Uppgift 2
inda13

Christopher Lillthors
911005 – 3817

Kurskod: DD1339
KTH – VT13
lillt@kth.se

30 januari 2014

Kungliga Tekniska Högskolan
School of Computer science and communication
Civilingenjörsprogrammet Datateknik
Christopher Lillthors

Inlämningsuppgift
30 januari 2014

1 Svar till frågor

- *9:11*
  Printer

- *9.12*
  Den egna. Printerns metod anropas med super.getName();

- *9.13*
  Inga felmeddelanden, men den kommer inte skriva ut något. Faktum är att den ärver den metoden av Object.

- *9.14*
  Ja det fungerar, men den kommer skriva ut objektets magic number, vilket är en identifikation för ett objekt.

- *9.15*
  Ja det kommer att fungera, använder man @Override så kommer den att köra på den metoden istället för den ärvda av Objekt. Detta anrop kommer att skriva ut alla studenters namn.

- *9.16*
  Vehicle car = new Car() Vehicle är nu car statiska variabeltyp och Car är car dynamiska variabeltyp.

- isHealthy
  Denna är O(1), kontrollerna är oberoende av längden på buffern.

- Size()
  O(1), ty den hämtar ett fält och returnerar det.

- LinkedList()
  O(1) sätter två fält till null.

- addFirst() addLast() getFirst() getLast() removeFirst() isEmpty()
  Alla dessa (är/tillhör) O(1) och det är dom eftersom att ingen av metoderna gör något som involverar speciellt många operationer. På alla här utom isEmpty lägger man om en pekare och sätter den på ett annat objekt, det är knappast att betrakta som O(n)

- toString()
  Denna är linjär, av den enkla anledningen att den loopar igenom n stycken element för att skriva ut n stycken element.

- get()
  Denna är också linjär, eftersom att den måste söka igenom alla element fram till det sökta elementet n. Om det sökta elementet har index n och vi börjar i index 0, inses det lätt att operationen är linjär.