

Pflicht-Offline-Aufgabe O 04-12 (INF & WI & MCD): Buchstaben im String umwandeln

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches einen einzeiligen Text vom Benutzer einliest. Leerzeichen, Satzzeichen, Zahlziffern und andere Zeichen seien im Eingabetext erlaubt.

Das Programm soll dann den Text gemäß der folgenden Regeln umwandeln:

- Aus jedem Kleinbuchstaben sollen zwei aufeinanderfolgende Exemplare dieses Buchstabens werden.
- Aus jedem Großbuchstaben sollen drei Exemplare dieses Buchstabens werden.
- Zahlziffern (0 bis 9) sollen in einen Punkt umgewandelt werden.
- Ausrufezeichen und Fragezeichen sollen wegfallen.
- Leerzeichen sollen in einen Unterstrich umgewandelt werden.
- Alle anderen Zeichen sollen unverändert in das Ergebnis übernommen werden.

Der Benutzer mache nur korrekte Eingaben.

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E?!123
Der Text nach der Umwandlung: AAA_bb_CCC_dd_EEE...
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? a
Der Text nach der Umwandlung: aa
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? !

Der Text nach der Umwandlung:

Drücken Sie eine beliebige Taste . . .

Kommentar: Am Ende der zweiten Zeile befindet sich ein Leerzeichen, welches man hier im Testlauf nicht sehen kann.

Bitte Text eingeben (ggfs. mit Leerzeichen): ? 1!2

Der Text nach der Umwandlung: ..

Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? && \$\$

Der Text nach der Umwandlung: &&_\$\$

Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ?

Der Text nach der Umwandlung:

Drücken Sie eine beliebige Taste . . .

Kommentar: Leere Eingabe für den Text. Am Ende der zweiten Zeile befindet sich ein Leerzeichen, welches man hier im Testlauf nicht sehen kann.

Freiwillige Offline-Aufgabe O 04-01 (INF & WI & MCD): Aufzählung rückwärts

(Schleife)

Schreiben Sie ein C++ Programm, welches die Zahlen von 50 bis 0 rückwärts in 5-er Schritten ausgibt.

Testlauf: *(keine Benutzereingaben)*

```
50
45
40
35
30
25
20
15
10
5
0
Drücken Sie eine beliebige Taste . . .
```

Freiwillige Offline-Aufgabe O 04-02 (INF & WI & MCD): Mehrere Aufzählungen

(Schleife)

Schreiben Sie ein C++ Programm, welches zeilenweise jeweils einen Zeilenzähler, einen mit jeder Zeile sich verdoppelnden Wert und einen mit jeder Zeile sich verdreifachenden Wert ausgibt. Der „Verdoppler“ und „Verdreifacher“ Wert starten jeweils mit dem Startwert 10.

Jeder neue „Verdoppler“ bzw. „Verdreifacher“ Wert werde durch Verdoppeln bzw. Verdreifachen des vorherigen „Verdoppler“ bzw. „Verdreifacher“ Werts berechnet. Die „Verdoppler“ bzw. „Verdreifacher“ Werte werden nicht aus dem Zeilenzähler berechnet!

Realisieren Sie dies über eine einzige `for`-Schleife, in dem Sie folgendes Programmgerüst ergänzen. Es darf nur die Zeile mit dem Kopf der `for`-Schleife angepasst werden.

Die Variablen `z`, `wert2` und `wert3` müssen somit im Initialisierungsteil der `for`-Schleife definiert werden.

```
#include <iostream>
using namespace std;

int main()
{
    for ( /* nur hier ändern */ )
    {
        cout << "Zeile: " << z
              << " Verdoppler: " << wert2
              << " Verdreifacher: " << wert3
              << endl;
    }

    system("PAUSE");
    return 0;
}
```

Testlauf: *(keine Benutzereingaben)*

```
Zeile: 1 Verdoppler: 10 Verdreifacher: 10
Zeile: 2 Verdoppler: 20 Verdreifacher: 30
Zeile: 3 Verdoppler: 40 Verdreifacher: 90
Zeile: 4 Verdoppler: 80 Verdreifacher: 270
Zeile: 5 Verdoppler: 160 Verdreifacher: 810
Zeile: 6 Verdoppler: 320 Verdreifacher: 2430
Zeile: 7 Verdoppler: 640 Verdreifacher: 7290
Zeile: 8 Verdoppler: 1280 Verdreifacher: 21870
Zeile: 9 Verdoppler: 2560 Verdreifacher: 65610
Zeile: 10 Verdoppler: 5120 Verdreifacher: 196830
Drücken Sie eine beliebige Taste . . .
```

Freiwillige Offline-Aufgabe O 04-03 (INF & WI & MCD): Rechteck zeichnen

(Schleifen)

Schreiben Sie ein C++ Programm, welches die Breite und Höhe eines Rechtecks als ganze Zahlen einliest und dann das entsprechende Rechteck mittels Sternchen ausgibt. Das Rechteck ist mit Pluszeichen gefüllt.

Hinweise:

Breite und Höhe des Rechtecks müssen mindestens 2 betragen. Ihr Programm kann davon ausgehen, dass die Benutzereingabe diesbezüglich korrekt ist. Innerhalb des Rechtecks gibt es keine Leerzeilen, auch wenn dies im Testlauf ggfs. so aussieht. Die Ausgaben Ihres Programms müssen exakt mit den Testläufen übereinstimmen.

Testläufe (Benutzereingaben unterstrichen):

```
Bitte geben Sie die Breite des Rechtecks ein: ? 10
Bitte geben Sie die Hoehe des Rechtecks ein: ? 5
*****
*++++++*
*++++++*
*++++++*
*++++++*
*****
Drücken Sie eine beliebige Taste . . .
```

```
Bitte geben Sie die Breite des Rechtecks ein: ? 2
Bitte geben Sie die Hoehe des Rechtecks ein: ? 2
**
**
```

Drücken Sie eine beliebige Taste . . .

```
Bitte geben Sie die Breite des Rechtecks ein: ? 4
Bitte geben Sie die Hoehe des Rechtecks ein: ? 3
****
*+++
****
```

Drücken Sie eine beliebige Taste . . .

Pflicht-Offline-Aufgabe O 04-04 (INF & WI & MCD): Parallelogramm zeichnen

(Schleifen)

Ein *Parallelogramm* ist ein Viereck, bei dem die beiden jeweils gegenüberliegenden Seiten parallel sind.

Schreiben Sie ein C++ Programm, welches die Breite und Höhe eines Parallelogramms als ganze Zahlen einliest und dann das entsprechende Parallelogramm mittels Sternchen ausgibt. Aufeinanderfolgende Zeilen des Parallelogramms sind jeweils „um ein Sternchen nach rechts versetzt“, was man auch an den Punkten sieht, die links vom Parallelogramm ausgegeben werden sollen. Das Parallelogramm ist mit Pluszeichen gefüllt.

Hinweise:

Breite und Höhe des Parallelogramms müssen mindestens 2 betragen. Ihr Programm kann davon ausgehen, dass die Benutzereingabe diesbezüglich korrekt ist. Schauen Sie sich den Testlauf gut an, auch betreffend **der (einen) Leerzeile unterhalb des Parallelogramms**. Innerhalb des Parallelogramms gibt es keine Leerzeilen, auch wenn dies im Testlauf ggfs. so aussieht. Die Ausgaben Ihres Programms müssen exakt mit den Testläufen übereinstimmen.

Testläufe (Benutzereingaben unterstrichen):

```
Bitte geben Sie die Breite des Parallelogramms ein: ? 10
Bitte geben Sie die Hoehe des Parallelogramms ein: ? 5
*****
.*+++++++*
..*+++++++*
...*+++++++*
....*****
```

Drücken Sie eine beliebige Taste . . .

```
Bitte geben Sie die Breite des Parallelogramms ein: ? 2
Bitte geben Sie die Hoehe des Parallelogramms ein: ? 2
**
.**
```

Drücken Sie eine beliebige Taste . . .

```
Bitte geben Sie die Breite des Parallelogramms ein: ? 4
Bitte geben Sie die Hoehe des Parallelogramms ein: ? 3
****
.*++*
..****
```

Drücken Sie eine beliebige Taste . . .

Freiwillige Offline-Aufgabe O 04-05 (INF & WI & MCD): Buchstaben eines Strings

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches zeilenweise die Buchstaben eines vom Benutzer eingegebenen einzeiligen Texts ausgibt, zusammen mit ihrer Position im String (nach maschineller Zählung, d.h. mit Zählung ab 0).

Es seien keine Leerzeichen in der Eingabe erlaubt. Die leere Eingabe (Benutzer drückt sofort ENTER / RETURN) soll aber erlaubt sein (siehe zweiter Testlauf).

Der Benutzer mache nur korrekte Eingaben.

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

```
Bitte einzeiligen Text eingeben (ohne Leerzeichen): ? AbCd123
Position: 0 Buchstabe: A
Position: 1 Buchstabe: b
Position: 2 Buchstabe: C
Position: 3 Buchstabe: d
Position: 4 Buchstabe: 1
Position: 5 Buchstabe: 2
Position: 6 Buchstabe: 3
Drücken Sie eine beliebige Taste . . .
```

```
Bitte einzeiligen Text eingeben (ohne Leerzeichen): ?
Drücken Sie eine beliebige Taste . . .
```

Freiwillige Offline-Aufgabe O 04-06 (INF & WI & MCD): String umgekehrt ausgeben

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches einen einzeiligen Text vom Benutzer einliest und diesen dann umgekehrt (von hinten nach vorne) ausgibt.

Es seien auch Leerzeichen in der Eingabe erlaubt.

Die leere Eingabe (Benutzer drückt sofort ENTER / RETURN) braucht von ihrem Programm nicht berücksichtigt zu werden, d.h. der Benutzer gebe keine leere Eingabe ein, so dass ihr Programm diesen Fall auch nicht bearbeiten muss.

Der Benutzer mache nur korrekte Eingaben.

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Eingabetext: A b C d E
Ergebnis: E d C b A
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? X
Eingabetext: X
Ergebnis: X
Drücken Sie eine beliebige Taste . . .
```

Pflicht-Offline-Aufgabe O 04-07 (INF & WI & MCD): Palindrom-Erkennung mit *flexibler* Wortlänge

(Schleife, C++ Strings)

Ein *Palindrom* ist eine Zeichenkette, die von vorn und von hinten gelesen gleich bleibt.

Beispiele: otto, maoam, hannah, lagerregal, reittier, reliefpfeiler, rentner, rotor, stets.

Schreiben Sie ein C++ Programm, welches eine einzeilige Zeichenkette einliest und prüft, ob es sich bei der Zeichenkette um ein Palindrom handelt. Das Programm soll dann eine entsprechende Meldung ausgeben (siehe Testläufe).

Leerzeichen, Satzzeichen, Zahlziffern etc. dürfen nicht Teil der Eingabe sein, d.h. bei einer Eingabe, die entsprechende Zeichen enthält, soll der Benutzer umgehend zu einer neuen Eingabe aufgefordert werden.

Das eingegebene Wort soll nur Kleinbuchstaben enthalten. Ihr Programm kann sich darauf verlassen, dass der Benutzer keine Großbuchstaben eingeben wird (aber ggfs. fehlerhafterweise Satzzeichen etc., siehe oben).

Testläufe: (Benutzereingaben sind zur Verdeutlichung unterstrichen)

Text: ? reit tier!

Text: ? reittier

Das eingegebene Wort ist ein Palindrom.

Drücken Sie eine beliebige Taste . . .

Text: ? aachen

Das eingegebene Wort ist KEIN Palindrom.

Drücken Sie eine beliebige Taste . . .

Freiwillige Offline-Aufgabe O 04-08 (INF & WI & MCD): String umgekehrt ausgeben, leere Eingabe erlaubt

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches einen einzeiligen Text vom Benutzer einliest und diesen dann umgekehrt (von hinten nach vorne) ausgibt.

Es seien auch Leerzeichen in der Eingabe erlaubt.

Im Gegensatz zur Offline-Aufgabe O-04-06 soll nun auch die leere Eingabe (Benutzer drückt sofort ENTER / RETURN) erlaubt sein (siehe zweiter Testlauf).

Hinweis: Dieser Fall kann trickreich werden, falls Sie die Position im Strings in einer Variable vom Typ `unsigned int` speichern wollen. Die leere Eingabe hat die Länge Null, und falls Sie von diesem Wert etwas subtrahieren wollen, kommt keine negative Zahl heraus, sondern eine sehr große positive Zahl.

Abhilfe: Mit `int` programmieren (dann tritt die Problematik gar nicht auf und die Lösung sollte einfach sein) oder beim Programmieren dieser Aufgabe gut aufpassen und die gültigen Positionen im String exakt prüfen (versuchen Sie mal, diesen „ambitionierteren“ Weg zu gehen ...)!

Der Benutzer mache nur korrekte Eingaben.

Testen Sie alle Testlauf-Fälle Ihres Programms erst auf dem eigenen Rechner, bevor Sie die Lösung in den Jenkins hochladen. Ist ihr Programm so fehlerhaft, dass es crashed, werden Sie auch vom Jenkins keine aussagekräftigen Fehlermeldungen bekommen!

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E

Eingabetext: A b C d E

Ergebnis: E d C b A

Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ?

Eingabetext:

Ergebnis:

Drücken Sie eine beliebige Taste . . .

Kommentar: Leere Eingabe. Am Ende der ersten drei Zeilen befindet sich jeweils ein Leerzeichen, welches man aber nicht sehen kann.

Bitte Text eingeben (ggfs. mit Leerzeichen): ? X

Eingabetext: X

Ergebnis: X

Drücken Sie eine beliebige Taste . . .

Freiwillige Offline-Aufgabe O 04-09 (INF & WI & MCD): Buchstaben in String finden

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches einen einzeiligen Text vom Benutzer einliest sowie einen Buchstaben.

Das Programm soll dann ausgeben, ob der Buchstabe in dem einzeiligen Text vorkommt oder nicht. Falls ja, so soll die **letzte** Position ausgegeben werden, an der dieser Buchstabe vorkommt.

Die Positionszählung beginne ab Null, d.h. maschinelle Positionszählung.

Es seien auch Leerzeichen im Eingabetext erlaubt. Ebenso sei das Leerzeichen als Suchzeichen erlaubt. Für den Eingabetext sei auch die leere Eingabe erlaubt.

Der Benutzer mache nur korrekte Eingaben.

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte Buchstaben eingeben: ? C
Der Buchstabe C kommt im Text vor, an Position 4.
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte Buchstaben eingeben: ? X
Der Buchstabe X kommt nicht im Text vor.
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ?
Bitte Buchstaben eingeben: ? x
Der Buchstabe x kommt nicht im Text vor.
Drücken Sie eine beliebige Taste . . .
```

Kommentar: Leere Eingabe für den Text. Am Ende der ersten Zeile befindet sich jeweils ein Leerzeichen, welches man aber hier im Testlauf nicht sehen kann.

Bitte Text eingeben (ggfs. mit Leerzeichen): ? b b b
Bitte Buchstaben eingeben: ? b
Der Buchstabe b kommt im Text vor, an Position 4.
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? a
Bitte Buchstaben eingeben: ? a
Der Buchstabe a kommt im Text vor, an Position 0.
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? h
Bitte Buchstaben eingeben: ? H
Der Buchstabe H kommt nicht im Text vor.
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte Buchstaben eingeben: ?
Der Buchstabe kommt im Text vor, an Position 7.
Drücken Sie eine beliebige Taste . . .

**Kommentar: Der Benutzer hat das Leerzeichen als Suchzeichen
eingegeben. Am Ende der zweiten Zeile befinden sich somit zwei
Leerzeichen (eines, welches sowieso immer hinter dem Fragezeichen ist,
sowie das Leerzeichen vom Suchzeichen), welche man aber hier im
Testlauf nicht sehen kann.**

Freiwillige Offline-Aufgabe O 04-10 (INF & WI & MCD): Buchstaben in String zählen

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches einen einzeiligen Text vom Benutzer einliest sowie einen Buchstaben als Suchzeichen.

Das Programm soll dann ausgeben, wie oft der Buchstabe in dem einzeiligen Text vorkommt.

Es seien auch Leerzeichen im Eingabetext erlaubt. Für den Eingabetext sei auch die leere Eingabe erlaubt. Das Suchzeichen sein ein Buchstabe, d.h. insbesondere kein Sonderzeichen und auch kein Leerzeichen.

Der Benutzer mache nur korrekte Eingaben.

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte Buchstaben eingeben: ? C
Der Buchstabe C kommt 1 mal im Text vor.
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte Buchstaben eingeben: ? X
Der Buchstabe X kommt 0 mal im Text vor.
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ?
Bitte Buchstaben eingeben: ? x
Der Buchstabe x kommt 0 mal im Text vor.
Drücken Sie eine beliebige Taste . . .
```

Kommentar: *Leere Eingabe für den Text. Am Ende der ersten Zeile befindet sich ein Leerzeichen, welches man hier im Testlauf nicht sehen kann.*

Bitte Text eingeben (ggfs. mit Leerzeichen): ? b b b
Bitte Buchstaben eingeben: ? b
Der Buchstabe b kommt 3 mal im Text vor.
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? a
Bitte Buchstaben eingeben: ? a
Der Buchstabe a kommt 1 mal im Text vor.
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? h
Bitte Buchstaben eingeben: ? H
Der Buchstabe H kommt 0 mal im Text vor.
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte Buchstaben eingeben: ?
Der Buchstabe kommt 4 mal im Text vor.
Drücken Sie eine beliebige Taste . . .

Kommentar: Der Benutzer hat das Leerzeichen als Suchzeichen eingegeben. Am Ende der zweiten Zeile befinden sich somit zwei Leerzeichen (eines, welches sowieso immer hinter dem Fragezeichen ist, sowie das Leerzeichen vom Suchzeichen), welche man aber hier im Testlauf nicht sehen kann.

Freiwillige Offline-Aufgabe O 04-11 (INF & WI & MCD): Buchstaben in String ersetzen

(Schleife, Strings)

Schreiben Sie ein C++ Programm, welches einen einzeiligen Text vom Benutzer einliest sowie nacheinander zwei Buchstaben.

Das Programm soll dann den Text erneut ausgeben, wobei alle Vorkommen des ersten Buchstabens durch den zweiten Buchstaben ersetzt sind.

Kommt der zu ersetzende Buchstabe gar nicht in dem einzeiligen Text vor, so bleibt der Text unverändert.

Es seien auch Leerzeichen im Eingabetext erlaubt. Für den Eingabetext sei auch die leere Eingabe erlaubt. Das zu ersetzende Zeichen und das Ersatzzeichen seien jeweils ein Buchstabe, d.h. insbesondere kein Sonderzeichen, kein Leerzeichen und auch nicht mehrere Buchstaben oder Zeichen.

Der Benutzer mache nur korrekte Eingaben.

Testläufe: (Benutzereingaben zur Verdeutlichung unterstrichen)

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte den zu ersetzenden Buchstaben eingeben: ? C
Bitte den Ersatz-Buchstaben eingeben: ? X
Der Text nach der Ersetzung: A b X d E
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? a b c d e
Bitte den zu ersetzenden Buchstaben eingeben: ? C
Bitte den Ersatz-Buchstaben eingeben: ? X
Der Text nach der Ersetzung: a b c d e
Drücken Sie eine beliebige Taste . . .
```

```
Bitte Text eingeben (ggfs. mit Leerzeichen): ? A b C d E
Bitte den zu ersetzenden Buchstaben eingeben: ? X
Bitte den Ersatz-Buchstaben eingeben: ? Y
Der Text nach der Ersetzung: A b C d E
Drücken Sie eine beliebige Taste . . .
```

Bitte Text eingeben (ggfs. mit Leerzeichen): ? h
Bitte den zu ersetzenden Buchstaben eingeben: ? H
Bitte den Ersatz-Buchstaben eingeben: ? X
Der Text nach der Ersetzung: h
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? a
Bitte den zu ersetzenden Buchstaben eingeben: ? a
Bitte den Ersatz-Buchstaben eingeben: ? b
Der Text nach der Ersetzung: b
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ? b b b
Bitte den zu ersetzenden Buchstaben eingeben: ? b
Bitte den Ersatz-Buchstaben eingeben: ? X
Der Text nach der Ersetzung: X X X
Drücken Sie eine beliebige Taste . . .

Bitte Text eingeben (ggfs. mit Leerzeichen): ?
Bitte den zu ersetzenden Buchstaben eingeben: ? a
Bitte den Ersatz-Buchstaben eingeben: ? b
Der Text nach der Ersetzung:
Drücken Sie eine beliebige Taste . . .

Kommentar: Leere Eingabe für den Text. Am Ende der ersten und vierten Zeile befindet sich jeweils ein Leerzeichen, welches man hier im Testlauf nicht sehen kann.
