Seminaropplegg uke 10

Tema for uken: repetisjon.

**Vise litt String-operasjoner:**

Gjerne vise dokumentasjonen.

* char charAt(int index);
* int length()
* String substring(int startIndeks);  *alt:* String substring(int fra, int til); #nb fra og med. Og til, ikke til og med(!).
* String[] split(String splittVed);

*Burde være kjent med allerede:*

int compareTo(String annenTekst);

boolean equals(String annenTekst)

**Oppgave 1:**

|  |
| --- |
| Gitt en streng “hallo i luken, og hei paa deg du er veldig kul og snill”.  Skriv ut ordet og det andre tegnet for hvert ord i strengen, dersom ordet er kortere enn 2 i lengde, skrives det ut at ordet er for kort.  *Utskrift:*  Ord: 'hallo'. Andre bokstav: a  Ord: 'i'. Ingen andre bokstav.  Ord: 'luken,'. Andre bokstav: u  Ord: 'og'. Andre bokstav: g  Ord: 'hei'. Andre bokstav: e  Ord: 'paa'. Andre bokstav: a  Ord: 'deg,'. Andre bokstav: e  Ord: 'du'. Andre bokstav: u  Ord: 'er'. Andre bokstav: r  Ord: 'veldig'. Andre bokstav: e  Ord: 'kul'. Andre bokstav: u  Ord: 'og'. Andre bokstav: g  Ord: 'snill.'. Andre bokstav: n |

**Oppgave 2:**

|  |
| --- |
| Du har et array med ord, du skal nå finne alle ord som begynner med “pe” og legge dem i en liste (her velger du beholder selv, og du kan godt benytte deg av en av de selvlagde beholderne fra feks oblig 3). Deretter skal du skrive ut alle elementene i listen.  *Utfordring: programmet ditt skal i tillegg ta høyde for forskjell i store og små bokstaver, altså skal “Penal” også gi treff i søket på “pe”.*  String [] ordliste = {“hatt”, “pensel”, “kake”, “peke”, “Penal”, “lampe”, “lese”, “pese”}; |

**Oppgave 3:**

|  |
| --- |
| Med utgangspunkt i arrayet over, så skal du nå finne alle ord som slutter på “e” og legge dem i en beholder. Skriv deretter ut alle ordene. |

**Beholderoppgave - generisk:**

**Oppgave 4**

Vi skal lage en beholder som skal kunne ta vare på et vilkårlig antall objekter, ved å bruke

ferdiglagde datastrukturer i Java.

a) Skriv en generisk beholder som kun skal inneholde en peker til et objekt av typen

ArrayList, samt metodene:

• public void put(E element) - et element legges til i beholderen.

• public E take() - fjerner og returnerer det første elementet som ble lagt til i beholderen. Hvis det ikke finnes noen elementer skal null returneres.

b) Test programmet på følgende måte:

• Opprett en beholder som skal ta vare på typen String.

• Bekreft at et kall på take() returnerer null.

• Legg til strengene "test1" og "test2".

• Bekreft at første kall på take() returnerer "test1", og at andre kall på take()

returnerer "test2".

• Gjør et siste kall på take() og bekreft at det returnerer null.

c) Vi skal nå bruke HashMap istedet for ArrayList. Typen String skal brukes som nøkkel, men

verdien kan være av vilkårlig type.

• public void put(String key, V value) - legger til verdien value i beholderen, asso-

siert med nøkkelen key.

• public E take(String key) - fjerner og returnerer verdien som er assosiert med

nøkkelen. Dersom ingen verdi er assosiert med nøkkelen skal null returneres.

d) Test programmet på følgende måte:

• Opprett en beholder som skal ta vare på typen Integer.

• Bekreft at et kall på take("test1") returnerer null.

• Legg til verdien 1 med nøkkelen "test1" og verdien 2 med nøkkelen "test2".

• Bekreft at et kall på take("test1") returnerer 1, og at et kall på take("test2")

returnerer 2.

• Bekreft at et kall på take("test1") returnerer null, og at et kall på take("test2")

returnerer null.

Tilleggsspørsmål: Hvorfor må vi bruke Integer i stedet for int når vi skal oppbevare heltall i

en generisk beholder?