

Subthieren mit zwei überlangen Zahlen:

```
int umdrehen = 0;
scanf("%s", s1); // Hier wird in s1 eingelesen → Eingabe: 149 → Ascii: 49|57|48
```

```
char s1[21];
```

49	52	57	\0	\0	\0	\0
s1[0]							s[20]

```
scanf("%s", s2); // Hier wird in s2 eingelesen → Eingabe: 19018 → Ascii: 49|52|57
```

```
char s2[21];
```

49	57	48	49	56	\0	\0
s1[0]							s[20]

```
if(strlen(s2) > strlen(s1))
{
    umdrehen = 1;
    strcpy(ss1,s2);

    for(int i = 0; i < strlen(s1); i++)
    {
        ss2[i + (strlen(s2) - strlen(s1))] = s1[i];
    }
    for( i = 0; i < (strlen(s2) - strlen(s1)); i++)
    {
        ss2[i] = 48;
    }
    ss2[strlen(ss1)] = '\0';
}
```

```
char ss1[21]; (entspricht Zahl: 19018)
```

49	57	48	49	56	\0	\0
ss1[0]						ss1[6]

```
char ss2[21]; (entspricht Zahl: 00149 → 2 NULLEN wurden vorangestellt !)
```

48	48	49	52	57	\0	\0
ss1[0]						ss1[6]

Errechnen der Subtraktion: Ergebnis wird in **ergebnis** geschrieben.

```
int uebertrag = 0;

for(int i = 0; i < strlen(ss1); i++)
{
    if ( ( (ss1[strlen(ss1)-1-i]-48) - (ss2[strlen(ss2)-1-i]-48) - uebertrag) < 0)
    {
        ergebnis[strlen(ss1)-i+2] = ( ((ss1[strlen(ss1)-1-i]-48) - (ss2[strlen(ss2)-1-i]-48) - uebertrag) + 10 )+48;
        uebertrag = 1;
    }
    else
    {
        ergebnis[strlen(ss1)-i+2] = ( (( ss1[strlen(ss1)-1-i]-48) - (ss2[strlen(ss2)-1-i]-48) - uebertrag))+48;
        uebertrag = 0;
    }
}
if(uebertrag > 0)
{
    ergebnis[1] = uebertrag + 48;
    if(umgedreht)
    {
        ergebnis[0] = '-';
    }
    else
    {
        ergebnis[0] = ' ';
    }
}
else
{
    ergebnis[0] = ' ';
    if(umgedreht)
    {
        ergebnis[1] = '-';
    }
    else
    {
        ergebnis[1] = ' ';
    }
}
}
```

Wenn s2 größer s1 ist, dann drehe ich die Zahlen einfach um, schreibe also s2 direkt in ss1. s1 hingegen muss ich dann, um die Differenz von (s1 - s2) versetzt, wie beim Addieren, allerdings jetzt in **ss2** schreiben (weil umgedreht.)
Sind die Zahlen gleich groß, muss ich prüfen ob s2[0] größer als s1[0] ist.
Wenn ja, dann muss ich ebenfalls umdrehen, schreibe also s1 in ss2 und s2 in ss1.

Anschließend beginne ich zu rechnen.
Ich prüfe ob die ss1[i] kleiner als ss2[i] ist.
Ist dies gegeben, dann subtrahiere ich ss2[i] von ss1[i] und addiere anschließend 10 dazu.
Da Ergebnis speichere ich schließlich in ergebnis[i+2].
Plus 2 deshalb, weil ich an der nullten Stelle (ergebnis[0]) einen Platz für das Minus und an der ersten Stelle (ergebnis[1]) einen Platz für den Übertrag freihalte.
Im Übertrag merke ich mir 1.

Am Schluss muss ich noch ein '-' vor das Ergebnis setzen, falls ich die Zahlen vorher umgedreht habe, was ich mir in der Variable **umgedreht** gemerkt habe.

char ergebnis[22]; (entspricht Zahl: 18869)

32	45	49	56	56	54	57	\0
[0]	[1]	[2]	[6]	[7]