

Handbuch zur C++ - Lektion "String v1.030", zu Deutsch "Zeichenfolgen"

Copyright by Hilfe & Service von EDV-Fachleuten

Dipl.-Phys Harald Schärfer

Reilstr. 22 (Eingang Mozartstr.)

D-06114 Halle (Saale)

Edit v1.020 from 2022-09-15 to 2024-04-09 by Harald Schärfer

Tel. (0345) 521 13 40

Funk (0177) 521 13 40

EM Vertrieb@HuSvEDVF.de

Fax (0345) 521 13 41

Nutzen: Erlernen, wie Zeichenfolgen mit Operationen verarbeitet und dargestellt werden können.

Inhaltsverzeichnis

<u>1.</u>	<u>Bestandteile</u>	2
<u>2.</u>	<u>Systemvoraussetzungen</u>	2
<u>3.</u>	<u>Installation</u>	2
<u>4.</u>	<u>Benutzung</u>	3
<u>4.1</u>	<u>Untersuchung</u>	3
<u>4.2</u>	<u>Ausgabe</u>	4

1. Bestandteile

Das Programm wurde mit der Entwicklungsumgebung "Code:Block" entwickelt besteht aus folgenden x Komponenten.

String

- *.EXE Das ausführbare Programm "String".
- *.PDF Dieses Handbuch als "Portable Document Format" (übertragbares Dokumentenformat).
- *.TXT Dieses Handbuch als Text.
- *.ZIP Alle Dateien in eine ZIP - Datei gepackt.

Work

- Change.LOG Chronologische Liste der Aenderungen des Programms.
- main.cpp Das Hauptprogramm in C++;
- main.hpp Die Kopfdatei zum Hauptprogramm in C++.
- Error.TXT Fehlermeldungen als Liste.
- Lizenz.TXT Lizenzbestimmungen mit Gewährleistungsbeschränkung, Nutzungsrecht und Vertrieb zum Programm.

2. Systemvoraussetzungen

Das Programm wurde erfolgreich Ausführung und Nutzung getestet unter

- Code::Block v20.03;
- Windows 7 Prof. 64Bit;
- Windows 10 Prof. 64Bit

3. Installation

Wie Code::Blocks oder eine andere Entwicklungsumgebung für C/C++ installiert wird, ist hier nicht das Thema. Kopieren Sie die Dateien in ein Projektverzeichnis mit den Namen "String" und benutzen Sie den Quelltext, um das was Sie darin gesucht haben, zu verstehen.

4. Benutzung

Hier soll gezeigt werden wie im Quelltext von "main.cpp" und "main.hpp" mit Zeichenketten und Strings umgegangen werden kann.

4.1 Untersuchung

Untersucht wurden folgende Methoden, Iteratoren von String:

- String Funktionen fuer die Analyse und Bearbeitung
 - strEins.append();
 - strEins.size();
 - strEins.length();
 - strEins.capacity();
 - strEins.front();
 - strEins.at(3);
 - strEins.back();
 - strEins.find(cZ);
 - strEins.find(cZ, 1+iP1);
 - strEins.compare(strZwei);
 - strEins="1. Teil" + "2. Teil ";
 - strEins.append(" alla Append!");
 - strEins.append(strZwei);
 - strZwei=strEins.substr(2, 6);
 - iL=strZwei.compare(4, 6, strEins, 6, 6);
 - strZwei=strEins.substr(2, 6);
 - strEins.replace(3, 4, ",.");
- Zahl runden und als String ausgeben!
 - << defaultfloat ...;
 - << setw(10) ...;
 - << std::right ...;
 - << setfill('.') ...;
 - << fixed ...;
 - << setprecision(5) ...;
 - << fZahl ...;
 - << dZahl ...;
 - =to_string(fZahl);
- Von Zahlen aus String und dessen Konvertierung!
 - =stoi(strEins);
 - =stof(strEins);

4.2 Ausgabe

Siehe Anlage [01.TXT!](#)

1. = Information durch zuweisen, anfüegen und Groeszenbestimmung

```

-----
*****
* String v1.035 *
* String-Funktionen ausprobieren und das Ergebnis anzeigen! *
* ===== *
* Copyright by Hilfe & Sercvice von EDV-Fachleuten Web www.HuSvEDVF.de *
* Dipl.-Phys. Harald Schaerfer Tel. (0345) 521 13 40 *
* Reilstr. 22 Funk (0177) 521 13 40 *
* EM Info@HuSvEDVF.de *
* D-06114 Halle(Saale) Fax (0345) 521 13 41 *
* Edit v1.035 from 2020-10-28 to 2024-04-09 by HSc *
*****

```

2. = Methoden des Strings

```

=====
Gegeben: 0123456789abcdefghijklmnopqrstuvwxyz0...abc
Gesucht:
- Laenge: 43
- Groesze: 43
- Platzbedarf: 43
- Maximale Groesze: 2147483647
- Front-Zeichen: 0
- 0, ..., 3. Zeichen: 3
- Letztes Zeichen: c

```

3. = Suchen und Finden

```

-----
Gegeben: 0123456789abcdefghijklmnopqrstuvwxyz0...abc
Gesucht:
- Position, von Links, des Zeichens >a<: 10
- Das naechste Zeichen >a< ist an Position 10
Das >A<: -1 oder -1 fuer nicht gefunden!
- Position, von Rechts, des >a<: 40

```

4. = Vergleich zweier Strings

```

-----
Vergleich auf gleiche Strings mit Beachtung der Grosz- und Kleinschreibweise!
1. Wort: Ott+Ã2,
2. Wort: Ott+û2,
Ergebnis: 1
Strings sind NICHT identisch!

```

5. = Drei Teile zu einem String zusammen kitten

```

-----
Vorher: Ott+Ã2
Nachher: 1. Teil + 2. Teil

```

6. = Anfüegen von einer Zeichenkette an einen String.

```

-----
Vorher: 1. Teil + 2. Teil
Zeichenkette: alla Append!
Nachher: 1. Teil + 2. Teil alla Append!
Nachher: 1. Teil + 2. Teil alla Append! Oder mit String.

```

7. = Teilstring ab dem 2., von 0 an, werden 6. Zeichen ausgeschnitten.

```

-----
Gegeben:
- strEins: 0123456789abcdefghijklmnopqrstuvwxyz0...abc
Ergebnis:
- strZwei: 234567

```

8. = Ist in einem Teil von EINS ein Teil von ZWEI enthalten?

```

-----
Gegeben: (-1=Weniger; 0=Gleich; +1=Mehr als gesucht!
- Eins: Heute geht es nach Haus.
- Zwei: ... geht etwas entzwei.
Ergebnis:
- Ist >... geht etwas entzwei.< im 1. enthalten: -1
- Ist >geht e< im 1. enthalten: 1
- Ist >geht e< in >geht e< enthalten: 0

```

9. = Ersetzen von Zeichen

```

-----
Ab der 3., von 0 an, bis zur 6 Position werden die Zeichen durch >.,< ersetzt.
Gegeben:
- strEins: 0123456789abcdefghijklmnopqrstuvwxyz0...abc

```

Ergebnis:

- strEins: 012,.9abcdefghijklmnopqrstuvwxy0...abc

10. = Zahl runden und als String ausgeben.

=====

1. Zahl: 12.34567890 und 13 moegliche Stelle bei
 - FLOAT: 12.34567928314 mit einer Genauigkeit von circa 7 Dezimalziffern;
 - DOUBLE: 12.3456789 mit einer Genauigkeit von circa 13 Dezimalziffern.
 2. Zahl = 12.3456789 auf 3 Stellen reduzieren! 12.3
 3. Zahl = 12.3 auf 4 Stellen reduzieren! 12.35
 4. Zahl = 12.3 auf ? Stellen reduzieren? 12.3

11. = Tausender Trennzeichen, je nach Systemeinstellung, ausgeben.

Vorgabe: 12stellige Zahl

- als FLOAT: 0.370370358229 und

- als DOUBLE: 1.23456789

0,370.370

1,234.568

Mit tausender Trenner:

- als FLOAT: 0,370.370 und

- als DOUBLE: 1,234.568

12. = Eine Quadratzahl im Fenster von 10 Zeichen, 5 Nachkommastellen, rechtsbuendig, ausgeben.

Ergebnis:

...0.00000

..1.10000

...4.40000

..9.90000

..17.60000

..27.50000

..39.60000

..53.90000

..70.40000

..89.10000

.110.00000

.133.10001

13. = Stelligkeit der Mantisse variieren.

0 Stellen: 314 und mit Fixed: 314 oder mit Standard: 3e+002 ausgeben.
 1 Stellen: 3e+002 und mit Fixed: 314.2 oder mit Standard: 3e+002 ausgeben.
 2 Stellen: 3.1e+002 und mit Fixed: 314.16 oder mit Standard: 3.1e+002 ausgeben.
 3 Stellen: 314 und mit Fixed: 314.159 oder mit Standard: 314 ausgeben.
 4 Stellen: 314.2 und mit Fixed: 314.1593 oder mit Standard: 314.2 ausgeben.
 5 Stellen: 314.16 und mit Fixed: 314.15927 oder mit Standard: 314.16 ausgeben.
 6 Stellen: 314.159 und mit Fixed: 314.159271 oder mit Standard: 314.159 ausgeben.
 7 Stellen: 314.1593 und mit Fixed: 314.1592712 oder mit Standard: 314.1593 ausgeben.
 8 Stellen: 314.15927 und mit Fixed: 314.15927124 oder mit Standard: 314.15927 ausgeben.
 9 Stellen: 314.159271 und mit Fixed: 314.159271240 oder mit Standard: 314.159271 ausgeben.

14. = Zahl im String zu einer reinen Zahl uebergeben.

=====

Vorgabe:

Zahl im String Nr.

- Eins ist -1234.567890123456789 und

- Zwei ist +1234567890.123456789567890123456789

- Zahl aus String Eins +1 als Integer: -1233

- Zahl aus String Eins +1.0010 als Flieszkomma: -1233.56689

- Zahl aus String Eins +1.0010 als doppelte Genauigkeit: -1233.56689

- Zahl aus String Eins +1.0010 als lange Zahl mit doppelte Genauigkeit: -1233.56689

- Zahl aus String Zwei +1.0010 als lange lange Zahl: 1234567891

- Zahl aus String Zwei +1.0010 als nur positive lange Zahl: 1234567891

- Zahl aus String Zwei +1.0010 als nur positive lange lange Zahl: 1234567891

15. = Programm String v1.035 schlieszen und Beenden.

=====

Ende bestaetigen!

Viel Erfolg beim studieren!

******* EOF *******