

Disciplina: Programarea Calculatorului

Probleme cu Funcții. Lucru individual

A efectuat student: _____ (Oprea Danu, gr. IA-203)

A controlat: _____ (lect.univ. Guțu Maria)

Probleme cu Funcții

1. De la tastatură se introduc patru numere a, b, c, d. Scrieți un program C ce va determina:

- a. Numărul mai mare dintre ele;
- b. Numărul mai mic dintre ele;
- c. Cel mai mare divizor comun al lor;
- d. Cel mai mic multiplu comun al lor;
- e. Dacă unul dintre numere este divizor pentru toate celelalte;
- f. Dacă unul dintre ele este multiplu al celorlalte numere.

Utilizați funcții pentru fiecare dintre cazuri.

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

```
#include <math.h>
```

```
int MAX(int a, int b, int c, int d){
```

```
    int max, i;
```

```
    int A[4] = {a, b, c, d};
```

```
    max = A[0];
```

```
    for( i = 0; i < 4; i++){
```

```
        if (max < A[i]){
```

```
            max = A[i];
```

```
        }
```

```
    }
```

```
    return max;
```

```
}
```

```
int MIN(int a, int b, int c, int d){
```

```
    int min, i;
```

```
    int A[4] = {a, b, c, d};
```

```
    min = A[i];
```

```

for( i = 0; i < 4; i++){
    if (min > A[i]){
        min = A[i];
    }
}
return min;
}

```

```

int DIVIZ(int a, int b, int c, int d){
    int div, i;

    for (i = 1; i <= a && i <= b && i <= c && i <= d; i++){
        if(a % i == 0 && b % i == 0 && c % i == 0 && d % i == 0)
            div = i;
    }

    return div;
}

```

```

int MULT(int a, int b, int c, int d){
    int div, mult, i;

    for (i = 1; i <= a && i <= b && i <= c && i <= d; i++) {

        if (a % i == 0 && b % i == 0 && c % i == 0 && d % i == 0)
            div = i;
    }

    mult = (a * b * c * d) / div;

    return mult;
}

```

```
}
```

```
int DIV_NUM(int a, int b, int c, int d){
```

```
    int div, i;
```

```
    for (i = 1; i <= a && i <= b && i <= c && i <= d; i++){
```

```
        if(a % i == 0 && b % i == 0 && c % i == 0 && d % i == 0)
```

```
            div = i;
```

```
            if (i == a || i == b || i == c || i == d){
```

```
                return 1;
```

```
            } else {
```

```
                return 0;
```

```
            }
```

```
    }
```

```
}
```

```
int MULT_NUM(int a, int b, int c, int d){
```

```
    return 0;
```

```
}
```

```
int main(){
```

```
    int a, b, c, d;
```

```
    printf("a = ");
```

```
    scanf("%d", &a);
```

```
    printf("b = ");
```

```
    scanf("%d", &b);
```

```
    printf("c = ");
```

```
    scanf("%d", &c);
```

```
    printf("d = ");
```

```
    scanf("%d", &d);
```

```
printf("d) Cel mai mic multiplu comun = %d \n", MULT(a, b, c, d));

printf("a) Max = %d \n", MAX(a, b, c, d));
printf("b) Min = %d \n", MIN(a, b, c, d));
printf("c) Cel mai mare divizor comun = %d \n", DIVIZ(a, b, c, d));

if (DIV_NUM(a, b, c, d) == 0){
    printf("e) Unul dintre numere nu este divizor pentru toate celelalte");
}else {
    printf("e) Unul dintre numere este divizor pentru toate celelalte");
}

return 0;
}
```

2. De la tastatură se introduc patru numere naturale a, b, c, d. Scrieți un program C ce va determina:

- a. Suma cifrelor fiecărui număr;
- b. Cifre cea mai mare din componența fiecărui număr;
- c. Numărul divizorilor fiecărui număr dat;
- d. Suma divizorilor fiecărui număr cu valoarea mai mică decât a numărului dat;
- e. Dacă printre numerele date sunt numere perfecte;
- f. Dacă numărul respectiv este prim sau nu.

Utilizați funcții pentru fiecare dintre cazuri.

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

```
#include <math.h>
```

```
int SUM(int a){  
    int t, r, sum = 0;  
  
    t = a;  
    while (t != 0){  
        r = t % 10;  
        sum += r;  
        t = t / 10;  
    }  
    return sum;  
}
```

```
int NR_MAX(int a){  
    int t, r, max;  
  
    t = a;  
    max = a % 10;  
    while (t != 0){  
        r = t % 10;  
        if (max < r)  
            max = r;  
        t = t / 10;  
    }  
}
```

```
        return max;
    }

    int SUM_DIV(int a){
        int div, i, sum = 0;

        for (i = 1; i <= a; i++){
            if(a % i == 0){
                div = i;
                sum += div;
            }
        }

        return sum;
    }
}
```

```
int main(){
    int a, b, c, d;
    printf("a = ");
    scanf("%d", &a);
    printf("b = ");
    scanf("%d", &b);
    printf("c = ");
    scanf("%d", &c);
    printf("d = ");
    scanf("%d", &d);

    printf("Ex2 a\n");
    printf("Suma %d = %d\n", a, SUM(a));
    printf("Suma %d = %d\n", b, SUM(b));
    printf("Suma %d = %d\n", c, SUM(c));
    printf("Suma %d = %d\n", d, SUM(d));
}
```

```
printf("Ex2 b\n");  
  
printf("Max din %d = %d\n", a, NR_MAX(a));  
printf("Max din %d = %d\n", b, NR_MAX(b));  
printf("Max din %d = %d\n", c, NR_MAX(c));  
printf("Max din %d = %d\n", d, NR_MAX(d));  
  
printf("Ex 2 d");  
  
printf("Suma divizorilor mai mici %d = %d\n", a, SUM_DIV(a));  
printf("Suma divizorilor mai mici %d = %d\n", b, SUM_DIV(b));  
printf("Suma divizorilor mai mici %d = %d\n", c, SUM_DIV(c));  
printf("Suma divizorilor mai mici %d = %d\n", d, SUM_DIV(d));  
  
return 0;  
}
```