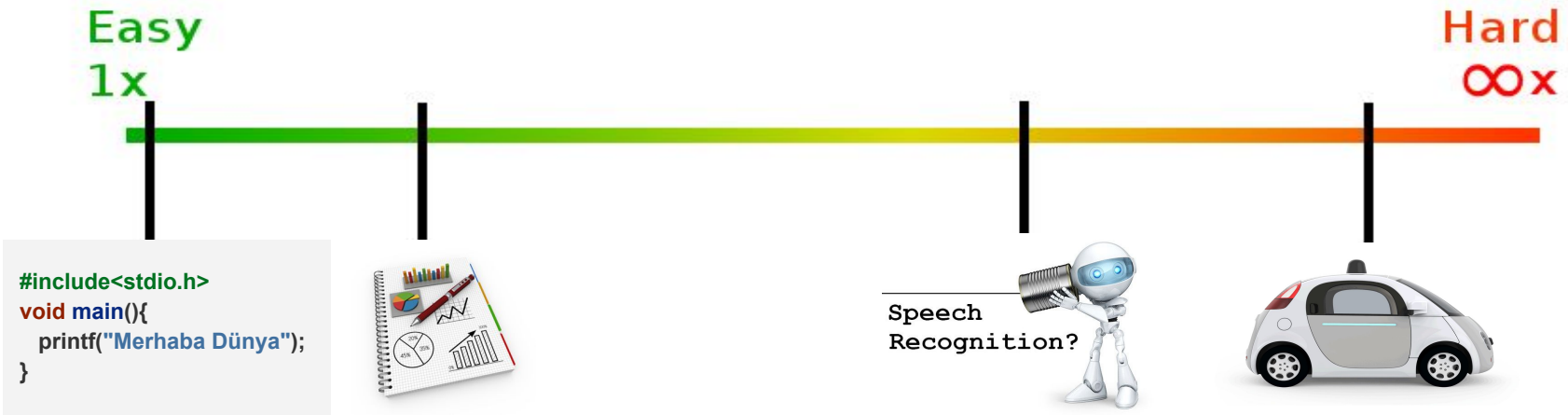
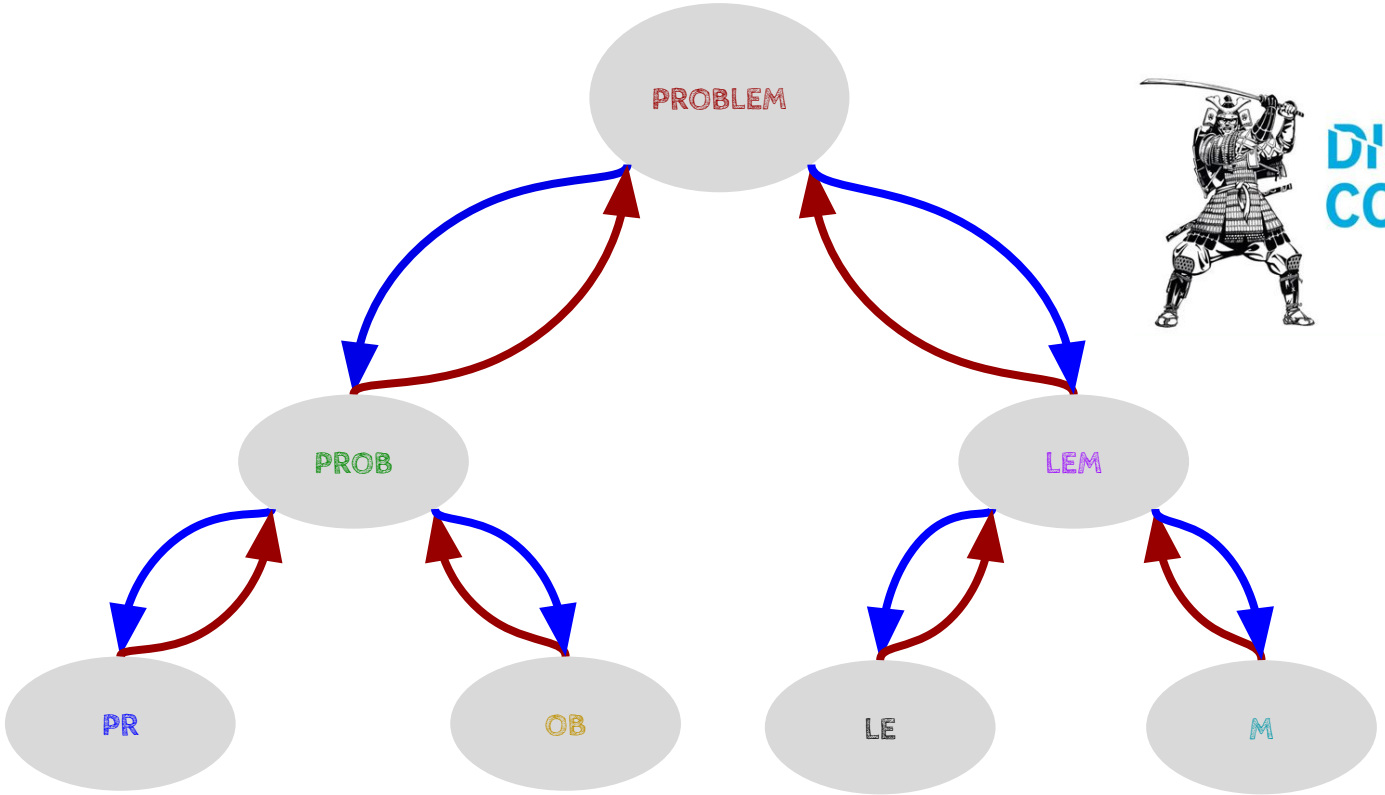
Fonksiyonlar

Suhap SAHIN
Onur GÖK

Fonksiyonlar



Fonksiyonlar

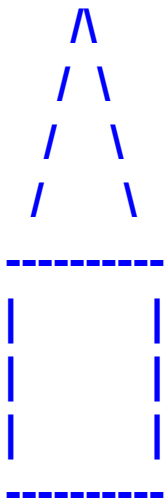


DIVIDE AND CONQUER

Fonksiyonlar

DIVIDE AND CONQUER

```
#include<stdio.h>
int main(        )
{




        return 0;
}
```



Fonksiyonlar

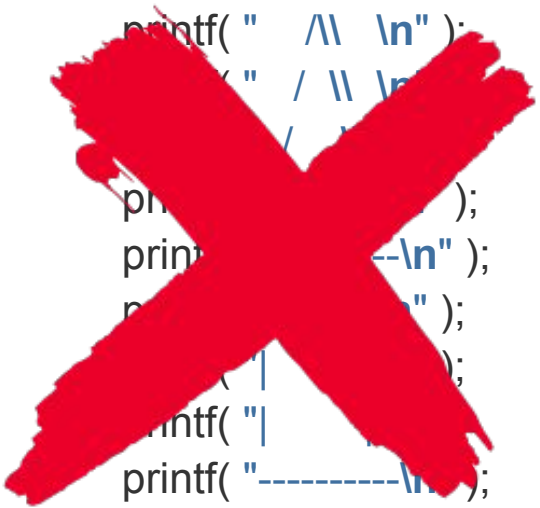

```
#include<stdio.h>
int main(        )
{
        printf( "    /\\   \n" );
        printf( "   /  \\  \n" );
        printf( "  /    \\ \n" );
        printf( " /      \\ \n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );

        return 0;
}
```

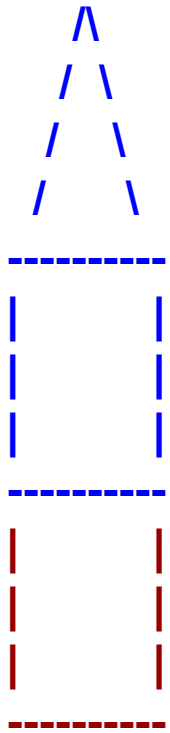
```
    /\
   /  \
  /    \
 /      \
----------
|        |
|        |
|        |
----------
```

Fonksiyonlar

DIVIDE AND CONQUER

```
#include<stdio.h>
int main(        )
{
        printf( "    /\\   \n" );
        printf( "   /  \\  \n" );
        printf( "  /    \\ \n" );
        printf( " ---------- \n" );
        printf( "|          |\n" );
        printf( "|          |\n" );
        printf( "|          |\n" );
        printf( " ---------- \n" );
        printf( "|          |\n" );
        printf( "|          |\n" );
        printf( "|          |\n" );
        printf( "----------\n" );

        return 0;
}
```

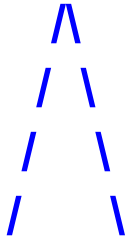




Fonksiyonlar

DIVIDE AND CONQUER

ÇATI



Fonksiyonlar

DIVIDE AND CONQUER

ÇATI

TABAN

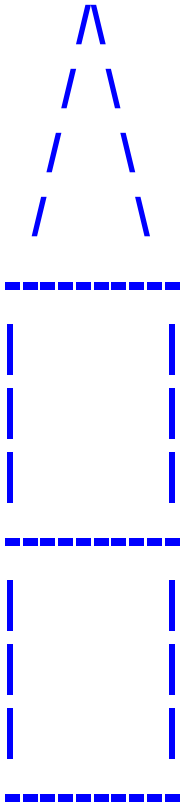
Fonksiyonlar

DIVIDE AND CONQUER

ÇATI

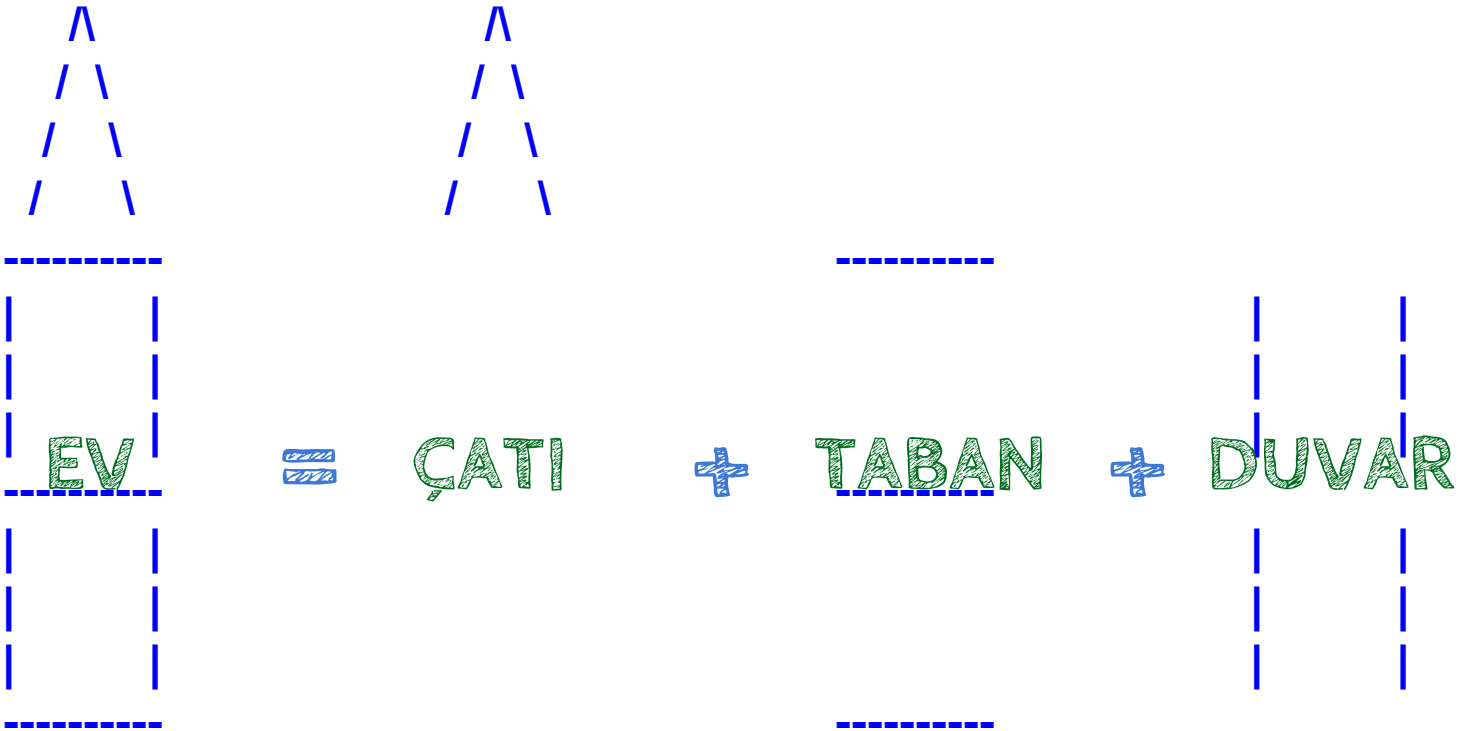
TABAN

DUVAR



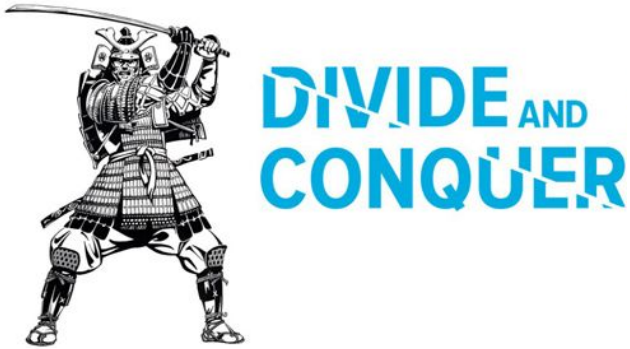
Fonksiyonlar

DIVIDE AND CONQUER



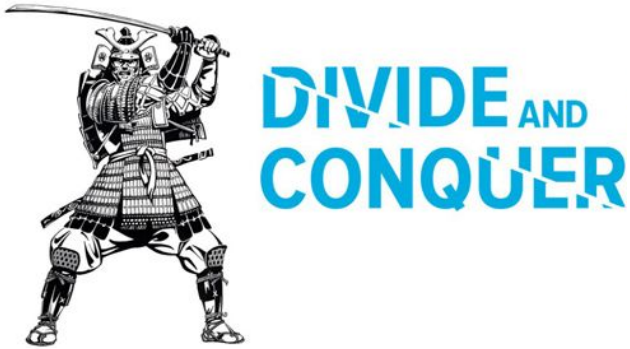
Fonksiyonlar

```
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
```



Fonksiyonlar

```
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
```



Fonksiyonlar

```
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

DIVIDE AND CONQUER

Fonksiyonlar


```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

```c
int main( )
{
    catiyi_ciz( );


return 0;
}
```

Fonksiyonlar


```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

```c
int main( )
{
    catiyi_ciz( );
    taban_ciz( );

return 0;
}
```

```
    /\
   /  \
  /    \
 /      \
----------
```

Fonksiyonlar


```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

```c
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );

return 0;
}
```

```
    /\
   /  \
  /    \
 /      \
----------
|        |
|        |
|        |
```

Fonksiyonlar


```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

```c
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );

return 0;
}
```

```
    /\
   /  \
  /    \
 /      \
----------
|        |
|        |
|        |
----------
```

Fonksiyonlar


```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

```c
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );

return 0;
}
```

```
    /\
   /  \
  /    \
 /      \
----------
|        |
|        |
|        |
----------
|        |
|        |
|        |
```

Fonksiyonlar


```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   /  \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
```

```c
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```

```
    /\
   /  \
  /    \
 /      \
----------
|        |
|        |
|        |
----------
|        |
|        |
|        |
----------
```

Fonksiyonlar

DIVIDE AND CONQUER

```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /   \\  \n" );
    printf( " /     \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );


return 0;
}
```

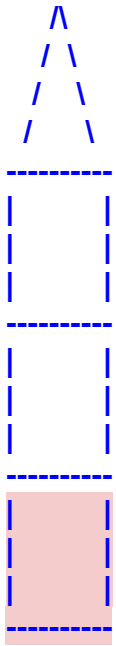
```
    /\
   /  \
  /    \
 /      \
----------
|        |
|        |
|        |
----------
|        |
|        |
|        |
----------
```

```c
#include<stdio.h>
int main( )
{
        printf( "    /\\    \n" );
        printf( "   / \\   \n" );
        printf( "  /   \\  \n" );
        printf( " /     \\ \n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );



        return 0;
}
```

Fonksiyonlar

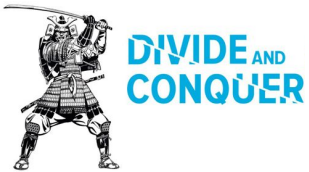
DIVIDE AND CONQUER

```
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /   \\  \n" );
    printf( " /     \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```

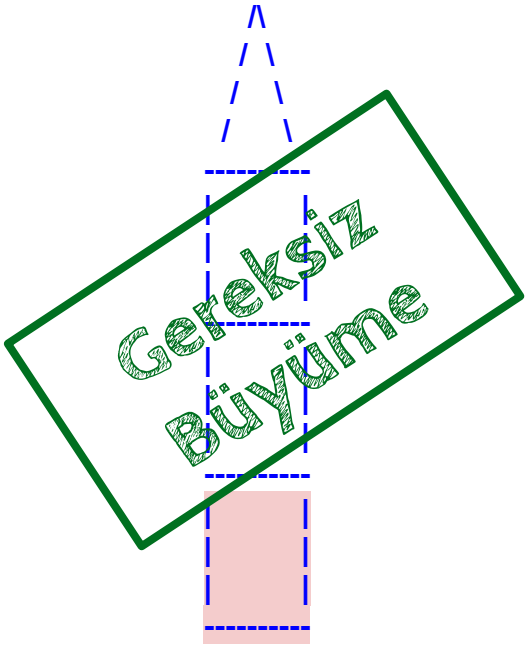


```
#include<stdio.h>
int main( )
{
        printf( "    /\\    \n" );
        printf( "   / \\   \n" );
        printf( "  /   \\  \n" );
        printf( " /     \\ \n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );

        return 0;
}
```

Fonksiyonlar



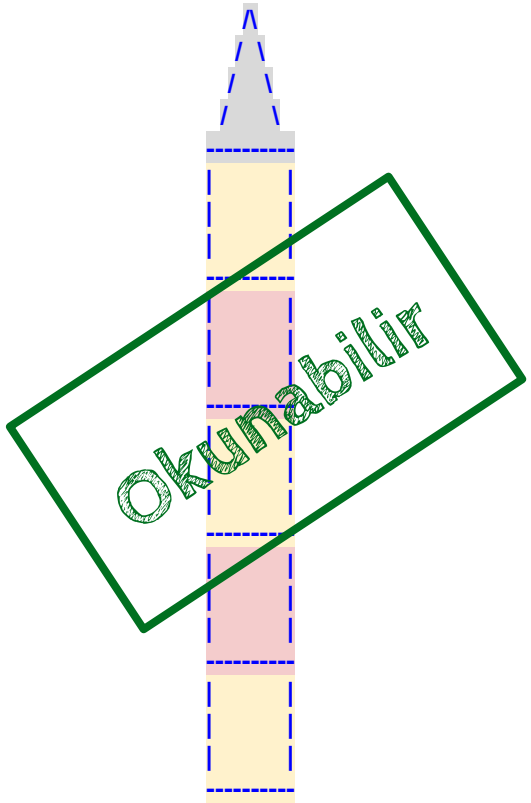

```c
#include<stdio.h>
void catiyi_ciz( )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( )
{
    printf( "----------\n" );
}
int main( )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```




```c
#include<stdio.h>
int main( )
{
        printf( "    /\\    \n" );
        printf( "   / \\   \n" );
        printf( "  /    \\  \n" );
        printf( " /      \\ \n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );

        return 0;
}
```

Fonksiyonlar

```c
#include<stdio.h>
void catiyi_ciz( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /   \\  \n" );
    printf( " /     \\ \n" );
}
void duvar_ciz( void )
{
    printf( "|       |\n" );
    printf( "|       |\n" );
    printf( "|       |\n" );
}
void taban_ciz( void )
{
    printf( "----------\n" );
}
int main( void )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```



DIVIDE AND CONQUER

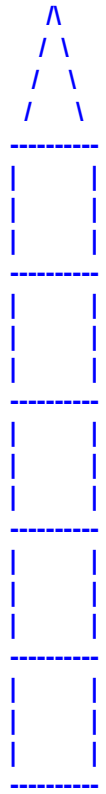
```c
#include<stdio.h>
int main( void )
{
        printf( "    /\\    \n" );
        printf( "   / \\   \n" );
        printf( "  /   \\  \n" );
        printf( " /     \\ \n" );
        printf( "----------\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "----------\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "----------\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "----------\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "----------\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "|       |\n" );
        printf( "----------\n" );
        return 0;
}
```

Fonksiyonlar

```c
#include<stdio.h>
void catiyi_ciz( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( void )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( void )
{
    printf( "----------\n" );
}
int main( void )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```

taban çizerim

fonksiyon çağırırım



her is yapılır

DIVIDE AND CONQUER

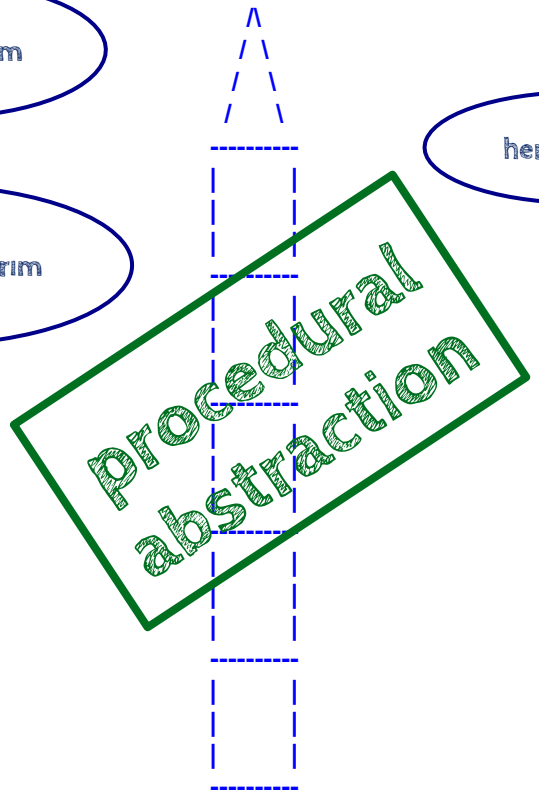
```c
#include<stdio.h>
int main( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    return 0;
}
```

Fonksiyonlar

```c
#include<stdio.h>
void catiyi_ciz( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( void )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( void )
{
    printf( "----------\n" );
}
int main( void )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```

taban çizerim

fonksiyon çagırırım



her is yapılır

DIVIDE AND CONQUER

```c
#include<stdio.h>
int main( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    return 0;
}
```

Fonksiyonlar

```c
#include<stdio.h>
void catiyi_ciz( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( void )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( void )
{
    printf( "----------\n" );
}
int main( void )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```

DÜZELT!!!

Kolay test & hata bulma

DIVIDE AND CONQUER

```c
#include<stdio.h>
int main( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "----------\n" );
    return 0;
}
```

DÜZELT!!!

DÜZELT!!!

DÜZELT!!!

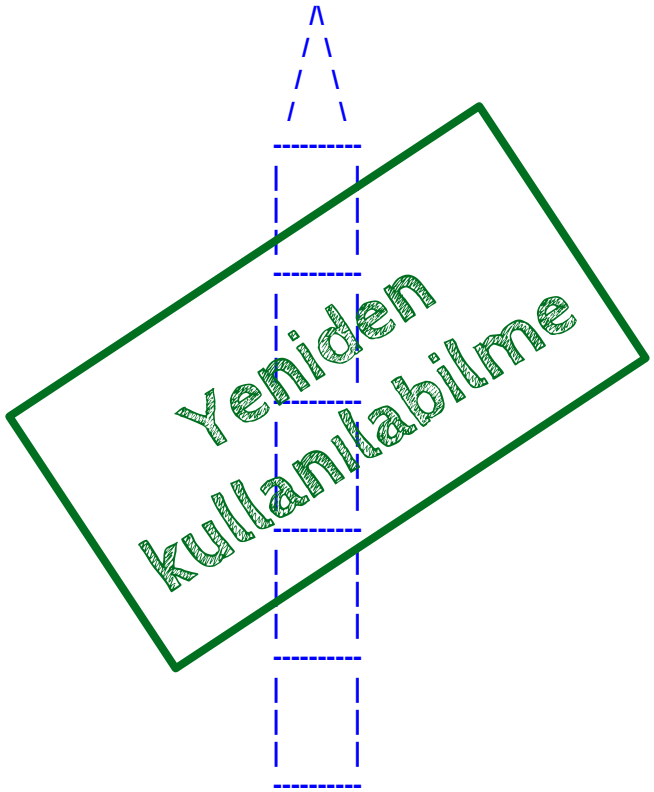
DÜZELT!!!

DÜZELT!!!

Fonksiyonlar

```c
#include<stdio.h>
void catiyi_ciz( void )
{
    printf( "    /\\    \n" );
    printf( "   / \\   \n" );
    printf( "  /    \\  \n" );
    printf( " /      \\ \n" );
}
void duvar_ciz( void )
{
    printf( "|        |\n" );
    printf( "|        |\n" );
    printf( "|        |\n" );
}
void taban_ciz( void )
{
    printf( "----------\n" );
}
int main( void )
{
    catiyi_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
    duvar_ciz( );
    taban_ciz( );
return 0;
}
```

1

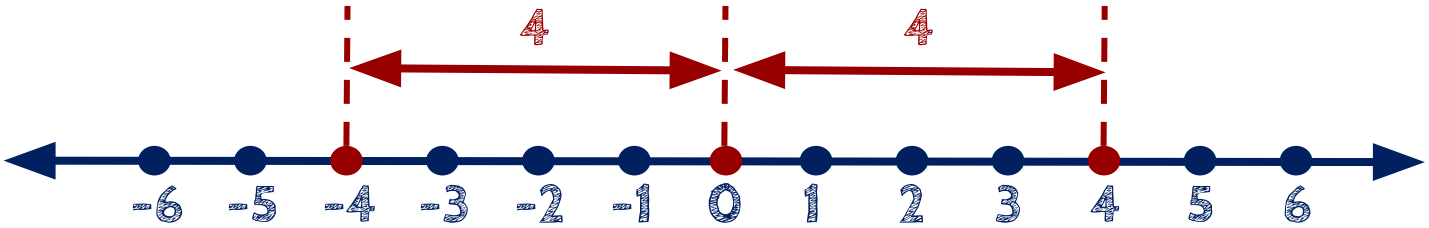


DIVIDE AND CONQUER

```c
#include<stdio.h>
int main( void )
{
        printf( "    /\\    \n" );
        printf( "   / \\   \n" );
        printf( "  /    \\  \n" );
        printf( " /      \\ \n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "|        |\n" );
        printf( "----------\n" );
        return 0;
}
```

1 2 3 4 5

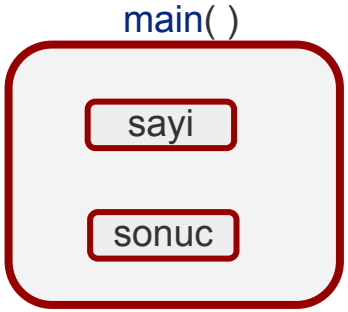
Mutlak Değer



$$|-4| = |4|$$

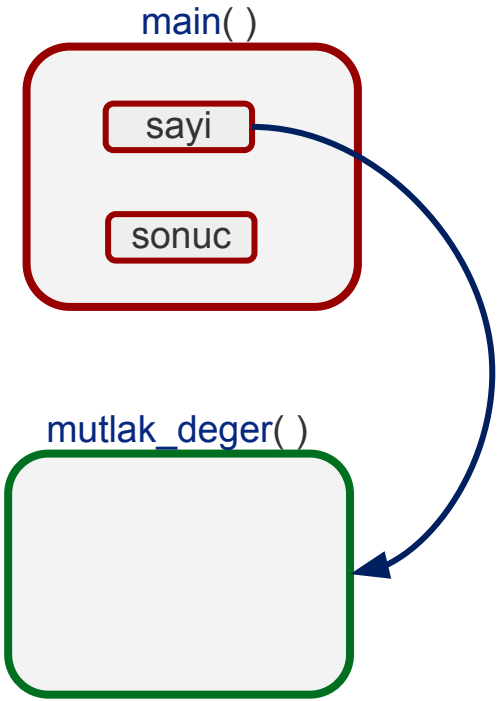
Mutlak Deger

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

int main( ) {
        int sayi, sonuc;
        printf("sayi girin:");
        scanf("%d", &sayi);



        return 0;
}
```



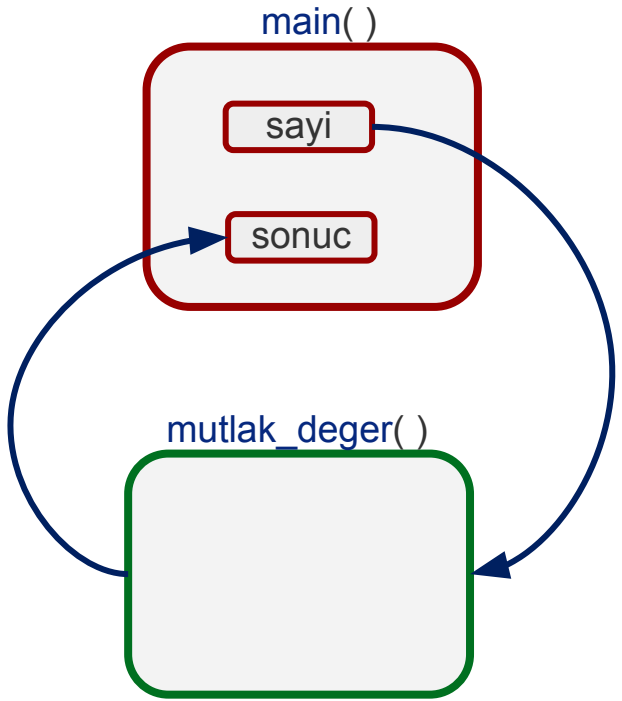
Mutlak Deger

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

int main( ) {
        int sayi, sonuc;
        printf("sayi girin:");
        scanf("%d", &sayi);
                mutlak_deger(sayi);
        printf("mutlak deger: %d\n", sonuc);
        return 0;
}
```



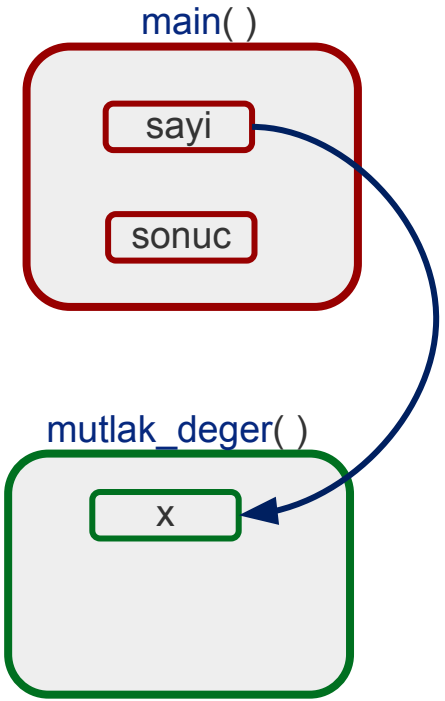
Mutlak Deger

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

int main( ) {
        int sayi, sonuc;
        printf("sayi girin:");
        scanf("%d", &sayi);
        sonuc = mutlak_deger(sayi);
        printf("mutlak deger: %d\n", sonuc);
        return 0;
}
```



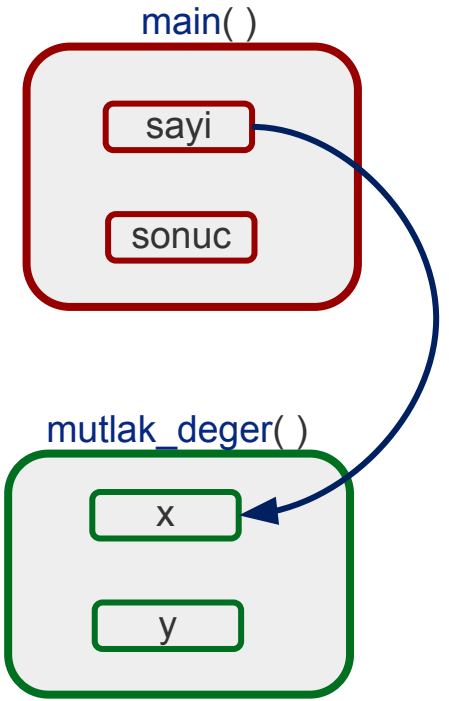
Mutlak Deger

```
int mutlak_deger( int x ) {




}
```



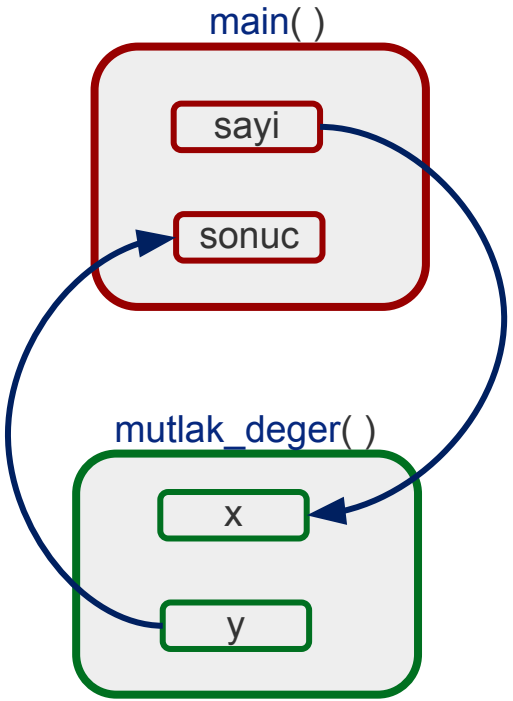
Mutlak Deger

```
int mutlak_deger( int x ) {
        int y;
        if (x >= 0)
        y = x;
        else
        y = -1 * x;

}
```



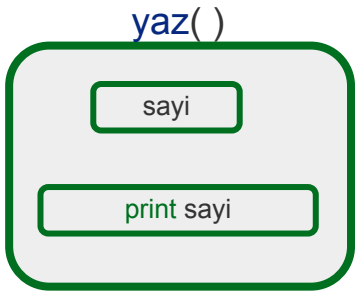
Mutlak Deger

```
int mutlak_deger( int x ) {
        int y;
        if (x >= 0)
        y = x;
        else
        y = -1 * x;
        return y;
}
```



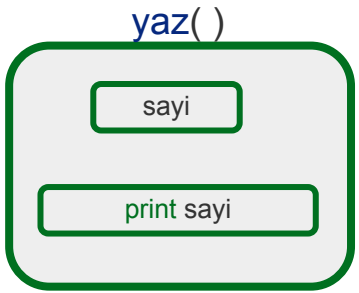
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
```



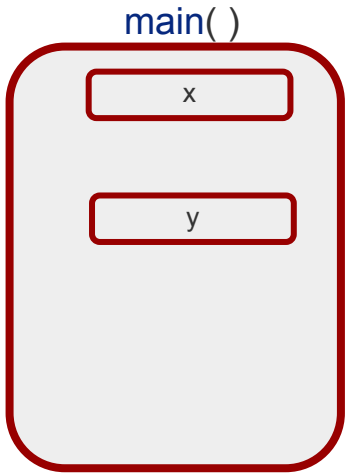
printf & scanf

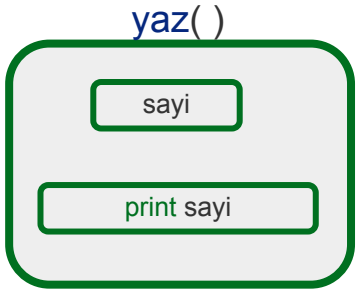
```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
```



printf & scanf

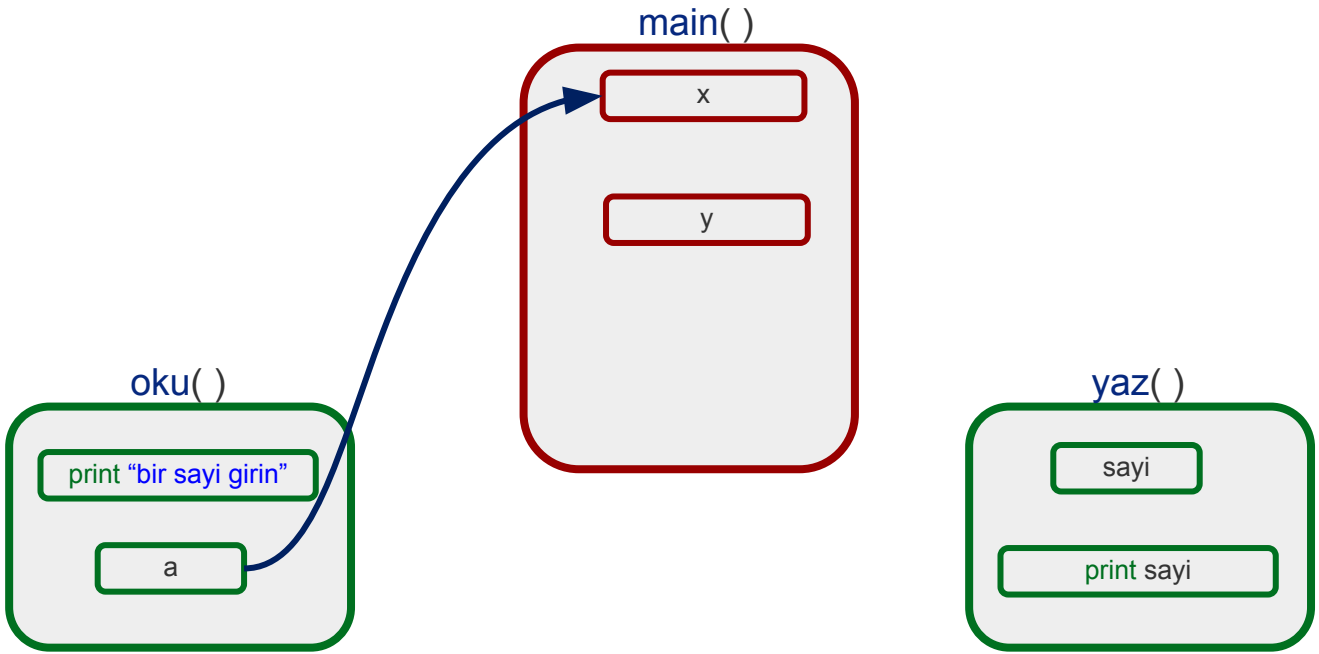
```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;



        return 0;
}
```





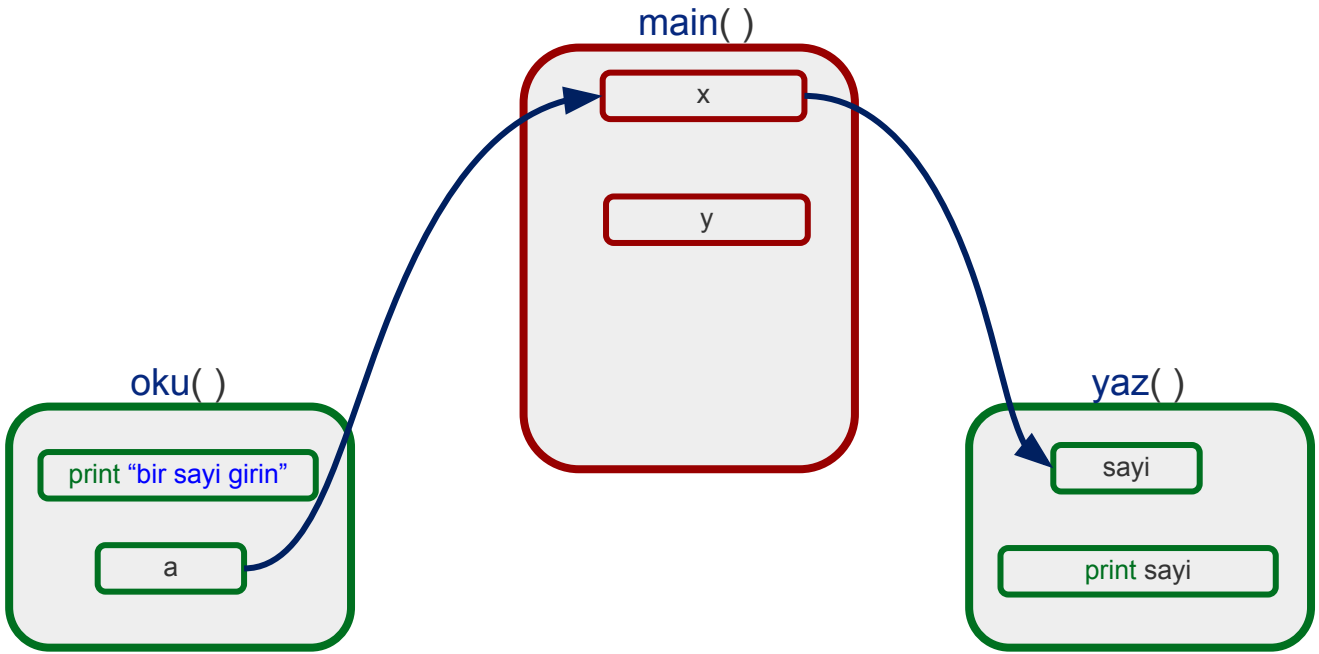
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();



        return 0;
}
```



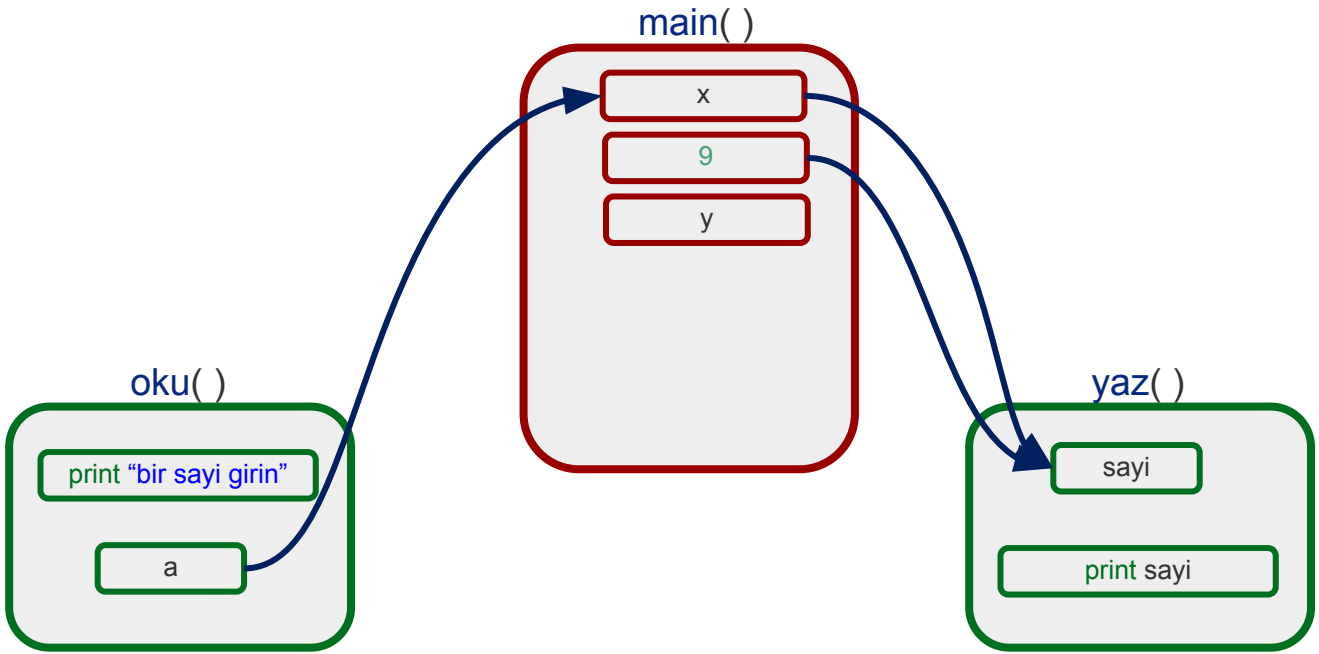
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();
        yaz(x);



        return 0;
}
```



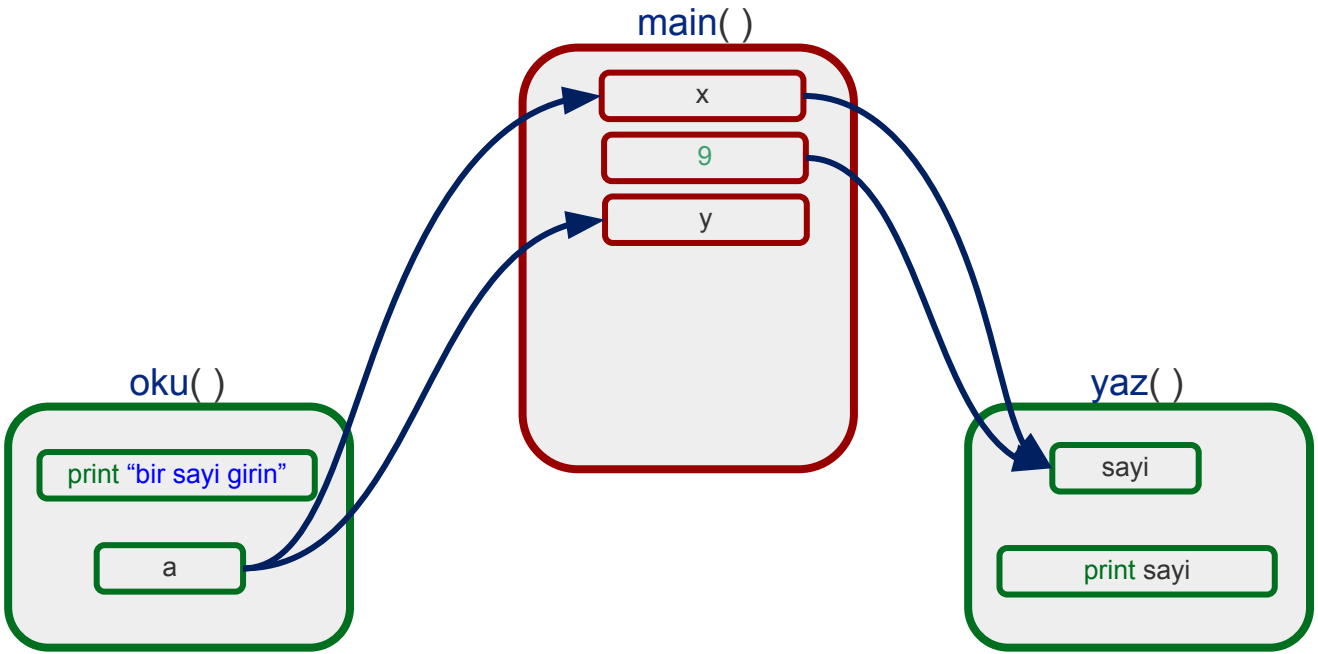
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();
        yaz(x);
        yaz(9);



        return 0;
}
```



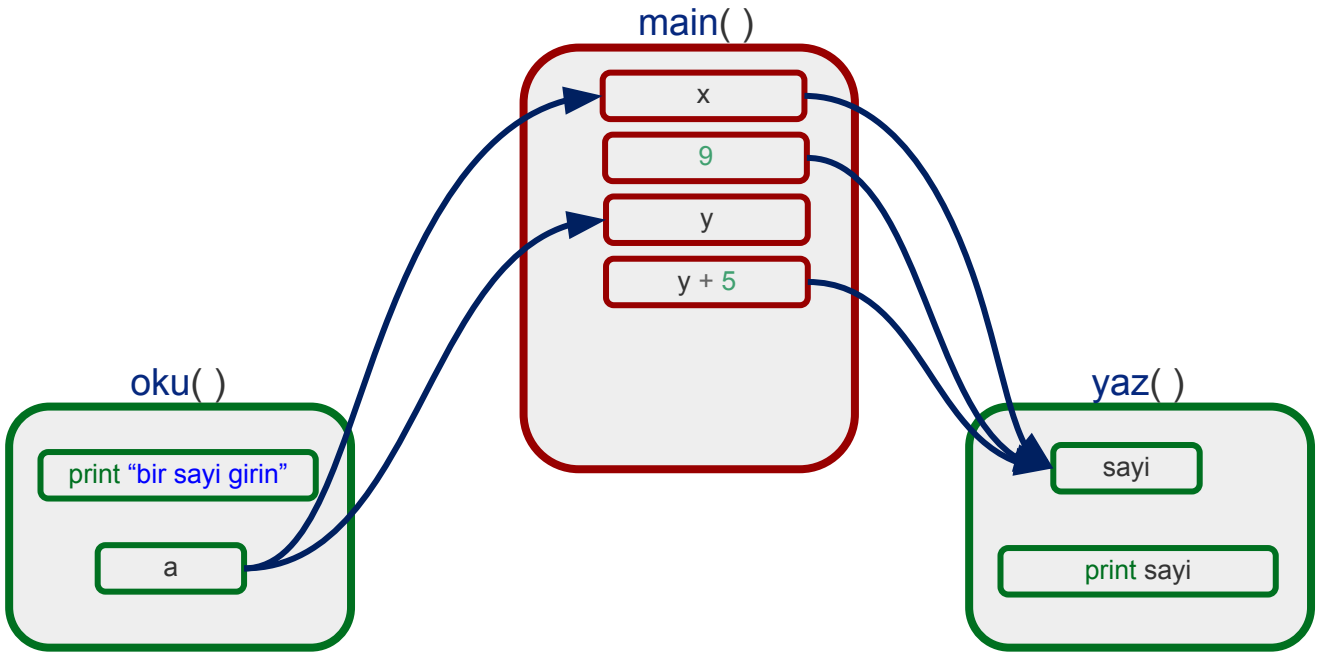
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();
        yaz(x);
        yaz(9);
        y = oku();



        return 0;
}
```



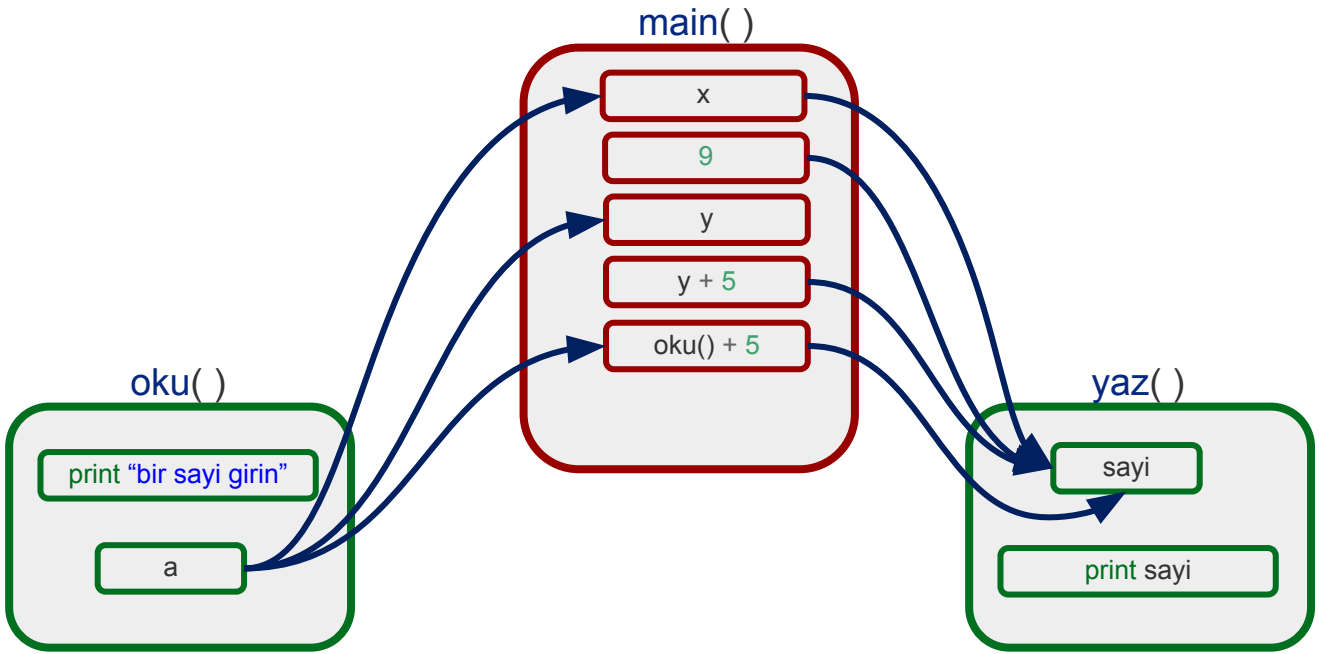
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();
        yaz(x);
        yaz(9);
        y = oku();
        yaz(y + 5);


        return 0;
}
```



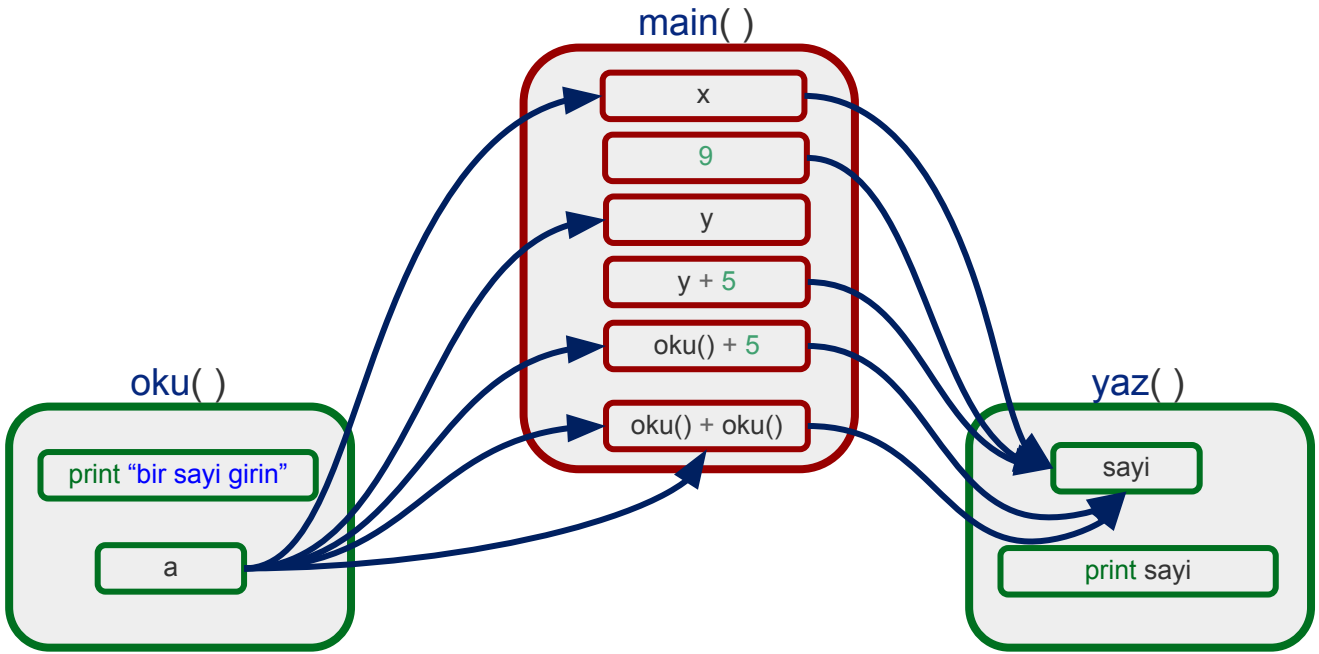
printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();
        yaz(x);
        yaz(9);
        y = oku();
        yaz(y + 5);
        yaz( oku() + 5);

        return 0;
}
```



printf & scanf

```c
#include <stdio.h>
#include <stdlib.h>
void yaz(int sayi) {
        printf("%d\n", sayi);
}
int oku() {
        int a;
        printf("bir sayi girin:");
        scanf("%d", &a);
        return a;
}
int main() {
        int x, y;
        x = oku();
        yaz(x);
        yaz(9);
        y = oku();
        yaz(y + 5);
        yaz( oku() + 5);
        yaz( oku() + oku() );
        return 0;
}
```



Ucgen

analiz:

1. s… 1 tane x
2. sa… 2 ta…
3. satir…
4. satirda … x
5. satirda … x
6. sati…
7. s… 2 t…
8. …da 3 tan…
9. satirda 4 tane x

x

xx

xxx

x

xx

x

xx

xxx

xxxx

DIVIDE AND CONQUER

Ucgen

analiz:

1. satirda	1 tane x
2. satirda	2 tane x
3. satirda	3 tane x

x

xx

xxx

x

xx

x

xx

xxx

xxxx

DIVIDE AND CONQUER

Ucgen

analiz:

1. satirda	1 tane x
2. satirda	2 tane x
3. satirda	3 tane x
1. satirda	1 tane x
2. satirda	2 tane x

x

xx

xxx

x

xx

x

xx

xxx

xxxx

DIVIDE AND CONQUER

Ucgen

analiz:

1. satirda	1 tane x
2. satirda	2 tane x
3. satirda	3 tane x
1. satirda	1 tane x
2. satirda	2 tane x
1. satirda	1 tane x
2. satirda	2 tane x
3. satirda	3 tane x
4. satirda	4 tane x

x

xx

xxx

x

xx

x

xx

xxx

xxxx

DIVIDE AND CONQUER

Hatırlatma

Hatırlatma

```
#include <stdio.h>
int main() {
        int i, j;
        for (i = 1 ; i <= 5 ; i++) {



                printf("\n");
        }
        return 0;
}
```

satir =5

sutun = 5

x					i=1;	j=1;	x=1
x	x				i=2;	j=1, j=2;	x=2
x	x	x			i=3;	j=1, j=2, j=3	x=3
x	x	x	x		i=4;	j=1, j=2, j=3, j=4	x=4
x	x	x	x	x	i=5;	j=1, j=2, j=3, j=4, j=5	x=5

Hatırlatma

```
#include <stdio.h>
int main() {
	int i, j;
	for (i = 1 ; i <= 5 ; i++) {
		for (j = 1 ; j <= i ; j++) {
			printf("x");
		}
		printf("\n");
	}
	return 0;
}
```

satir =5

sutun = 5

x					i=1;	j=1;	x=1
x	x				i=2;	j=1, j=2;	x=2
x	x	x			i=3;	j=1, j=2, j=3	x=3
x	x	x	x		i=4;	j=1, j=2, j=3, j=4	x=4
x	x	x	x	x	i=5;	j=1, j=2, j=3, j=4, j=5	x=5

Ucgen

```
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
        int i, j;
        for (i = 1 ; i <= x ; i++) {
                for (j = 1 ; j <= i ; j++) {
                        printf("*");
                }
                printf("\n");
        }
}
```

```
x
xx
xxx
x
xx
x
xx
xxx
xxxx
```

DIVIDE AND CONQUER

Ucgen

```
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
        int i, j;
        for (i = 1 ; i <= x ; i++) {
                for (j = 1 ; j <= i ; j++) {
                        printf("*");
                }
                printf("\n");
        }
}

int main() {
        ucgen_ciz(3);


        return 0;
}
```

```
x
xx
xxx
x
xx
x
xx
xxx
xxxx
```

DIVIDE AND CONQUER

Ucgen

```
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
        int i, j;
        for (i = 1 ; i <= x ; i++) {
                for (j = 1 ; j <= i ; j++) {
                        printf("*");
                }
                printf("\n");
        }
}

int main() {
        ucgen_ciz(3);
        ucgen_ciz(2);

        return 0;
}
```

```
x
xx
xxx
x
xx
x
xx
xxx
xxxx
```

DIVIDE AND CONQUER

Ucgen

```
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
        int i, j;
        for (i = 1 ; i <= x ; i++) {
                for (j = 1 ; j <= i ; j++) {
                        printf("*");
                }
                printf("\n");
        }
}

int main() {
        ucgen_ciz(3);
        ucgen_ciz(2);
        ucgen_ciz(4);
        return 0;
}
```

```
x
xx
xxx
x
xx
x
xx
xxx
xxxx
```

DIVIDE AND CONQUER

Ucgen

analiz:

1. satirda	1 tane x
2. satirda	2 tane x
3. satirda	3 tane x
4. satirda	4 tane x

x

xx

xxx

xxxx

xxx

xx

x

DIVIDE AND CONQUER

Ucgen

analiz:

1. satirda	1 tane x
2. satirda	2 tane x
3. satirda	3 tane x
4. satirda	4 tane x
1. satirda	3 tane x
2. satirda	2 tane x
3. satirda	1 tane x

x

xx

xxx

xxxx

xxx

xx

x

DIVIDE AND CONQUER

Ucgen

```c
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
	int i, j;
	for (i = 1 ; i <= x ; i++) {
		for (j = 1 ; j <= i ; j++) {
			printf("*");
		}
		printf("\n");
	}
}

int main() {
	ucgen_ciz(4);

	return 0;
}
```

```
x
xx
xxx
xxxx
xxx
xx
x
```

DIVIDE AND CONQUER

Ucgen

```c
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
	int i, j;
	for (i = 1 ; i <= x ; i++) {
		for (j = 1 ; j <= i ; j++) {
			printf("*");
		}
		printf("\n");
	}
}
void ters_ucgen_ciz(int x) {
	int i, j;
	for (i = 1 ; i <= x ; i++) {



		printf("\n");
	}
}
int main() {
	ucgen_ciz(4);
	ters_ucgen_ciz(3);
	return 0;
}
```

```
x
xx
xxx
xxxx
xxx
xx
x
```

DIVIDE AND CONQUER

Ucgen

```c
#include <stdio.h>
#include <stdlib.h>
void ucgen_ciz(int x) {
	int i, j;
	for (i = 1 ; i <= x ; i++) {
		for (j = 1 ; j <= i ; j++) {
			printf("*");
		}
		printf("\n");
	}
}
void ters_ucgen_ciz(int x) {
	int i, j;
	for (i = 1 ; i <= x ; i++) {
		for (j = 1 ; j <= x-i+1 ; j++) {
			printf("*");
		}
		printf("\n");
	}
}
int main() {
	ucgen_ciz(4);
	ters_ucgen_ciz(3);
	return 0;
}
```

```
x
xx
xxx
xxxx
xxx
xx
x
```

DIVIDE AND CONQUER

Double Equality

```
#include <stdio.h>
#include <stdlib.h>
int main() {
        double x = 1.2;
        double y = 5.3;
        double z = y - 4.1;




        return 0;
}
```

Double Equality

```
#include <stdio.h>
#include <stdlib.h>
int main() {
        double x = 1.2;
        double y = 5.3;
        double z = y - 4.1;
        if ( z == x)
                printf("esit\n");
        else
                printf("esit degil\n");
        return 0;
}
```

Double Equality

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

int main() {
	double x = 1.2;
	double y = 5.3;
	double z = y - 4.1;

	return 0;
}
```

HATA PAYI

0.0001

Double Equality

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

HATA PAYI

0.0001

```c
int main() {
	double x = 1.2;
	double y = 5.3;
	double z = y - 4.1;
	if ( esit_mi(x, z, 0.0001) == 1)
		printf("esit\n");
	else
		printf("esit degil\n");
	return 0;
}
```

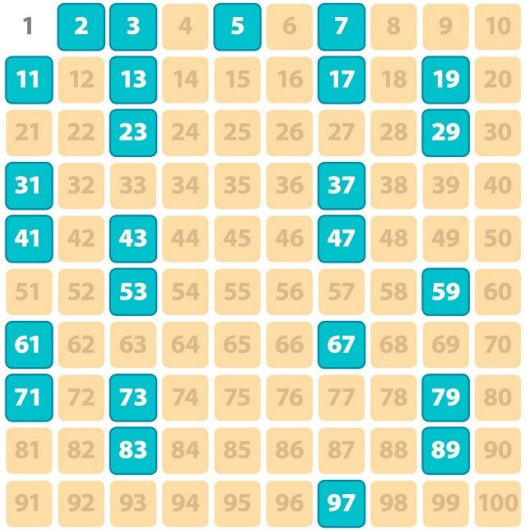
Double Equality

```c
#include <stdio.h>
#include <stdlib.h>
int esit_mi(double A, double B, double hata_payi) {
	if (fabs(A - B) < hata_payi) {
		return 1;
	} else {
		return 0;
	}
}
int main() {
	double x = 1.2;
	double y = 5.3;
	double z = y - 4.1;
	if ( esit_mi(x, z, 0.0001) == 1)
		printf("esit\n");
	else
		printf("esit degil\n");
	return 0;
}
```

HATA PAYI

0.0001

Asal Sayı



23 asal mıdır?

23 >= P X P

23 >= 4 X 4

23 <= 4 X 4

2 <= 23/2

3 <= 23/3

4 <= 23/4

5 <= 23/5

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 ...

Asal Sayı

```c
#include <stdio.h>
#include <stdlib.h>
int asal_mi(int sayi) {
    int j,i=sayi;
        for(j = 2; j <= (i/j); j++)
            if(!(i%j)) break;
        if(j > (i/j)) return 1;
    return 0;
}
```

23 <= 4 X 4

j 2 <= (i/j) 23/2

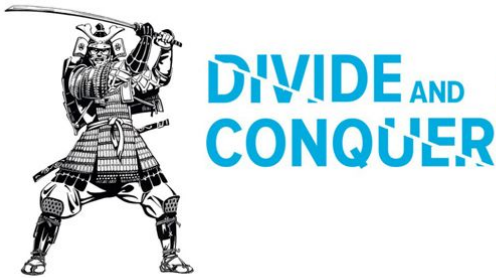
j 3 <= (i/j) 23/3

j 4 <= (i/j) 23/4

j 5 <= (i/j) 23/5

Asal Sayı

```c
#include <stdio.h>
#include <stdlib.h>
int asal_mi(int sayi) {
    int j,i=sayi;
        for(j = 2; j <= (i/j); j++)
            if(!(i%j)) break;
        if(j > (i/j)) return 1;
    return 0;
}
int main() {
    int a, b;
    printf("iki sayi girin: ");
    scanf("%d %d", &a, &b);
    int sayi;




    return 0;
}
```



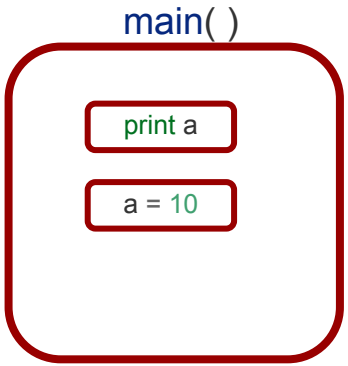
Asal Sayı

```c
#include <stdio.h>
#include <stdlib.h>
int asal_mi(int sayi) {
    int j,i=sayi;
        for(j = 2; j <= (i/j); j++)
            if(!(i%j)) break;
        if(j > (i/j)) return 1;
    return 0;
}
int main() {
    int a, b;
    printf("iki sayi girin: ");
    scanf("%d %d", &a, &b);
    int sayi;
    for (sayi = a ; sayi <= b ; sayi++) {
        int sonuc = asal_mi(sayi);
        if (sonuc == 1) printf("asal: %d\n", sayi);
    }
    return 0;
}
```

DIVIDE AND CONQUER

main & fonksiyon

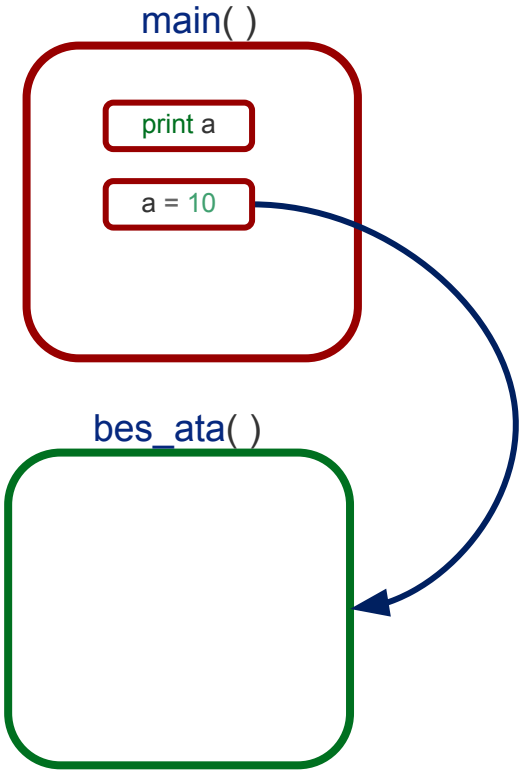
```c
#include <stdio.h>
```




```c
int main() {
    int a = 10;
    printf("gonderilmeden once -> %d\n\n\n", a);


    return 0;
}
```

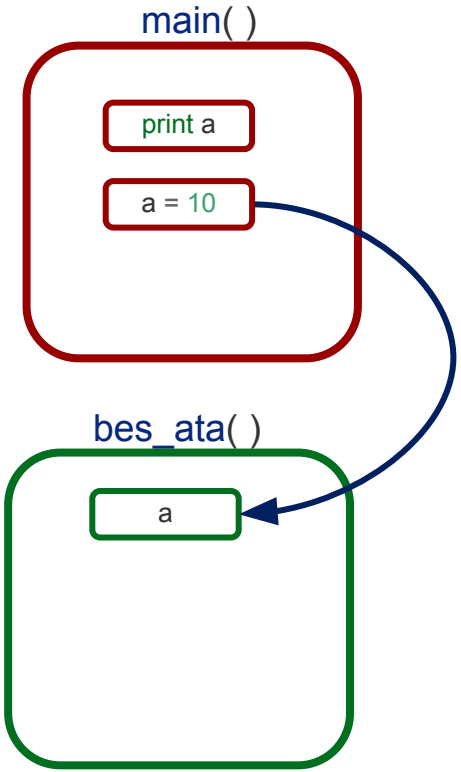
main & fonksiyon

```c
#include <stdio.h>

int main() {
    int a = 10;
    printf("gonderilmeden once -> %d\n\n\n", a);
    bes_ata(a);

    return 0;
}
```



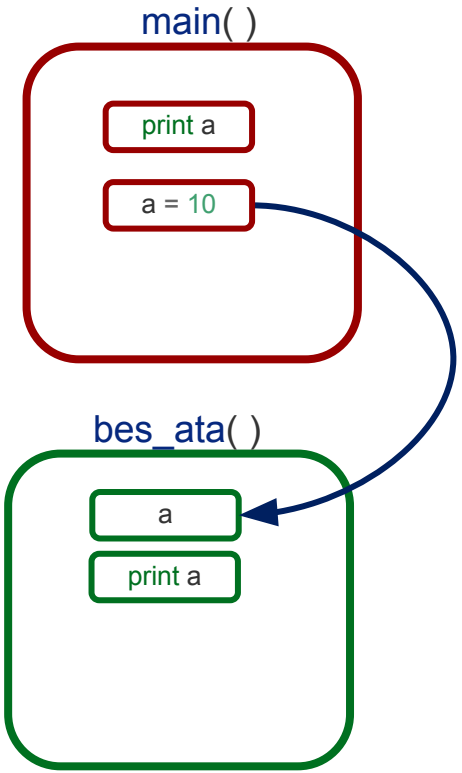
main & fonksiyon

```c
#include <stdio.h>
void bes_ata(int a) {



}

int main() {
    int a = 10;
    printf("gonderilmeden once -> %d\n\n\n", a);
    bes_ata(a);

    return 0;
}
```



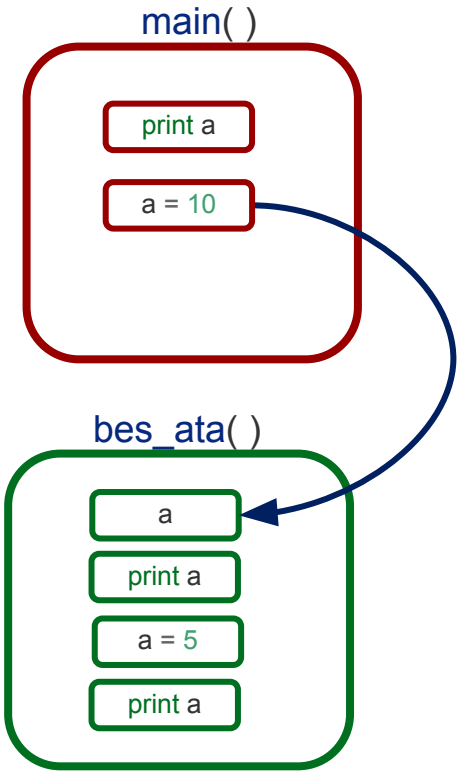
main & fonksiyon

```c
#include <stdio.h>
void bes_ata(int a) {
    printf("alınan -> %d\n", a);


}

int main() {
    int a = 10;
    printf("gonderilmeden once -> %d\n\n\n", a);
    bes_ata(a);

    return 0;
}
```



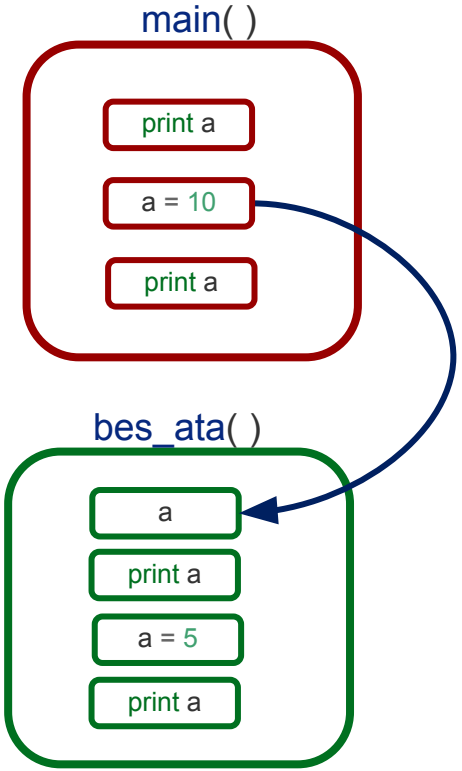
main & fonksiyon

```c
#include <stdio.h>
void bes_ata(int a) {
    printf("alınan -> %d\n", a);
    a = 5;
    printf("degistirildi -> %d\n\n\n", a);
}

int main() {
    int a = 10;
    printf("gonderilmeden once -> %d\n\n\n", a);
    bes_ata(a);

    return 0;
}
```



main & fonksiyon

```c
#include <stdio.h>
void bes_ata(int a) {
    printf("alınan -> %d\n", a);
    a = 5;
    printf("degistirildi -> %d\n\n\n", a);
}

int main() {
    int a = 10;
    printf("gonderilmeden once -> %d\n\n\n", a);
    bes_ata(a);
    printf("gonderildikten sonra -> %d\n\n\n", a);
    return 0;
}
```



Sorular