

# Programiranje 2

## *Bitovski operatori*

### 1 ZADACI

**Zadatak 1** Napisati funkciju `print_bits` koja štampa bitove u zapisu datog celog broja  $x$ .

Primer 1:	Primer 2:
56 00000000000000000000000000000000111000	-9876 111111111111111101100101101100

**Zadatak 2** Napisati program koji proverava da li je na  $k$ -tom mestu jedinica u celom broju  $x$ . Celi brojevi  $k$  (neoznačen ceo broj) i  $x$  se unose sa standarnog ulaza. Pozicija se broji od najmanje težine ka najvećoj počevši od 0. Ukoliko pozicija nije u dobrom opsegu ispisati poruku o grešci.

Primer 1:	Primer 2:	Primer 3:	Primer 4:	Primer 5:
78 1 jeste	78 0 nije	78 3 jeste	78 -3 pozija nije u dobrom opsegu	78 32 pozija nije u dobrom opsegu

**Zadatak 3** Napisati program koji postavlja na  $k$ -to mestu jedinicu u celom broju  $x$ . Celi brojevi  $k$  (neoznačen ceo broj) i  $x$  se unose sa standarnog ulaza. Pozicija se broji od najmanje težine ka najvećoj počevši od 0. Ukoliko pozicija nije u dobrom opsegu ispisati poruku o grešci.

Primer 1:	Primer 2:	Primer 3:
78 4 94	78 3 78	78 32 pozija nije u dobrom opsegu

**Zadatak 4** Napisati funkciju `suma_bitova` koja izračunava sumu bitova datog broja. Testirati funkciju pozivom u `main`-u, broj  $x$  se unosi sa standarnog ulaza.

Primer 1:	Primer 2:	Primer 3:	Primer 4:
78 4	126789 11	1024 1	-3 31

**Zadatak 5** Napisati funkciju `unsigned get_bits(int x, int p, int n)` koja vraća broj dobijen od broja  $x$  tako da mu je izdvojeno  $n$  bita počev od pozicije  $p$ , a ostalo su 0. Testirati funkciju pozivom u `main`-u, pri čemu se brojevi  $x$ ,  $p$  i  $n$  unose sa standarnog ulaza. Proveriti da li su brojevi  $p$  i  $p+n$  u odgovarajućem opsegu i ispisati poruku o grešci ako nisu. Na standarni izlaz ispisati bitove unesenog broja  $x$  i bitove novodobijenog broja.

**Zadatak 6** Napisati funkciju `unsigned set_bits(unsigned x, int p, int n, unsigned y)` koja vraća modifikovano  $x$  tako što mu je izmenjeno  $n$  bitova počevši od pozicije  $p$  i na ta mesta je upisano poslednjih  $n$  bitova broja  $y$ . Testirati funkciju pozivom u `main-u`, pri čemu se brojevi  $x$ ,  $y$ ,  $p$  i  $n$  unose sa standarnog ulaza. Proveriti da li su brojevi  $p$  i  $p + n$  u odgovarajućem opsegu i ispisati poruku o grešci ako nisu. Na standarni izlaz ispisati bitove unesenog broja  $x$  i  $y$  i bitove novodobijenog broja.

**Zadatak 7** Napisati funkciju `unsigned invert_bits(unsigned x, int p, int n)` koja invertuje  $n$  bitova broja  $x$  počevši od pozicije  $p$ . Testirati funkciju pozivom u `main`-u, pri čemu se brojevi  $x$ ,  $p$  i  $n$  unose sa standarnog ulaza. Proveriti da li su brojevi  $p$  i  $p + n$  u odgovarajućem opsegu i ispisati poruku o grešci ako nisu. Na standarni izlaz ispisati bitove unesenog broja  $x$  i bitove novodobijenog broja.

<b>Primer 1:</b> Unesi brojeve 12568 Unesite n i p 10 3	<b>Primer 2:</b> Unesi brojeve -1348 Unesite n i p 8 15
0000000000000000000011000100011000 0000000000000000000001011101110000	1111111111111111111111111101010111100 1111111110000000111101010111100

**Zadatak 8** Napisati funkciju `unsigned right_rotate(unsigned x, int n)` koja vrši rotaciju neoznačenog broja  $x$  za  $n$  pozicija u desno. Testirati funkciju pozivom u `main`-u. Brojevi  $x$  i  $n$  se unose sa standarnog ulaza. Na standarni izlaz ispisati binarnu reprezentaciju broja  $x$  i novodobijnog broja. Ukoliko je broj  $n$  negativan ispisati poruku o grešci.

<b>Primer 1:</b>	<b>Primer 2:</b>	<b>Primer 3:</b>
Unesi broj 12589	Unesi broj 5689	Unesi broj 5689
Unesite n 10	Unesite n 33	Unesite n -3
0000000000000000000011000100101101 0100101101000000000000000000001100	00000000000000000000000000001011000111001 1000000000000000000000000000101100011100	greska u stepenu rotacije

**Zadatak 9** Napisati funkciju `unsigned mirror(unsigned x)` koja obrće zapis broja  $x$ . Testirati funkciju pozivom u `main`-u. Broj  $x$  se unosi sa standarnog ulaza. Na standarni izlaz ispisati binarnu reprezentaciju broja  $x$  i novodobijnoj broja.

<b>Primer 1:</b>	Unesi broj 698 00000000000000000000000001010111010 0101110101000000000000000000000000000
------------------	---

**Zadatak 10** Napisati funkciju koja ispituje da li je dati ceo broj n stepen broja 2. Koristiti bitovske operatore.

<b>Primer 1:</b> 1024 jeste	<b>Primer 2:</b> 15 nije
-----------------------------------	--------------------------------

**Zadatak 11** Napisati program koja broji koliko se puta kombinacija 11 (dve uzastopne jedinice) pojavljuje u binarnom zapisu celog broja x. Broj x se unosi sa standarnog ulaza, a rezultat se ispisuje na standarni izlaz.

<b>Primer 1:</b> 1548 2	<b>Primer 2:</b> -56 25
-------------------------------	-------------------------------

**Zadatak 12 a)** Napisati program koji određuje najveći pozitivan broj koji se može zapisati istim binarnim ciframa kao dati broj. Koristiti bitovske operatore. Sa standarnog ulaza se unosi broj, a na standarni izlaz se ispisuje rezultat.

<b>Primer 1:</b> 125489 2143289344	<b>Primer 2:</b> -13 2147483646
--	---------------------------------------

**b)** Napisati funkciju koja određuje najmanji pozitivan broj koji se može zapisati istim binarnim ciframa kao dati broj. Koristiti bitovske operatore. Sa standarnog ulaza se unosi broj, a na standarni izlaz se ispisuje rezultat.

<b>Primer 1:</b> 125489 511	<b>Primer 2:</b> -13 1073741823
-----------------------------------	---------------------------------------

**Zadatak 13** Napisati program koji ispisuje zapis datog broja u pokretnom zarezu (napomena: može se koristit unija koja sadrži float i long).

<b>Primer 1:</b> 13.4 010000010101011001100110011001100110	<b>Primer 2:</b> -5692.36 11000101101100011110001011100001
--	--

**Zadatak 14** Napisati funkciju koja na osnovu neoznačenog broja x formira nisku s koji sadrži heksadekadni zapis broja x, koristeći algoritam za brzo prevodenje binarnog u heksadekadni zapis (svake 4 binarne cifre se zamenjuju jednom odgovarajućom heksadekadnom cifrom).

**Primer 1:**

56987

0000DE9B

**Primer 2:**

259774

0003F6BE

**Zadatak 15** Sa standardnog ulaza se unosi ceo broj  $n$  ( $n \leq 32$ ), a zatim i niz od  $n$  neoznačenih celih brojeva. Formirati neoznačeni ceo broj  $x$  tako što se na  $i$ -ti bit broja  $x$  postavlja vrednost  $i$ -tog bita  $i$ -tog broja niza. Broj  $x$  ispisati na standardni izlaz. Bitove broja čitati od pozicija manje težine ka pozicijama veće težine. Podrazumevana vrednost bitova broja  $x$  je 0. U slučaju greške ispisati  $-1$  na standardni izlaz.

**Primer 1:**

7 12 45 72 415 146 333 85

88

**Primer 2:**

5 1024 64 128 31 511

24

**Primer 3:**

5 127 0 0 63 128

9

**Primer 4:**

41

-1

**Zadatak 16** Sa standarnog ulaza učitava se neoznačen ceo broj  $x$ , neoznačen ceo broj  $n$  i niz od  $n$  celih neoznačenih brojeva ( $n \leq 32$ ). Odrediti neoznačen ceo broj  $y$  koji se dobija na sledeći način: porede se  $i$ -ti bit broja  $x$  i  $i$ -ti bit  $i$ -tog broja niza. Ukoliko su jednaki na  $i$ -to mesto broja  $y$  se postavlja bit 1, a inače se postavlja 0 (ukoliko  $i$ -ti broj niza ne postoji, podrazumevati da je vrednost odgovarajućeg bita 0). Ispisati broj  $y$  na standardni izlaz. U slučaju greške ispisati  $-1$  na standardni izlaz.

**Primer 1:**

1023 7 0 0 1023 1023 0 0 0

4294966284

**Primer 2:**

348712 4 1235 964914 24214 4212

4294618576

**Primer 3:**

12345 0

4294954950

**Primer 4:**

726431 2 4967 349672

4294240865