

Aufgabe 1: (10)

Finden Sie im folgenden Quellcode alle Fehler, die einen Compiler an der einwandfreien Übersetzung des Quellendes hindern.

Begründen Sie jeweils knapp warum Sie an der entsprechenden Stelle einen Fehler vermuten.

```
#include <stdio.h>
#include <stdlib.h>

int main
{
    int eingabe = 4096; teiler = 4096;

    While( teiler > 0 ){

        if( Eingabe \ teiler < 10 ){
            print("%d ", Eingabe / teiler)
        }
        else(

            switch(Eingabe / teiler){

                case 10: printf("A");
                    break;
                case 11: printf("B");
                    break;
                case 12: printf("C");
                    break;
                case 13: printf("D");
                    break;
                case 14: printf("E");
                    break;
                case 15: printf("F");
                    break;
                default: printf("Fehler");
            }
        )
        eingabe = eingabe%teiler;
        teiler / 16 = teiler;
    };

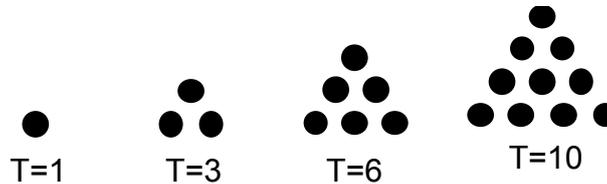
    printf("\n");
}
```

Name:

Klassenarbeit INFT
Programmieren in C

Aufgabe 2: (8)

Die Griechen des Altertums haben Zahlen geometrisch klassifiziert. Eine Zahl wurde zum Beispiel „triangulär“ genannt, wenn sich die entsprechende Anzahl der Steine zu einem symmetrischen Dreieck anordnen lässt. Die ersten triangulären Zahlen lauten 1, 3, 6, 10 usw.



Erstellen Sie ein Struktogramm, das die ersten 10 „triangulären“ Zahlen **berechnet** und ausgibt.

Aufgabe 3: (8)

Lassen Sie von einer eingegebenen maximal zehnstelligen Ganzzahl angeben, wie oft eine bestimmte einstellige Zahl darin vorkommt.

Lassen Sie die benötigten Zahlenwerte (maximal zehnstellige Ganzzahl und Suchzahl) einlesen.

Erstellen Sie den Programmablaufplan.

Aufgabe 4: (9)

Begründen Sie die Ausgabe, die der folgende Quellcodeausschnitt liefert.

```
int a = 8, b = 3, c = 2;
a = b++ + ( a * --c );
printf(" a = %d, b = %d, c = %d\n", a, b, c);
b = --a * ++c - ( b * c );
printf(" a = %d, b = %d, c = %d\n", a, b, c);
c = a-- * --b - ( a + c );
printf(" a = %d, b = %d, c = %d\n", a, b, c);
```

Aufgabe 5: (10)

Schreiben Sie ein Programm, das eine beliebig lange - vom Benutzer eingegebene - Ganzzahl in ihrer Reihenfolge umdreht.

Beispiel: Aus 123456789 wird 987654321

Gute Gelingen!