

++++ 001 +++++

Napisati program koji učitava šesteroznamenasti broj. Lijevu polovicu čini broj sastavljen od prve tri znamenke, a desnu broj od zadnje tri znamenke. (Npr. lijeva polovica broja 123456 je 123, a desna 456). Ako je razlika lijeve i desne polovice parna, program treba ispisati "paran", inače "neparan" (navodnike ne ispisivati). Npr. za broj 123456 program mora ispisati "neparan".

```
#include <stdio.h>

int main(void)
{
    unsigned int number;

    scanf("%u", &number);

    if ((number/1000)-(number%1000))%2 == 0)
        printf("paran\n");
    else
        printf("neparan\n");

    return 0;
}
```

++++ 002 +++++

Učitavaju se koordinate dviju točaka u koordinatnom sustavu - prvo koordinate x i y prve točke, a zatim koordinate x i y druge točke i to u formatu %d %d %d %d (između svakog %d i %d nalazi se razmak te nema "\n" na kraju). Koordinate su cijeli brojevi. Provjeriti jesu li točke centralno simetrične s obzirom na ishodište te, ako jesu, ispisati znak "H" (navodnike ne ispisivati) te onda provjeriti i jesu li osnosimetrične s obzirom na apscisu (u tom slučaju ispisati znak "T" (navodnike ne ispisivati)) i, konačno, provjeriti i jesu li osnosimetrične s obzirom na ordinatu (ispisati znak "G" (navodnike ne ispisivati)) (slova H, T odnosno G pri ispisu potrebno odvojiti praznim znakom (razmakom) ako je potrebno ispisati više od jednog slova, npr. H T). U slučaju da točke nisu centralno simetrične s obzirom na ishodište, treba samo ispisati koordinate (prvo koordinatu x, a zatim y međusobno odvojene praznim znakom (razmakom) u formatu %f) polovišta dužine koje tvore dvije točke.

Prilikom ispisa više znakova u programu, potrebno je svaki znak ispisati u zasebnom redu.

```
#include<stdio.h>

int main(void)
{
    int x1, y1, x2, y2;

    scanf("%d %d %d %d", &x1, &y1, &x2, &y2);

    if (x1 == -x2 && y1 == -y2)
    {
        printf("H ");
        if (x1 == x2 && y1 == -y2)
            printf("T ");
        if (x1 == -x2 && y1 == y2)
            printf("G ");
    }
    else
        printf("%f %f", (x1 + x2) /2., (y1 + y2) /2.);

    return 0;
}
```

++++ 003 +++++

Napisati program koji učitava četveroznamenkasti cijeli pozitivni broj. Program treba provjeriti je li umnožak prve i četvrte znamenke (znamenke se broje s lijeva na desno) djeljiv sa zbrojem druge i treće znamenke te u tom slučaju ispisati prve dvije znamenke broja. U protivnom treba ispisati druge dvije znamenke učitano broj. U slučaju da je zbroj druge i treće znamenke jednak nuli program treba ispisati 0.

Primjeri

za broj 8765 program treba ispisati: 65

za broj 8555 program treba ispisati: 85

Program će biti testiran samo s pozitivnim četveroznamenkastim cijelim brojevima, odnosno nije potrebno provjeravati ispravnost učitano broj.

```
#include<stdio.h>

int main(void)
{
    int num;

    scanf("%d", &num);
    if ((num/100%10)+(num/10%10) == 0)
        printf("0");
    else if (((num/1000)*(num%10)) % ((num/100%10)+(num/10%10)) == 0)
        printf("%d", num/100);
    else
        printf("%d", num%100);

    return 0;
}
```

++++ **004** +++++

Napisati program koji učitava šest cijelih brojeva po formatu %d %d %d %d %d %d koji predstavljaju koordinate triju točaka, redom x1, y1, x2, y2, x3, y3. Provjeriti jesu li točke vrhovi jednakokračnog trokuta. Ako su točke vrhovi jednakokračnog trokuta ispisati "Tocke su vrhovi jednakokračnog trokuta", inače ispisati "Tocke nisu vrhovi jednakokračnog trokuta".

Na primjer:

- za ulaz 0 0 1 3 2 0 potrebno je ispisati "Tocke su vrhovi jednakokračnog trokuta"

- za ulaz 0 0 0 3 2 0 potrebno je ispisati "Tocke nisu vrhovi jednakokračnog trokuta"

```
#include<stdio.h>

int main(void)
{
    int x1, y1, x2, y2, x3, y3;
    float dist1, dist2, dist3;

    scanf("%d %d %d %d %d %d", &x1, &y1, &x2, &y2, &x3, &y3);
    dist1 = (float)((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1));
    dist2 = (float)((x1-x3)*(x1-x3)+(y1-y3)*(y1-y3));
    dist3 = (float)((x3-x2)*(x3-x2)+(y3-y2)*(y3-y2));
    if (dist1 == dist2 || dist2 == dist3 || dist1 == dist3)
        printf("Tocke su vrhovi jednakokračnog trokuta");
    else
        printf("Tocke nisu vrhovi jednakokračnog trokuta");

    return 0;
}
```

++++ 005 ++++

Napisati program koji učitava pozitivni cijeli broj u dekadskom zapisu. Ako je broj izvan intervala [16, 255], program treba ispisati "Izvan intervala". Ako je broj unutar intervala [16,255], program treba učitani broj pretvoriti u heksadekadski zapis i ispisati ga. Znamenke heksadekadskog broja potrebno je ispisaati u formatu "%c%c" (bez razmaka). Heksadekadske znamenke A, B, C, D, E i F treba ispisati isključivo kao velika slova. Npr. za zadni broj 168, program treba ispisati A8 jer jer 168 po bazi 10 = A8 po bazi 16.

```
#include<stdio.h>

int main(void)
{
    unsigned int number;
    char first, second;

    scanf("%d", &number);
    if (number < 16 || number > 255)
        printf("Izvan intervala");
    else
    {
        first = number / 16;
        second = number % 16;
        printf("%c%c", (first>9)?64+first-9:48+first, (second>9)?64+second-9:48+second);
    }

    return 0;
}
```

++++ 006 ++++

Program kao ulaz s tipkovnice učitava četveroznamenkasti cijeli broj (pozitivan ili negativan). Potrebno je izbaciti neparne znamenke s početka broja te ispisati tako dobiveni cijeli broj s istim (početnim) predznakom. Početak broja je znamenka najveće težine. Ukoliko nakon izbacivanja ne ostane niti jedna znamenka ili ostane jedna ili više nula smatrati da je dobiveni cijeli broj 0. Također, za nule ne treba ispisivati predznak.

Pimjer:

za učitani broj 2468 ispisati 2468

za učitani broj 5768 ispisati 68

za učitani broj -5998 ispisati -8

za učitani broj 5769 ispisati 69

za učitani broj 1379 ispisati 0

za učitani broj 1000 ispisati 0

za učitani broj -1000 ispisati 0

**NAPOMENA:** Kaže kako program ne šljaka dobro za npr. sljedeće brojeve (ne znam zašto, a prema tekstu zadatka moji rezultati mi izgledaju valjano): 3012 rez. 2; 7703 rez. 0; -1001 rez. 0; -3090 rez. 0.

```
#include <stdio.h>

int main(void)
{
    int number;

    scanf("%d", &number);
    if ((number/1000%2 == 0) && (number/1000 != 0))
        printf("%d", number);
    else if ((number%1000/100%2 == 0) && (number%1000/100 != 0))
        printf("%d", number%1000);
    else if ((number%100/10%2 == 0) && (number%100/10 != 0))
        printf("%d", number%100);
}
```

```

else if ((number%10%2 == 0) && (number%10 != 0))
    printf("%d", number%10);
else
    printf("0");

return 0;
}

```

++++ **007** +++++

Napisati program koji učitava četiri znaka po formatu %c %c %c %c. Korisnik u ta četiri znaka mora učitati barem jedno slovo i barem jedan broj.

Ako nije učitano niti jedan broj treba ispisati poruku: "Nije učitano niti jedan broj."

Ako nije učitano niti jedno slovo treba ispisati poruku: "Nije učitano niti jedno slovo."

Ako su učitani ispravni znakovi treba ispisati poruku: "Znakovi su ispravni."

Na primjer:

- za ulaz a b C D potrebno je ispisati Nije učitano niti jedan broj.
- za ulaz 1 2 ! ? potrebno je ispisati Nije učitano niti jedno slovo.
- za ulaz + - \* / potrebno je ispisati Nije učitano niti jedan broj. Nije učitano niti jedno slovo.
- za ulaz 1 2 a B potrebno je ispisati Znakovi su ispravni.

```

#include <stdio.h>
#include <ctype.h>

int main(void)
{
    char first, second, third, fourth;

    scanf("%c %c %c %c", &first, &second, &third, &fourth);
    if (isalnum(first) || isalnum(second) || isalnum(third) || isalnum(fourth))
    {
        if (!isdigit(first) && !isdigit(second) && !isdigit(third) && !isdigit(fourth))
            printf("Nije učitano niti jedan broj.");
        else if (!isalpha(first) && !isalpha(second) && !isalpha(third) && !isalpha(fourth))
            printf("Nije učitano niti jedno slovo.");
        else if (isalnum(first) && isalnum(second) && isalnum(third) && isalnum(fourth))
            printf("Znakovi su ispravni.");
    }
    else
        printf("Nije učitano niti jedan broj. Nije učitano niti jedno slovo.");

    return 0;
}

```

++++ **008** +++++

Napisati program koji se za unesene koordinate dviju točaka u ravnini ispituje nalaze li se u istom te u kojem kvadrantu.

Ako se nalaze u istom kvadrantu, ispisati poruku: "Nalaze se u istom (tu upisati: prvom, drugom, trecem ili cetvrtom) kvadrantu."

U suprotnom ispisati poruku: "Ne nalaze se u istom kvadrantu. Prva je u (tu upisati: pravom, drugom, trecem ili cetvrtom) kvadrantu. Druga je u (tu upisati: prvom, drugom trecem ili cetvrtom) kvadrantu." Pretpostaviti da se niti jedna točka ne nalazi na nekoj od koordinatnih osi.

Primjeri:

Za učitane 7 8 9 5, program ispisuje "Nalaze se u istom prvom kvadrantu."

Za učitane -5 6 9 -8, program ispisuje "Ne nalaze se u istom kvadrantu. Prva je u drugom kvadrantu, Druga je u cetvrtom kvadrantu."

```

#include <stdio.h>

int main(void)
{
    int x1, y1, x2, y2;

```

```

int qFirst, qSecond;

scanf("%d %d %d %d", &x1, &y1, &x2, &y2);
if (x1 > 0 && y1 > 0)
    qFirst = 1;
else if (x1 < 0 && y1 > 0)
    qFirst = 2;
else if (x1 < 0 && y1 < 0)
    qFirst = 3;
else
    qFirst = 4;

if (x2 > 0 && y2 > 0)
    qSecond = 1;
else if (x2 < 0 && y2 > 0)
    qSecond = 2;
else if (x2 < 0 && y2 < 0)
    qSecond = 3;
else
    qSecond = 4;

if (qFirst == qSecond)
    printf("Nalaze se u istom %s kvadrantu.",
        (qFirst==1 || qFirst==2)? (qFirst==1?"prvom":"drugom") : (qFirst==3?"trecem":"cetvrtom"));
else
{
    if (qFirst==1)
        printf("Ne nalaze se u istom kvadrantu. Prva je u prvom kvadrantu. Druga je u %s kvadrantu.",
            (qSecond==1 || qSecond==2)? (qSecond==1?"prvom":"drugom") :
(qSecond==3?"trecem":"cetvrtom"));
        else if (qFirst==2)
            printf("Ne nalaze se u istom kvadrantu. Prva je u drugom kvadrantu. Druga je u %s kvadrantu.",
                (qSecond==1 || qSecond==2)? (qSecond==1?"prvom":"drugom") :
(qSecond==3?"trecem":"cetvrtom"));
        else if (qFirst==3)
            printf("Ne nalaze se u istom kvadrantu. Prva je u trecem kvadrantu. Druga je u %s kvadrantu.",
                (qSecond==1 || qSecond==2)? (qSecond==1?"prvom":"drugom") :
(qSecond==3?"trecem":"cetvrtom"));
        else
            printf("Ne nalaze se u istom kvadrantu. Prva je u cetvrtom kvadrantu. Druga je u %s kvadrantu.",
                (qSecond==1 || qSecond==2)? (qSecond==1?"prvom":"drugom") :
(qSecond==3?"trecem":"cetvrtom"));
    }

    return 0;
}

```

++++ 009 +++++

Napisati glavni program koji učitavan jedan znako binarne logičke operacije koja se mora obaviti (u varijablu tipa char) te prvi i drugi operand binarne logičke operacije koja se mora obaviti (u varijable tipa int).

Binarna logičke operacije koje mora podržati su & (binarni AND), | (binarni OR) i ^ (binarni XOR).

Nakon što su učitana ova tri parametra program mora izračunati rezultat izraza u kojem se zadana binarna operacija primjenjuje na učitane operande. Ukoliko je rezultat tog izraza paran potrebno je ispisati znamenku (dekadskog broja) na drugoj najmanje značajnoj poziciji izračunatog izraza, a u protivnom znamenku na trećoj najmanje značajnoj poziciji izračunatog izraza. Ispis znamenke obaviti u formatu "%d";

Ukoliko se zada neadekvatna binarna operacija (ona koja nije ni AND ni OR ni XOR ispisati "Greska");

Primjer: Ukoliko korisnik redom utipka:

```

|
4
8

```

Računa se izraz  $4|8$  i odbiva se 12 (0100|1000 = 1100). Budući da je rezultat paran broj (12 je paran broj) ispisuje se znamekna na drugoj najmanje značajnoj poziciji izraza, a to je 1 (gleda se broj 12 zdesna, na drugoj poziciji je znamenka 1).

```

#include <stdio.h>

int main(void)
{
    char binLogOperator;
    int first, second, result;

    scanf("%c %d %d", &binLogOperator, &first, &second);
    if (binLogOperator == '&')
    {
        result = first & second;
        if (result%2==0)
            printf("%d", result%100/10);
        else
            printf("%d", result%1000/100);
    }
    else if (binLogOperator == '|')
    {
        result = first | second;
        if (result%2==0)
            printf("%d", result%100/10);
        else
            printf("%d", result%1000/100);
    }
    else if (binLogOperator == '^')
    {
        result = second ^ first;
        if (result%2==0)
            printf("%d", result%100/10);
        else
            printf("%d", result%1000/100);
    }
    else
        printf("Greska");

    return 0;
}

```

++++ **010** +++++

Program učitavan peteroznamenasti prirodni broj. Ispitati je li broj osno simetričan s obzirom na srednju znamenku. Ako je, ispisati sumu parnih znamenki u broju, a ako nije, sumu neparnih znamenki u broju.

Primjer: za broj 54345 koji je osno simetričan s obzirom na srednju znamenku 3, program treba ispisati broj 8, a za broj 24684 treba ispisati broj 0.

```

#include <stdio.h>

int main(void)
{
    unsigned int number, sum = 0;

    scanf("%u", &number);
    printf("%5u %5u %5u %5u\n", number/10000, number%10, number%10000/1000, number%100/10);
    if (((number/10000) == (number%10)) && ((number%10000/1000) == (number%100/10)))
    {
        while(number != 0)
        {
            if ((number%10)%2 == 0)
                sum += number%10;
            number /= 10;
        }
    }
}

```

```

else
{
    while(number != 0)
    {
        if (!(number%10%2 == 0))
            sum += number%10;
        number /= 10;
    }
}
printf("%d", sum);

return 0;
}

```

++++ **011** +++++

Napisati program koji učitavan tri cijela broja po formatu %d %d %d. Program treba provjeriti je li neki od učitana tri broja umnožak kvadrata preostala dva broja. Ako jest ispisati "DA", a ako nije ispisati "NE".

Primjer:

Ulaz: 3 144 4 Izlaz: DA

Ulaz: 900 5 6 Izlaz: DA

Ulaz: 144 16 9 Izlaz NE

```

#include <stdio.h>

int main(void)
{
    int num1, num2, num3;

    scanf("%d %d %d", &num1, &num2, &num3);
    if (num1*num1*num2*num2 == num3)
        printf("DA");
    else if (num1*num1*num3*num3 == num2)
        printf("DA");
    else if (num2*num2*num3*num3 == num1)
        printf("DA");
    else
        printf("NE");

    return 0;
}

```

++++ **012** +++++

Učitati četveroznamenasti prirodni broj n, te nakon toga učitati cijeli broj m. Ako je zadani broj m u intervalu [1,4], potrebno je znamenku na mjestu m u broju n zamijeniti sa znamenkom na mjestu m+1. Znamenka koja se nalazi na mjestu jedinica ima mjesto 1, desetica 2 itd. Ako ne postoji znamenka na mjestu (m+1), potrebno je sve znamenke pomaknuti ciklički za jedno mjesto ulijevo. Ako je zadani broj m izvan intervala [1,4], potrebno je ispisati sumu znamenaka broja n.

Primjer: Program za ulaze 1234 i 3 ispisuje 2134, za ulaz 1234 1 ispisuje 1243, a za ulaze 1234 i 8 ispisuje 10.

```

#include <stdio.h>

int main(void)
{
    unsigned int numN;
    int numM;

    scanf("%u %d", &numN, &numM);
    if (numM == 1)
        printf("%u", numN/100*100 + numN%10*10 + numN%100/10);
    else if (numM == 2)

```

```

        printf("%u", numN/1000*1000 + numN%100/10*100 + numN%1000/100*10 + numN%10);
    else if (numM == 3)
        printf("%u", numN%1000/100*1000 + numN/1000*100 + numN%100);
    else if (numM == 4)
        printf("%u", numN%1000/100*1000 + numN%100/10*100 + numN%10*10 + numN/1000);
    else
        printf("%u", numN/1000 + numN%1000/100 + numN%100/10 + numN%10);

    return 0;
}

```

++++ **013** +++++

Napisati program koji učitava četveroznamenasti pozitivni cijeli broj N i dvoznamenkasti cijeli broj M. Program treba provjeriti je li M podniz od N. Ako jest, program mora ispisati "DA", a ako nije "NE".

Primjer:

Ulaz: 1234 23 Izlaz: DA

Ulaz: 1234 24 Izlaz: NE

Ulaz: 1234 25 Izlaz: NE

```

#include <stdio.h>

int main(void)
{
    unsigned int numN;
    int numM;

    scanf("%u %d", &numN, &numM);
    if (numN/1000 == numM/10)
    {
        if (numN%1000/100 == numM%10)
            printf("DA");
        else
            printf("NE");
    }
    else if (numN%1000/100 == numM/10)
    {
        if (numN%100/10 == numM%10)
            printf("DA");
        else
            printf("NE");
    }
    else if (numN%100/10 == numM/10)
    {
        if (numN%10 == numM%10)
            printf("DA");
        else
            printf("NE");
    }
    else
        printf("NE");

    return 0;
}

```

++++ **014** +++++

Napisati program koji učitavan tri cijela broja. Program treba provjeriti je li neki od tri unesena broja produkt (umnožak) preostala 2. Ako jest ispisati "DA", a ako nije ispisati "NE".



Primjer:

Ulaz: 6 54 9 Izlaz: DA

Ulaz: 3 8 5 Izlaz: NE

Ulaz: 81 9 9 Izlaz: DA

```
#include <stdio.h>

int main(void)
{
    int numA, numB, numC;

    scanf("%d %d %d", &numA, &numB, &numC);
    if (numA*numB == numC)
        printf("DA");
    else if (numA*numC == numB)
        printf("DA");
    else if (numB*numC == numA)
        printf("DA");
    else
        printf("NE");

    return 0;
}
```

++++ **015** +++++

Napisati program koji učitavan troznamenkasti pozitivni cijeli broj u dekadskom zapisu, pretvara ga u oktalni i zapisuje. Npr. za zadani dekadski broj 368, program treba ispisati 560 jer je 368 u bazi deset jednako 560 u bazi osam.

Ne treba provjeravati je li učitani broj ispravno zadan (tj. pretpostaviti da je ispravno zadano).

```
#include <stdio.h>

int main(void)
{
    int number;

    scanf("%d", &number);
    printf("%o", number);

    return 0;
}
```

++++ **016** +++++

Učitati četveroznamenkasti prirodni broj  $n$ , te nakon toga učitati cijeli broj  $m$ . Ako postoji znamenka na mjestu  $m$  u broju  $n$ , promijeniti broj  $n$  na način da se iz njega izbaci znamenka na  $m$ -tom mjestu i ispisati promijenjeni broj. Ako ne postoji  $m$ -ta znamenka u broju  $n$ , potrebno je ispisati sumu znamenaka učitanoog broja  $n$ . Mjesta znamenaka u broju se broje s desna na lijevo, tj. mjest 4 predstavlja znamenku s najvećom težinom u broju.

Primjer: Program za ulaze 5897 i 3 ispisuje 597, a za ulaze 8765 i -6 ispisuje 26.

```
#include <stdio.h>

int main(void)
{
```

```

unsigned int numN;
int numM;

scanf("%u %d", &numN, &numM);
if (numM == 1)
    printf("%u", numN/10);
else if (numM == 2)
    printf("%u", numN/100*10 + numN%10);
else if (numM == 3)
    printf("%u", numN/1000*100 + numN%100);
else if (numM == 4)
    printf("%u", numN%1000);
else
    printf("%u", numN/1000 + numN%1000/100 + numN%100/10 + numN%10);

return 0;
}

```

++++ **017** +++++

Napisati program koji učitava pozitivan troznamenkasti broj. Potrebno je ispisati broj koji bi nastao ukoliko bi se znamenke učitano broj silazno sortirale.

Na primjer:

- za učitani broj 132 potrebno je ispisati 321
- za učitani broj 312 potrebno je ispisati 321
- za učitani broj 304 potrebno je ispisati 430
- itd.

```

#include <stdio.h>

int main(void)
{
    int number, a, b, c;

    scanf("%d", &number);
    a = number/100;
    b = number%100/10;
    c = number%10;

    if (a >= b && a >= c)
    {
        if (b >= c)
            printf("%d", number);
        else
            printf("%d", number/100*100 + number%10*10 + number%100/10);
    }
    else if (b >= a && b >= c)
    {
        if (a >= c)
            printf("%d", number%100/10*100 + number/100*10 + number%10);
        else
            printf("%d", number%100/10*100 + number%10*10 + number/100);
    }
    else if (c >= a && c >= b)
    {
        if (a >= b)
            printf("%d", number%10*100 + number/100*10 + number%100/10);
        else
            printf("%d", number%10*100 + number%100/10*10 + number/100);
    }
}

```

```

    return 0;
}

```

#### ++++ 018 +++++

Napisati program koji učitava dva pozitivna cijela dvoznamenkasta broja. Program treba ispisati presjek skupa znamenki učitanih brojeva, odnosno znamenke koje se nalaze u oba broja. Svaka zajednička znamenka ispisuje se samo jednom. Ukoliko postoji više od jedne zajedničke znamenke, treba ispisati prvo manju pa onda veću znamenku, a znamenke međusobno odvojiti jednim razmakom. Ako učitani brojevi nemaju zajedničkih znamenki, program treba ispisati "Nema".

Npr. za ulaz "22 i 23" treba ispisati "2", a za ulaz "14 41" treba ispisati "1 4" (navodnike ne ispisivati).  
Ne treba provjeravati jesu li učitani brojevi ispravno zadani (tj. pretpostaviti da su ispravno zadani).

```

#include <stdio.h>

int main(void)
{
    unsigned int numOne, numTwo;

    scanf("%u %u", &numOne, &numTwo);
    if (numOne/10 == numTwo/10 || numOne/10 == numTwo%10 || numOne%10 == numTwo/10 || numOne%10 == numTwo%10)
    {
        if (numOne/10 == numOne%10)
        {
            if (numOne/10 == numTwo/10)
                printf("%u", numTwo/10);
            else
                printf("%u", numTwo%10);
        }
        else
        {
            if (numOne/10 > numOne%10)
                printf("%u %u", numOne%10, numOne/10);
            else
                printf("%u %u", numOne/10, numOne%10);
        }
    }
    else
        printf("Nema");

    return 0;
}

```

#### ++++ 019 +++++

Napisati program koji učitava četveroznamenkasti prirodni broj. Ako je umnožak znamenki broja paran, program treba ispisati znamenku najmanje težine. Znamenka najmanje težine je znamenka jedinice. Ako je umnožak broja neparan, program treba ispisati znamenku najveće težine. Znamenka najveće težine je znamenka tisućica.

Npr. ako se učitava broj 6871, program ispisuje znamenku 1 jer je umnožak znamenki broja paran i jer je znamenka 1 znamenka najmanje težine.

```

#include <stdio.h>

int main(void)
{
    unsigned int number;

    scanf("%u", &number);
    if (((number/1000)*(number%1000/100)*(number%100/10)*(number%10))%2 == 0)
        printf("%u", number%10);
    else
        printf("%u", number/1000);

    return 0;
}

```

++++ 020 ++++

Napisati program koji učitava četveroznamenasti pozitivni cijeli broj. Program treba provjeriti je li učitani broj djeljiv sa zbrojem svojih znamenki. Ako je djeljiv onda ispisati rezultat dijeljenja, a ako nije ispisati "NE".

Primjer:

Ulaz: 4550 Izlaz: 325 (4+5+5+0=14, 4550/14=325)

Ulaz: 2546 Izlaz: NE

```
#include <stdio.h>

int main(void)
{
    unsigned int number;

    scanf("%u", &number);
    if ( (number%((number/1000)+(number%1000/100)+(number%100/10)+(number%10))) == 0)
        printf("%u", number/((number/1000)+(number%1000/100)+(number%100/10)+(number%10)));
    else
        printf("NE");

    return 0;
}
```

++++ 021 ++++

Napisati program koji učitava šest realnih brojeva po formatu %f %f %f %f %f %f koji predstavljaju koeficijente triju jednadžbi pravaca u eksplicitnom obliku ( $y=kx+l$ ). Koeficijenti se unose redom  $k_1, l_1, k_2, l_2, k_3, l_3$ .

Potrebno je ispitati je li trokut kojeg omeđuju uneseni pravci pravokutan te ispisati odgovarajuću poruku. Ukoliko je trokut pravokutan ispisati "DA", u protivnom ispisati "NE". Ukoliko su uneseni pravci međusobno paralelni (dva ili sva tri) ispisati poruku "Pravci su paralelni!".

Primjer:

- za ulaz 1 1 1 2 3 4 potrebno je ispisati Pravci su paralelni!

- za ulaz 1 2 -1 2 3 4 potrebno je ispisati DA

- za ulaz 1 2 3 4 -0.5 6 potrebno je ispisati NE

```
#include <stdio.h>

int main(void)
{
    float k1, l1, k2, l2, k3, l3;
    float x1, y1, x2, y2, x3, y3;
    int sideA, sideB, sideC;

    scanf("%f %f %f %f %f %f", &k1, &l1, &k2, &l2, &k3, &l3);

    x1 = (l2 - l1)/(k1 - k2);
    y1 = k1*x1 + l1;
    x2 = (l3 - l1)/(k1 - k3);
    y2 = k1*x2 + l1;
    x3 = (l3 - l2)/(k2 - k3);
    y3 = k3*x3 + l3;

    sideA = (int)((x2-x1)*(x2-x1) + (y2-y1)*(y2-y1));
    sideB = (int)((x3-x1)*(x3-x1) + (y3-y1)*(y3-y1));
    sideC = (int)((x2-x3)*(x2-x3) + (y2-y3)*(y2-y3));

    if (k1 == k2 || k1 == k3 || k2 == k3)
        printf("Pravci su paralelni!");
}
```

```

else if ((sideA + sideB == sideC) ||
        (sideA + sideC == sideB) ||
        (sideB + sideC == sideA))
    printf("DA");
else
    printf("NE");

return 0;
}

```

#### ++++ 022 +++++

Napisati program koji učitava dva pozitivna peteroznamenkasta broja. Ako je prvi učitani veći od drugog, ispisati sumu znamenki na neparnim pozicijama oba učitana broja. Ako je drugi učitani veći od prvog ispisati sumu znamenki na parnim pozicijama. Također, ako je veći učitani broj paran, ispisati razliku brojeva. Ako pak nije paran, program neće ispisati razliku brojeva. Ako su brojevi jednaki, program neće ništa ispisati.

Primjer:

Za učitane brojeve 59786 (5 je na prvoj, 9 na drugoj poziciji itd.) te 24893 redom, program treba ispisati sumu znamenki na neparnim pozicijama, tj. 31 što je  $5+6+5+2+8+3$  te razliku brojeva  $59786-24893 = 34893$ . Dakle, izlaz je: 31 34893. Za učitane brojeve 24893 te 59786 redom, program treba ispisati sumu znamenki na parnim pozicijama, tj. 30 što je  $4+9+9+8$  te razliku brojeva  $59786-24893=34893$ . Dakle, izlaz je: 30 34893.

```

#include <stdio.h>

int main(void)
{
    unsigned int num1, num2;

    scanf("%u %u", &num1, &num2);

    if (num1 > num2)
    {
        printf("%u", num1/10000 + num1%1000/100 + num1%10 + num2/10000 + num2%1000/100 + num2%10);
        if (num1%2 == 0)
            printf(" %u", num1-num2);
    }
    else if (num1 < num2)
    {
        printf("%u", num1%10000/1000 + num1%100/10 + num2%10000/1000 + num2%100/10);
        if (num2%2 == 0)
            printf(" %u", num2-num1);
    }

    return 0;
}

```

#### ++++ 023 +++++

Napisati program koji učitava dva pozitivna cijela četveroznamenkasta broja (vodeće znamenke mogu biti 0 (nula)). Prvo se učitava kontrolni broj k čije su znamenke isključivo 0 i 1, a zatim broj x nad kojim je potrebno napraviti preinaku i to na slijedeći način. Znamenke za koje vrijedi da je znamenka na ekvivalentnoj poziciji u kontrolnom broju jednaka znamenici 1 treba uvećati za jedan (ako je znamenka koju treba uvećati 9 tada ona postaje znamenka 0). Znamenke za koje vrijedi da je znamenka na ekvivalentnoj poziciji u kontrolnom broju 0 se ne mijenjaju. Program kao izlaz treba ispisati tako promijenjeni broj x (bez vodećih 0 (nula)).

Primjer:

- za ulaze k=1011 i x=2345 program treba ispisati broj 3356
- za ulaze k=1001 i x= 9034 program treba ispisati broj 35
- za ulaze k=0001 i x=9034 program treba ispisati broj 9035 (PAŽNJA!)

```

#include <stdio.h>

```

```

int main(void)
{
    unsigned int num1, num2;

    scanf("%u %u", &num1, &num2);

    if ((num1/1000 == 1) && (num2/1000 != 9))
        num2 += 1000;
    else if ((num1/1000 == 1) && (num2/1000 == 9))
        num2 = num2%1000;

    if ((num1%1000/100 == 1) && (num2%1000/100 != 9))
        num2 += 100;
    else if ((num1%1000/100 == 1) && (num2%1000/100 == 9))
        num2 = num2/1000*1000 + num2%100;

    if ((num1%100/10 == 1) && (num2%100/10 != 9))
        num2 += 10;
    else if ((num1%100/10 == 1) && (num2%100/10 == 9))
        num2 = num2/100*100 + num2%10;

    if ((num1%10 == 1) && (num2%10 != 9))
        num2 += 1;
    else if ((num1%10 == 1) && (num2%10 == 9))
        num2 = num2/10*10;

    printf("%u", num2);

    return 0;
}

```

++++ **024** +++++

Napisati program koji učitava dva pozitivna troznamenkasta cijela broja po formatu %d %d. Potrebno je ispisati broj koji bi nastao tako da se sve znamenke prvog učitanoog broja koje su manje od znamenki na istom mjestu drugog učitanoog broja zamijene sa znamenkama drugog učitanoog broja.

Na primjer:

- za ulaz 271 315 potrebno je ispisati 375
- za ulaz 512 331 potrebno je ispisati 532
- za ulaz 123 456 potrebno je ispisati 456
- itd.

```

#include <stdio.h>

int main(void)
{
    int num1, num2, numX;

    scanf("%d %d", &num1, &num2);

    if (num1/100 >= num2/100)
        numX = num1/100*100;
    else
        numX = num2/100*100;

    if (num1%100/10 >= num2%100/10)
        numX += num1%100/10*10;
    else
        numX += num2%100/10*10;
}

```

```

    if (num1%10 >= num2%10)
        numX += num1%10;
    else
        numX += num2%10;

    printf("%d", numX);

    return 0;
}

```

++++ **025** +++++

Učitati četveroznamenkasti prirodni broj *n*, te nakon toga učitati jednoznamenkasti broj *m*. Ako u broju *n* postoji znamenka koja odgovara broju *m*, ispisati na kojim se sve mjestima javlja. Ako broj *n* ne sadrži znamenku *m*, ispisati "0".

Program će biti testiran samo s četveroznamenkastim i jednoznamenkastim cijelim brojem, odnosno nije potrebno provjeravati ispravnost učitanih varijabli.

Primjer: Program za ulaze 5897 i 3 ispisuje 0, za ulaz 5897 i 5 ispisuje 1, a za ulaze 5888 i 8 ispisuje 234.

```

#include <stdio.h>

int main()
{
    unsigned int numN;
    int numM;

    scanf("%u %d", &numN, &numM);
    if ((numM == numN/1000) || (numM == numN%1000/100) || (numM == numN%100/10) || (numM == numN%10))
    {
        if (numM == numN/1000)
            printf("1");
        if (numM == numN%1000/100)
            printf("2");
        if (numM == numN%100/10)
            printf("3");
        if (numM == numN%10)
            printf("4");
    }
    else
        printf("0");

    return 0;
}

```

++++ **026** +++++

Napisati program koji po formatu %d %d učitava koordinate točke te ispisuje u kojem se kvadrantu ta točka nalazi. Ako se točka nalazi na nekoj od osi ili u ishodištu program ispisuje "OS".

Npr.:

ULAZ	IZLAZ
3 2	1
-1 2	2
3 -5	4
0 0	OS
1 0	OS

```

#include <stdio.h>

int main(void)

```

```

{
    int x, y;

    scanf("%d %d", &x, &y);

    if (x > 0)
    {
        if (y > 0)
            printf("1");
        else
            printf("4");
    }
    else if (x < 0)
    {
        if (y > 0)
            printf("2");
        else
            printf("3");
    }
    else
        printf("05");

    return 0;
}

```

++++ **027** +++++

Napisati program koji učitava dva pozitivna troznamenkasta cijela broja po formatu %d %d. Izračunati i ispisati treći troznamenkasti broj čije se znamenke dobiju zbrajanjem odgovarajućih znamenki učitanih brojeva. Prilikom zbrajanja znamenki, ako je rezultat veći od 9, uzima se samo znamenka jedinica.

Na primjer:

- za ulaz 123 456 potrebno je ispisati 579
- za ulaz 265 475 potrebno je ispisati 630
- za ulaz 222 999 potrebno je ispisati 111
- itd.

```
#include <stdio.h>
```

```

int main(void)
{
    unsigned int numOne, numTwo, numX;

    scanf("%u %u", &numOne, &numTwo);

    numX = (numOne/100 + numTwo/100) %10*100;
    numX += (numOne%100/10 + numTwo%100/10) %10*10;
    numX += (numOne%10 + numTwo%10) %10;
    printf("%u", numX);

    return 0;
}

```

++++ **028** +++++

Napisati program koji redom učitava dva znaka (char) koji predstavljaju dva jednoznamenkasta broja, te znak c koji predstavlja računsku operaciju koju treba obaviti nad dva učitana znaka, odnosno broja (zbrajanje (+), oduzimanje (-), množenje(\*), dijeljenje(/) ili ostatak cjelobrojnog dijeljenja (%)). Znakove je potrebno učitati u jednoj scanf naredbi i to u



formatu "%c %c %c" (između svakog %c i %c se nalazi razmak te nema \n na kraju). Ako je sadržaj varijable c jednak '+' učitane brojeve treba zbrojiti. Ako je sadržaj varijable c jednak '-' treba oduzeti drugi broj od prvog. Ako je sadržaj varijable c jednak '\*' učitane brojeve treba pomnožiti. Ako je sadržaj varijable c jednak '/' treba izračunati rezultat cjelobrojnog dijeljenja prvog broja sa drugim. Ako je sadržaj varijable c jednak '%' treba izračunati ostatak cjelobrojnog dijeljenja prvog broja sa drugim. Ako je unesena neispravna operacija (koja nije među već navedenima), program treba ispisati "Neispravna". Inače, program treba ispisati rezultat zadane računske operacije kao cijeli broj.

```
#include <stdio.h>

int main(void)
{
    char num1, num2, op;

    scanf("%c %c %c", &num1, &num2, &op);

    if (op == '+')
        printf("%d", (num1-'0')+(num2-'0'));
    else if (op == '-')
        printf("%d", (num1-'0')-(num2-'0'));
    else if (op == '*')
        printf("%d", (num1-'0')*(num2-'0'));
    else if (op == '/')
        printf("%d", (num1-'0')/(num2-'0'));
    else if (op == '%')
        printf("%d", (num1-'0')%(num2-'0'));
    else
        printf("Neispravna");

    return 0;
}
```

++++ **029** +++++

Napisati program koji učitava četveroznamenkasti prirodni broj. Lijevu polovicu čini broj sastavljen od prve dvije znamenke, a desnu broj sastavljen od zadnje dvije znamenke. (Npr. lijeva polovica broja 1756 je broj 17, a desna broj 56). Ako je suma lijeve i desne polovice broja djeljiva sa lijevom polovicom, program treba ispisati "Jest", inače "Nije" (navodnike ne ispisivati).

```
#include <stdio.h>

int main(void)
{
    unsigned int number;

    scanf("%u", &number);
    if ((number%100)%((number/100)+(number%100)) == 0)
        printf("Jest");
    else
        printf("Nije");

    return 0;
}
```

++++ **030** +++++

Napisati program koji učitava četveroznamenkasti pozitivni cijeli broj, svaku neparnu znamenku učitano broj zamjenjuje prvom većom parnom znamenkom i ispisuje novi broj. Znamenku 9 treba zamijeniti znamenkom 0. Ako su sve znamenke učitano broj parne, program treba ispisati originalni broj.

Npr. za zadan broj 4379, program treba ispisati 4480. Početne nule ne ispisivati npr. 9999 daje kao rezultat 0 (jedna ispisana znamenka), ne 0000 (4 ispisane znamenke).

```
#include <stdio.h>

int main(void)
{
    unsigned int number, numX;

    scanf("%u", &number);

    if (number/1000%2 != 0)
        numX = (number/1000+1)%10 *1000;
    else
        numX = number/1000*1000;

    if (number%1000/100%2 != 0)
        numX += (((number%1000/100)+1)%10) *100;
    else
        numX += number%1000/100*100;

    if (number%100/10%2 != 0)
        numX += (number%100/10+1)%10 *10;
    else
        numX += number%100/10*10;

    if (number%10%2 != 0)
        numX += (number%10+1)%10;
    else
        numX += number%10;
    printf("%u", numX);

    return 0;
}
```

++++ **031** +++++

Napisati program koji učitava šest cijelih brojeva po formatu %d %d %d %d %d %d koji predstavljaju koordinate triju točaka, redom x1, y1, x2, y2, x3, y3. Provjeriti jesu li točke tri vrha pravokutnika.

Ako točke jesu vrhovi pravokutnika ispisati "Tocke su tri vrha pravokutnika". Inače ispisati "Tocke nisu tri vrha pravokutnika".

Na primjer:

- za ulaz 0 0 0 3 2 3 potrebno je ispisati Tocke su tri vrha pravokutnika
- za ulaz 0 -2 0 0 -1 3 potrebno je ispisati Tocke nisu tri vrha pravokutnika

```
#include <stdio.h>

int main(void)
{
    int x1, y1, x2, y2, x3, y3;
    int sideA, sideB, sideC;

    scanf("%d %d %d %d %d %d", &x1, &y1, &x2, &y2, &x3, &y3);
    sideA = (x2-x1)*(x2-x1) + (y2-y1)*(y2-y1);
    sideB = (x3-x1)*(x3-x1) + (y3-y1)*(y3-y1);
    sideC = (x3-x2)*(x3-x2) + (y3-y2)*(y3-y2);
```

```

if (sideA+sideB==sideC || sideA+sideC==sideB || sideB+sideC==sideA)
    printf("Tocke su tri vrha pravokutnika");
else
    printf("Tocke nisu tri vrha pravokutnika");

return 0;
}

```

++++ **032** +++++

Napisati program koji učitava dva troznamenkasta pozitivna cijela dekadaska broja te ispisuje manji od dva učitana broja u oktalnom zapisu. Ako su učitani brojevi jednaki, program treba ispisati prvi broj u oktalnom zapisu.

Npr. za zdane brojeve 368 i 455, program treba ispisati 560 jer je 368 po bazi 10 jednako 560 po bazi osam; za zadane brojeve 155 i 155, program treba ispisati 233 jer je 155 po bazi deset jednako 233 po bazi osam.

```

#include <stdio.h>

int main(void)
{
    unsigned int num1, num2;

    scanf("%d %d", &num1, &num2);
    num1==num2 ? printf("%o", num1) : (num1>num2 ? printf("%o", num2) : printf("%o", num1));

    return 0;
}

```

++++ **033** +++++

Napisati program koji učitava tri pozitivna (>0) cijela broja po formatu %d %d %d. Provjeriti jesu li uneseni brojevi Pitagorini, odnosno vrijedi li za njih (u bilo kojem redosljedu):  $a^2 + b^2 = c^2$

Ako su Pitagorini ispisati sumu manja dva od njih (u formuli: a i b). Ako nisu za manja dva od njih izračunati zbroj njihovi kvadrata.

Na primjer:

- za ulaz 3 4 5 potrebno je ispisati 7
- za ulaz 5 3 4 potrebno je ispisati 7
- za ulaz 5 14 12 potrebno je ispisati 169
- itd.

```

#include <stdio.h>

int main(void)
{
    int x, y, z;

    scanf("%d %d %d", &x, &y, &z);
    if (x*x+y*y==z*z || x*x+z*z==y*y || y*y+z*z==x*x)
    {
        if (x >= y)
        {
            if (x <= z)
                printf("%d", y+x);
            else
                printf("%d", y+z);
        }
        else
        {
            if (y >= z)

```

```

        printf("%d", x+z);
    else
        printf("%d", x+y);
    }
}
else
{
    if (x >= y)
    {
        if (x <= z)
            printf("%d", y*y+x*x);
        else
            printf("%d", y*y+z*z);
    }
    else
    {
        if (y >= z)
            printf("%d", x*x+z*z);
        else
            printf("%d", x*x+y*y);
    }
}

return 0;
}

```

++++ **034** +++++

Napisati program koji učitava troznamenkasti cijeli pozitivni broj x. Program treba provjeriti vrijedi li tvrdnja da je kvadrat prve znamenke broj x jednak broju kojega čine druge dvije znamenke broj x. Znamenke se numeriraju s lijeva na desno (u broju 416, prva znamenka jest 4, druga 1, a treća 6).

U slučaju da zadana tvrdnja vrijedi, program treba ispisati broj koji se dobije obrtanjem redoslijeda znamenaka broja x. (za broj 416, broj koji se dobiva obrtanjem redoslijeda znamenaka jest 614).

U slučaju da zadana tvrdnja ne vrijedi, program treba ispisati:

- Broj 1 ako je učitani broj x u intervalu [150, 450]
- Broj 7 ako je učitani broj x u intervalu [451, 700]
- Inače sumu znamenaka učitano broj x

```

#include <stdio.h>

int main(void)
{
    int numX;

    scanf("%d", &numX);
    if (numX/100*numX/100 == numX%100)
    {
        if (numX%100/10 == 0)
            printf("%d", numX%100*100 + numX/100);
        else
            printf("%d", numX%100*10 + numX/100);
    }
    else
    {
        if (numX >= 150 && numX <= 450)
            printf("1");
        else if (numX >= 451 && numX <= 700)
            printf("7");
        else

```

```

    printf("%d", numX/100 + numX%100/10 + numX%10);
}

return 0;
}

```

#### ++++ 035 +++++

Napisati program koji redom učitava dva prirodna cijela broja, te znak c koji predstavlja računsku operaciju koju treba obaviti nad dva učitana broja (zbrajanje (+), oduzimanje (-), množenje (\*), dijeljenje(/) ili ostatak cjelobrojnog dijeljenja (%)). Brojeve i znak potrebno je učitati u jednoj scanf naredbi i to u formatu %d %d %c (između %d i %d , te %d i %c se nalazi razmak, te nema \n na kraju), Ako je sadržaj varijable c jednak '+', učitane brojeve treba zbrojiti. Ako je sadržaj varijable c jednak '-', treba oduzeti drugi broj od prvoga. Ako je sadržaj varijable c jednak '\*', učitane brojeve treba pomnožiti. Ako je sadržaj varijable c jednak '/', treba izračunati rezultat cjelobrojnog dijeljenja prvog broja s drugim. Ako je sadržaj varijable c jednak '%', treba izračunati ostatak cjelobrojnog dijeljenja prvog broja s drugim. Ako je unesena neispravna operacija (koja nije među navedenima), program treba ispisati "Neispravna". Inače program treba ispisati rezultat zadane računske operacije.

Uputa: pretpostavite da neće doći dijeljenja s 0.

```

#include <stdio.h>

int main(void)
{
    int numX, numY;
    char op;

    scanf("%d %d %c", &numX, &numY, &op);
    if (!(op=='+' || op=='-' || op=='*' || op=='/' || op=='%'))
        printf("Neispravna");
    else
    {
        if (op=='+')
            printf("%d", numX+numY);
        else if (op=='-')
            printf("%d", numX-numY);
        else if (op=='*')
            printf("%d", numX*numY);
        else if (op=='/')
            printf("%d", numX/numY);
        else if (op=='%')
            printf("%d", numX%numY);
    }

    return 0;
}

```

#### ++++ 036 +++++

Napisati program koji učitava znakove a i b te cijeli broj d, gdje a i b predstavljaju znamenke broja s bazom d. Znakove i cijeli broj potrebno je učitati u jednoj scanf naredbi i to u formatu "%c %c %d" (između %c i %c te između %c i %d nalazi se razmak te nema "\n" na kraju). Prvi znak (a) predstavlja znamenku veće težine, a drugi znak (b) znamenku manje težine, odnosno znamenku jedinica. Program treba ispisati broj kojeg čine znamenke a i b u dekadskom zapisu. Pretpostaviti da se znamenke A, B, C, D, E, F, G... uvijek zadaju kao velika slova te da su poredane prema ASCII tablici, a da neće nikada biti veći od 32.

Npr. ako se učitaju znakovi P i 2 te broj 31, program treba ispisati broj 777 jer jer P2 u bazi 32 jednako 777 u bazi 10.

Nije potrebno kontrolirati korisnički unos.

```

#include <stdio.h>
#include <ctype.h>

int main(void)
{
    char ch1, ch2;
    int base;

    scanf("%c %c %d", &ch1, &ch2, &base);
    if (isalpha(ch1) && isalpha(ch2))
        printf("%d", (ch1-55)*base + (ch2-55));
    else if (!isalpha(ch1) && isalpha(ch2))
        printf("%d", (ch1-'0')*base + (ch2-55));
    else if (isalpha(ch1) && !isalpha(ch2))
        printf("%d", (ch1-55)*base + (ch2-'0'));
    else
        printf("%d", (ch1-'0')*base + (ch2-'0'));

    return 0;
}

```

++++ **037** +++++

Učitava se troznamenkasti prirodni broj *n*, zatim prirodni broj *m* u rasponu [1,3] (s uključenim granicama intervala), te znak *c* koji označava smjer rotacije ('L' označava rotaciju ulijevo, a 'R' rotaciju udesno). Brojeve i znak potrebno je učitati u jednoj scanf naredbi i to u formatu "%d %d %c" (između svakog %d i %c nalazi se razmak). Potrebno je broj *n* rotirati *m* puta u smjeru određenom znakom *c*, te ga zatim ispisati. Ako učitani *n* nije broj veći od 0 ili je učitani *m* izvan intervala [1,3] ili ako *c* nije 'L' ili 'R' potrebno je ispisati poruku "Error" (navodnike ne ispisivati).

Uputa: Rotacija mijenja poziciju znamenki u broju tako da ih ciklički gura na zadanu stranu. Ukoliko se znamenku nema kamo gurnuti ona prelazi na "drugnu stranu" troznamenkastog broja.

Npr.

- za *n*=123, *m*=1 i *c*='L' rezultat rotacije je 231 (sve znamenke su se pomaknule za jedno mjesto ulijevo, a znamenka 1 je otišla na kraj)
- za *n*=123, *m*=2 i *c*='R' rezultat rotacije je 231 (znamenka 1 se pomaknula za dva mjesta udesno, znamenka 3 je dospjela u sredinu, a znamenka 2 na početak)

```

#include <stdio.h>

int main(void)
{
    int numN, numM;
    char direction;

    scanf("%d %d %c", &numN, &numM, &direction);
    if (!(numN>0 && (numM>=1 && numM<=3) && (direction=='L' || direction=='R')))
        printf("Error");
    else
    {
        if (direction=='R')
        {
            if (numM==1)
                printf("%d", numN%10*100 + numN/100*10 + numN%100/10);
            else if (numM==2)
                printf("%d", numN%100*10 + numN/100);
            else
                printf("%d", numN);
        }
        else
        {
            if (numM==1)

```

```

        printf("%d", numN%100*10 + numN/100);
    else if (numM==2)
        printf("%d", numN%10*100 + numN/100*10 + numN%100/10);
    else
        printf("%d", numN);
    }
}

return 0;
}

```

++++ **038** ++++

Program kao ulaz s tipkovnice prima četveroznamenkasti cijeli broj. Potrebno je sve parne znamenke tog broja zamijeniti sa znamenkom 0 te ispisati tako dobiven cijeli broj.

Primjer: za učitani broj 8586, treba ispisati broj 500, jer su znamenke 8, 8 i 6 parne.

```

#include <stdio.h>

int main(void)
{
    int number, arrNum[4], i, z;

    scanf("%d", &number);

    for (i=3; i>=0; i--)
    {
        arrNum[i] = number%10;
        number /= 10;
    }

    for (i=0; i<4; i++)
    {
        if (arrNum[i]%2 == 0)
            arrNum[i] = 0;
    }

    number = 0;
    for (i=3, z=1; i>=0; i--, z=z*10)
    {
        number += arrNum[i]*z;
    }
    printf("%d", number);

    return 0;
}

```

**ILI** \*\*\*\*\* se prva petlja koja ubacuje brojeve u array zamijeni sa sprintf funkcijom koju je napomenu kolega futo, ali onda nije više int array nego char array, tj. string

```

#include <stdio.h>

int main(void)
{
    int number, i, z;
    char arrNum[4];

    scanf("%d", &number);

    sprintf(arrNum, "%d", number);
}

```

```

for (i=0; i<4; i++)
{
    if ((arrNum[i]-'0')%2 == 0)
        arrNum[i] = '0';
}

number = 0;
for (i=3, z=1; i>=0; i--, z=z*10)
{
    number += (arrNum[i]-'0')*z;
}
printf("%d", number);

return 0;
}

```

+++++ **039** +++++

Napisati program koji učitava pozitivan četveroznamenasti broj. Pronaći najmanju znamenku i "preseliti" je na najznačajniju poziciju (na mjesto tisućica). Ukoliko više znamenki učitano broj ima jednaku najmanju vrijednost smatrati da je najmanja ona koja se nalazi na najznačajnijoj poziciji (najznačajniju poziciju četveroznamenastom broju ima znamenka na mjestu tisućica, nakon nje znamenka na mjestu stotica itd.) Potrebo je ispisati nastali broj.

Na primjer:

- za učitani broj 5321 potrebno je ispisati 1532
- za učitani broj 7634 potrebno je ispisati 3764
- za učitani broj 6131 potrebno je ispisati 1631
- itd.

```

#include <stdio.h>

int main(void)
{
    int number, i, z, flag;
    char arrNum[4], temp, temp2;

    scanf("%d", &number);
    sprintf(arrNum, "%d", number);

    if ((arrNum[0]==arrNum[3] || arrNum[1]==arrNum[2] || arrNum[0]==arrNum[2] || arrNum[1]==arrNum[3] ||
        arrNum[0]==arrNum[1] || arrNum[2]==arrNum[3]))
    {
        for (i=3; i>0; i--)
        {
            if (arrNum[i]<arrNum[i-1])
            {
                temp = arrNum[i];
                flag = i;
            }
            else
            {
                temp = arrNum[i-1];
                flag = i-1;
            }
        }
        temp2 = arrNum[0];
        arrNum[0] = arrNum[flag];
        arrNum[flag] = temp2;

        number = 0;
        for (i=3, z=1; i>=0; i--, z=z*10)
        {
            number += (arrNum[i]-'0')*z;
        }
    }
}

```



```
}
else
{
    if (arrNum[3]<arrNum[2] && arrNum[3]<arrNum[1] && arrNum[3]<arrNum[0])
        number = (arrNum[3]-'0')*1000 + (arrNum[0]-'0')*100 + (arrNum[1]-'0')*10 + (arrNum[2]-'0');
    else if (arrNum[2]<arrNum[3] && arrNum[2]<arrNum[1] && arrNum[2]<arrNum[0])
        number = (arrNum[2]-'0')*1000 + (arrNum[0]-'0')*100 + (arrNum[1]-'0')*10 + (arrNum[3]-'0');
    else if (arrNum[1]<arrNum[2] && arrNum[1]<arrNum[3] && arrNum[1]<arrNum[0])
        number = (arrNum[1]-'0')*1000 + (arrNum[0]-'0')*100 + (arrNum[2]-'0')*10 + (arrNum[3]-'0');
}

printf("%d", number);

return 0;
}
```