PG2100 – Programmering 2
3.6.15
Løsningsforslag

Oppgave 1 a)

```
kate
kate-L
kate-L locke-F

jack
kate-L jack-L
kate-L jack-L locke-F

jack
kate-L jack-L
sawyer-F kate-L jack-L locke-F

kate
kate-L
kate-F
```

Oppgave 1 b)

```
static int count(ArrayList<String> list, int n) {
  int count = 0;
  for (int i = 0; i < list.size(); i++) {
    if (list.get(i).length() == n) count++;
  }
  return count;
}
```

Oppgave 1 c)

```
static void replace(ArrayList<String> list) {
  for (int i = 0; i < list.size(); i++) {
    String replacement = "";
    String element = list.get(i);
    for (int j = 0; j < element.length(); j++) {
      replacement += "*";
    }
    list.set(i, replacement);
  }
}
```

Oppgave 2 a)

```
public class PhoneCard {
  private double credit;
  private double charge;
  private double minutes;
  private String id;

  public PhoneCard() {
    this(0.0, 0.0, "");
  }

  public PhoneCard(double credit, double charge, String id) {
    setCredit(credit);
    setCharge(charge);
    setId(id);
    setMinutes(0);
  }
```

```
  public double getMinutes() {
    return minutes;
  }

  public void setMinutes(double minutes) {
    this.minutes = minutes;
  }

  public String getId() {
    return id;
  }

  public void setId(String id) {
    this.id = id;
  }

  public double getCredit() {
    return credit;
  }

  public void setCredit(double credit) {
    this.credit = credit;
  }

  public double getCharge() {
    return charge;
  }

  public void setCharge(double charge) {
    this.charge = charge;
  }

  public String toString() {
    return "Credit left: " + getCredit()
         + "\nMinute charge: " + getCharge() +
         "\nMinutes used: " + getMinutes()
         + "\nId: (" + getId() + ")";
  }

  public boolean equals(Object other) {
    if (!(other instanceof PhoneCard)) return false;
    if (other == this) return true;
    PhoneCard pc = (PhoneCard) other;
    return getId().equals(pc.getId());
  }

  public double call(double minutes) {
    double spent = minutes * charge;
    if (getCredit() >= spent) {
       setCredit(getCredit() - spent);
       setMinutes(getMinutes() + minutes);
       return spent;
    } else {
       setMinutes(getMinutes() + getCredit() / getCharge());
       double rest = getCredit();
       setCredit(0);
       return rest;
    }
  }
}
```

Oppgave 2 b)

```
public class Oppgave2 {

  public static void main(String[] args) {
    int [] minutes = {10, 5, 1000};
    double credit = 200;
    double charge = 0.5;
    String id = "EasyPhoning 1234";
    System.out.println("Constructs a phone card (credit = " + credit
        + " kr, minute charge = " + charge
        + " kr, id = " + id + ")");
    PhoneCard card = new PhoneCard(credit, charge, id);

    for (int i = 0; i < minutes.length ; i++) {
      int min = minutes[i];
      System.out.println("\nMakes a " + min + " minutes call");
      System.out.println("Forbruk kr. " + card.call(min));
      System.out.println("Credit left: " + card.getCredit());
      System.out.println("Minutes used (total): " + card.getMinutes());
          }
    System.out.println("\nCard info:\n" + card.toString());
  }
}
```

Oppgave 3)

```
//1
JPanel pnlBrikker = new JPanel();
pnlBrikker.setLayout(new GridLayout(4, 4));
btnBrikker = new JButton[16];
btnOmigjen = new JButton("Prøv på nytt");
btnOmigjen.addActionListener(this);
tallene = new int[15];
//2
btnBrikker[i].addActionListener(this);
pnlBrikker.add(btnBrikker[i]);

//3
add(pnlBrikker, BorderLayout.CENTER);
add(btnOmigjen, BorderLayout.SOUTH);

setDefaultCloseOperation(EXIT_ON_CLOSE);
setSize(500, 500);
setVisible(true);

//4
JButton klikket = (JButton) e.getSource();
if (klikket == btnOmigjen) {
  reset();
} else {
  String text = klikket.getText();
  klikket.setText("");
  btnBlank.setText(text);
  btnBlank = klikket;
}
```