IN1010 - OBJEKTORIENTERT PROGRAMMERING

VELKOMMEN TIL SYVENDE REPETISJONSTIME

Timen starter 12:15

REPETISJON

- Lenkeliste
- Stabel
- Kø
- Prioritetskø
- Iterable og Iterator

EKSEMPLER OG OPPGAVER

- En lenkeliste kan være enveis eller toveis
- Man trenger en start-peker, og det er ofte lurt med en slutt-peker
- Det kan være effektivt å ha en størrelse



LIVE-PROGRAMMERING: LINJER OG STOPP

Lag en klasse Linje

Lag en indre klasse Stopp i Linje med navnet på stoppet og en referanse til neste stopp

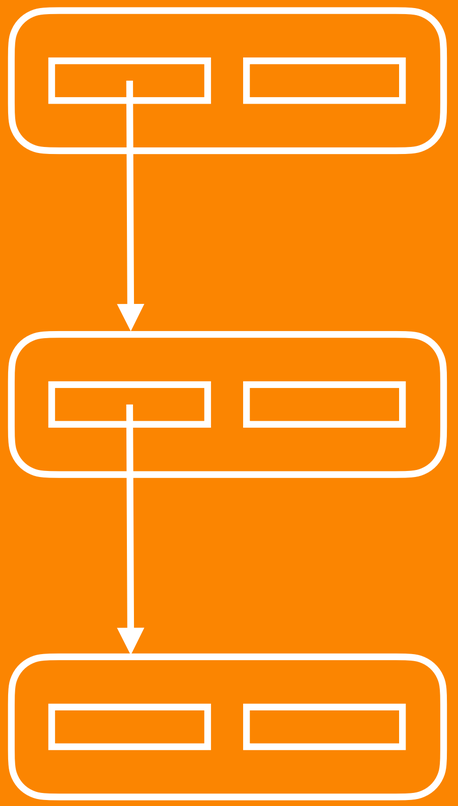
Linje-klassen skal ha en referanse til start-stoppet og en metode for å legge til stopp

REPETISJON

- Lenkeliste
- Stabel
- Kø
- Prioritetskø
- Iterable og Iterator

EKSEMPLER OG OPPGAVER

- En stabel kan være en variant av en lenkeliste
- Reglene for en stabel handler om hvordan å legge til og ta ut elementer
- Det siste elementet som ble lagt til, er det første som skal bli tatt ut, og det er kun lov å legge til og ta ut fra toppen av stabelen
- Blir ofte betegnet som LIFO, altså Last In, First Out



REPETISJON

- Lenkeliste
- Stabel
- **Kø**
- Prioritetskø
- Iterable og Iterator

EKSEMPLER OG OPPGAVER

- En kø kan også være en variant av en lenkeliste
- Igjen handler det om hvordan man legger til og tar ut elementer
- Det første elementet som ble satt inn, er det første som skal bli tatt ut, og det er kun lov å legge til på slutten av køen og å ta ut fra starten av køen
- På denne måten vil det elementet som har "ventet" lengst, blir tatt ut først
- Blir ofte betegnet som FIFO, altså First In, First Out

REPETISJON

- Lenkeliste
- Stabel
- Kø
- Prioritetskø
- Iterable og Iterator

EKSEMPLER OG OPPGAVER

- En prioritetskø er en variant av en kø
- Elementetene i køen er sortert etter prioritet
- En måte å løse dette på er å sette inn elementet på "riktig" plass ved innsetting, fordi da kan man alltid ta ut elementer fra starten av køen
- Dette er mulig å få til hvis man kan sammenligne elementene
- For at to objekter av en klasse skal kunne sammenlignes, kan klassen implementere interfacet Comparable<T> og dermed metoden compareTo(T o)

LIVE-PROGRAMMERING: PASIENTER

Lag en klasse Pasient som implementerer Comparable-interfacet

Klassen skal ta vare på en beskrivelse og en prioritet

Implementer compareTo-metoden fra Comparable-interfacet slik at det er mulig å sammenligne pasienter

REPETISJON

- Lenkeliste
- Stabel
- Kø
- Prioritetskø
- Iterable og Iterator

EKSEMPLER OG OPPGAVER

- For å kunne ta i bruk en forenklet for-løkke for å iterere over en lenkeliste, kan en benytte seg av intefacene Iterable<T> og Iterator<T>
- Klassen det skal itereres over implementerer Iterable<T> og må dermed override'e metoden iterator( )
- Så kan man ha en indre iterator-klasse som er spesifikk for klassen den er inni
- Den indre klassen implementerer Iterator<T> og må dermed override'e metodene hasNext( ) og next( )

SPØRSMÅLSRUNDE

VI SEES

HADE 👋