

FUNDAMENTELE PROGRAMĂRII

1. Implementați următoarele funcții:

- funcție care crește valoarea unui număr întreg x cu un număr întreg y ;
- funcție care elimină ultima cifră dintr-un număr natural n ;
- funcție care, folosind apeluri utile ale celor două funcții definite anterior, calculează suma cifrelor unui număr natural n ;
- funcție care furnizează cifra de control ("numărul destinului" dacă numărul este o dată de naștere) a unui număr natural n (se calculează, în mod repetat, suma cifrelor sumei cifrelor numărului n până când se obține o singură cifră).

Folosind funcția definită la punctul d), scrieți un program care afișează numerele naturale cuprinse între două numere naturale a și b cu proprietatea că au cifra de control egală cu o cifră dată c .

Rezolvare:

$n = 29111997 \Rightarrow sc = 2+9+1+1+1+9+9+7 = 39 \Rightarrow sc = 3 + 9 = 12 \Rightarrow sc = 1 + 2 = 3$
 \Rightarrow cifra de control a numărului $n = 29111997$ este 3

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

```
void incrementare(unsigned int *x, unsigned int y)
{
    *x += y; // *x = *x + y
}
```

```
void eliminare_ultima_cifra (unsigned long long int *n)
{
    *n /= 10; // *n = *n / 10;
}
```

```

unsigned int suma_cifre (unsigned long long int n)
{
    unsigned int suma = 0;

    while (n)
    {
        incrementare(&suma, n%10); // suma = suma + n%10;
        eliminare_ultima_cifra(&n); // n = n / 10;
    }
    return suma;
}

unsigned int cifra_control (unsigned long long int n)
{
    unsigned int cif = suma_cifre(n);

    while(cif > 9)
        cif = suma_cifre(cif);

    return cif;
}

int main()
{
    unsigned long long int a, b, i;
    unsigned int c;
    printf("a (marginea inferioara) = ");
    scanf("%I64u", &a);
    printf("b (marginea superioara) = ");
    scanf("%I64u", &b);
    printf("c (\"numarul destinului\") = ");
    scanf("%u", &c);

    printf("Numerele care au %u ca cifra de control sunt:\n ", c);
    for(i = a; i <= b; i++)
        if(cifra_control(i) == c)
            printf("%I64u ", i);
    return 0;
}

```

2. Implementați următoarele funcții:

- a) funcție care calculează răsturnatul/oglinditul unui număr natural n (răsturnatul numărului 21753 este 35712, iar răsturnatul numărului 1000 este 1);

b) funcție care elimină toate cifrele egale cu o cifră c dintr-un număr natural n (dacă $n = 2122753$ și $c=2 \Rightarrow n = 1753$);

Folosind funcția definită la punctul b), scrieți un program care afișează numerele naturale cuprinse între două numere naturale a și b cu proprietatea că:

- nu conțin o cifră c dată;
- sunt formate din cifre egale;
- sunt formate din exact două cifre distincte.

Rezolvare:

Numărul n	Răsturnatul r al lui n
21753	0
2175	$0 * 10 + 3 = 3$
217	$3 * 10 + 5 = 35$
21	$35 * 10 + 7 = 357$
2	$357 * 10 + 1 = 3571$
0	$3571 * 10 + 2 = 35712$

Numărul n	Variabila aux
2122753	0
212275	$0 * 10 + 3 = 3$
21227	$3 * 10 + 5 = 35$
2122	$35 * 10 + 7 = 357$
212	357
21	357
2	$357 * 10 + 1 = 3571$
0	3571

Acum trebuie să calculez răsturnatul lui aux = 3571.

Numărul n	Variabila aux
2120027500022	0
2120027500022	Eliminăm cifrele egale cu c de la sfârșitul lui n
21200275000	Eliminăm cifrele egale cu 0 de la sfârșitul lui n și le numărăm
21200275	Eliminăm cifrele egale cu c din n
57001	aux

10075	Răsturnăm aux
10075000	Adăugăm la sfârșit cele 3 zerouri eliminate