

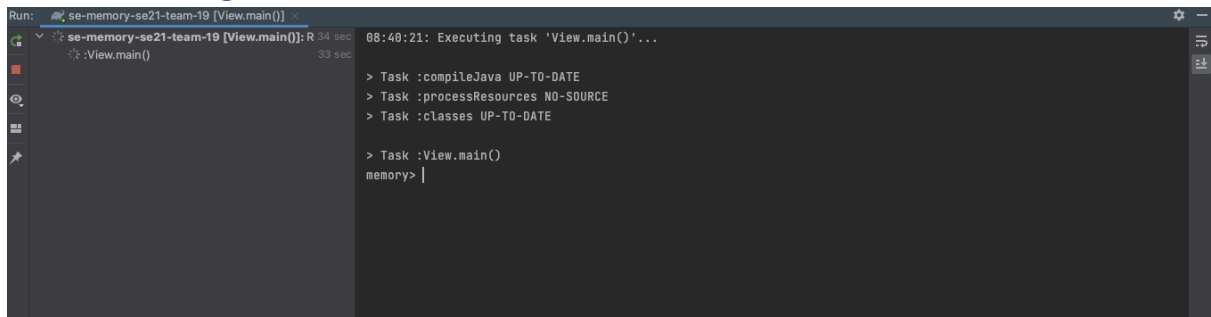
# Manuelles Testen zu Iteration-1

## Inhalt

<b>KORREKTE AUSGABEN .....</b>	<b>2</b>
<b>STARTEN DES PROGRAMMS .....</b>	<b>2</b>
<b>FALSCH EINGABEN .....</b>	<b>2</b>
<b>EINE KARTE AUFDECKEN .....</b>	<b>3</b>
<b>ZWEI UNTERSCHIEDLICHE KARTEN AUFDECKEN .....</b>	<b>3</b>
<b>NICHT GEFUNDENES PAAR WIRD WIEDER UMGEDREHT .....</b>	<b>4</b>
<b>GEFUNDENES PAAR WIRD ENTFERNT .....</b>	<b>4</b>
<b>ZWEIMAL DIE GLEICHE KARTE AUSGEWÄHLT.....</b>	<b>5</b>
<b>VERSUCH BEREITS GEFUNDEN KARTE AUFZUDECKEN MIT ERSTEN ZUG .....</b>	<b>5</b>
<b>ALLE KARTEN WURDEN GEFUNDEN .....</b>	<b>6</b>
 <b>FEHLER DIE BERICHTIGT WERDEN IN ITERATION-3 .....</b>	 <b>7</b>
<b>FALL 1: BEREITS GEFUNDENE KARTE WIRD AUSGEWÄHLT IM ERSTEN ZUG→NEUE KARTE WIRD AUSGEWÄHLT FÜR DEN ERSTEN ZUG →BEREITS GEFUNDENE KARTE WIRD AUSGEWÄHLT FÜR ZWEITEN ZUG .....</b>	<b>7</b>
<b>FALL 2: NEUE KARTE WIRD AUSGEWÄHLT IN ERSTEN ZUG→BEREITS GEFUNDENE KARTE WIRD AUSGEWÄHLT IM ZWEITEN ZUG→KARTE AUS ZUG 1 WIRD ERNEUT AUSGEWÄHLT.....</b>	<b>8</b>
<b>FALL 3: NEUE KARTE WIRD AUSGEWÄHLT IN ZUG 1→GLEICHE KARTE WIRD IN ZUG 2 AUSGEWÄHLT→NEUE KARTE AUSWÄHLEN FÜR ZUG 2 .....</b>	<b>9</b>
<b>FALL 4: NEUE KARTE WIRD AUSGEWÄHLT IN ERSTEN ZUG→GLEICHE KARTE WIRD IN ZUG 2 AUSGEWÄHLT→BEREITS GEFUNDENE KARTE WIRD ALS NÄCHSTES AUSGEWÄHLT .....</b>	<b>9</b>
<b>FALL 5: NEUE KARTE WIRD AUSGEWÄHLT IN ZUG 1→KARTE DIE BEREITS GEFUNDEN WURDE WIRD IN ZUG 2 AUSGEWÄHLT→NEUE KARTE WIRD AUSGEWÄHLT (NICHT PASSEND)→NEUER ZUG.....</b>	<b>10</b>
 <b>ANMERKUNGEN ZU BUGS .....</b>	 <b>12</b>

## Korrekte Ausgaben

### Starten des Programms



```
Run: se-memory-se21-team-19 [View.main()] x
se-memory-se21-team-19 [View.main()]: R 34 sec
  :View.main() 33 sec
08:48:21: Executing task 'View.main()'...
> Task :compileJava UP-TO-DATE
> Task :processResources NO-SOURCE
> Task :classes UP-TO-DATE
> Task :View.main()
memory> |
```

- Keine Probleme beim starten

### Falsche Eingaben

```
> Task :View.main()
memory> X
Error! Have not received correct number of parameters
memory> 1
Error! Have not received correct number of parameters
memory> X 1
Error! First entry was out of range
memory> 1 X
Error! Second entry was out of range
memory> X X
Error! First entry was out of range
memory> 9 9
Error! First entry was out of range
memory> 1 9
Error! Second entry was out of range
memory> 9 1
Error! First entry was out of range
memory> 1 1 1
Error! Have not received correct number of parameters
memory>
Error! Have not received correct number of parameters
memory> 1 1 X
Error! Have not received correct number of parameters
```

- Korrekte Fehlermeldung bei Eingabe von nicht akzeptierten Werten

Eine Karte aufdecken

```
memory> 1 1
  0 1 2 3
0 X X X X
1 X 2 X X
2 X X X X
3 X X X X
memory>
```

- Keine Probleme

Zwei unterschiedliche Karten aufdecken

```
memory> 1 1
  0 1 2 3
0 X X X X
1 X 2 X X
2 X X X X
3 X X X X
memory> 0 1
  0 1 2 3
0 X 6 X X
1 X 2 X X
2 X X X X
3 X X X X
memory>
```

- Keine Probleme

Nicht gefundenes Paar wird wieder umgedreht

```
memory> 1 1
  0 1 2 3
0 X X X X
1 X 2 X X
2 X X X X
3 X X X X
memory> 0 1
  0 1 2 3
0 X 6 X X
1 X 2 X X
2 X X X X
3 X X X X
memory> 0 0
  0 1 2 3
0 1 X X X
1 X X X X
2 X X X X
3 X X X X
memory> |
```

- Korrekte Ausgabe

Gefundenes Paar wird entfernt

```
memory> 1 1
  0 1 2 3
0 X X X X
1 X 2 X X
2 X X X X
3 X X X X
memory> 0 3
You've found a pair!
  0 1 2 3
0 X X X
1 X  X X
2 X X X X
3 X X X X
memory>
```

- Korrekte Ausgabe

Autor/Tester: Florian Muhrer

Zweimal die gleiche Karte ausgewählt

```
memory> 0 0
  0 1 2 3
0 1 X X X
1 X X X X
2 X X X X
3 X X X X
memory> 0 0
You've selected the same card twice!
memory>
```

- Korrekte Ausgabe

Versuch bereits gefunden Karte aufzudecken mit ersten Zug

```
memory> 0 3
The selected card was already found!
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X X
memory>
```

- Korrekte Ausgabe

Alle Karten wurden gefunden

```
memory> 2 1
  0 1 2 3
0
1
2   8   X
3
memory> 2 3
You've found a pair!
  0 1 2 3
0
1
2
3
You won!

BUILD SUCCESSFUL in 1h 24m 49s
2 actionable tasks: 1 executed, 1 up-to-date
10:05:12: Task execution finished 'View.main()'.
```

- Korrekte Beendung des Spiels wenn alle Kartenpaare gefunden wurden

## Fehler die berichtigt werden in Iteration-3

Fall 1: Bereits gefundene Karte wird ausgewählt im ersten Zug → Neue Karte wird ausgewählt für den ersten Zug → Bereits gefundene Karte wird ausgewählt für zweiten Zug

```
memory> 0 3
The selected card was already found!
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X X
memory> 0 0
  0 1 2 3
0 1 X X
1 X   X X
2 X X X X
3 X X X X
memory> 0 3
  0 1 2 3
0 1 X X
1 X   X X
2 X X X X
3 X X X X
memory> 0 1
  0 1 2 3
0 1 6 X
1 X   X X
2 X X X X
3 X X X X
memory> 0 2
```

```
  0 1 2 3
0 1 6 4
1 X   X X
2 X X X X
3 X X X X
memory> 1 0
  0 1 2 3
0 1 X X
1 5   X X
2 X X X X
3 X X X X
memory>
```

- Fehlermeldung bei Auswahl in Zug 2 fehlt
- Nach zweiten Korrekten Zug werden Karten nicht umgedreht
- Position 1 1 bleibt bestehen → Muss umgedreht sein

Fall 2: Neue Karte wird ausgewählt in ersten Zug → Bereits gefundene Karte wird ausgewählt im zweiten Zug → Karte aus Zug 1 wird erneut ausgewählt

```
memory> 3 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1
memory> 0 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1
memory> 3 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1
```

- Fehlermeldungen fehlen für Wahl der bereits aufgedeckten Karte und für Wahl der gleichen Karte



Fall 3: Neue Karte wird ausgewählt in Zug 1 → Gleiche Karte wird in Zug 2 ausgewählt → Neue Karte auswählen für Zug 2

```
memory> 3 1
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X 3 X X

memory> 3 1
You've selected the same card twice!
memory> 3 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1
```

- Karte aus Zug 1 muss umgedreht bleiben
- Erneuter erster Zug soll nicht erfolgen sondern der 2.

Fall 4: Neue Karte wird ausgewählt in ersten Zug → Gleiche Karte wird in Zug 2 ausgewählt → Bereits gefundene Karte wird als nächstes ausgewählt

```
memory> 3 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1

memory> 3 3
You've selected the same card twice!
memory> 0 3
The selected card was already found!
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X X
```

- Karte aus ersten Zug muss umgedreht bleiben
- Zug 2 soll ausgeführt werden können

Fall 5: Neue Karte wird ausgewählt in Zug 1 → Karte die bereits gefunden wurde wird in Zug 2 ausgewählt → Neue Karte wird ausgewählt (nicht passend) → Neuer Zug

```
memory> 0 0
  0 1 2 3
0 3   X
1 X X X X
2 X X X X
3 X X X X
memory> 0 1
  0 1 2 3
0 3   X
1 X X X X
2 X X X X
3 X X X X
memory> 0 2
  0 1 2 3
0 3   5
1 X X X X
2 X X X X
3 X X X X
memory> 1 0
  0 1 2 3
0 3   5
1 8 X X X
2 X X X X
3 X X X X
memory> |
```

- Fehlermeldung fehlt bei Versuch bereits gefundene Karte aufzudecken
- Nach nicht-finden müssen Karten wieder verdeckt werden

Fall 6: Neue Karte wird ausgewählt in Zug 1 → Karte die bereits gefunden wurde wird in Zug 2 ausgewählt → Neue Karte wird ausgewählt (passend) → Neuer Zug

```
memory> 3 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1
memory> 0 3
  0 1 2 3
0 X X X
1 X   X X
2 X X X X
3 X X X 1
memory> 0 0
  0 1 2 3
0 1 X X
1 X   X X
2 X X X X
3 X X X 1
memory> 0 1
  0 1 2 3
0 1 6 X
1 X   X X
2 X X X X
3 X X X 1
```

.....

```
memory> 0 2
  0 1 2 3
0 X X 4
1 X   X X
2 X X X X
3 X X X 1
memory>
```

- Fehlermeldung fehlt bei Versuch bereits gefundene Karte aufzudecken
- Meldung bei gefundenen Paar fehlt
- Gefundenes Paar wird nicht entfernt
- Gefundenes Paar wird fälschlicherweise wieder umgedreht

## Anmerkungen zu Bugs

Die in der manuellen Testung gefundenen Fehler werden im Rahmen der 3. Iteration behoben. Die Tasks hierzu werden erstellt und in das Kanban-Board der 3. Iteration hinzugefügt.