Exercise 3

```
for (int i = 0; i < 4; ++i) {
   for (int j = 1; j < 4; ++j) {
      a[i + 2][j - 1] = b * a[i][j] + 4;
   }
}
```

(2, 0) -> I = 0 und j = 2 und I = 2 und j = 1

Some true dependencies here:

a[i + 2][j - 1] = b * a[i][j] + 4;

from lecture

The **dependence direction vector** $D(\vec{i},\vec{j})$ is a vector of length n such that

- $D(\vec{i},\vec{j})_k = \begin{cases} \text{"<"} \ if \ \left(d(\vec{i},\vec{j})_k > 0\right) \\ \text{"="} \ if \ \left(d(\vec{i},\vec{j})_k = 0\right) \\ \text{">"} \ if \ \left(d(\vec{i},\vec{j})_k < 0\right) \end{cases}$
- With $d(\vec{i},\vec{j})_k = j_k - i_k$

Dep	Source	Type	Sink	Dist Vec	Dir Vec	Loop carried	Loop ind.	Dependence carried by loop
Dep1	a[i+2][j-1]	True dependency	a[i][j]	(2,-1)	(<, >)	x		i, j -loop

A[i+2][j-1] which are actually being read later on: i in interval [0,4] and j in interval [1,4]

	j=1	2	3	4
i = 0	A[2][0]	A[2][1]	A[2][2]	A[2][3]
1	A[3][0]	A[3][1]	A[3][2]	A[3][3]
2	A[4][0]	A[4][1]	A[4][2]	A[4][3]
3	A[5][0]	A[5][1]	A[5][2]	A[5][3]
4	A[6][0]	A[6][1]	A[6][2]	A[6][3]

The green statements

are not dependent and can be parallelized, the red ones can't.