

ALGORITMA DAN PEMROGRAMAN I

PROSEDUR

Rosa Ariani Sukamto

ROSA ARIANI SUKAMTO

Blog: <http://hariiniadalahhadiah.wordpress.com>


Facebook: <https://www.facebook.com/rosa.ariani.sukamto>

Email: rosa_if_itb_01@yahoo.com

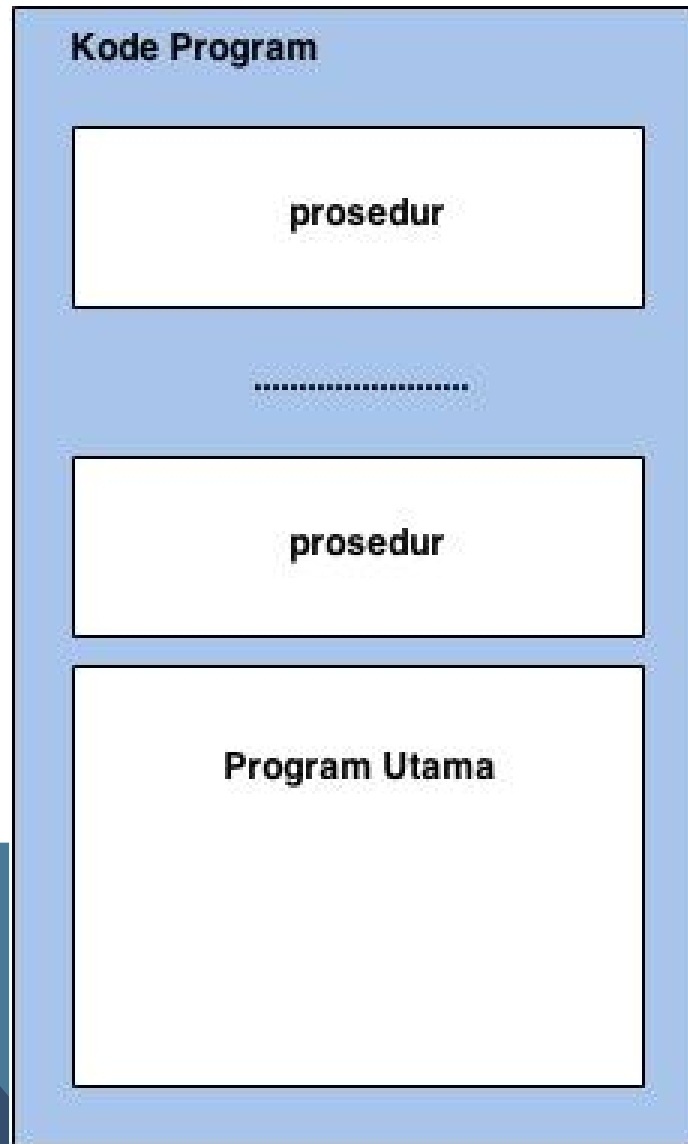
Website: <https://rosa-as.id>



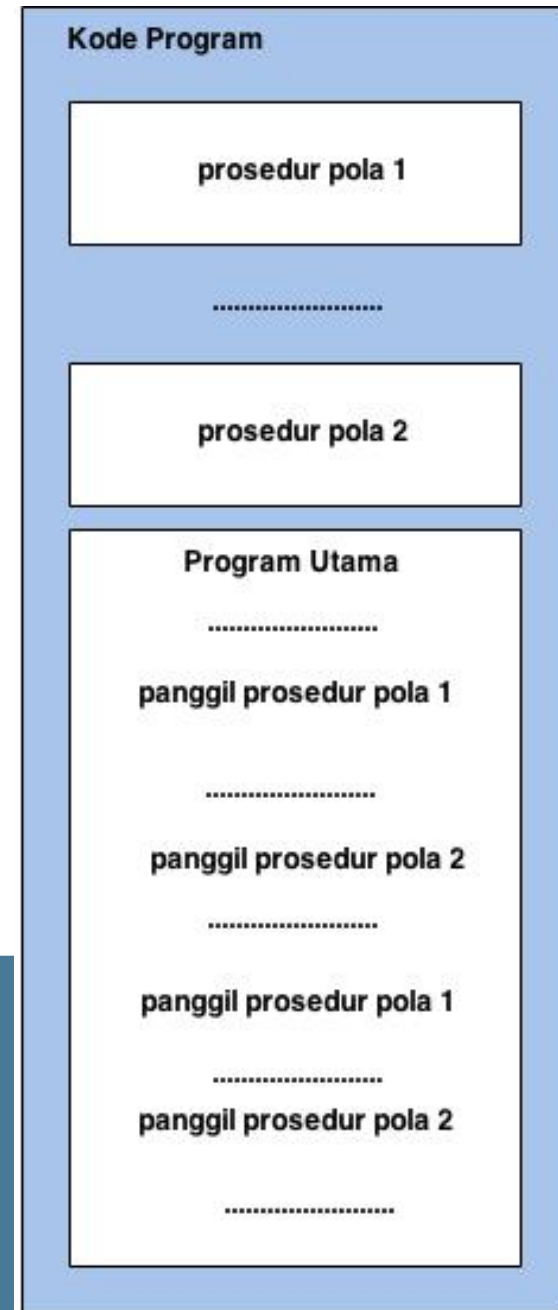
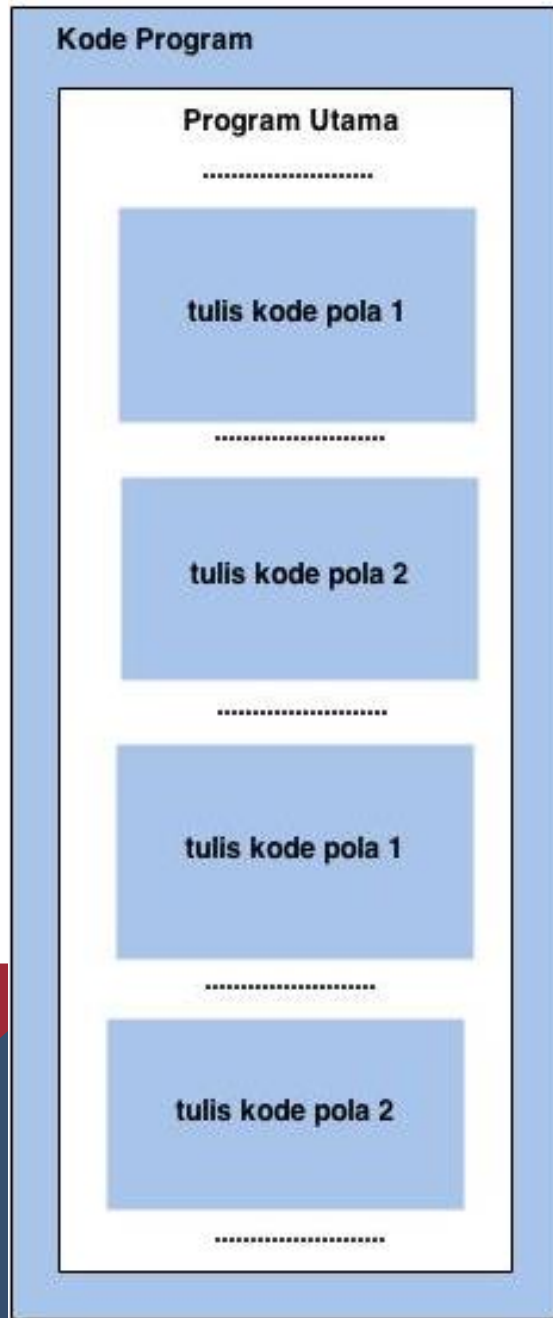
PROSEDUR (1)

- **Prosedur adalah sebuah blok program tersendiri yang merupakan bagian dari program lain yang lebih besar**
 - **Prosedur dapat dipanggil oleh program utamanya ataupun oleh prosedur lain yang juga merupakan bagian dari program utamanya (masih dalam satu kode program)**
 - **Sebuah program yang memiliki prosedur biasanya terdiri dari satu atau lebih prosedur dan satu program utama**
- 

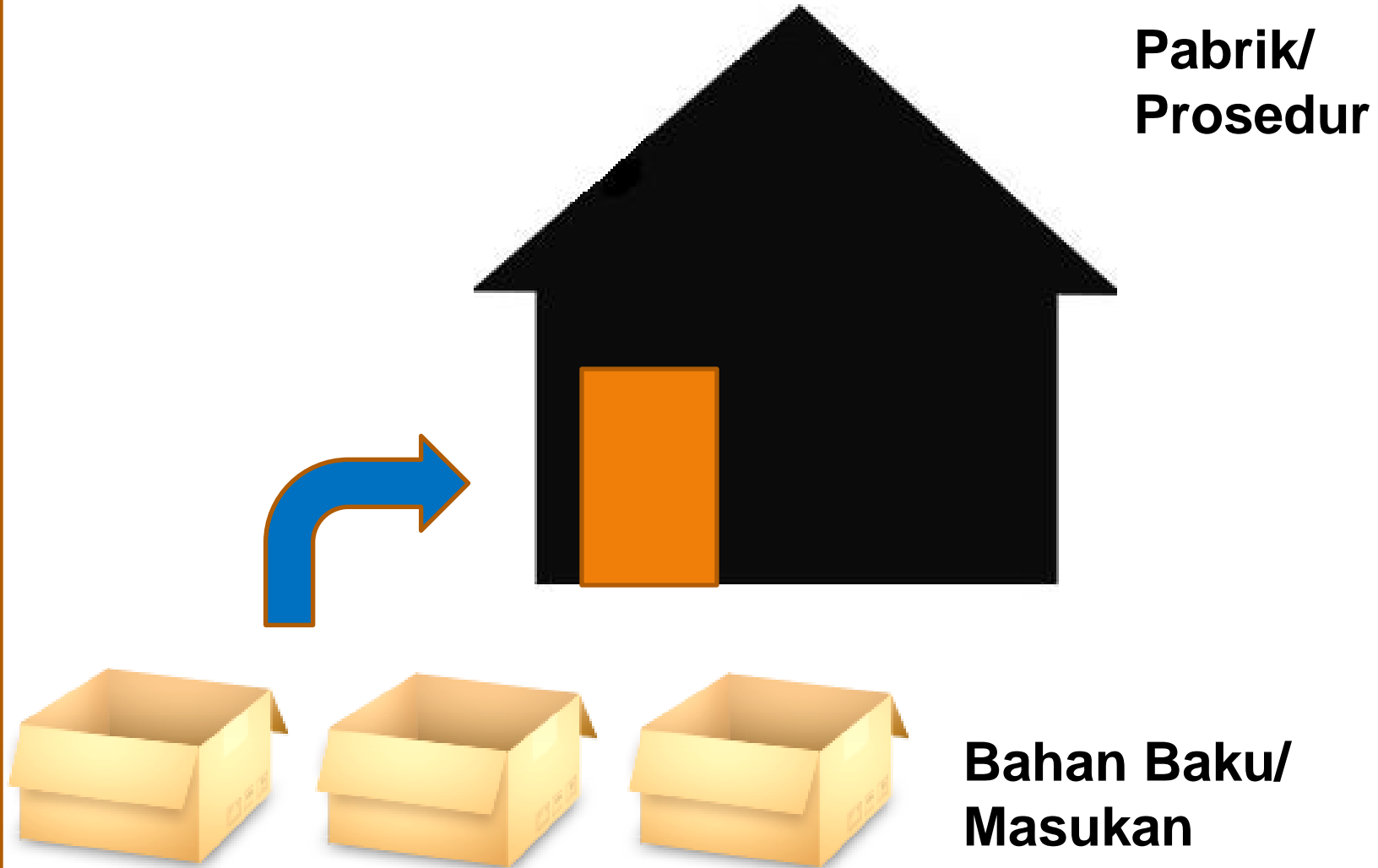
PROSEDUR (2)



PROSEDUR (3)



PROSEDUR (4)



PROSEDUR POLA (1)

* * *

* * *

* * *

* * * *

* * * *

* * * *

* * * *

* * * * *

* * * * *

* * * * *

* * * * *

* * * * *

PROSEDUR POLA (2)

```
#include <stdio.h>

int main() {
    int n, i, j;
    scanf("%d", &n);
    for(i=1; i<=n; i++) {
        for(j=1; j<=n; j++) {
            printf("*");
        }
        printf("\n");
    }
    for(i=1; i<=(n+1); i++) {
        for(j=1; j<=n; j++) {
            printf("*");
        }
        printf("\n");
    }
}
```

```
for(i=1; i<=(n+2); i++) {
    for(j=1; j<=n; j++) {
        printf("*");
    }
    printf("\n");
}
return 0;
}
```


PROSEDUR POLA (3)

```
#include <stdio.h>

void pola(int n){
    int i, j;
    for(i=1;i<=n;i++){
        for(j=1; j<=n; j++){
            printf("*");
        }
        printf("\n");
    }
}
```

```
int main(){
    int n;
    scanf("%d", &n);

    pola(n);
    pola(n+1);
    pola(n+2);

    return 0;
}
```

3 FILE (1)

Penulisan kode program menggunakan prosedur yang baik dapat menggunakan 3 *file* agar struktur kode program lebih baik

Yang membedakan seseorang yang belajar pemrograman dengan otodidak dan dengan kurikulum pendidikan yang baik adalah dari bagaimana menuliskan kode program yang baik dan efisiensi dari program yang dibuat... sehingga tidak sekedar asal jalan

Struktur *file* adalah sebagai berikut:

- *Header* dengan ekstensi .h yang merupakan *file* yang berfungsi sebagai daftar isi semua yang ada di mesin
- *Mesin* dengan ekstensi .c merupakan *file* yang berisi semua implementasi dari daftar isi
- *Main* dengan nama biasanya main.c merupakan *file* yang berisi program utama yang memanfaatkan mesin

3 FILE (2)

Penulisan kode program dalam bahasa C menggunakan 3 *file* juga merupakan konsep untuk membuat pustaka/*library* kita sendiri seperti halnya pustaka kompilator seperti

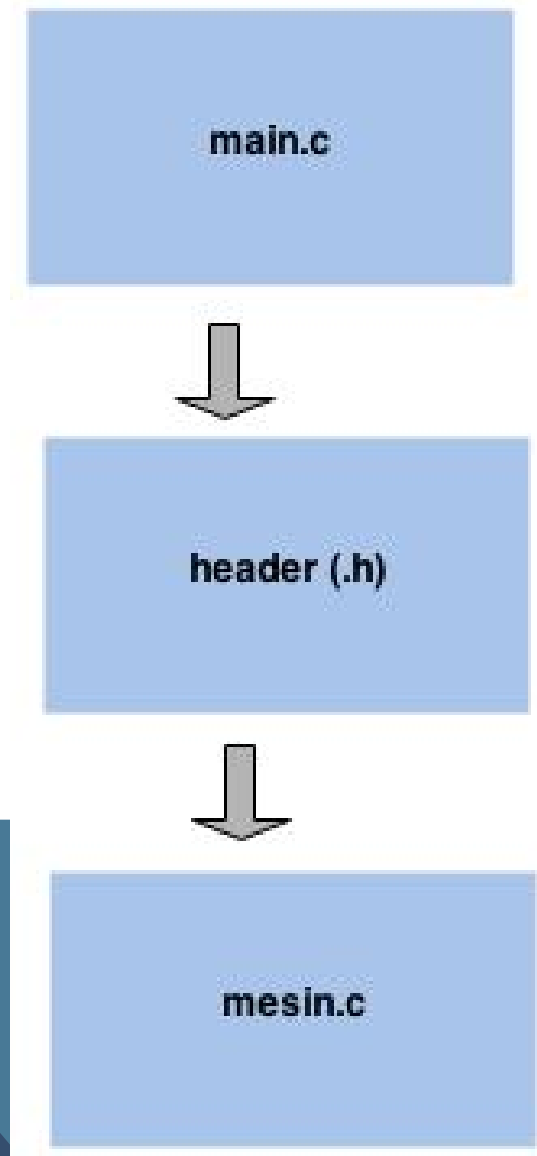
`stdio.h`

atau

`string.h` dan lain sebagainya



3 FILE (3)



HEADER (1)

- Semua include dari pustaka kompilator
- Semua typedef struct
- Semua variabel global
- Semua deklarasi prosedur/fungsi yang dapat digunakan oleh main.c

HEADER (2)

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

```
void pola(int) ;
```



MESIN (1)

Semua implementasi dari prosedur atau fungsi yang telah disebutkan di *header*



MESIN (2)

```
#include "pola.h"
```

```
void pola(int n) {  
    int i, j;  
    for(i=1; i<=n; i++) {  
        for(j=1; j<=n; j++) {  
            printf("*");  
        }  
        printf("\n");  
    }  
}
```


MAIN

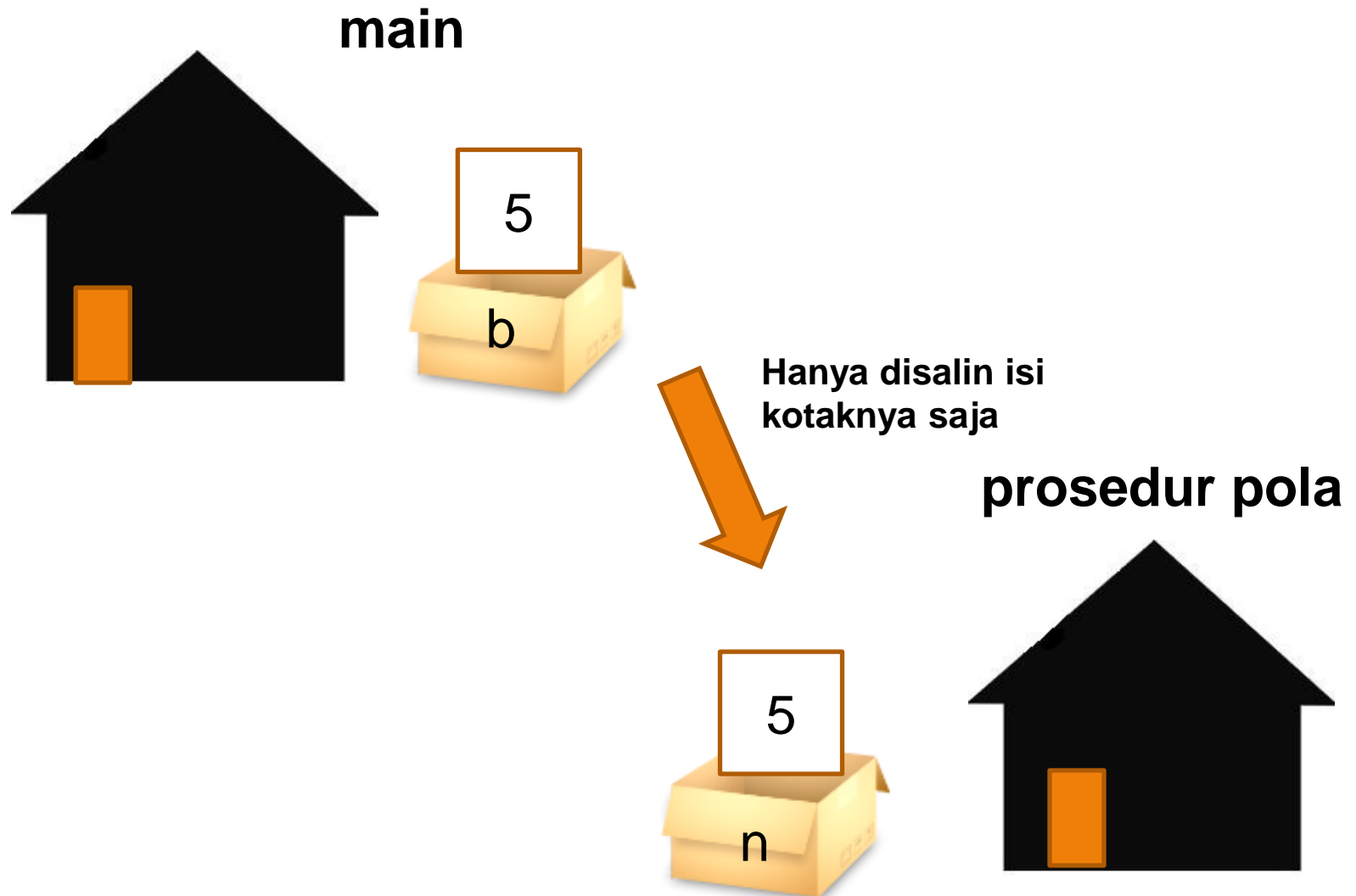
```
#include "pola.h"

int main() {
    int b;
    scanf("%d", &b) ;

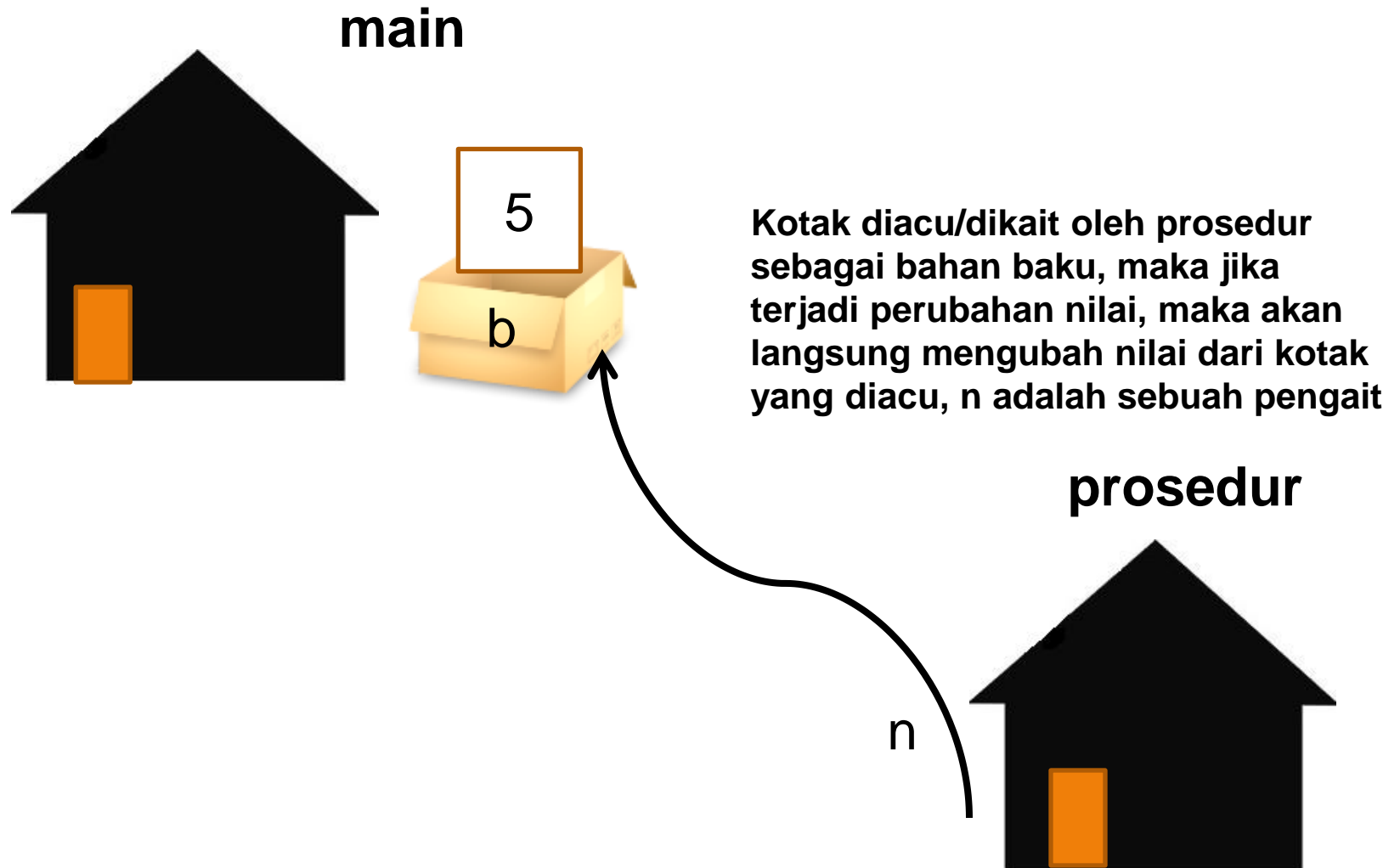
    pola(b) ;
    pola(b+1) ;
    pola(b+2) ;

    return 0;
}
```

MEKANISME PARAMETER PASS BY VALUE



MEKANISME PARAMETER PASS BY REFERENCE



KOMPILASI DAN EKSEKUSI

Masukkan semua kode program dalam satu direktori
Kompilasi

Masuk ke direktori yang berisi 3 *file*

```
gcc -c *.c
```

Membuat eksekusi


```
gcc *.c -o nama_file_eksekusi.exe
```

CONTOH HEADER

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

```
void pola(int) ;
```

```
void tulisArrayInt(int n, int arr[n]) ;
```



CONTOH MESIN

```
#include "pola.h"

void pola(int n){
    int i, j;
    for(i=1;i<=n;i++){
        for(j=1; j<=n; j++){
            printf("*");
        }
        printf("\n");
    }
}
```

```
void tulisArrayInt(int n,
    int arr[n]){
    int i;
    for(i=0;i<n;i++){
        printf("%d\n",
            arr[i]);
    }
}
```

CONTOH MAIN

```
#include "pola.h"
```

```
int main(){
```

```
    int b;
```

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

```
    pola(b);
```

```
    pola(b+1);
```

```
    pola(b+2);
```

```
    int kumpulan[b];
```

```
    int i;
```

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

```
        scanf("%d",  
        &kumpulan[i]);
```

```
    }
```

```
    tulisArrayInt(b,  
    kumpulan);
```

```
    return 0;
```

```
}
```

MARI MENCOBA

Diberikan 3 buah array. Ambillah nilai rata-rata nilai separuh elemen array bagian depan dan bagian belakang (diambil hanya integer tanpa koma). Lalu tampilkan secara urut dari nilai yang besar ke kecil dari semua array (gunakan if untuk mengurutkan 3 bilangan dari besar ke kecil). Banyaknya elemen array selalu genap.



DAFTAR PUSTAKA

