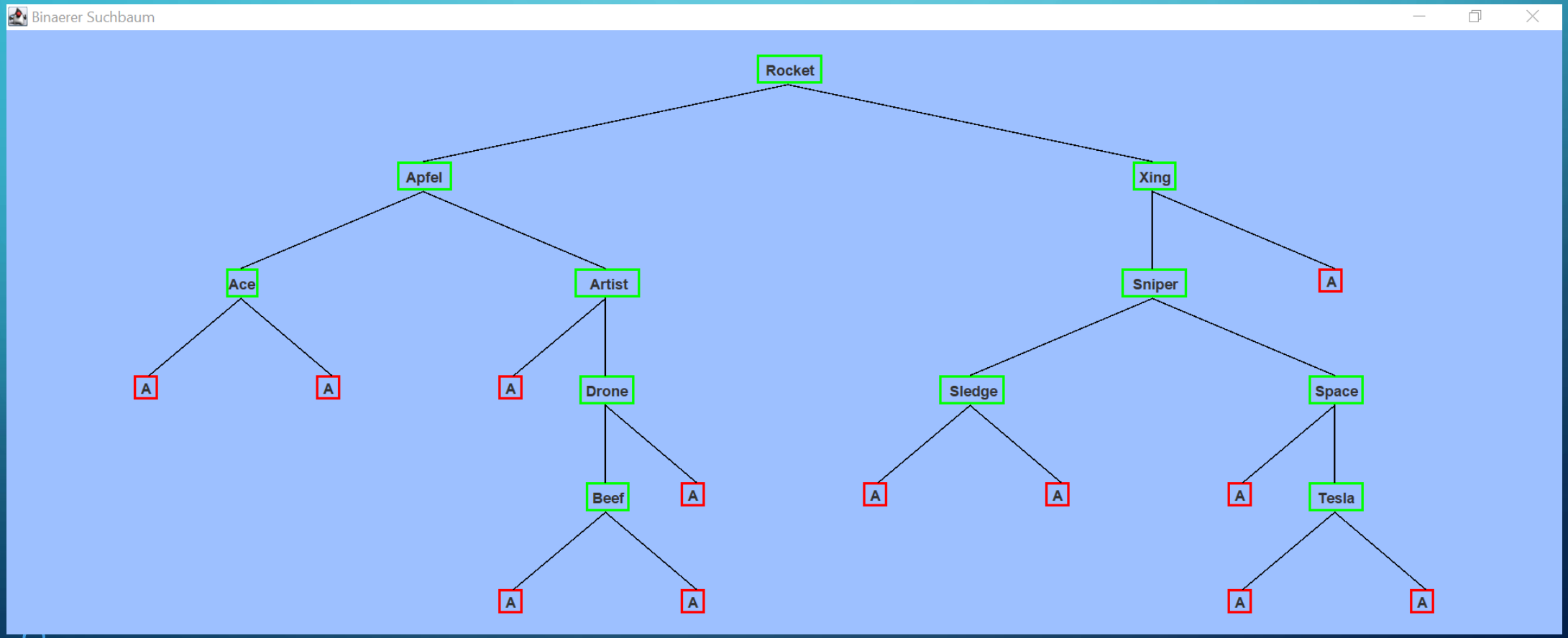
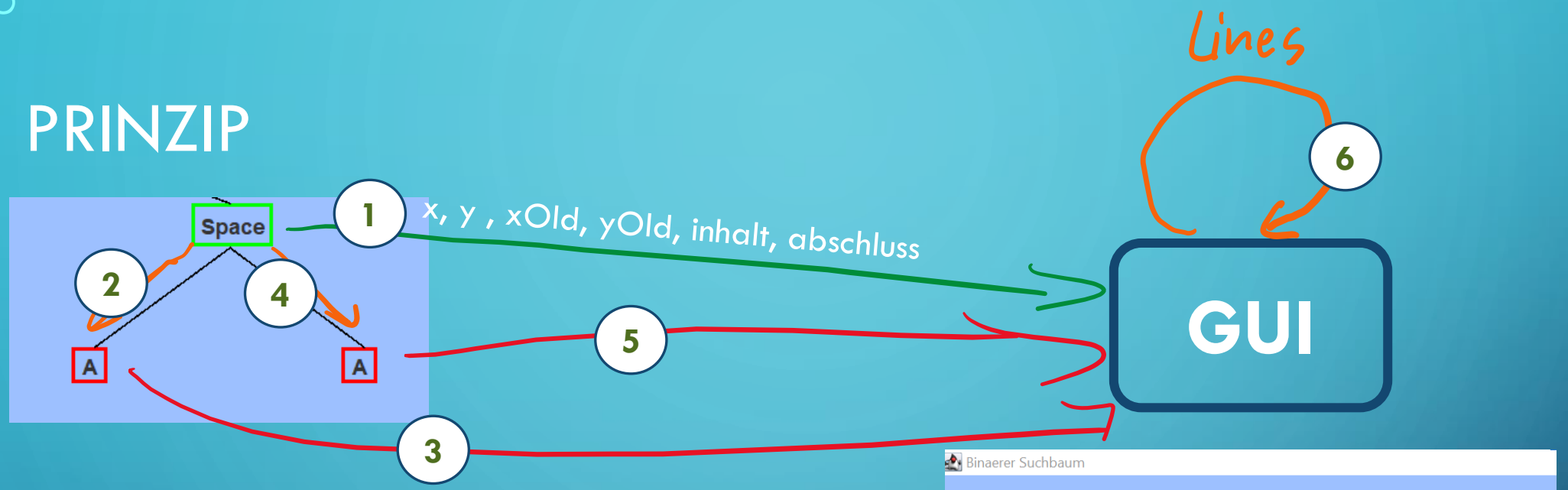
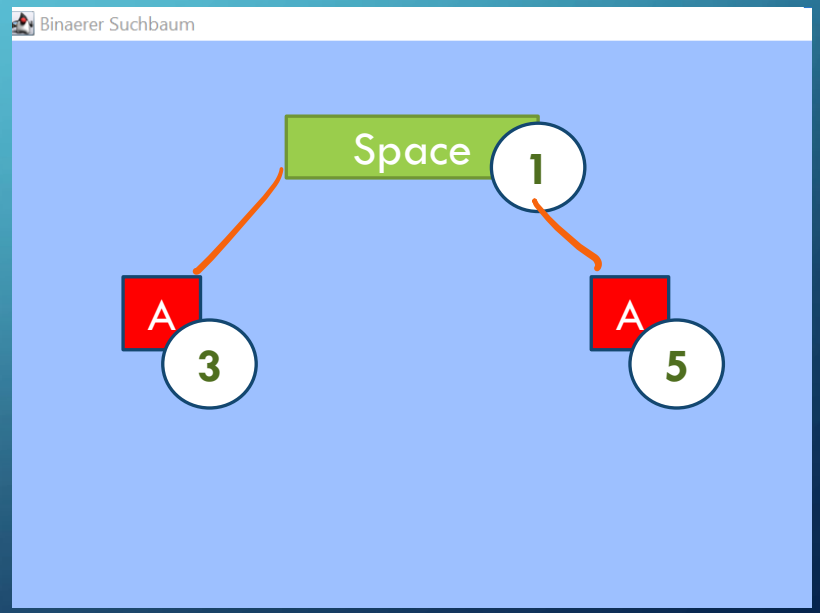
VISUALISIERUNG: BINÄRER SUCHBAUM

ZIEL

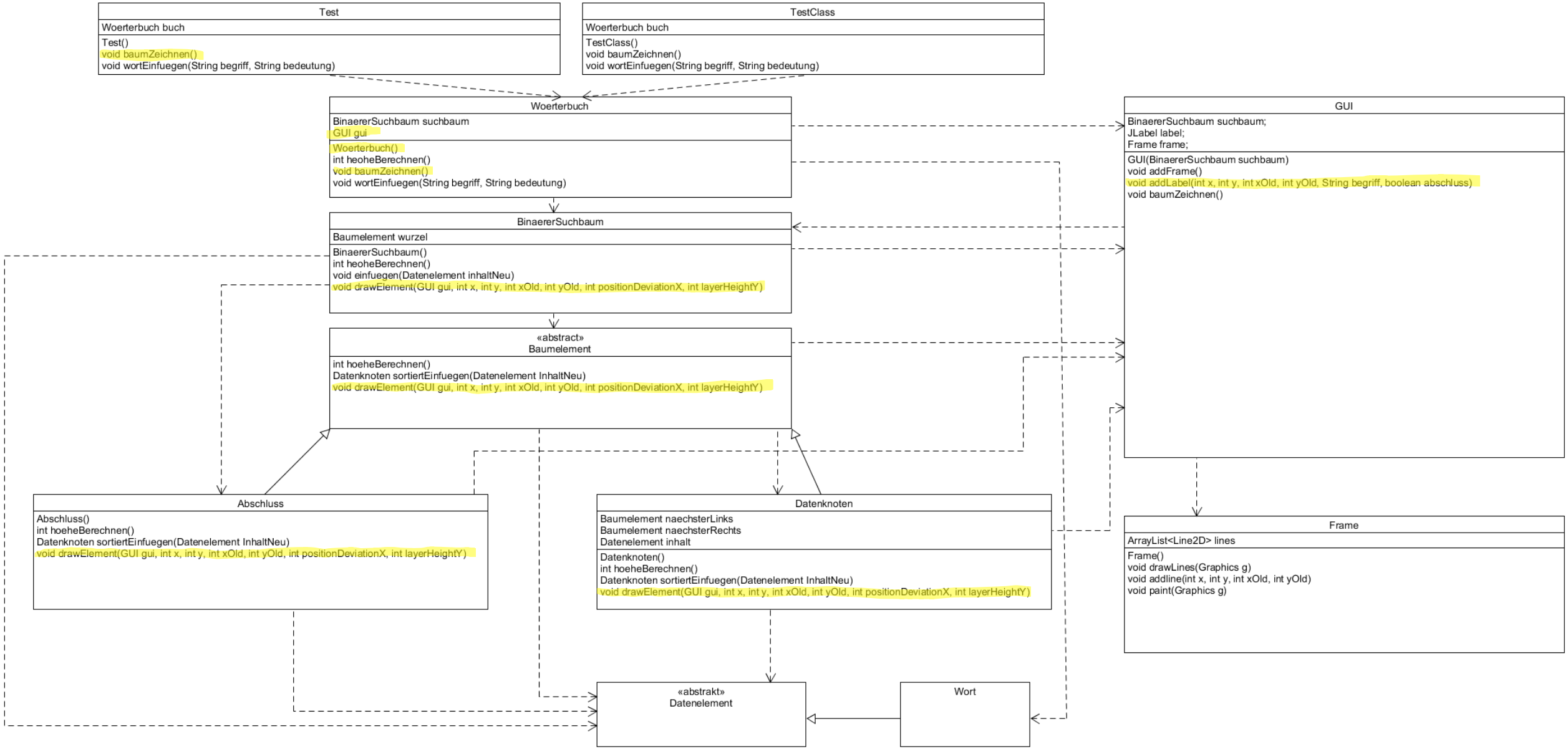

Binaerer Suchbaum
Rocket
Apfel
Xing
Ace
Artist
Sniper
A
A
A
A
Drone
Sledge
Space
A
A
A
Beef
A
Tesla
A
A
A
A

PRINZIP





2 gui, x, y, xOld, yOld, poitionDeviationX, layerHeightY


Test
Woerterbuch buch
Test()
void baumZeichnen()
void wortEinfuegen(String begriff, String bedeutung)
TestClass
Woerterbuch buch
TestClass()
void baumZeichnen()
void wortEinfuegen(String begriff, String bedeutung)
Woerterbuch
BinaererSuchbaum suchbaum
GUI gui
Woerterbuch()
int heoheBerechnen()
void baumZeichnen()
void wortEinfuegen(String begriff, String bedeutung)
GUI
BinaererSuchbaum suchbaum;
JLabel label;
Frame frame;
GUI(BinaererSuchbaum suchbaum)
void addFrame()
void addLabel(int x, int y, int xOld, int yOld, String begriff, boolean abschluss)
void baumZeichnen()
BinaererSuchbaum
Baumelement wurzel
BinaererSuchbaum()
int heoheBerechnen()
void einfuegen(Datenelement inhaltNeu)
void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY)
«abstract»
Baumelement
int hoeheBerechnen()
Datenknoten sortiertEinfuegen(Datenelement InhaltNeu)
void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY)
Abschluss
Abschluss()
int hoeheBerechnen()
Datenknoten sortiertEinfuegen(Datenelement InhaltNeu)
void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY)
Datenknoten
Baumelement naechsterLinks
Baumelement naechsterRechts
Datenelement inhalt
Datenknoten()
int hoeheBerechnen()
Datenknoten sortiertEinfuegen(Datenelement InhaltNeu)
void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY)
Frame
ArrayList<Line2D> lines
Frame()
void drawLines(Graphics g)
void addline(int x, int y, int xOld, int yOld)
void paint(Graphics g)
«abstrakt»
Datenelement
Wort

Klasse: Woerterbuch

```
private GUI gui;// Achtung: Nicht verändern

public Woerterbuch() {
    suchbaum = new BinaererSuchbaum();

    gui = new GUI(suchbaum);// Achtung: Nicht verändern
}

public void baumZeichnen(){ // Achtung: Nicht verändern
    gui.baumZeichnen();
}
```

Klasse: Testablauf

```
public void baumZeichnen(){
    buch.baumZeichnen();
}
```

Klasse: BinaererSuchbaum

```
public void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY){
    wurzel.drawElement(gui, x,  y, xOld, yOld, positionDeviationX, layerHeightY);
}
```

Klasse: Baumelement

```
public abstract void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY);
```

Klasse: Datenknoten

```
public void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY){

    int deviationLeft = positionDeviationX;
    int deviationRight = positionDeviationX;

    if(naechsterRechts instanceof Abschluss && naechsterLinks instanceof Abschluss){
        deviationRight = deviationLeft = positionDeviationX;
    }else if(naechsterRechts instanceof Abschluss){
        deviationRight = 0;
        positionDeviationX *= 2;
    }else if(naechsterLinks instanceof Abschluss){
        deviationLeft = 0;
        positionDeviationX *= 2;
    }

    gui.addLabel(x, y, xOld, yOld, inhalt.begriffGeben(),false);
    naechsterLinks.drawElement(gui, x-deviationRight, y+layerHeightY, x, y, positionDeviationX/2, layerHeightY);
    naechsterRechts.drawElement(gui, x+deviationLeft, y+layerHeightY, x, y, positionDeviationX/2, layerHeightY);

}
```

Klasse: Abschluss

```
public void drawElement(GUI gui, int x, int y, int xOld, int yOld, int positionDeviationX, int layerHeightY){
    gui.addLabel(x, y, xOld, yOld, "A",true);
}
```

Klasse: Datenelement

```
public abstract String begriffGeben();
```

Link

https://github.com/dancing-shark/Binaerer_Suchbaum