

# შესავალი დაპროგრამებაში

ლექტორი:

სალომე ონიანი

ტელ. 571 39 40 22

ელ.-ფოსტა [salome.oniani@iliauni.edu.ge](mailto:salome.oniani@iliauni.edu.ge)



**if(ლოგიკური პირობა){**

**შესასრულებელი მოქმედებები**

**}**

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

```
int main(){  
printf("Please, enter your birth date with format dd/mm/yyyy ");  
int day, month, year;  
scanf("%d/%d/%d", &day, &month, &year);  
  
if(day>0 && day<=31 && month>0 && month<13 ){  
    printf("Thank you, your data is saved... ");  
}  
  
return 0;  
}
```



**if(ლოგიკური პირობა){**

შესასრულებელი მოქმედებები 1

**} else {**

შესასრულებელი მოქმედებები 2

**}**

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

```
int main(){
```

```
printf("Please, enter your birth date with format dd/mm/yyyy ");
```

```
int day, month, year;
```

```
scanf("%d/%d/%d", &day, &month, &year);
```

```
if(day>0 && day<31 && month>0 && month<13 ){
```

```
    printf("Thank you, your data is saved... ");
```

```
} else{
```

```
    printf("Sorry, your data isn't correct, please run the program again... ");
```

```
}
```

```
return 0;
```

```
}
```



## ლოგიკური პირობა? შესასრულებელი მოქმედებები 1 : შესასრულებელი მოქმედებები 2

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

```
int main(){  
    printf("Please, enter your birth date with format dd/mm/yyyy ");  
    int day, month, year;  
    scanf("%d/%d/%d", &day, &month, &year);  
  
    year%4 == 0? printf("February is with 29"):printf("February is with 28");  
  
    return 0;  
}
```



დავწეროთ ეროვნული გამოცდების დაფინანსების პროგრამა. პროგრამა უნდა მოიცავდეს შემდეგ ფუნქციებს:

1. პროგრამაში ოპერატორი შეიტანს სტუდენტის ჯამურ ქულას;
2. თუ ჯამური ქულა მეტია 350-ზე მაშინ სტუდენტი მიიღებს 100% დაფინანსებას;
3. თუ ჯამური ქულა მეტია 300-ზე მაშინ სტუდენტი მიიღებს 70% დაფინანსებას;
4. თუ ჯამური ქულა მეტია 250-ზე მაშინ სტუდენტი მიიღებს 50% დაფინანსებას;
5. თუ ჯამური ქულა მეტია 220-ზე მაშინ სტუდენტი მიიღებს 30% დაფინანსებას;
6. თუ ჯამური ქულა ნალები ან ტოლია 220-ის მაშინ სტუდენტი დაფინანსებას ვერ მიიღებს.
7. წელიწადში სწავლის საფასურია 2250 ლარი;
8. პროგრამა ოპერატორს გამოუთვლის ოთხი წლის მანძილზე სტუდენტის გადასახდელ თანხას

# სასწავლო კვირა IV

- ციკლური პროცესები
- ციკლის ჩაწერა ბლოკსქემებით
- ციკლის ოპერატორი while და მისი შედარება if ოპერატორთან
- ციკლის ოპერატორი do while. მისი შედარება while ოპერატორთან
- შემთხვევები, როდესაც შესაძლებელია while ოპერატორის ნაცვლად გამოყენებულ იქნას do while ოპერატორი

# მონაცემთა ტიპები

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

```
int main(){
```

```
short int n1 = -3;
```

```
unsigned short int n2 = 20;
```

```
unsigned int n3 = 25;
```

```
long int n4 = 390;
```

```
unsigned long int n5 = 39023;
```

```
long long int n6 = 212331;
```

```
unsigned long long int n7 = 12345698789;
```

```
printf("short int %hd --> unsigned short int %hu --> unsigned int %u --  
> long int %ld --> undigned long int %lu --> long long int %lld --  
> unsigned long long int %llu", n1, n2, n3, n4, n5, n6, n7);
```

```
return 0;
```

```
}
```

მონაცემთა ტიპი	ფორმატი
short int	%hd
unsigned short int	%hu
unsigned int	%u
long int	%ld
unsigned long int	%lu
long long int	%lld
unsigned long long int	%llu

**while**(ლოგიკური პირობა){

შესასრულებელი მოქმედებები

}

ეს არის ოპერატორი, რომელშიც ვიმეორებთ ერთიდაიგივე მოქმედებებს მანამ სანამ ლოგიკური პირობა ჭეშმარიტია





დავბეჭდოთ რიცხვები 0-დან 10-მდე



```
#include <stdio.h>

int main(){

int c=0;

while (c<10)
{
    printf("c=%d\n", c);
    c++;
}

return 0;
}
```



დავბეჭდოთ ყველა ორნიშნა 5-ის ჯერადი რიცხვი



```
#include <stdio.h>

int main(){

int c=10;

while (c<100)
{
    printf("c=%d\n", c);
    c +=5;
}

return 0;
}
```



დავბეჭდოთ ყველა სამნიშნა 3-ის ჯერადი რიცხვი



დაწერეთ პროგრამა რომელიც დაბეჭდავს ფიბონაჩის  
მიმდევრობის პირველ 100 რიცხვის



დაწერეთ პროგრამა, რომელიც დაბეჭდავს თუ რამდენ  
ნიშნა რიცხვი შემოიტანა მომხმარებელმა



დაწერეთ პროგრამა რომელიც დაეხმარება მომხმარებელს გაიგოს მის მიერ შეტანილი რიცხვი არის თუ არა არმსტრონგის რიცხვი.

არმსტრონგის რიცხვის მაგალითებია:

9 არის არმსტრონგის რიცხვი, რადგან  $9^1 = 9$

10 არ არის არმსტრონგის რიცხვი, რადგან  $1^2 + 0^2 = 1$

153 არის არმსტრონგის რიცხვი, რადგან  $1^3 + 5^3 + 3^3 = 1 + 125 + 27 = 153$

154 არ არის არმსტრონგის რიცხვი, რადგან  $1^3 + 5^3 + 4^3 = 1 + 125 + 64 = 190$

## არმსტრონგის რიცხვები



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

int main(){

    int number;
    printf("Enter the number ");
    scanf("%d", &number);
    int m1 = number, m2 = number;
    double sum = 0;
    int c = 0;

    while (m1>0)
    {
        m1 /=10;
        c++;
    }

    while (m2>0)
    {
        sum += pow (m2%10, c);
        m2 /=10;
    }

    if(number == sum) printf("It is Armstrong Number");
    else printf("It is not Armstrong Number");

    return 0;
}
```

# pow ();

ნებისმიერი რიცხვის ნებისმიერ ხარისხში აყვანა

pow ( ) ფუნქციის გამოსაყენებლად აუცილებელია #include<math.h>

```
double pow(double val1, double val2);
```

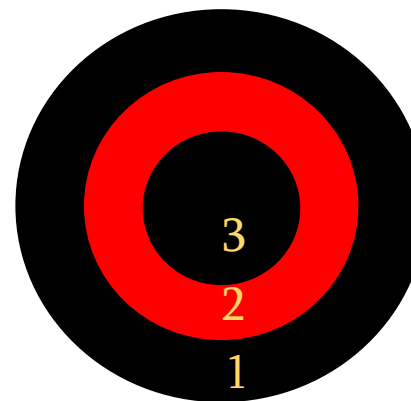
```
#include<stdio.h>
#include<math.h>
int main() {
    double x = 5.5;
    double y = 4.0;
    double p;
    p = pow(x, y);
    printf("The value : %lf", p);
    return 0;
}
```





დაწერეთ დართის ქულების გამოსათვლელი პროგრამა, სადაც მოქმედებს  
შედეგი წესები:

- თუ მოთამაშემ ააცილა 0 ქულა;
- თუ ისარი მოხვდა პირველ წრეში 1 ქულა;
- თუ ისარი მოხვდა მეორე წრეში 5 ქულა;
- თუ ისარი მოხვდა მესამე წრეში 10 ქულა;



- ✓ პირველი წრის რადიუსია 10
- ✓ მეორე წრის რადიუსია 5
- ✓ მესამე წრის რადიუსია 1
- ✓ ყველა წრის ცენტრი მოტავსებულია  $(0,0)$  კოორდინატზე.

პროგრამამ უნდა წაიკითხოს ისრის მოხვედრის კოორდინატები  $(x,y)$  და  
გამოთვალოს მოთამაშის მიღებული ქულა

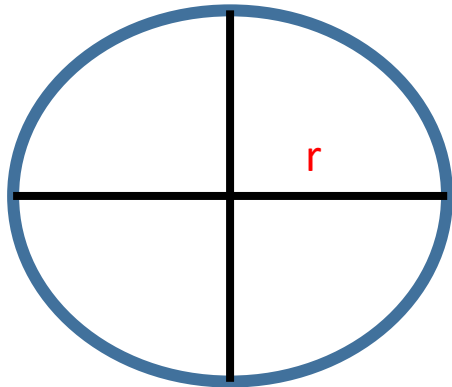


ამოცანა შედგება რამოდენიმე ნაბიჯისაგან

1. გავიგოთ რომელ წრეში მოხვდა ისარი;
2. გამოვთვალოთ ქულა;

ცენტრის კოორდინატები (0,0)

ისრის კოორდინატები (x,y)



იმისათვის რომ გავიგოთ რაიმე წერტილი მდებარეობს წრეში, წრეწირზე ან წრის გარეთ უნდა გამოვთვალოთ მანძილი ამ წერტილიდან წრის ცენტრამდე ფორმულით

$$d = \sqrt{(0 - x)^2 + (0 - y)^2}$$

- თუ  $d = r$  მაშინ წერილი მდებარეობს წრეწირზე
- თუ  $d < r$  მაშინ წერილი მდებარეობს წრეში
- თუ  $d > r$  მაშინ წერილი მდებარეობს წრის გარეთ



1. გამოცანა შედგება რამოდენიმე ნაბიჯისაგან  
გავიგოთ რომელ წრეში მოხვდა ისარი;

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

int main(){

int r1 = 10, r2 = 5, r3 = 1;
printf("Enter point coordinates  ");
int x,y;
scanf("%d %d", &x, &y);

double d = sqrt(x*x +y*y);

if(d > r1) printf("no");
else if(d>r2 && d<=r1) printf("1");
else if(d>r3 && d<=r2) printf("2");
else
{
    printf("3");
}

return 0;
}
```



## ამოცანა შედგება რამოდენიმე ნაბიჯისაგან 2. გამოვთვალოთ ქულა;

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

int main(){

    int r1 = 10, r2 = 5, r3 = 1;
    printf("Enter point coordinates  ");
    int x,y;
    scanf("%d %d", &x, &y);

    double d = sqrt(x*x +y*y);

    if(d > r1) printf("0 pont");
    else if(d>r2 && d<=r1) printf("1 point");
    else if(d>r3 && d<=r2) printf("5 point");
    else
    {
        printf("3 point");
    }

    return 0;
}
```

# sqrt();

ნებისმიერ რიცხვიდან კვადრატული ფესვის ამოღება

sqrt() ფუნქციის გამოსაყენებლად აუცილებელია #include<math.h>

```
double sqrt(double arg);
```

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

```
int main() {
    double number, squareRoot;

    printf("Enter a number: ");
    scanf("%lf", &number);
    squareRoot = sqrt(number);

    printf("Square root of %.2lf = %.2lf", number, squareRoot);

    return 0;
}
```

Do {

შესასრულებელი მოქმედებები

} while(ლოგიკური პირობა)

ეს არის ოპერატორი, რომელშიც ვიმეორებთ ერთიდაიგივე მოქმედებებს მანამ სანამ ლოგიკური პირობა ჭეშმარიტია



## დავბეჭდოთ რიცხვები 0-დან 10-მდე



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

int main(){

int c=0;
do
{
    printf("c = %d\n",c);
    c++;
} while (c<10);

return 0;
}
```

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

```
int main(){
```

```
int c=11;
```

```
do
```

```
{
```

```
    c++;
```

```
    printf("c = %d\n",c);
```

```
} while (c<10);
```

```
printf("Difference");
```

```
c = 11;
```

```
while (c<10)
```

```
{
```

```
    c++;
```

```
    printf("c = %d\n",c);
```

```
}
```

```
return 0;
```

```
}
```

do while და while, მათ შორის განსხვავება





დაწერეთ პროგრამა, რომელიც მომხმარებელს სთხოვს ორი რიცხვის შემოტანას, რის შემდეგაც პროგრამა პირველ რიცხვს აიყვანს მეორე რიცხვის ტოლ ხარისხში და დაბეჭდავს მიღებულ შედეგს.



დაწერეთ პროგრამა, სადაც მომხმარებელი შეიტანს ნებისმიერ მთელ რიცხვს  
და პროგრამა დაბეჭდავს მომხმარებლის მიერ შეტანილი რიცხვის ყველა  
გამოყოფს



დაწერეთ პროგრამა, რომელიც დაბეჭდავს 0-დან მომხმარებლის მიერ  
შეტანილ რიცხვამდე ყველა ხუთის ჯერად რიცხვს



### 3 ქულიანი დავალება

დავალების შესრულების ვადაა 24/10/2020 20:00 საათი.

ამოცანა 4

დაწერეთ პროგრამა, რომელიც მომხმარებელს დაეხმარება ნებისმიერ ნიშნა რიცხვის თანრიგებად დაშლაში. ანუ მომხმარებელი შემოიტანს რაიმე მთელ რიცხვს და პროგრამამ უნდა დაბეჭდოს ამ რიცხვის თანრიგებად დაშლის გამოსახულება

შეფასების კრიტერიუმები და დავალების გამოგზავნის ინსტრუქცია იხილეთ შემდეგ გვერდზე



## 3 ქულიანი დავალება

დავალების შესრულების ვადაა  
24/10/2020 20:00 საათი.

### შეფასება

1. პროგრამის მენიუ გასაგებია და ოპერატორი მარტივად ხვდება პროგრამაში შესატან ინფორმაციას, კომპილატორის შეცდომები არაა - 1 ქულა
2. პროგრამის შედეგი გათვლილია ნებისმიერ ნიშნა რიცხვზე - 1 ქულა
3. პროგრამას გამოაქვს შეტანილი რიცხვის თანრიგებად დაშლის გამოსახულება - 1 ქულა

შესრულებული დავალების პროგრამული კოდი დააკოპირეთ ვორდის ფაილში (გამოიყენეთ 3 ქულიანი დავალების წარდგენის ფორმა) და ატვირთეთ Elearning.iliauni.edu.ge-ზე დავალების მიზმის ველში.

გმადლობთ ყურადღებისთვის!