

Бинарни литерали

Свака команда и податак се мора превести у бинарни облик (битови 0 и 1) када је потребно да се обраде у процесору рачунара.

Подаци и вредности се могу представити у неколико бројних система: бинарни (основа 2), децимални (основа 10), октални (основа 8) и хексадецимални (основа 16).

Овде су приказана два низа вредносно истих бинарних и хексадецималних вредности:

```
int[] binarneVrednosti = { 0b00110001, 0b01100010, 0b11000100,
                           0b10001001 };
int[] heksaVrednosti = { 0x31, 0x62, 0xC4, 0x89 };
```

Овде су приказана два низа вредносно истих децималних и бинарних вредности:

```
int[] decimalneVrednosti = { 1, 2, 4, 8, 16 };
int[] binarneVrednosti = { 0b0001, 0b0010, 0b00100, 0b0001000,
                           0b00010000 };
```

Код хексадецималних вредности се користи као префикс 0x како би компајлер препознао бројеве у хексадецималном бројном систему.

Код бинарних вредности се користи као префикс 0b како би компајлер препознао бројеве у бинарном бројном систему.

Пошто се у бинарном облику пише велики број 1 и 0, може се користити сепаратор цифара:

```
int[] decimalneVrednosti = { 32, 64, 128 };
int[] binarneVrednosti = { 0b0010_0000, 0b0100_0000, 0b1000_0000 };
```

Постављање цифара бинарног броја у групе помаже читљивости бројева и управљању кодом.

Сепаратори цифара се могу користити и код других бројних система: код int, decimal, float, double.

```
const double Pi = 3.141_592_653_589_793_238_462_643_383_279_502;
```

Стринг литерали

Ескејп секвенце за стринг литерале

ескејп секвенце	реализација
\'	један апостроф
\"	наводници
\\	коса линија (backslash)
\0	null
\a	звучни сигнал

\b	поље уназад (backspace)
\f	form feed
\n	прелазак у нову линију
\r	carriage return
\t	хоризонтални таб
\v	вертикални таб

Дословно специфицирање стринга значи да су сви знаци наведени унутар пара наводника укључени у стринг, укључујући карактере краја линије.

То се изводи навођењем карактера @ пре стринга:

@ "Doslovni string literal"

@ "Kratka lista:

stvar 1

stvar 2"

Ако је потребно у коду навести потпуно име фајла, треба користити двоструке косе линије унутар стринга да би се правилно идентификовале косе линије као део имена:

"C:\\Temp\\MyDir\\MyFile.doc"

Употребом дословног литерала није неопходно користити двоструке косе линије:

@ "C:\\Temp\\MyDir\\MyFile.doc"

Изрази

У C# се користе неколико оператора за извођење операција над вредностима.

Комбиновањем промењивих и литералних вредности (операнди), креирају се изрази (expressions).

Оператори се деле на математичке, логичке и релацијске.

Математички оператори

оператор	категорија	пример	резултат
+	бинарни	a = 5 + 6	a = 11
-	бинарни	a = 5 - 6	a = -1
*	бинарни	a = 5 * 6	a = 30
/	бинарни	a = 5 / 6	a = 0.833333

%	бинарни	$a = 10 \% 3$	$a = 1$
+	унарни	$a = +(-5)$	$a = -5$
-	унарни	$a = -(-5)$	$a = 5$

Употреба + оператора зависи од типа података који се користе као операнди.

Ако су операнди подаци типа `char`, сабирање знакова такође даје као резултат цео број пошто се `char` подаци смештају као целобројни подаци у меморији (преко ASCII кода).

Оператори инкрементирања и декрементирања

оператор	категорија	пример	резултат
++	унарни	$a = ++b;$	$b = b + 1; a = b + 1;$
--	унарни	$a = --b;$	$b = b - 1; a = b - 1;$
++	унарни	$a = b++;$	$a = b; b = b + 1;$
--	унарни	$a = b--;$	$a = b; b = b - 1;$