

# Дэд ба рекурсив алгоритм

Б.Наранчимэг

Мэдээлэл, компьютерийн ухааны тэнхим

ХШУИС, МУИС

[naranchimeg@seas.num.edu.mn](mailto:naranchimeg@seas.num.edu.mn)

# Өмнөх хичээлээр

- Мөр хэмжигдэхүүн (string)
- Утга нь тэмдэгтүүдийн дараалал байдаг хэмжигдэхүүнийг текстэн буюу мөр хэмжигдэхүүн гэнэ.
- Мөр хэмжигдэхүүнийн бодлогууд
  - Тэмдэгт дарааллын утгыг өөрчлөх
  - Тэмдэгт дараалал палиндром мөн эсэхийг шалгах
  - Тэмдэгт дарааллаас үсэг, үг, өгүүлбэр хайх

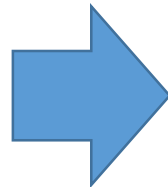
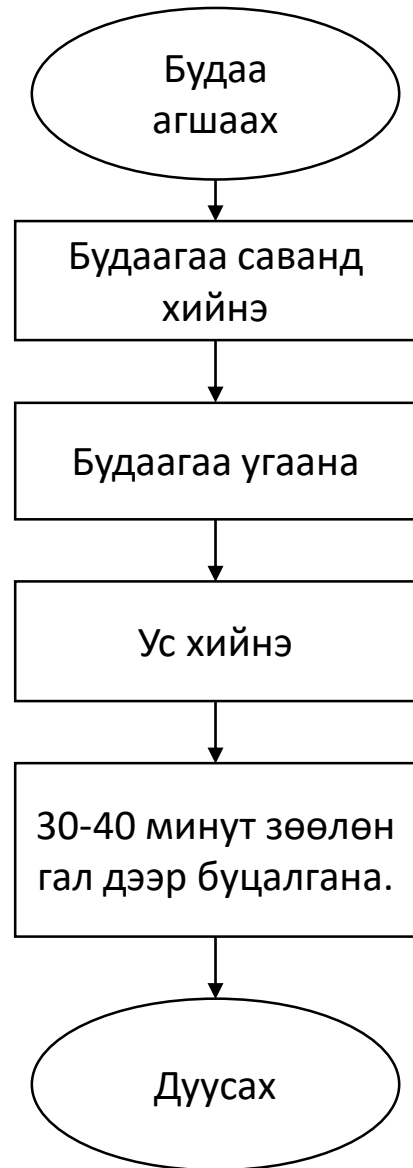
# Дээрээс-доош алгоритмын зохиомж

- Аливаа нэг асуудлыг жижиг, дэд асуудлуудад хувааж зохиомжлохыг “Top-down зохиомж”, “Бүтэцлэгдсэн зохиомж” гэх мэт нэрлэнэ.
- Зорилго
  - Алдаагүй алгоритм бичих
  - Өмнө нь зохиогоод, шалгагдсан ө.х зөв ажиллаж буй алгоритмыг дахин ашиглах ([reusability](#))
  - Хоорондоо холбоотой дэд-алгоритмуудтай байж болно.

# Дэд алгоритм

- Дэд алгоритм
  - Өөр алгоритмаас дуудахад гүйцэтгэгддэг алгоритмын блок юм.
- Дэд алгоритмыг хэрхэн ашиглах вэ
  - Дэд алгоритмыг тодорхойлж, зохиох, бичих
  - Дэд алгоритмыг дуудаж (CALL) ажиллуулах алгоритм бичих
- Дэд алгоритмыг програмчлалын хэлэнд функц (function) гэнэ

# Дэд алгоритм



# Дэд алгоритм

- Үндсэн алгоритмаас утга дамжих эсэхээс шалтгаалж
  - Параметертэй дэд алгоритм
  - Параметергүй дэд алгоритм
- Үндсэн алгоритмд утга дамжуулах эсэхээс шалтгаалж
  - Утга буцаадаггүй дэд алгоритм
  - Утга буцаадаг дэд алгоритм

# Параметергүй дэд алгоритм

```
SUBALGORITHM subalgorithm-name ( )  
    Statements  
END subalgorithm-name
```

↑  
Хоосон

- ( ) хоосон байна.

# Параметертэй дэд алгоритм

**SUBALGORITHM** subalgorithm-name (*parameter-list*)

Statements

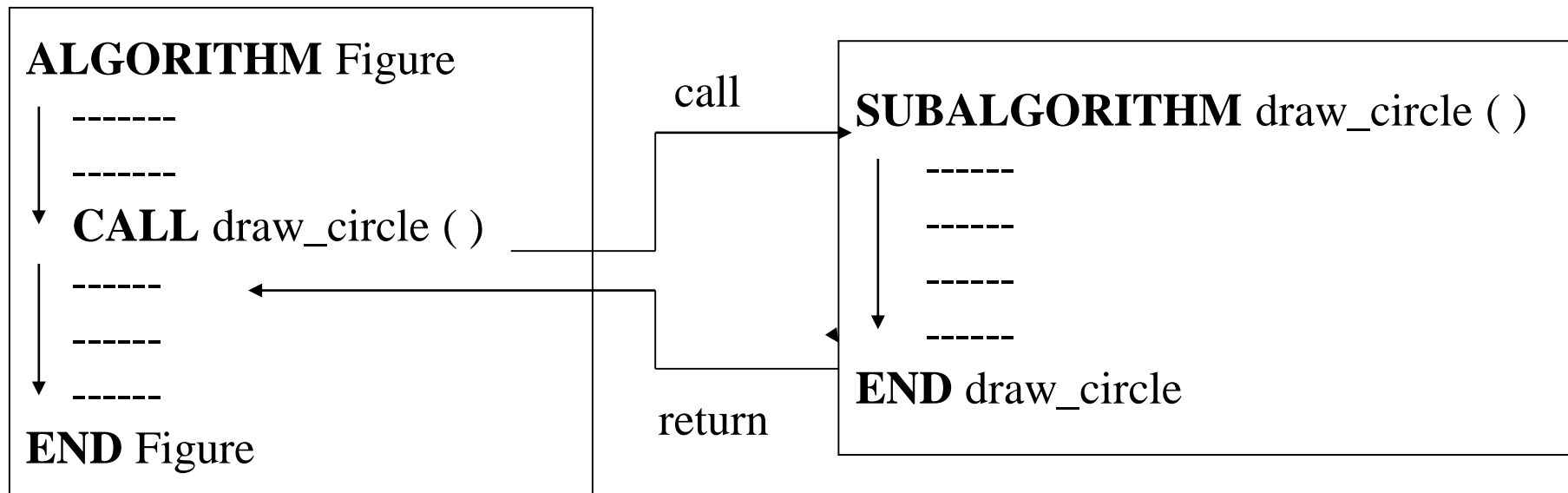
**END** subalgorithm-name



Параметрууд



# Дэд алгоритмыг дуудах



# Жишээ

## Syntax ( in Algorithm ):

```
SubAlgorithm print_box ()  
    output “* * * * * *\n”  
    output “* “, 50 , “ * \n”  
    output “* * * * * *\n”  
End print_box
```

## **Algorithm** Test

```
begin  
    print_box ()  
    output “\n”  
    print_box ()  
End Test
```

## Syntax ( in C++ ):

```
void print_box ()  
{  
    cout << “* * * * * *\n” ;  
    cout << “* “ << 50 << “ * \n” ;  
    cout << “* * * * * *\n” ;  
}
```

```
void main()  
{  
    print_box ();  
    cout<<endl;  
    print_box ();  
}
```

# Жишээ

## Syntax ( in Algorithm ):

```
SubAlgorithm print_box ( n)
    output "*" * 7 * "\n"
    output "*" " , n , " * "\n"
    output "*" * 7 * "\n"
End print_box
```

## **Algorithm** Test

### **Begin**

Output " Enter integer number"

Input a

print\_box (a)

print\_box (5)

### **End** Test

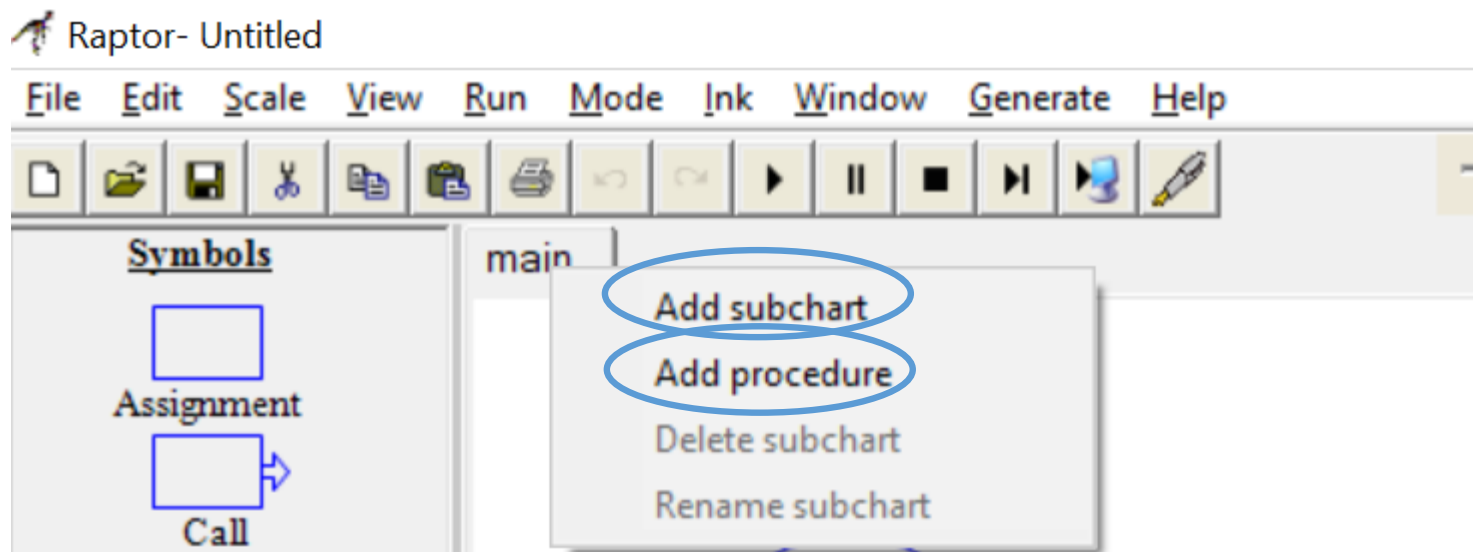
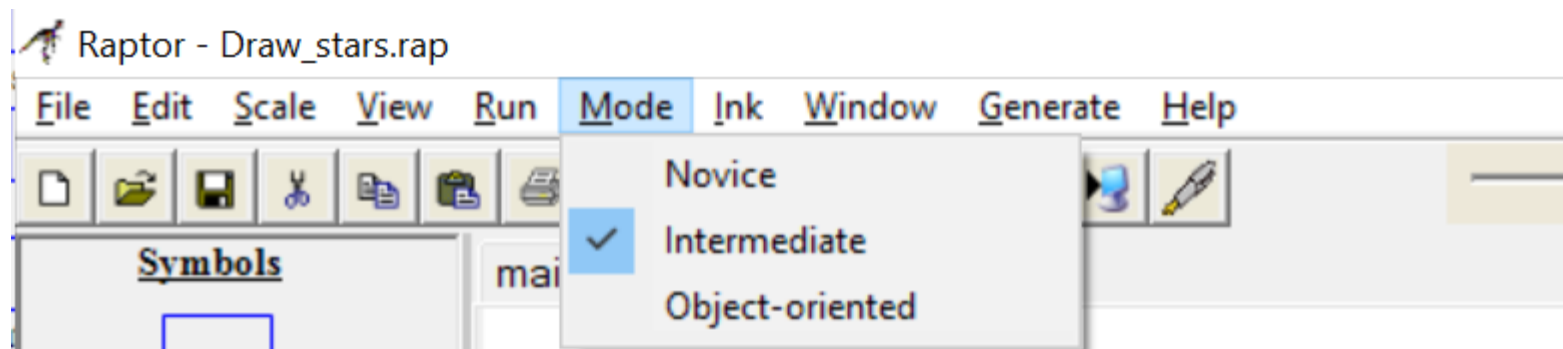
## Syntax ( in C++ ):

```
void print_box (int n)
{
    cout << "*" * 7 * "\n" ;
    cout << "*" " << n << " * "\n" ;
    cout << "*" * 7 * "\n" ;
}
```

## **void** main()

```
{
    int a;
    cout<<"\n Enter integer number";
    cin>>a;
    print_box (a);
    cout<<endl;
    print_box (5);
}
```

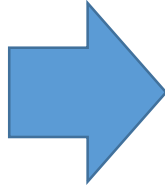
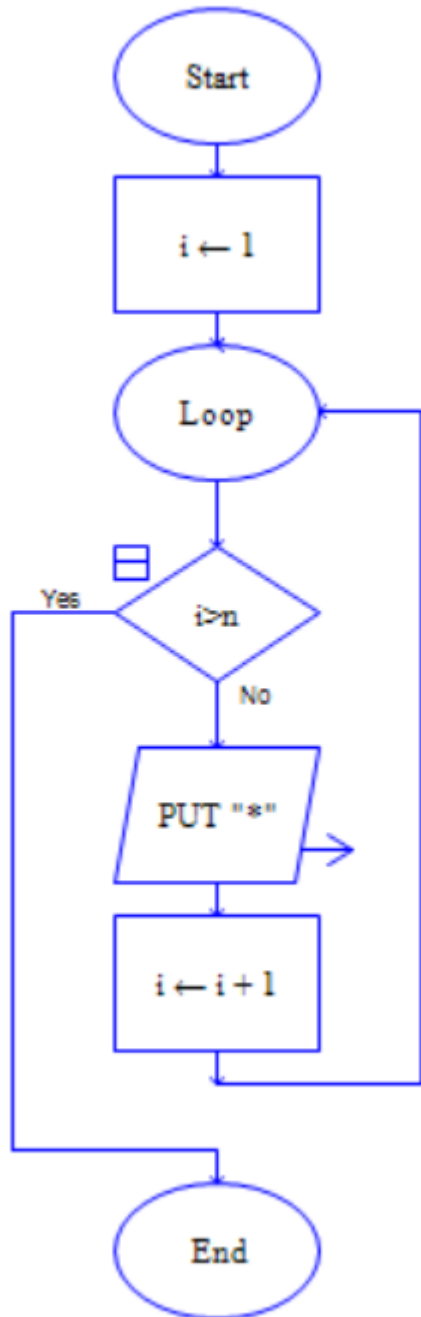
# Дэд алгоритмыг Raptor дээр зохиомжлох



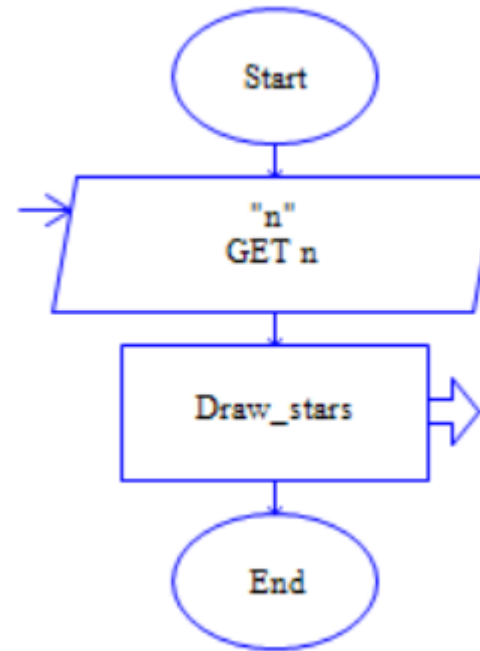
Параметергүй дэд алгоритм

Параметертэй дэд алгоритм

## Draw\_stars



## main



# Утга буцаадаг алгоритм

*f*type **SUBALGORITHM** subalgorithm-name (*parameter-list*)

Statements

Return variable\_name

**END** subalgorithm-name

- Ftype – int, float гэх мэт байна.

# Жишээ

## Syntax ( in Algorithm ):

```
INT SubAlgorithm sum ()  
    x ← 5  
    y ← 7  
    return x+y  
End sum
```

## **Algorithm** Test

### **Begin**

Output “sum=“ ,sum()

a ← sum()

Output “sum=“ , a

### **End** Test

## Syntax ( in C++ ):

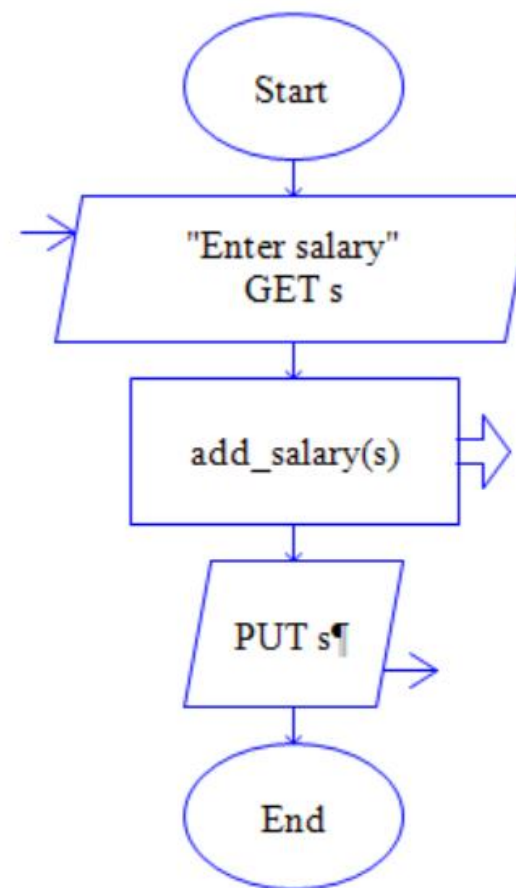
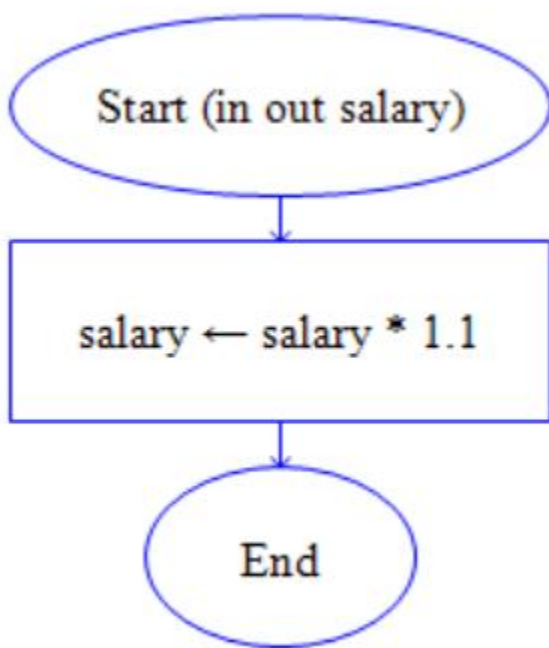
```
int sum ()  
{  
    int x=5, y=7 ;  
    return x+y ;  
}
```

## **void** main()

```
{  
    int a;  
    cout<<“sum=“<<sum()<<endl;  
    a=sum();  
    cout<<“sum=“<<a<<endl;  
}
```

# Дэд алгоритм

add\_salary





# Рекурсив алгоритм

- Өөрийгөө дууддаг алгоритм (дэд програм)-ыг рекурсив гэнэ.
- Өөрөө өөртөө хандсан дэд алгоритмыг рекурсив алгоритм гэнэ.

# Факториалыг рекурсивээр бодох

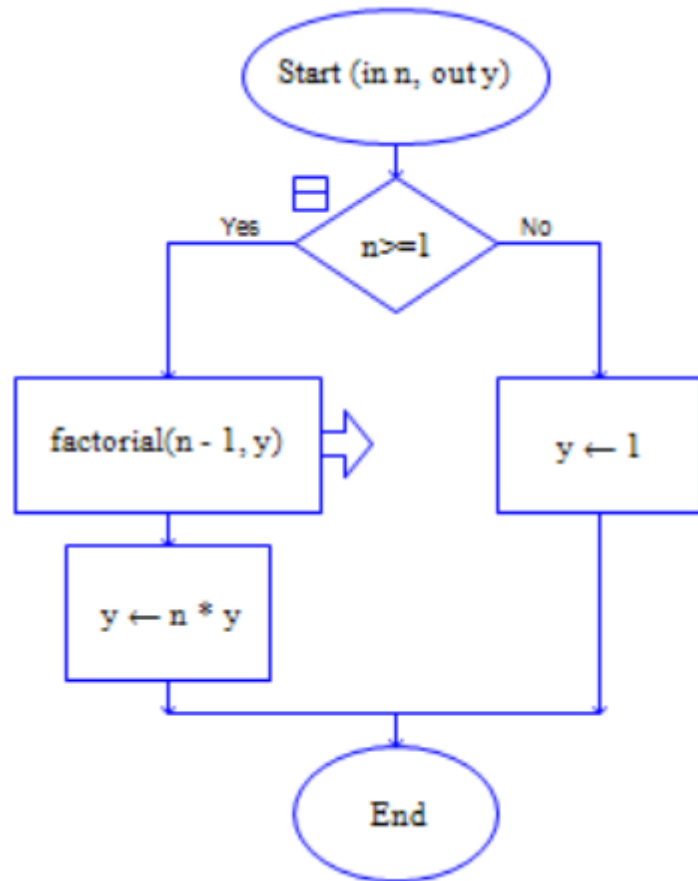
$$\begin{aligned} f(n) &= n! = (n-1)! * n = f(n-1)n \\ f(n-1) &= (n-1)! = (n-2)! * (n-1) = f(n-2)(n-1) \end{aligned}$$

$$f(2) = 2! = \overset{\dots}{1!} * 2 = f(1)2$$

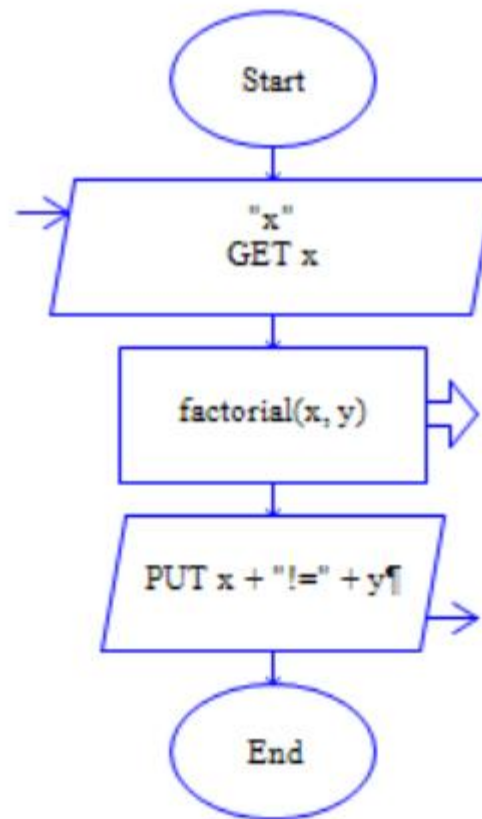
$$\begin{aligned} f(n) &= f(n-1)n \quad (n \geq 2) \\ f(1) &= 1 \end{aligned}$$

# Факториалыг рекурсивээр бодох

factorial



main



```
MasterConsole
Font Font Size Edit Help
5!=120
----Run complete. 34 symbols evaluated.----
```

Асуулт?