

Ein Beispiel für alles: Suche nach der größten ganzen Zahl

von
Irene Rothe

Zi. B 241
irene.rothe@h-brs.de
Instagram: irenerothesdesign





von
Irene Rothe

Zi. B 241

irene.rothe@h-brs.de

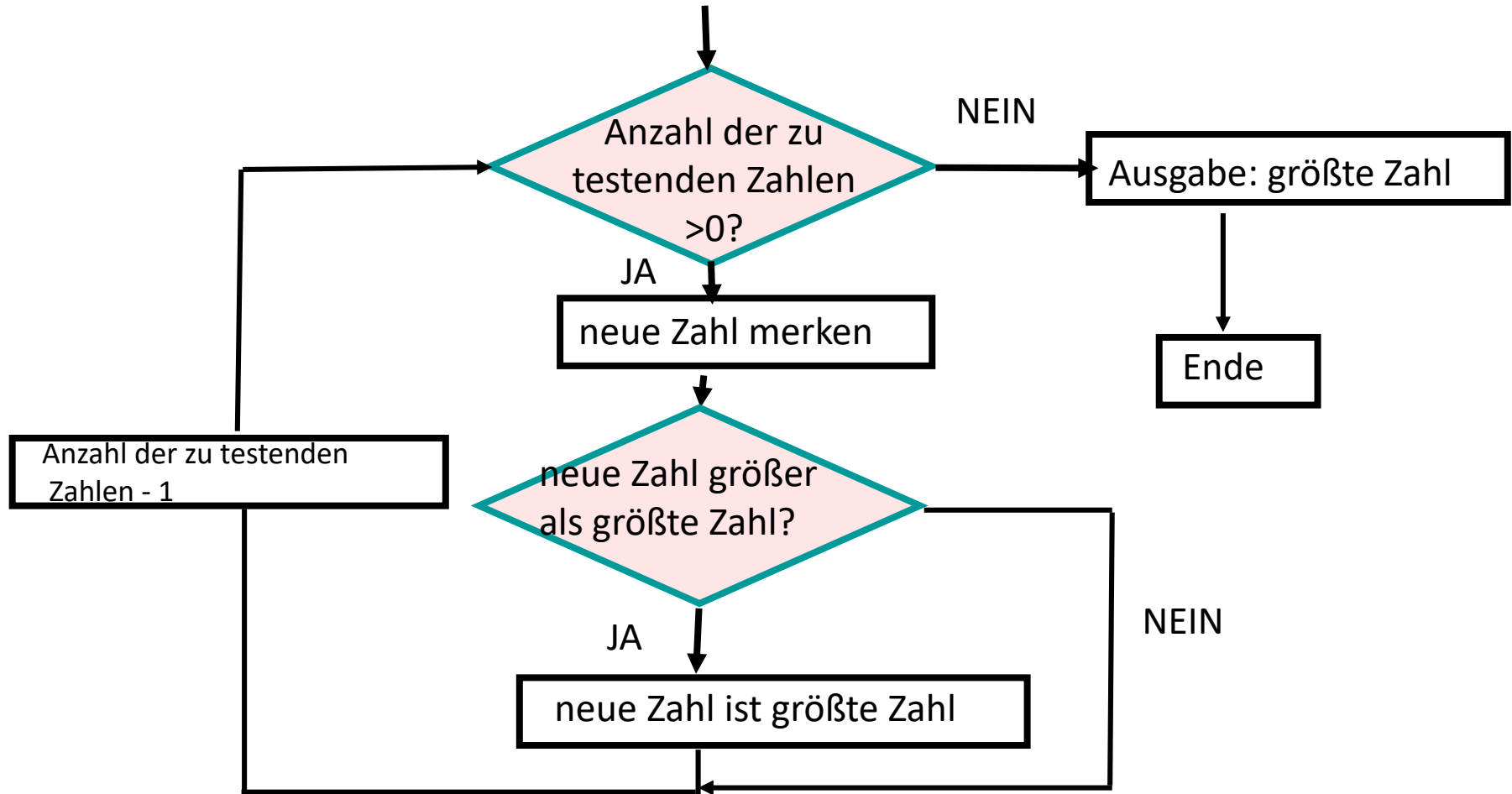
Instagram: irenerothesdesign

Herausforderung

1. **gegeben:** es werden 10 Zahlen nacheinander einzeln gezeigt
2. **gesucht:** größte Zahl
3. Verstehen des Problems
4. Entwurf einer Lösung → Algorithmus → Flussdiagramm
5. **Programmierung:** Aufschreiben des Algorithmus in einer anderen Sprache (Programmiersprache)
6. Übersetzen in Maschinensprache
7. Testen

Flussdiagramm: Beispiel - größte Zahl

erste Zahl ist größte Zahl und Anzahl der zu testenden Zahlen ist 10



Beispiel: Größte Zahl - Anfang

```
//Autor: Irene Rothe  
//Programm zum Finden der größten ganzen Zahl  
//Test: Eingabe: 3,5,1,9,5,7,8,3,10,4  
//Ausgabe: 10
```

Beispiel: Größte Zahl - Ende

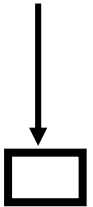
Start



```
#include <stdio.h>
int main() {
```

```
    return 0;
```

```
}
```



Größte Zahl - Deklaration

Start
↓

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;

    return 0;
}
```

↓
Ende



Größte Zahl - Zuweisung

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i;
    //zehn Zahlen sind gegeben
    i=10;

    return 0;
}
```

Start



Anzahl der zu testenden Zahlen i ist 10



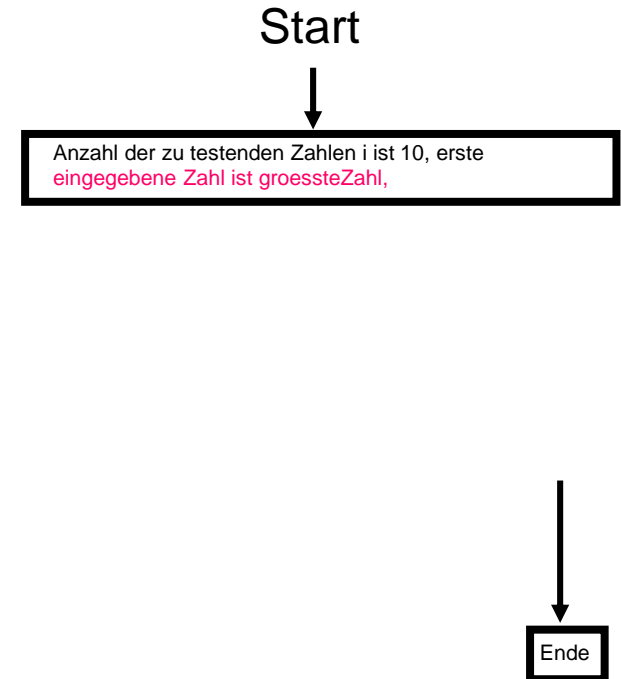
Ende



Größte Zahl - Eingabe

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i;
    //zehn Zahlen sind gegeben
    i=10;
    //Eingabe
    scanf("%i", &groessteZahl);

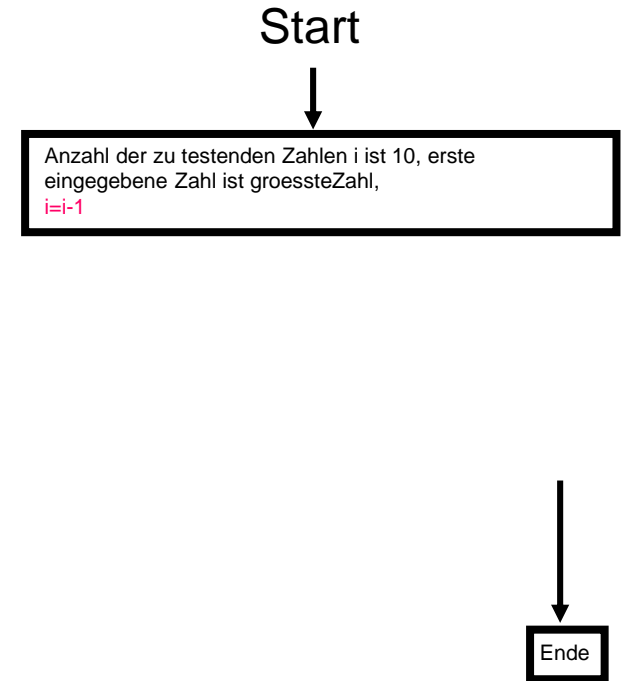
    return 0;
}
```



Größte Zahl - Eingabe

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i;
    //zehn Zahlen sind gegeben
    i=10;
    //Eingabe
    scanf("%i", &groessteZahl);
    i=i-1;

    return 0;
}
```

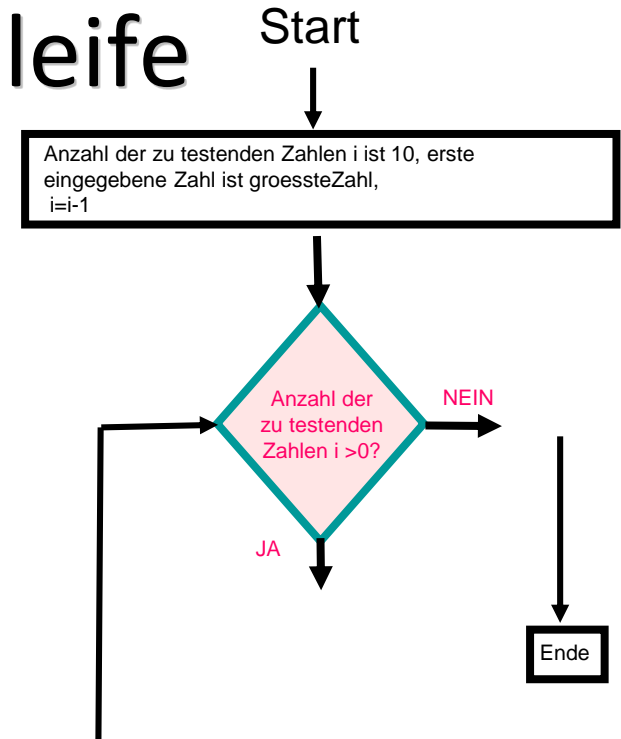


Größten Zahl - while-Schleife

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i;
    //zehn Zahlen sind gegeben
    i=10;
    //Eingabe
    scanf("%i",&groessteZahl);
    i=i-1;
    //Schleife
    while ( i > 0 ) {

    }

    return 0;
}
```

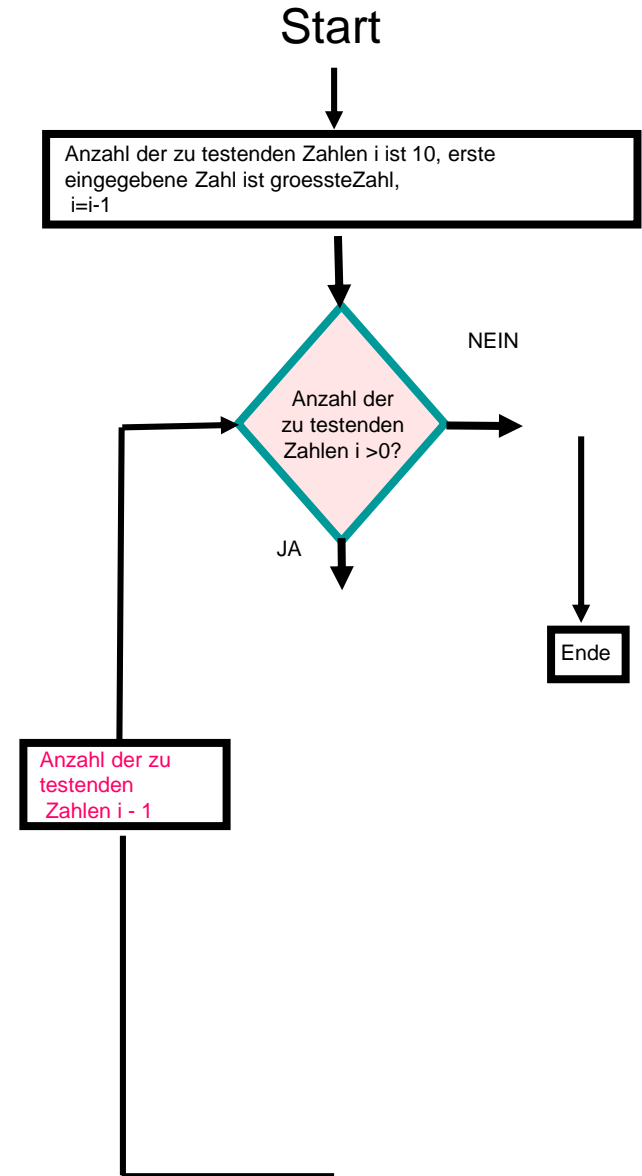


Größte Zahl - Abbruch

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i=10;
    //Eingabe
    scanf("%i",&groessteZahl);
    i=i-1;
    //Schleife
    while ( i > 0 ) {

        i=i-1;

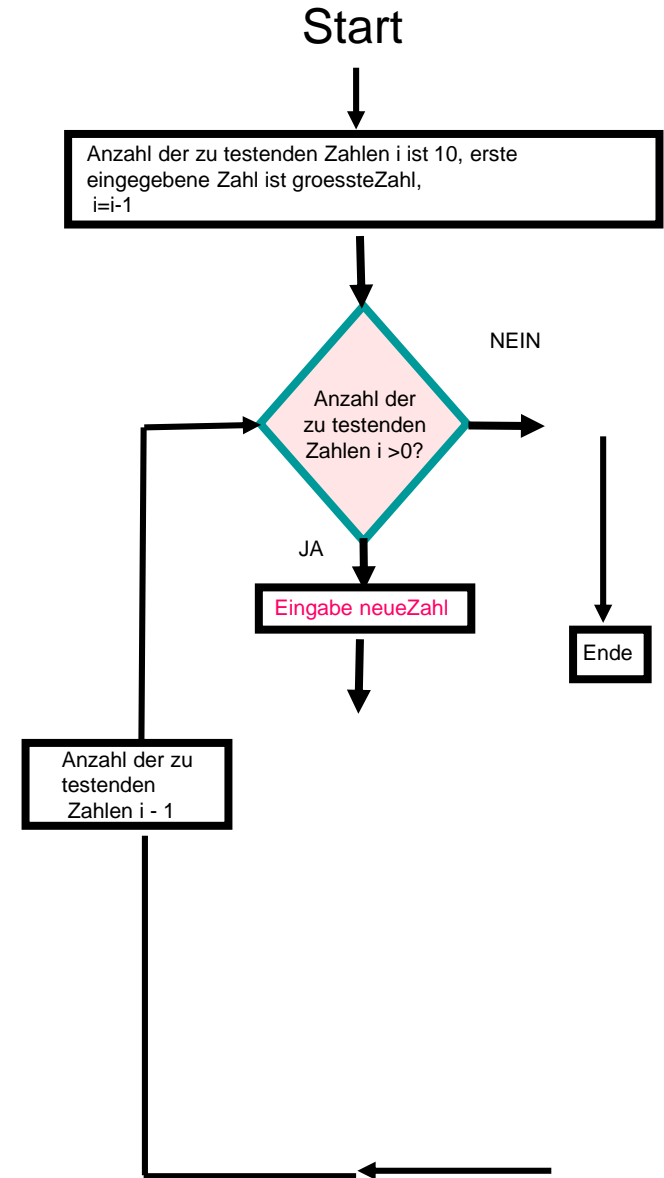
    }
    return 0;
}
```



Größte Zahl

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i=10;
    //Eingabe
    scanf("%i",&groessteZahl);
    i=i-1;
    //Schleife
    while ( i > 0 ) {
        //Eingabe
        scanf("%i",&neueZahl);

        i=i-1;
    }
    return 0;
}
```

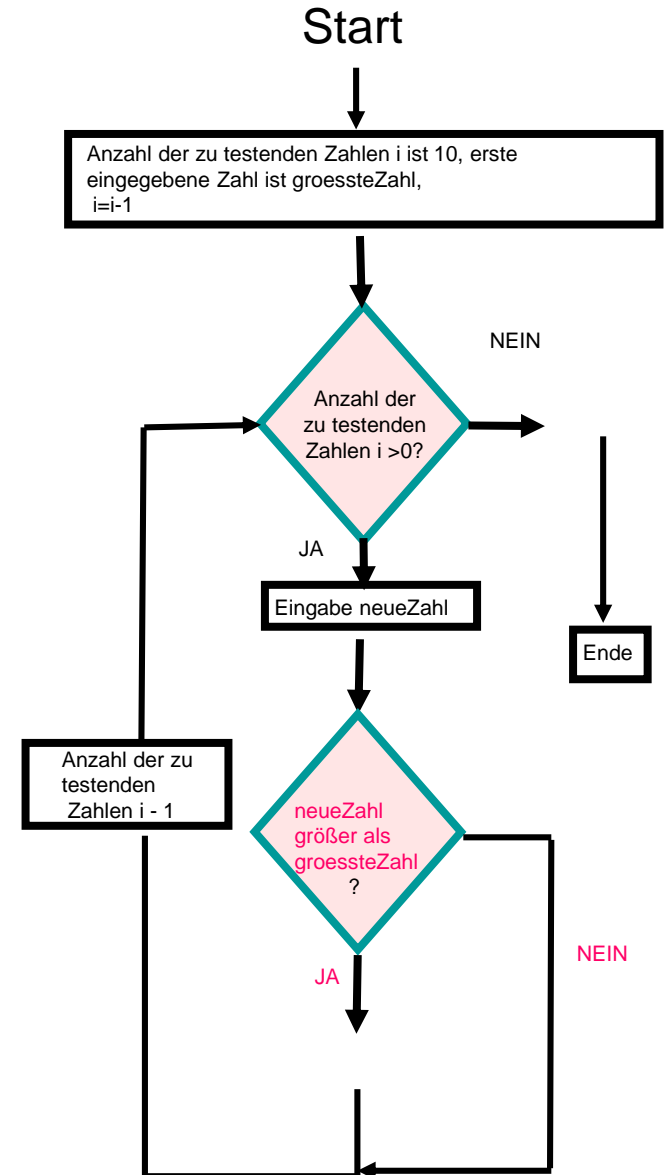


Größte Zahl - if/else

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i=10;
    //Eingabe
    scanf("%i",&groessteZahl);
    i=i-1;
    while ( i > 0 ) {
        //Eingabe
        scanf("%i",&neueZahl);
        if(neueZahl > groessteZahl){

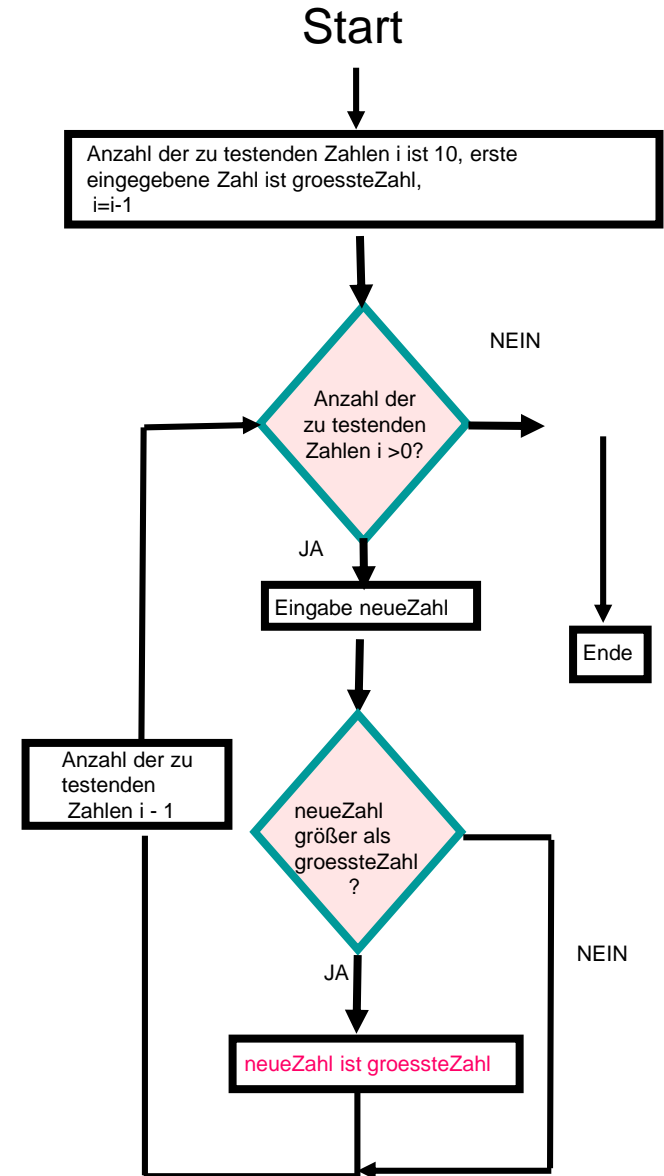
        }
        else{

        }
        i=i-1;
    }
    return 0;
}
```



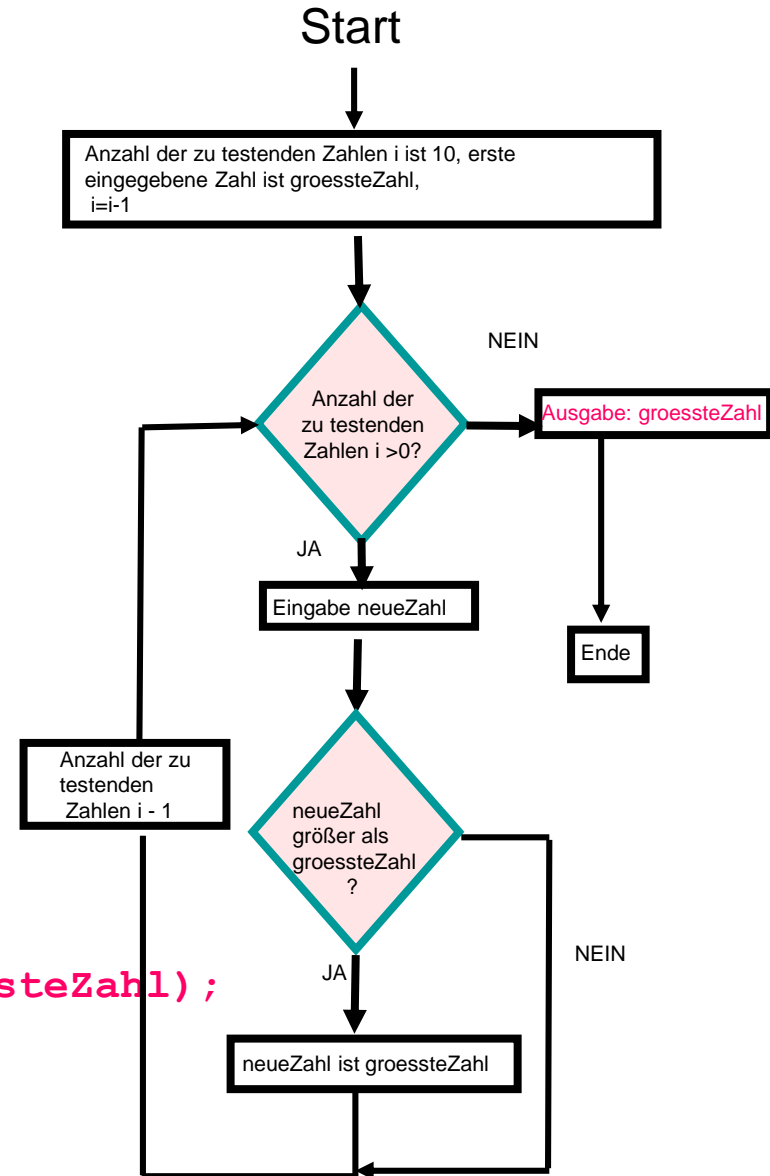
Größte Zahl

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i;
    i=10;
    //Eingabe
    scanf("%i",&groessteZahl);
    i=i-1;
    //Schleife
    while ( i > 0 ) {
        scanf("%i",&neueZahl);
        if(neueZahl > groessteZahl){
            groessteZahl = neueZahl;
        }
        else{}
        i=i-1;
    }
    return 0;
}
```



Größte Zahl - Ausgabe

```
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl;
    int i=10;
    //Eingabe
    scanf("%i",&groessteZahl);
    i=i-1;
    while ( i > 0 ) {
        scanf("%i",&neueZahl);
        if(neueZahl > groessteZahl){
            groessteZahl=neueZahl;
        }
        i=i-1;
    }
    printf("Die groesste Zahl ist %i",groessteZahl);
    return 0;
}
```



Noch schöner mit do/while

```
//Autor: Irene Rothe
//Programm zum Finden der größten ganzen Zahl
//Test: Eingabe: 3,5,1,9,5,7,8,3,10,4
//Ausgabe: 10
#include <stdio.h>
int main() {
    //Deklaration
    int neueZahl;
    int groessteZahl=0;
    int i=10;
    do {
        scanf("%i",&neueZahl);
        if(neueZahl > groessteZahl){
            groessteZahl=neueZahl;
        }
        i=i-1;
    } while ( i > 0 ) ;
    printf("Die groesste Zahl ist %i",groessteZahl);
    return 0;
}
```

Mit Funktion (als Arbeit mit Kopien - call by value) und return

```
#include <stdio.h>

int berechneGroessteZahl(int anzahl){
    int zahl,gz; //lokale Variablen
    printf("Bitte geben Sie nacheinander Zahlen ein:\n");
    scanf("%i",&zahl);
    while ( anzahl > 0 ) {
        scanf("%i",&zahl);
        if(zahl > gz){
            gz=zahl;
        }
        anzahl=anzahl-1;
    }
    return gz; //Rückgabewert
}

int main() {
    int ergebnis, anzahlzahlen=10;
    ergebnis = berechneGroessteZahl(anzahlzahlen); //Aufruf der Funktion
    printf("Die groesste Zahl ist %i\n", ergebnis);
    return 0;
}
```

Mit Feld (array)

```
#include <stdio.h>

int main() {
    int zahlen[10]; //Laenge muss hier fest angegeben werden!
    int groesstezahl;
    int i;
    printf("Bitte geben Sie nacheinander Zahlen ein:\n");
    for (i=0;i<10;i++){
        scanf("%i",&zahlen[i]);
    }
    groesstezahl=zahlen[0];
    for(i=1;i<10;i++){
        if(zahlen[i] > groesstezahl){
            groesstezahl=zahlen[i];
        }
    }
    printf("Die groesste Zahl ist %i\n", groesstezahl);
    return 0;
}
```

Mit Feld und Funktion (call by reference)

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

```
int berechneGroessteZahl(int feld[], int anzahl){
```

```
    int gz,i;
```

```
    gz=feld[0];
```

```
    for(i=0;i<anzahl;i++){
```

```
        if(feld[i] > gz){
```

```
            gz=feld[i];
```

```
        }
```

```
    }
```

```
    return gz;
```

```
}
```

```
int main() {
```

```
    int zahlen[10];
```

```
    int ergebnis;
```

```
    int i, anzahl=10;
```

```
    printf("Bitte geben Sie nacheinander %i Zahlen ein:\n", anzahl);
```

```
    for (i=0;i<anzahl;i++){
```

```
        scanf("%i",&zahlen[i]);
```

```
    }
```

```
    ergebnis = berechneGroessteZahl(zahlen, 10); //call by reference
```

```
    printf("Die groesste Zahl ist %i\n", ergebnis);
```

```
    return 0;
```

```
}
```



Mit Funktion und extra noch call by reference

```
#include <stdio.h>

void berechneGroessteZahl(int feld[], int anzahl, int *gz){ //Übergabe eines Zeigers
    *gz=feld[0]; //Inhaltsoperator
    int i;
    for(i=1;i<anzahl;i++){
        if(feld[i] > *gz){
            *gz=feld[i];
        }
    }
}

int main() {
    int zahlen[10];
    int ergebnis;
    int i, anzahl=10;
    printf("Bitte geben Sie nacheinander %i Zahlen ein:\n", anzahl);
    for (i=0;i<anzahl;i++){
        scanf("%i",&zahlen[i]);
    }
    berechneGroessteZahl(zahlen, 10, &ergebnis); //call by reference: Übergabe einer Adresse
    printf("Die groesste Zahl ist %i\n", ergebnis);
    return 0;
}
```



Mit Strukturen

```
#include <stdio.h>

struct Studierender{
    char Name[20];
    int Alter;
};

int main(){
    struct Studierender studies[]={{"Max",20},{ "Coco",30},{ "Irene",42},{ "Linda",18}};
    int indexmerker=0, aeltester=0;
    int i, anzahl=4;
    aeltester=studies[0].Alter;
    for(i=1;i<anzahl;i++){
        if(studies[i].Alter > aeltester){
            aeltester=studies[i].Alter;
            indexmerker=i;
        }
    }
    printf("%s ist mit %i der aelteste Studierende.\n",
           studies[indexmerker].Name,aeltester);

    return 0;
}
```



Mit Listen

```
...
struct Studierendenliste{
    char vorname[15];
    int alter;
    struct Studierendenliste *naechsterStudierender;
};

int berechneAeltesteStudierende(struct Studierendenliste *liste){
    liste = liste->naechsterStudierender;
    int aeltesteStudierende=0;
    while(liste != NULL) {
        if(liste->alter>aeltesteStudierende){
            aeltesteStudierende=liste->alter;
        }
        liste = liste->naechsterStudierender;
    }
    return aeltesteStudierende;
}

int main(){
    struct Studierendenliste *listeSoSe23;
    ...
    aeltesteStudierende=berechneAeltesteStudierende(listeSoSe23);
    printf("Die aelteste Studierende ist %i Jahre alt.",aeltesteStudierende);

    return 0;
}
```

