Liebe Kolleginnen und Kollegen,

der Dreiecksrechner ist bei unserer Kundin sehr gut angekommen und wir haben einen Folgeauftrag erhalten.

Wir sollen eine Anwendung erstellen, die Multiplikationstabellen von 1*1 bis 20*20 anzeigt.

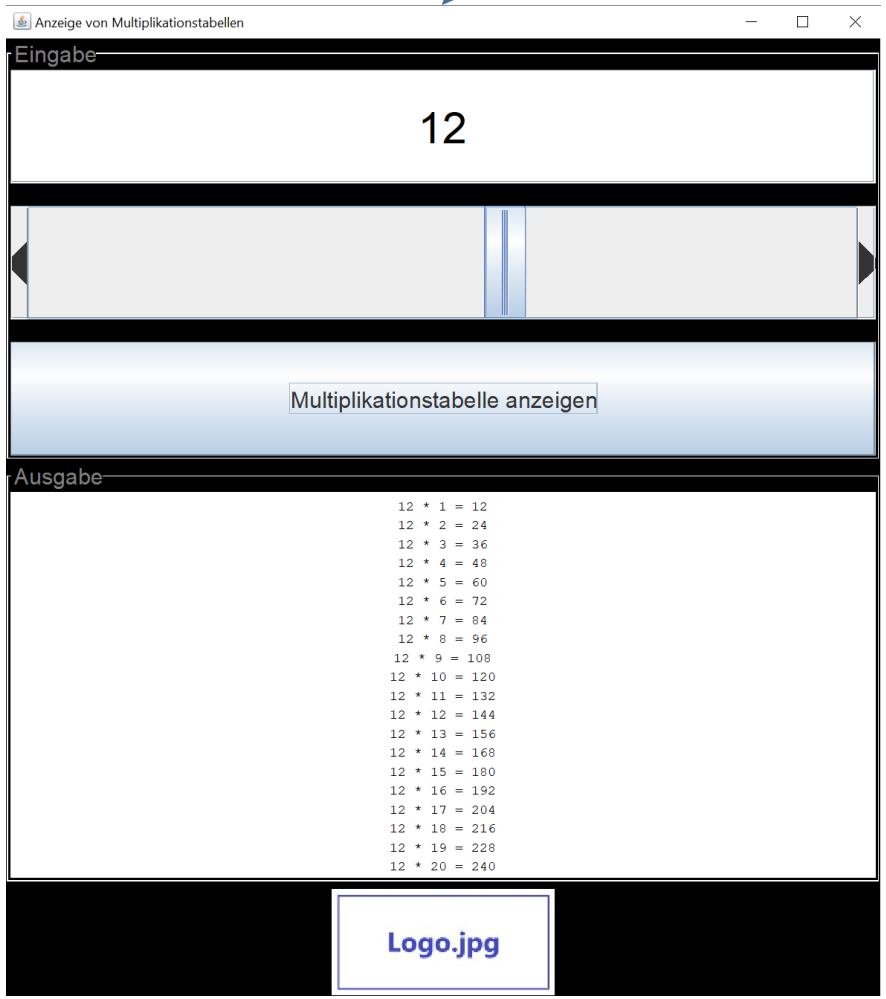
Die Benutzeroberfläche habe ich schon programmiert, es können bereits Daten angezeigt werden, es fehlt nur noch der eigentliche Algorithmus, um die Multiplikationstabellen zu berechnen und ein zweidimensionales Array damit zu befüllen.

Bitte geht entsprechend vor. Um den Algorithmus zu planen, bitte ich Euch, diesen vorher als Programmablaufplan (PAP), Struktogramm oder Pseudocode zu erstellen.

Bitte denkt auch bei dieser Anwendung daran, den Algorithmus entsprechend zu kommentieren.

Entwickelt auch hier geeignete Testfälle, um unsere Anwendung zu testen.

Benutzeroberfläche mit Dummydaten




```java
public class MultiplikationstabellenModel {

    //Konstanten der Instanz
    private int[][] multiplikationstafel = new int[20][20];

    /**
     * Erstellt die Multiplikationstafel.
     * @return Multiplikationstafel als int[][]
     */
    public int[][] ersteMultiplikationstafel() {

        //Hier bitte den Algorithmus für die Multiplikationstabelle von 1*1 bis 20*20 als zweidimensionales Array implementieren.

        return multiplikationstafel;
    }

    public Image liesBildEin(String bildPfad) {
        Image retImage = null;
        try {
            retImage = ImageIO.read(new File(bildPfad));
        } catch (IOException ioe) {
            System.out.println("Bild fehlt!");
        }
        return retImage;
    }
}
```

Bitte fügt hier den Algorithmus ein, der das zweidimensionale Array befüllt.

So soll das zweidimensionale Array befüllt aussehen.

Zeilenindex (äußeres Array)

Spaltenindex (inneres Array)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	2	4	6	8	10	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40
2	3	6	9	12	15	18	21	24	27	30	33	36	39	42	45	48	51	54	57	60
3	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60	64	68	72	76	80
4	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	85	90	95	100
5	6	12	18	24	30	36	42	48	54	60	66	72	78	84	90	96	102	108	114	120
6	7	14	21	28	35	42	49	56	63	70	77	84	91	98	105	112	119	126	133	140
7	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120	128	136	144	152	160
8	9	18	27	36	45	54	63	72	81	90	99	108	117	126	135	144	153	162	171	180
9	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150	160	170	180	190	200
10	11	22	33	44	55	66	77	88	99	110	121	132	143	154	165	176	187	198	209	220
11	12	24	36	48	60	72	84	96	108	120	132	144	156	168	180	192	204	216	228	240
12	13	26	39	52	65	78	91	104	117	130	143	156	169	182	195	208	221	234	247	260
13	14	28	42	56	70	84	98	112	126	140	154	168	182	196	210	224	238	252	266	280
14	15	30	45	60	75	90	105	120	135	150	165	180	195	210	225	240	255	270	285	300
15	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240	256	272	288	304	320
16	17	34	51	68	85	102	119	136	153	170	187	204	221	238	255	272	289	306	323	340
17	18	36	54	72	90	108	126	144	162	180	198	216	234	252	270	288	306	324	342	360
18	19	38	57	76	95	114	133	152	171	190	209	228	247	266	285	304	323	342	361	380
19	20	40	60	80	100	120	140	160	180	200	220	240	260	280	300	320	340	360	380	400