5,4,3,2,1



Tietojen tallennus

Yleistä tietoa

Tietoja voidaan tallentaa eri sovelluskohtaisilla tavoilla. Suurin osa esimerkkitapauksista on tosielämän projekteista, joihin Centria on osallistunut.

Muut kuin tietokantaratkaisut

Ne eivät perustu perinteisiin tietokantoihin.

Esimerkkejä tyypeistä:

- **Laskentataulukot (Spreadsheets)** (esim. Excel) – Tiedot esitetään taulukkomuodossa.
 - o Esimerkkejä tapauksista:
 1. Työntekijöiden päivittäiset sairauspoissaolot → hyvinvoinnin parantaminen
 2. Rakennuksen lämmitys-, ilmanvaihto-, ilmastointi- → sähkö tiedot optimoitu energiankulutus
- **Merkintäkieleen perustuva (Markup-language-based)** (esim. XML) – Tiedot ovat joustavia ja voivat olla hierarkkisia.
 - o Esimerkki tapauksesta:
Säähavaintohistoria → säästä riippuvat toiminnot, sääennusteet

Relaatiotietokannat

Nämä järjestävät tiedot taulukoihin, joissa on rivejä ja sarakkeita niiden suhteiden perusteella.

Esimerkkejä tyypeistä:

- **SQL** (esim. MySQL) – Tiedot tallennetaan perinteisesti SQL:llä, jota käytetään kyselyihin.
 - o Esimerkki tapauksesta:
Kotihoidon työntekijöiden päivittäinen hyvinvointi → hyvinvoinnin parantaminen
- **Pylväsmäinen (Columnar)** (esim. Amazon Redshift) – Tämä tietomuoto on tarkoitettu analyyttisille kuormituksille.
 - o Esimerkki tapauksesta:
varastohallinta → varastoinnin, logistiikan ja myynnin optimointi
- **Kaavio (Graph)** (esim. Neo4j) – Graafitietokannat kaappaavat monimutkaisia suhteita.
 - o Esimerkki tapauksesta:
Kotiautomaatiojärjestelmän anturisolmut → elinolojen parantamiseen/personointiin, energiansäästöön.
- **Aikasarjat (timeseries)** (esim. InfluxDB) – ajallisille (aikasarja)tiedoille
 - o Esimerkkejä tapauksista:
 1. Verkkoliikenne kyberhyökkäysten havaitsemisen → aikana
 2. Ajoneuvojen läsnäolo satamassa → liikenteen/logistiikan optimointi



- **Spatiaalinen (Spatial)** (esim. PostGIS) – paikkatiedot
 - o Esimerkkejä tapauksista:
 1. Nurmikon laatu paikoissa → optimoidaan maataloustyötä
 2. Drooneilla mitatut radiosignaalit paikoissa → joissa tietoliikenneverkkoja parannetaan
- **Monimalli (Multimodel)** (esim. ArangoDB) – useita tietomalleja
 - o Esimerkki tapauksesta:
useista lähteistä peräisin olevien tietojen hallinta → esimerkiksi monimutkaisen IT-infrastruktuurin analytiikan parantaminen/optimointi

Ei-relaatiotietokannat

Ne tarjoavat joustavuutta perinteisiä malleja enemmän.

Esimerkkejä tyypeistä:

- **JSON** (esim. MongoDB) – asiakirjapohjainen
 - o Esimerkki tapauksesta:
Ilmanlaatuanturien lukemat → parantavat sisäilman laatua.
- **Redis** – muistissa oleva rakennevarasto
 - o Esimerkki tapauksesta:
rasvan ja veden määrä rasvakaivossa → jätehuollon optimointi
- **RESTful-API-pohjainen** (esim. Couchbase) – Tietoja käytetään API:en kautta.
 - o Esimerkki tapauksesta:
Blogin/sosiaalisen median sisällönhallinta → trendianalyysi, turvallisuus, käyttäjien käyttäytymisen analysointi

Seuraava vaihe

[Siirry tietojen analysointimenetelmiin](#)

Viittaukset

Amplitude. 2024. *Advanced Data Structures Explained*. Available at:

<https://amplitude.com/explore/data/what-how-data-structure>. Accessed 10 June 2024.

Bible, P. W. & Moser, L. 2023. *An Open Guide to Data Structures and Algorithms*. USA: PALNI Press. Available at:

<https://pressbooks.palni.org/anopenguidetodatastructuresandalgorithms/>. Accessed 10 June 2024. Licence: CC 4.0 BY.

GeeksForGeeks. 2024a. *Data Structures Tutorial*. Available at:

<https://www.geeksforgeeks.org/data-structures/>. Accessed 10 June 2024.



**Euroopan unionin
osarahjoittama**



**Kokkola
Karleby**

CENTRIA
University of Applied Sciences

kpedu

GeeksForGeeks. 2024b. *Difference Between Stack and Queue Data Structures*. Available at: <https://www.geeksforgeeks.org/difference-between-stack-and-queue-data-structures/>. Accessed 10 June 2024.



**Euroopan unionin
osarahoittama**



**Kokkola
Karleby**

CENTRIA
University of Applied Sciences

