

Formátovanie štandardného výstupu

Nasledujúci dokument vysvetľuje, ako používať formátovacie špecifikátory pre tlač formátovaného výstupu na obrazovku. Dokument zahŕňa základy funkcie `printf()`, formátovacie špecifikátory, rôzne typy formátovania a formátovanie reťazcov.

Funkcia `printf()`

Funkcia `printf()` nie je súčasťou syntaxe jazyka C, pretože v jazyku C nie je definovaný samotný vstup alebo výstup. Funkcia `printf()` je iba užitočnou funkciou štandardnej knižnice, ktorej funkcie sú prístupné pre programy jazyka C. Správanie funkcie `printf()` je definované v štandarde ANSI. Ak je používaný prekladač v súlade s týmto štandardom, potom by všetky vlastnosti danej funkcie mali byť prístupné pre programátora.

Formátovacie špecifikátory

V jazyku C existuje veľa rôznych špecifikátorov pre formátovanie. Medzi základné patrí:

- `%i` alebo `%d` *int*
- `%c` *char*
- `%f` *float*
- `%lf` *double*¹
- `%s` *string*

Pozrite si zdrojový kód a výstup funkcie `formattedOutputExample()` projektu `printf() Examples`². Prvý zápis funkcie `printf()` vytlačí desiatkovú číslicu. Druhý zápis funkcie `printf()` vytlačí tú istú číslicu, ale so šírkou 3 (`%3d`), čo znamená, že pre daný výstup sú rezervované tri cifry (pozície). Výsledkom zápisu sú potom dva prázdne znaky umiestnené pred samotným znakom 7. Tretí zápis funkcie `printf()` je podobný tomu predošlému, avšak namiesto prázdnych znakov sa vyplnia dané cifry znakom 0. Štvrtý zápis funkcie `printf()` vytlačí *float*, pričom rezervuje tri pozície pred (šírka) a dve pozície za desatinnou čiarkou (presnosť).

¹Použité vo význame *long float*

²Projekt `printf() Examples` si môžete stiahnuť na stránke cvičenia č. 4 predmetu Programovanie

Každý uvedený znak `\n` sa nazýva úniková sekvencia (angl. *Escape Sequence*), pričom v uvedenom príklade reprezentuje znak nového riadku. Keby sa v zápisoch daný znak nenachádzal, potom by každá ďalšia funkcia `print()` vytlačila výstup na ten istý riadok (hneď za doposiaľ vytlačené znaky).

Medzi najčastejšie únikové sekvencie patrí:

<code>\n</code>	nový riadok
<code>\t</code>	tabulátor
<code>\v</code>	vertikálny tabulátor
<code>\f</code>	nová strana
<code>\b</code>	krok späť o jednu pozíciu
<code>\r</code>	„návrat vozíka” (angl. <i>Carriage Return</i>), teda presun kurzora na začiatok daného riadku

Pozrite si zdrojový kód a výstup funkcie `fahrenheitExample()` projektu `printf() Examples`. Na štandardný výstup sú vytlačené teploty v °C o šírke 3 pozície a teploty v °F o šírke 6 pozícií, s presnosťou 3 pozície za desatinnou čiarkou (DČ). Súhrnom teda je:

<code>%d</code>	vytlačí desiatkové celé číslo
<code>%6d</code>	vytlačí desiatkové celé číslo so šírkou aspoň 6 pozícií
<code>%f</code>	vytlačí desatinné číslo
<code>%4f</code>	vytlačí desatinné číslo so šírkou aspoň 4 pozície (pred DČ)
<code>%.4f</code>	vytlačí desatinné číslo s presnosťou 4 pozície (za DČ)
<code>%3.2f</code>	vytlačí desatinné číslo so šírkou aspoň 3 pozície (pred DČ) s presnosťou 2 pozície (za DČ)

Rôzne typy formátovania

Doteraz sa v príkladoch vyskytovali iba desiatkové a desatinné čísla. Existujú však aj iné typy, pozrite si preto zdrojový kód a výstup funkcie `formattingOtherTypesExample()` projektu `printf() Examples`.

Všimnite si predposledný zápis funkcie `printf()`. Na štandardný výstup je vyladený iba jeden znak `%`, pričom číslo 10 sa vo výstupe nenachádza.

Naopak, v poslednom zápise funkcie `printf()` je uvedené aj číslo 10. Všimnite si preto rozdiel spomínaných zápisov³.

Formátovanie reťazcov

Oproti doteraz spomínaným možnostiam formátovania je formátovanie reťazcov odlišné. Pozrite si zdrojový kód a výstup funkcie `formattingStrings-Example()` projektu `printf() Examples`.

Všimnite si, že formátovanie reťazcov sa správa inak ako formátovanie čísiel (vysvetlenia sú uvedené podľa číslovania na výstupe):

1. Zápis tlačí reťazec (nič zvláštne sa nedeje).
2. Zápis tlačí reťazec na 15 znakov. Ak je reťazec dlhší, vytlačí sa celý. Ak je reťazec kratší, chýbajúce pozície sa vyplnia prázdnyimi znakmi (pred začiatkom reťazca).
3. Zápis tlačí iba prvých 10 znakov reťazca, teda ak je reťazec dlhší, zvyšné znaky sa nevytlačia. Ak je reťazec kratší, prázdne znaky sa nedopĺňajú.
4. Zápis tlačí aspoň 10 znakov reťazca. Teda ak je reťazec dlhší, vytlačí sa celý. Ak je reťazec kratší, chýbajúce pozície sa vyplnia prázdnyimi znakmi (na konci reťazca).
5. Zápis tlačí aspoň 15 znakov reťazca. Keďže uvádzaný reťazec je kratší, na jeho koniec sú pridané prázdne znaky.
6. Zápis tlačí iba prvých 15 znakov reťazca, teda ak je reťazec dlhší, zvyšné znaky sa nevytlačia. Keďže uvádzaný reťazec je kratší, vytlačí sa celý.
7. Zápis tlačí iba prvých 10 znakov reťazca, pričom na jeho začiatok pridá prázdne znaky dopočítaním do 15.
8. Zápis tlačí iba prvých 10 znakov reťazca, pričom na jeho koniec pridá prázdne znaky dopočítaním do 15⁴.

³Upravte hodnotu 10 uvedenú v poslednom zápise funkcie `printf()` na hodnotu 100. Všimnite si, že vytlačená hodnota je formátovaná použitím `%2d`. Aká hodnota bude vytlačená na štandardný výstup? Vašu odpoveď si overte preložením a spustením programu.

⁴Zdroj dokumentu:

<http://www.cplusplus.com/reference/cstdio/printf/>

<http://www.codingunit.com/printf-format-specifiers-format-conversions-and-formatted-output>