



6. Bitovne operacije - 1

1. Izvršite bitovnu AND operaciju nad sledećim parovima heksadekadnih brojeva. Odgovor zapišite heksadekadno.

- a) 0FF00 AND 0FF _____ b) 0F00F AND 1234 _____
c) 4321 AND 1234 _____ d) 4765 AND 6543 _____

2. Izvršite bitovnu OR operaciju nad sledećim parovima heksadekadnih brojeva. Odgovor zapišite heksadekadno.

- a) FF00 OR 0FF _____ b) F00F OR 1234 _____
c) 4321 OR 1234 _____ d) 4765 OR 6543 _____

3. Izvršite bitovnu XOR operaciju nad sledećim parovima heksadekadnih brojeva. Odgovor zapišite heksadekadno.

- a) FF00 XOR 0FF _____ b) F00F XOR 1234 _____
c) 4321 XOR 1234 _____ d) 4765 XOR 6543 _____

4. Izvršite bitovnu SHL operaciju (logičko pomeranje ulevo za 1 bit) nad sledećim heksadekadnim brojevima. Odgovor takođe zapišite heksadekadno.

- a) FFFF _____ b) FF88 _____
c) FF00 _____ d) F00F _____

5. Izvršite bitovnu SHR operaciju (logičko pomeranje udesno za 1 bit) nad sledećim heksadekadnim brojevima. Odgovor takođe zapišite heksadekadno.

- a) FFFF _____ b) FF88 _____
c) FF00 _____ d) F00F _____
e) 4321 _____ f) 8080 _____

6. Izvršite bitovnu SAR operaciju (aritmetičko pomeranje udesno za 1 bit) nad sledećim heksadekadnim brojevima. Odgovor takođe zapišite heksadekadno.

- a) FFFF _____ b) FF88 _____
c) FF00 _____ d) F00F _____

7. Izvršite bitovnu ROL operaciju (cikličko pomeranje ulevo za 1 bit) nad sledećim heksadekadnim brojevima. Odgovor takođe zapišite heksadekadno.

a) FFFF _____ b) FF88 _____

c) FF00 _____ d) F00F _____

8. Izvršite bitovnu ROR operaciju (cikličko pomeranje udesno za 1 bit) nad sledećim heksadekadnim brojevima. Odgovor takođe zapišite heksadekadno.

a) FFFF _____ b) FF88 _____

c) FF00 _____ d) F00F _____

Primer: Pretpostavimo da slog **datum** (d, m, g) pamtimo u 16-bitnom registru na sledeći način:

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
D	D	D	D	D	M	M	M	M	G	G	G	G	G	G	G

Da bi se izdvojilo polje **dan** iz sloga **datum** potrebno je maskirati sve ostale bitove:

datum AND 1111 1000 0000 0000b, odnosno datum AND 0F800h.

Slično se korišćenjem logičke operacije AND mogu izdvojiti i polja za mesec i godinu.

Umetanje nove vrednosti za **mesec** (ili neko drugo) polje u okviru upakovanog sloga nešto je komplikovanije nego izdvajanje. Prvo se stara vrednost polja obriše sa AND:

datum AND 0F87Fh, zatim se vrednost za mesec (neoznačen ceo broj) pomeri na svoju poziciju operacijom SHL: SHL mesec,7 i konačno se upakuje u datum korišćenjem logičke operacije OR: datum OR mesec

Napomena: Assembler zahteva da se ispred heksadekadnih brojeva koji počinju slovom kao npr. F800 obavezno stoji cifra 0 kako bi ga razlikovao od imena promenljive.

9. Konvertujte sledeće datume u pakovani format kao prethodnom primeru i zapišite u heksadekadnom obliku vrednosti odgovarajućih 16-bitnih registara.

22/8/29 _____ 23/7/32 _____

25/1/77 _____ 9/10/82 _____

10. Ako sledeći heksadekadni brojevi predstavljaju sadržaje registara u koje su upisani pakovani datumi kao u prethodnim primerima, dekodirajte o kojim se datumima radi.

BABA _____ DACA _____

3254 _____ AC60 _____



7. Bitovne operacije - 2

1. Ako se datum pamti kao u primeru iz vežbe 6, objasnite kako se od pojedinačnih polja dan, mesec i godina može formirati pakovani 16-bitni podatak datum.
2. Pretpostavimo da je cifra celobrojni podatak iz intervala 0..9 koji se čuva u jednom bajtu. Objasnite kako se od njega može formirati ASCII karakter korišćenjem bitovnih operacija.
- 3*. Sledeća C++ funkcija vraća poziciju prvog setovanog (jednakog 1) bita u BitMap parametru počev od pozicije 0. Ako nijedan bit nije postavljen, funkcija vraća vrednost -1. Detaljno opisati kako funkcija radi i napisati odgovarajuću funkciju u TURBO PASCAL-u.

```
int FindFirstSet(un signed BitMap, un signed start)
{
    un signed Mask = (1 << start);
    while (Mask)
    {
        if (BitMap & Mask) re turn start;
        ++start;
        Mask <<= 1;
    }
    re turn -1;
}
```

- 4*. Sledeća C++ funkcija je inverzna prethodnoj, vraća poziciju prve binarne 0 u BitMap parametru. Detaljno opisati kako funkcija radi i napisati odgovarajuću funkciju u TURBO PASCAL-u.

```
int FindFirstClr(un signed BitMap, un signed start)
{
    re turn FindFirstSet(~BitMap, start);
}
```

5. Jezik C++ ne specifikuje koliko bitova ima u neoznačenom celom broju. Objasnite kako prethodne dve funkcije rade nezavisno od broja bitova u neoznačenom celom broju. Kakva ograničenja postoje u TURBO PASCAL-u?

- 6*. Sledeće dve funkcije postavljaju, odnosno brišu bit sa naznačene pozicije u parametru BitMap i vraćaju novu vrednost parametra. Detaljno opisati kako funkcije rade i napisati odgovarajuće funkcije u TURBO PASCAL-u.

```
un signed SetBit(un signed BitMap, un signed po si tion)
{
    re turn BitMap | (1 << po si tion);
}

un signed ClrBit(un signed BitMap, un signed po si tion)
{
    re turn BitMap & ~(1 << po si tion);
}
```

7. Objasnite šta se dešava u prethodnim funkcijama ako parametri start i position sadrže vrednosti veće ili jednake broju bitova u neoznačenom celom broju.

- 8*. Napisati funkciju CharToInt koja prenosi string i vraća odgovarajuću celobrojnu vrednost. Nemojte koristiti ugrađene funkcije poput atoi(C) ili strtoint (Pascal) za ovu konverziju. Treba

obraditi svaki karakter ulaznog stringa, konvertovati ga u numeričku vrednost korišćenjem logičkih operacija i prevesti tako dobijen niz cifara u jedinstven broj. (U Pascalu je dozvoljeno korišćenje funkcije ord).

9*. Napisati UDatum funkciju koja prihvata tri parametra: dan, mesec i godinu i vraća 16-bitni pakovani datum korišćenjem formata kao u primeru iz vežbe 7. Napisati i funkcije IzdvojDan, IzdvojMesec i IzdvojGodinu koje na osnovu parametra Datum izdvajaju polje dan, odnosno mesec ili godina. Funkcija UDatum treba automatski da konvertuje datume iz oblika 1900-1999 u 0..99.

10*. Napisati CnBits funkciju koja vraća broj postavljenih bitova u 16-bitnom celobrojnom podatku. Zabranjeno je korišćenje ugrađenih funkcija.

11*. Napisati TestBit funkciju koja prihvata dva 16-bitna parametra. Prvi parametar je 16-bitna vrednost za testiranje, a drugi je broj iz intervala 0..15 koji zadaje redni broj bita koji treba testirati. Funkcija treba da vrati vrednost true ako je bit na zadatoj poziciji jednak 1, a false u suprotnom. Ukoliko je drugi parametar izvan opsega 0..15, funkcija treba da vrati false.

12*. Turbo Pascal u C/C++ imaju ugrađene operatore za šiftovanja (SHL/SHR u Pascalu i "<<" / ">>" u C/C++), međutim nemaju operatore za rotaciju. Napisati funkcije ROL i ROR koje obezbeđuju rotaciju. Uputstvo: Funkcijom TestBit iz prethodnog zadatka testirajte najstariji bit, a zatim odgovarajućim operacijama šiftovanja i OR realizujte rotaciju.

13*. Napisati aplikaciju LOGCALC koja izračunava vrednosti različitih logičkih funkcija. Aplikacija treba da omogući unošenje 8-bitnih binarnih ili heksadekadnih vrednosti i izbor logičke operacije AND, OR, XOR, NAND, NOR, EQU ili NOT i prikaže odgovarajući rezultat (i binarno i heksadekadno). Ukoliko se izabere unarna operacija (NOT) program treba da onemogući unošenje drugog argumenta. Ukoliko se unose binarni argumenti, program automatski treba da generiše odgovarajuće heksadekadne vrednosti i obrnuto i da spreči zadavanje nekorektnih argumenata.

14*. Napisati aplikaciju SHIFTCALC koja izračunava vrednosti različitih pomeranja sadržaja 8-bitnog registra za 1. Aplikacija treba da omogući unošenje 8-bitne binarne ili heksadekadne vrednosti i izbor operacije pomeranja SHL, SHR, SAL, SAR, ROL, ROR, RCL i RCR i prikaže odgovarajući rezultat (i binarno i heksadekadno), kao i vrednost koja ostaje u bitu prenosa (CF). Program inicijalno treba da postavi vrednost CF na 0, a zatim da računa sa vrednosću CF koja je dobijena prethodnom operacijom. Ukoliko se unose binarni argumenti, program automatski treba da generiše odgovarajuće heksadekadne vrednosti i obrnuto i da spreči zadavanje nekorektnih argumenata.

15*. Napisati aplikaciju DATUM koja sa ulaza prihvata tri podatka: dan, mesec, godina (u dekadnom brojnom sistemu) i na osnovu njih formira i u prikazuje pakovani datum u dva oblika: ddddmmmmggggggg i mmmmddddggggggg. Pakovani datumi treba da se prikazuju i binarnom i uheksadekadnom obliku. Aplikacija takođe treba da omogućava i suprotan proces: unošenje pakovanog datuma u bilo kojem od polja - binarnom ili heksadekadnom jednog od dva oblika pakovanja i da na osnovu njega prikazuje sadržaje ostala tri pakovana oblika i njegove komponente dan, mesec i godina u dekadnom zapisu.