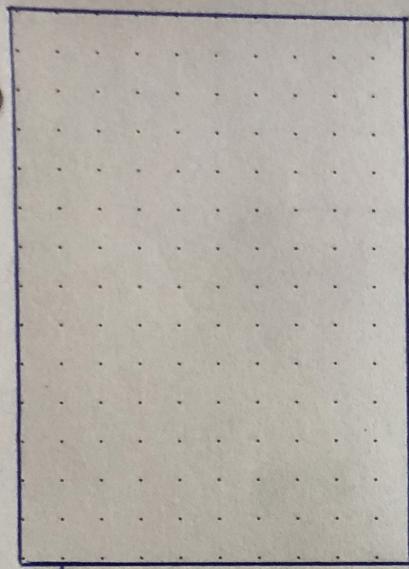
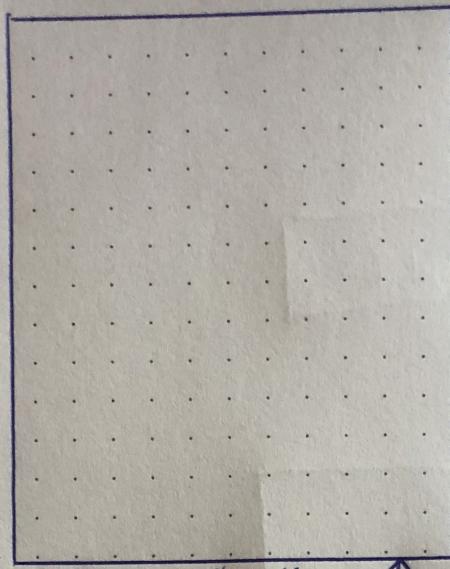


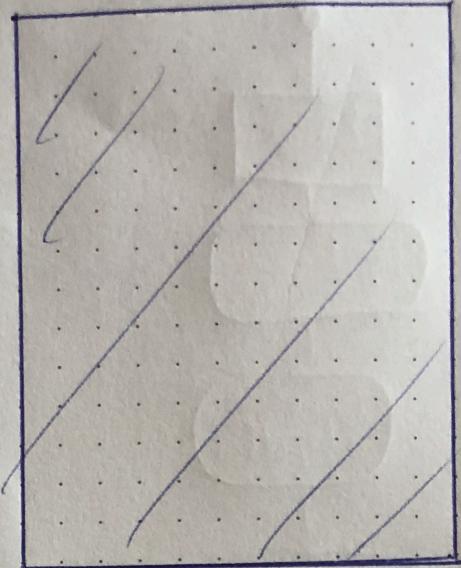
## Slizze



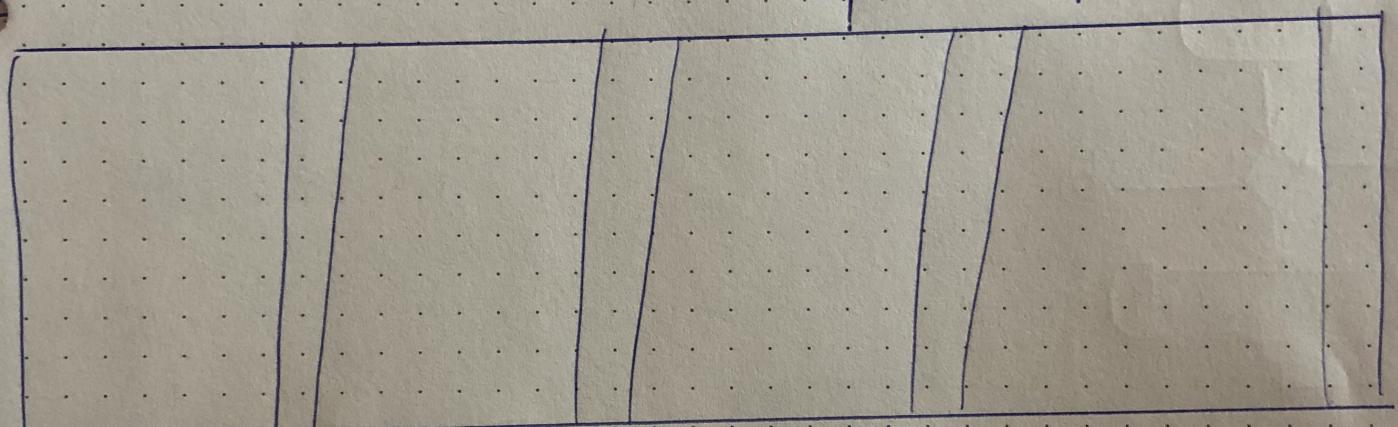
```
<div id="Deck">  
  ondlich(drawCard);</div>
```



```
<div id="Hand">
```



Wert  
Symbol > Vergleich mit - Spieldat  
Tippkarte - Ja/Nein



```
<div class="Hand">
```

Sor hren

Button ~~class~~  
 ondlich(sortHand);</button>

Hand.sort(jewelsila, {  
 return a-b?});

# Anwendungsfall-Diagramm // Aufgabe 3 // Auflegestapel

Interface Karten

Wert: number

Symbol: number

Wert: 7, 8, 9, 10, 11, 12, 13, 14

Symbol:  $\diamond, \heartsuit, \clubsuit, \spadesuit$

Benutzer      Interaktion



Mac-Mac System

Karte ziehen

1x p.  $P_1$

Karte legen

1x p.  $P_2$

Karten sammeln  
(Hand)

Jederzeit

Karte ziehen

$n =$  zufällige Number,  
Abhängig von Anzahl  
Karten im Deck  $F_1$

Füge Deck[n]  
unserer Hand[I] hinzu

Entferne Karte Deck[n]  
aus Deck[]

Hand sortieren

Sortiere Hand[]  
aufgrund von Wert

Sortiere Hand[]  
aufgrund von Symbol

Karte legen

Handkarten[I]  
Handkarten[Ex]

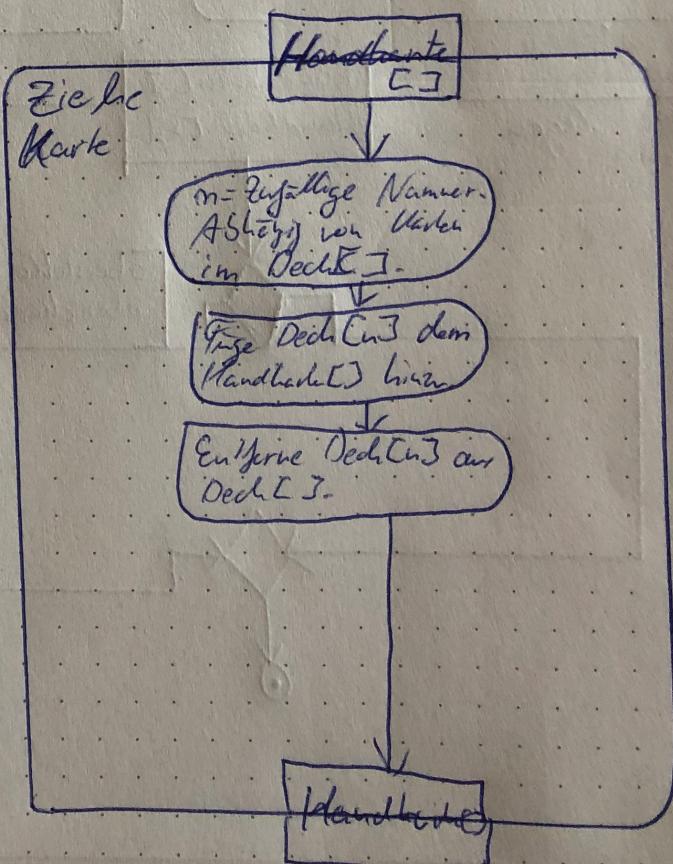
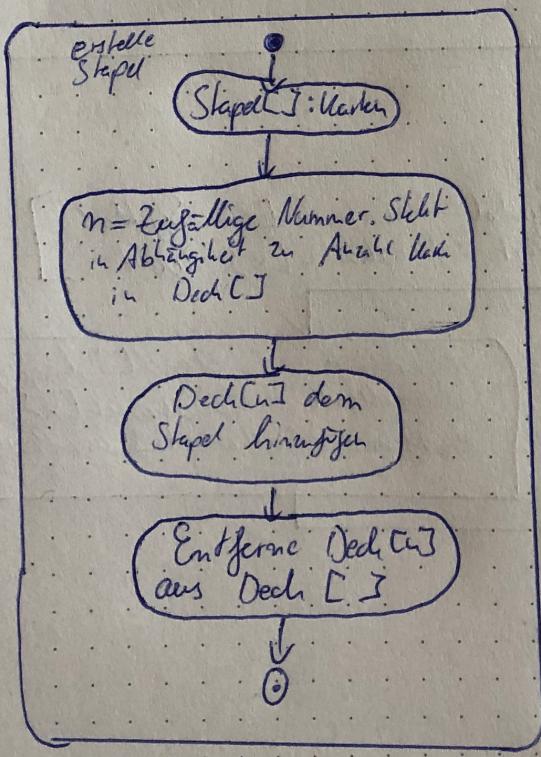
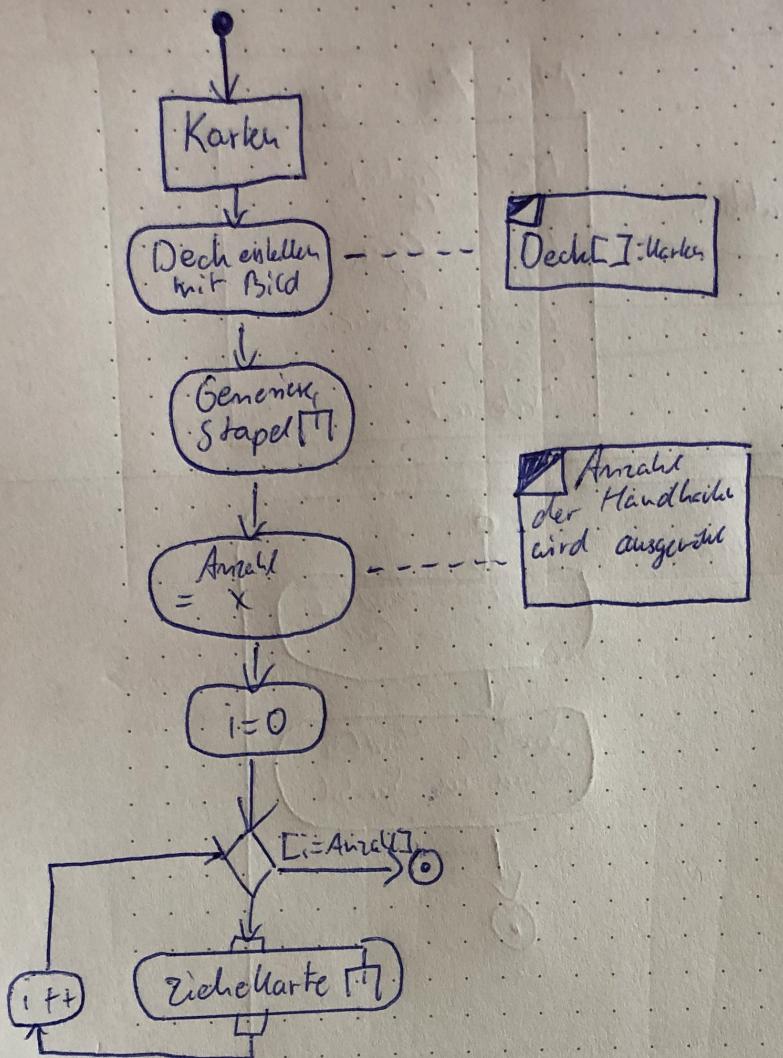
obenste Karte Symbol = Handkarten[Ex]. Symbol[]  
obenste Karte Wert = Handkarten[Ex]. Wert[]

obenste Karte wird Stapel[] hinzugefügt;  
obenste Karte = Handkarten[Ex];  
entferne Handkarten[Ex] aus Handkarten[]

Aufgabe

1 Maa-Ma.

II Nachzieh-Skript



> Klick auf Karte

Karte ziehen

> Klick auf Handkarte

Prüfen ob man legen darf

Karte ablegen

> Klick auf Sortieren

Karten sortieren