

Aufgabe 7.5

Schreiben Sie die folgenden Funktionen zur Zeichenkettenbearbeitung.

- Schreiben Sie eine Funktion `int AnzahlZeichen (char K[] , char c)`, die überprüft, wie oft ein Zeichen `c` in einer übergebenen Zeichenkette enthalten ist, und die diese Anzahl zurückgibt.

Beispiel:

```
char Kette[20 + 1] = "Uebungen sind schoen";
cout << AnzahlZeichen ( Kette , 'e' );    // gibt 3 aus.
```

- Schreiben Sie eine Funktion `bool Palindrom (char K [])`, die überprüft, ob eine übergebene Zeichenkette ein Palindrom ist. Der Rückgabewert ist entweder `true` oder `false`.

Ein Wort heißt Palindrom, wenn es rückwärts gelesen immer noch dasselbe Wort ist – beispielsweise das Wort „otto“.

Beispiel:

```
char Kette[4 + 1] = "otto";
if ( Palindrom ( Kette ) == true )
    cout << Kette << " ist ein Palindrom";
```

- Schreiben Sie eine Funktion `void Umdrehen (char K [])`, die eine übergebene Zeichenkette umdreht.

Beispiel:

```
char Kette[5 + 1] = "Hallo";
Umdrehen ( Kette );
cout << Kette; // gibt "ollaH" auf dem Bildschirm aus
```

- Schreiben Sie eine Funktion `long Char2Int (char K [])`, die eine übergebene Zeichenkette in einen Long-Integer-Wert umwandelt und diesen zurückgibt. Die Funktion soll Nichtziffern ignorieren.

Beispiele:

```
long x;
x = Char2Int ( "1234" );           // x hat den Wert 1234
x = Char2Int ( "01234" );          // x hat den Wert 1234
x = Char2Int ( "xyz123" );          // x hat den Wert 123
x = Char2Int ( "123xyz" );          // x hat den Wert 123
x = Char2Int ( "1:v;2##3+4" );     // x hat den Wert 1234
```