

Zadatak #1

Učitati deset cijelih brojeva. Ako je korisnik unio broj koji nije veći od nule ispisati poruku o pogrešci i završiti program. Inače ispisati koliko je brojeva djeljivo sa najmanjim učitanim brojem.

Ulaz:

7 10 5 12 15 3 9 13 17 14

Izlaz:

Najmanji: 3

Djeljivo: 4

Rješenje

```
#include <stdio.h>

#define MAXP 10

int main(void) {
    int i, min, djeljivih;
    int polje[MAXP];

    for (i = 0; i < MAXP; ++i) {
        scanf("%d", &polje[i]);
        if (polje[i] <= 0) {
            printf("Brojevi moraju biti veći od nule\n");
            return 1;
        }
    }

    min = polje[0];
    for (i = 1; i < MAXP; ++i) {
        if (polje[i] < min)
            min = polje[i];
    }

    djeljivih = 0;
    for (i = 0; i < MAXP; ++i) {
        if (polje[i] % min == 0)
            ++djeljivih;
    }

    printf("Najmanji: %d\nDjeljivo: %d\n", min, djeljivih);
    return 0;
}
```

Zadatak #2

Učitavati cijele brojeve dok se ne unese broj nula. Pretpostaviti da neće biti uneseno više od 50 brojeva i da će zadnji uneseni broj uvijek biti nula. Treba izračunati aritmetičku sredinu unesenih brojeva (uzeti u obzir i zadnju nulu), te ispisati koliko je brojeva manje od aritmetičke sredine, koliko jednako, a koliko veće. Kod računanja izbjeći cjelobrojno dijeljenje.

Ulaz:

1 2 3 4 5 6 7 8 9 10 0

Izlaz

Manjih: 5

Vecih: 5

Jednakih: 1

Rješenje

```
#include <stdio.h>

#define MAXP 50

int main(void) {
    int i, brojeva, suma, manjih, jednakih, vecih;
    float sredina;
    int polje[MAXP];

    suma = 0;
    brojeva = 0;
    for (i = 0; i < MAXP; ++i) {
        scanf("%d", &polje[i]);
        ++brojeva;
        suma += polje[i];
        if (polje[i] == 0)
            break;
    }

    sredina = (float)suma / brojeva;
    manjih = jednakih = vecih = 0;

    for (i = 0; i < brojeva; ++i) {
        if (polje[i] < sredina)
            ++manjih;
        else if (polje[i] > sredina)
            ++vecih;
        else
            ++jednakih;
    }
    printf("Manjih: %d\nVecih: %d\nJednakih: %d\n",
        manjih, vecih, jednakih);
    return 0;
}
```

Zadatak #3

Učitavati znakove dok se ne unese znak točka ('.'). Pretpostaviti da neće biti uneseno više od 30 znakova i da će zadnji znak uvijek biti točka. Ispisati sve suglasnike obrnutim redoslijedom.

Ulaz:

F@d!AEkj?HOL\$zt.

Izlaz:

tzLHjkdF

Rješenje

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

#define MAXP 30

int main(void) {
    int i, j;
    char niz[MAXP];

    i = 0;
    do {
        scanf("%c", &niz[i]);
        ++i;
    } while (niz[i - 1] != '.');

    for (j = i - 2; j >= 0; --j) {
        char znak = toupper(niz[j]);
        if (isalpha(znak) && znak != 'A' && znak != 'E' &&
            znak != 'I' && znak != 'O' && znak != 'U') {

            printf("%c", niz[j]);
        }
    }
    return 0;
}
```

Zadatak #4

Učitati dva niza znakova koristeći funkciju `gets`. Pretpostaviti da nizovi neće biti duži od 30 znakova. Ispisati koliko se puta drugi niz pojavljuje unutar prvoga.

Ulaz:

petar petri petlju plete

pe

Izlaz:

Broj pojavljivanja: 3

Rješenje

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

#define MAXL 30

int main(void) {
    char niz[MAXL + 1], podniz[MAXL + 1], *polozaj;
    int brojac;

    gets(niz);
    gets(podniz);

    brojac = 0;
    polozaj = &niz[0];
    while (1) {
        polozaj = strstr(polozaj, podniz);
        if (polozaj == NULL) break;
        ++polozaj;
        ++brojac;
    }

    printf("Broj pojavljivanja: %d\n", brojac);
    return 0;
}
```

Zadatak #5

Napisati funkciju koja prima dva cijela broja i vraća dvije vrijednosti: njihov zbroj (preko `return-a`) i umnožak (preko pokazivača). Napisati primjer pozivanja funkcije iz glavnog programa.

Rješenje

```
#include <stdio.h>

int zbrojiPaMnozi(int prvi, int drugi, int *produkt) {
    int suma;
    suma = prvi + drugi;
    *produkt = prvi * drugi;
    return suma;
}

int main(void) {
    int a, b;
    int zbroj, umnozak;

    scanf("%d %d", &a, &b);
    zbroj = zbrojiPaMnozi(a, b, &umnozak);
    printf("Zbroj: %d\nUmnozak: %d\n", zbroj, umnozak);
    return 0;
}
```

Zadatak #6

Učitavati broj redaka i stupaca matrice (neće biti veći od 20) sve dok korisnik ne unese valjane brojeve, a zatim učitati elemente matrice. Ispisati najmanji i najveći element matrice.

Ulaz:

2 3
1 2 3
4 5 6

Izlaz:

Najmanji: 1
Najveci: 6

Rješenje

```
#include <stdio.h>

#define MAXR 20
#define MAXS 20

int main(void) {
    int i, j, redaka, stupaca;
    int min, max;
    int matrica[MAXR][MAXS];

    do {
        scanf("%d %d", &redaka, &stupaca);
    } while (redaka < 1 || redaka > MAXR ||
            stupaca < 1 || stupaca > MAXS);

    for (i = 0; i < redaka; ++i) {
        for (j = 0; j < stupaca; ++j)
            scanf("%d", &matrica[i][j]);
    }

    min = max = matrica[0][0];
    for (i = 0; i < redaka; ++i) {
        for (j = 0; j < stupaca; ++j) {
            if (matrica[i][j] > max)
                max = matrica[i][j];
            else if (matrica[i][j] < min)
                min = matrica[i][j];
        }
    }

    printf("Najmanji: %d\nNajveci: %d\n", min, max);
    return 0;
}
```

Zadatak #7

Učitati veličinu kvadratne matrice (neće biti veća od 20), a zatim elemente matrice. Ispisati sumu elemenata na sporednoj dijagonali i umnožak elemenata na donjoj polovici glavne dijagonale.

00	01	02	03	04	05	
10	11	12	13	14	15	Sporedna dijagonala
20	21	22	23	24	25	
30	31	32	33	34	35	Donja polovica glavne dijagonale
40	41	42	43	44	45	
50	51	52	53	54	55	

Rješenje

```
#include <stdio.h>

#define MAXD 20

int main(void) {
    int i, j, n;
    int sumaSporedne, umnozakPolaGlavne;
    int matrica[MAXD][MAXD];

    scanf("%d", &n);
    for (i = 0; i < n; ++i) {
        for (j = 0; j < n; ++j)
            scanf("%d", &matrica[i][j]);
    }

    umnozakPolaGlavne = 1;
    sumaSporedne = 0;

    for (i = 0, j = n - 1; i < n && j >= 0; ++i, --j)
        sumaSporedne += matrica[i][j];

    /* Alternativni način:
    for (i = 0; i < n; ++i)
        sumaSporedne += matrica[i][n - i - 1];
    */

    for (i = n / 2; i < n; ++i)
        umnozakPolaGlavne *= matrica[i][i];

    printf("Suma: %d\nUmnozak: %d\n", sumaSporedne, umnozakPolaGlavne);
    return 0;
}
```

Zadatak #8

Učitati broj redaka i stupaca matrice (neće biti veći od 20), a zatim elemente matrice. Konstruirati novu matricu na temelju prve tako da je element (r, s) koji se nalazi u retku r i stupcu s jednak umnošku elemenata koji se nalaze „desno“ od pozicije (r, s) u prvoj matrici (uključujući i element na toj poziciji).

Prva matrica: Druga matrica:

3	5	1	15	5	1
4	2	6	48	12	6
2	7	3	42	21	3

Rješenje

```
#include <stdio.h>

#define MAXR 20
#define MAXS 20

int main(void) {
    int i, j, k, redaka, stupaca;
    int prva[MAXR][MAXS], druga[MAXR][MAXS];

    scanf("%d %d", &redaka, &stupaca);
    for (i = 0; i < redaka; ++i) {
        for (j = 0; j < stupaca; ++j)
            scanf("%d", &prva[i][j]);
    }

    for (i = 0; i < redaka; ++i) {
        for (j = 0; j < stupaca; ++j) {
            druga[i][j] = 1;
            for (k = j; k < stupaca; ++k)
                druga[i][j] *= prva[i][k];
        }
    }

    for (i = 0; i < redaka; ++i) {
        for (j = 0; j < stupaca; ++j) {
            printf("%d ", druga[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```