

1.) Felder

a) Deklarieren Sie eine Variable `feld` als ein 2-dimensionales Zahlen-Array und initialisieren Sie diese mit folgenden Werten:

3	5	7	9
2	4	6	8
1	3	5	7

`int feld [3][4] = {{3,5,7,9},{2,4,6,8},{1,3,5,7}};`

b) Eine Funktion `summe` soll geschrieben werden, welche alle Zahlen des Arrays addiert und das Ergebnis als Rückgabewert zurückliefert. Das Array soll als Parameter an die Funktion übergeben werden.

```
int summe ( int feld[3][4] )
{
    int zeile;
    int spalte;
    int erg=0;

    for (zeile = 0; zeile < 3; zeile++)
    {
        for (spalte = 0; spalte < 4; spalte++)
        {
            erg = erg + feld[zeile][spalte];
        }
    }
    return erg;
}
```

c) Zeigen Sie, wie diese Funktion `summe` aufgerufen werden kann, um die Summe aller Zahlen der Variable `feld` anzuzeigen.

`printf(" %d " , summe(feld));`

2.) Strukturen

a) Definieren Sie eine Struktur `Punkt`, welche die Koordinaten X und Y eines Punktes aufnehmen kann.

```
struct Punkt
{
    double x;
    double y;
};
```

b) Eine Funktion `abstand` soll 2 Punkte als Parameter erhalten und den Abstand der Punkte als Rückgabewert zurückliefern.

```
double Abstand( struct Punkt A, struct Punkt B )
{
    //Pythagoras
    return sqrt( pow( B.y - A.y , 2) + pow( B.x - A.x , 2) );
}
```

c) Wie kann die Funktion `abstand` aufgerufen werden?

```
int main(void)
{
    struct Punkt p1,p2;

    p1.x=1;
    p1.y=2;

    p2.x=3;
    p2.y=4;

    printf("Abstand: %.31f\n", Abstand(p1,p2));

    return 0;
}
```

3.) Kommandozeilen-Parameter, strtol

Schreiben Sie ein Programm, das eine ganze Zahl als Kommandozeilenparameter erhält. Es ist auszugeben, ob die angegebene Zahl gerade oder ungerade ist. Es ist eine Fehlerüberprüfung durchzuführen.

Usage: prog.exe zahl

Beispiele:

Eingabe: prog.exe 2	→ Ausgabe: 2 ist gerade
Eingabe: prog.exe 3	→ Ausgabe: 3 ist ungerade
Eingabe: prog.exe	→ Ausgabe: Usage: prog.exe zahl
Eingabe: prog.exe abc	→ Ausgabe: Usage: prog.exe zahl

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
void Usage()
{
    printf("Usage: prog.exe zahl\n");
    exit(EXIT_FAILURE);
}

int main( int argc, char* argv[])
{
    char *eptr;
    int zahl;
    int rest;

    //überprüfen, ob ein Kommandozeilenparameter angegeben wurde

    if (argc != 2)
    {
        Usage();
    }

    printf("%s ", argv[1] );

    zahl = (int) strtol(argv[1] & eptr, 10);

    //eptr überprüfen, ob der angegebene Parameter eine Zahl ist

    if ( *eptr != '\0' )
    {
        Usage();
    }
    else
    {
        if (zahl%2 == 0)
        {
            printf("ist gerade\n");
        }
        else
        {
            printf("ist ungerade\n");
        }
    }

    return 0;
}
```