

## 9. Dodatne vježbe

1. S tipkovnice učitati cijeli broj `n` koji mora biti između 0 i 16 (uključivo s granicama). Ako broj nije ispravan, ispisati odgovarajuću poruku. Nakon toga učitati `n` binarnih znamenki i ispisati dekadski ekvivalent učitano binarnog broja (ne primjenjuje se tehnika dvojnog komplementa, pa je dekadski ekvivalent sigurno pozitivan broj).

Npr., ako je korisnik upisao

```
4
1
1
0
1
```

program treba ispisati 13.

Npr., ako je korisnik upisao

```
0
```

program treba ispisati 0.

Riješiti pomoću petlje s ispitivanjem uvjeta na početku.

2. Načinite program za izračunavanje  $n!$ . Vrijednost za `n` učitati s tipkovnice. Ako broj nije ispravan (mora biti cijeli broj veći ili jednaki 0), ispisati odgovarajuću poruku. Riješiti pomoću petlje s ispitivanjem uvjeta na početku.
3. Načinite program za pretvaranje nenegativnog cijelog dekadskog broja u heksadekadski oblik (uzastopnim dijeljenjem sa 16). Znamenke heksadekadskog broja smijete ispisati "obrnutim" redoslijedom. Npr. za uneseni dekadski broj 725, program treba ispisati 5D2. Za uneseni broj 0, program treba ispisati 0. Riješiti pomoću petlje s ispitivanjem uvjeta na početku.
4. Isto kao prethodni zadatak, ali za učitani dekadski broj treba ispisati oktalni broj.
5. U varijablu tipa `int` učitati cijeli broj. Na zaslon ispisati binarni sadržaj registra te varijable (binarne znamenke se moraju ispisati ispravnim redoslijedom). Za izdvajanje bitova koristiti bitovni operator `&` i operator za posmak bitova. Može se pretpostaviti da se za pohranu `int` tipa podatka koristi 32 bita.

Primjer: ako je korisnik upisao broj 18, na zaslon treba ispisati

```
000000000000000000000000000000010010
```

**Rješenja svih zadataka provjeriti prevođenjem i testiranjem vlastitih programa!**

## Rješenja: NE GLEDATI prije nego sami pokušate riješiti zadatke

### Rješenje 1. zadatka

```
#include <stdio.h>
int main () {
    int n, znamenka, dekadski = 0;
    scanf("%d", &n);
    if (n < 0 || n > 16) {
        printf("Upisali ste neispravan broj\n");
    }
    else {
        while (n > 0) {
            scanf("%d", &znamenka);
            dekadski = dekadski*2 + znamenka;
            n--;
        }
        printf("%d\n", dekadski);
    }
    return 0;
}
```

### Rješenje 2. zadatka

```
#include <stdio.h>
int main () {
    int n, fakt = 1;
    scanf("%d", &n);
    if (n < 0) {
        printf("Upisali ste neispravan broj\n");
    }
    else {
        while (n > 1) {
            fakt *= n;
            n--;
        }
        printf("%d\n", fakt);
    }
    return 0;
}
```

### Rješenje 3. zadatka

```
#include <stdio.h>
int main () {
    int n, ostatak;
    scanf("%d", &n);
    if (n == 0) {
        printf("0\n");
    }
    else {
        while (n > 0) {
            ostatak = n % 16;
            if (ostatak < 10)
                printf("%d", ostatak);
            else
                printf("%c", 'A' + ostatak - 10);
            n = n / 16;
        }
    }
    return 0;
}
```

## Rješenje 4. zadatka

```
#include <stdio.h>
int main () {
    int n, ostatak;
    scanf("%d", &n);
    if (n == 0)
        printf("0\n");
    else
        while (n > 0) {
            ostatak = n % 8;
            printf("%d", ostatak);
            n = n / 8;
        }
    return 0;
}
```

## Rješenje 5. zadatka

```
#include <stdio.h>
int main () {
    int broj;
    int kolikoPosmaknutiDesno, pomocna;
    scanf("%d", &broj);
    printf("Upisali ste broj %d\n", broj);
    kolikoPosmaknutiDesno = 31;
    while (kolikoPosmaknutiDesno >= 0) {
        pomocna = broj >> kolikoPosmaknutiDesno;
        /* sada se u varijabli pomocna, na poziciji nultog bita (najmanje
           znacajnog bita) nalazi bit koji se u varijabli broj nalazi na poziciji
           kolikoPosmaknutiDesno. Vrijednost tog bita, 0 ili 1, moze se dobiti tako da se
           obavi operacija: pomocna (and po bitovima) 00000000000000000000000000000001 */
        printf("%d", pomocna & 1);
        kolikoPosmaknutiDesno--;
    }
    printf("\n");
    return 0;
}
```