**TUGAS PRAKTIKUM 16**

**DASAR FUNGSI**



**Nama : Devafilla Rizqy Santosa**

**Kelas : 1 D4 Teknik Informatika A**

**NRP : 3120600026**

1. Tugas Pendahuluan.
   1. A. Listing

#include <stdio.h>

#include <stdlib.h>

void

fungsi(void);

main()

{

fungsi();

}

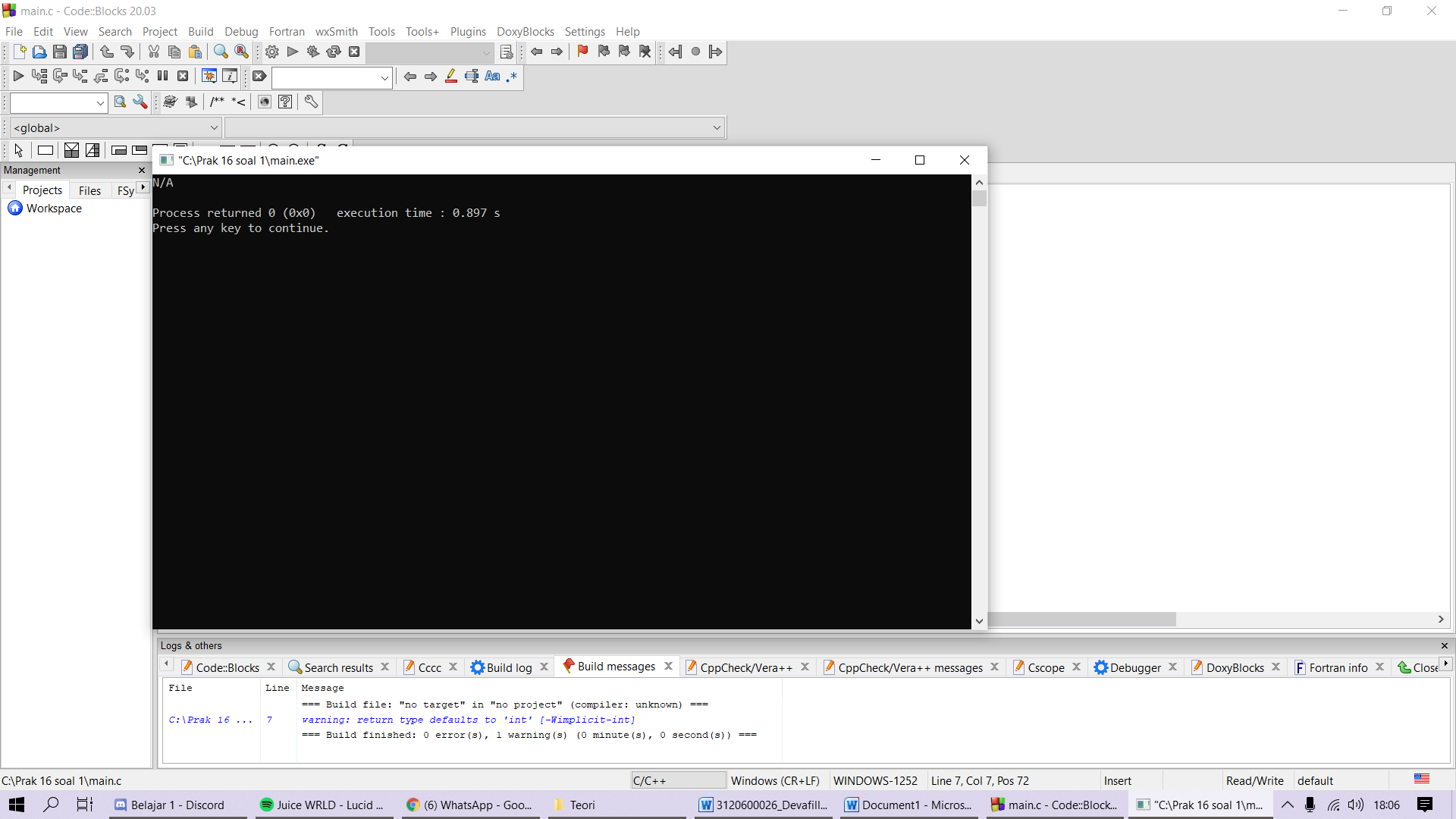
void fungsi(void)

{

puts("N/A");

}

1. Output



1. Analisa

Dari percobaan program diatas terbukti bahwa fungsi tanpa return value masih bisa berjalan apabila jika kita ingin menampilkan sebauh kalimat.

* 1. A. Listing

#include <stdio.h>

#include <stdlib.h>

int fungsi();

main()

{

fungsi();

printf("Nilainya adalah %d", fungsi());

}

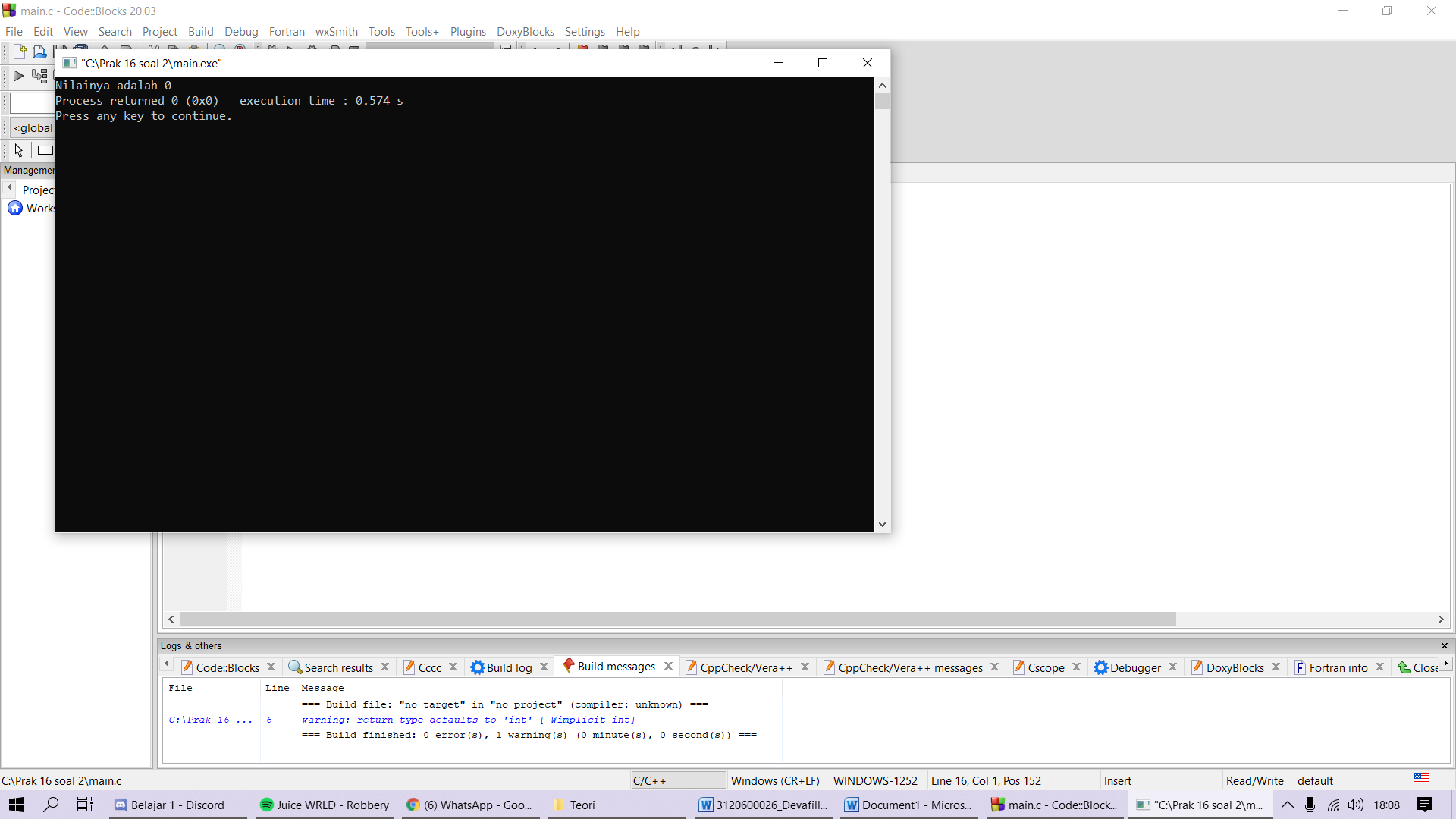
int fungsi()

{

return 0;

}

1. Output



1. Analisa

Dari percobaan program diatas terbukti bahwa jika kita memberi return value tetapi tidak diassign kan disebuah variable apapun program masih dapat membaca return jika kita ingin menampilkannya ( %d)

* 1. A. Listing

#include <stdio.h>

#include <stdlib.h>

void

fungsi(float, float);

main()

{

float bil\_1, bil\_2, nilai;

printf("Masukan bil\_1 : ");

scanf("%f", &bil\_1);

printf("Masukan bil\_2 : ");

scanf("%f", &bil\_2);

nilai = fungsi(bil\_1, bil\_2);

printf("%f", nilai);

}

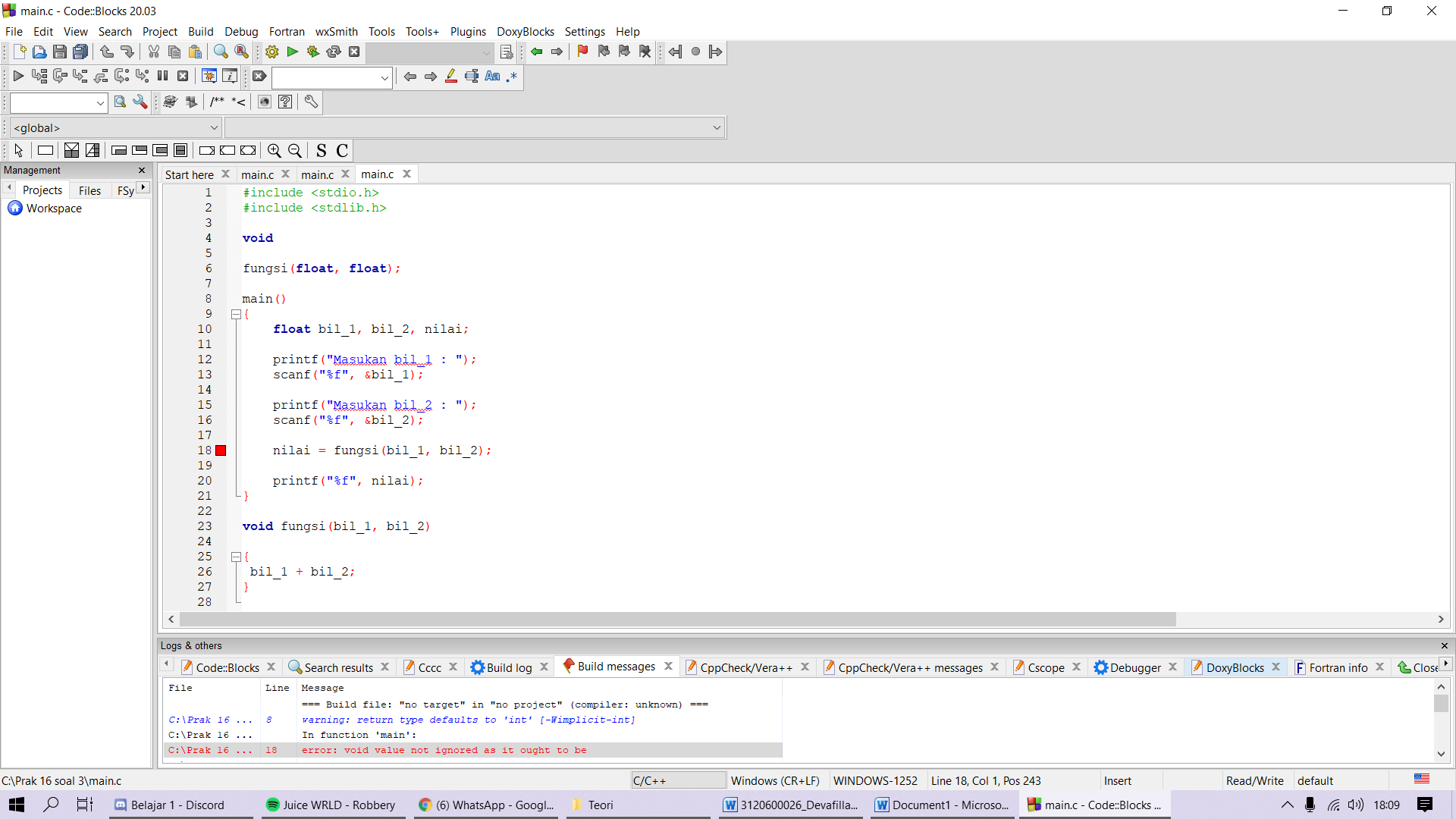
void fungsi(bil\_1, bil\_2)

{

bil\_1 + bil\_2;

}

B. Output



C.Analisa

Dari percobaan program diatas terbukti bahwa jika fungsi diassign kan ke sebuah variabel dan fungsi tidak memiliki return value, maka program tersebut dapat menampilkan hasil di fungsi tetapi dalam main int tidak dapat ditampilkan hasil setelah pengubahan nilai variabel.

* 1. A. Listing

#include <stdio.h>

#include <stdlib.h>

int OddEvenTest(int);

main()

{

int a, hasil;

a = 5;

hasil = OddEvenTest(a);

printf("a=%d; hasil=%d\n",a,hasil);

}

OddEvenTest(int b)

{

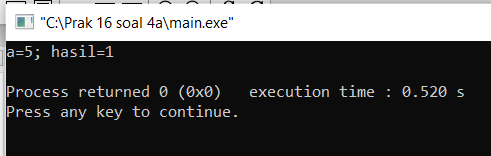
int a;

a = b % 2;

return a;

}

B. Output



|  |  |  |
| --- | --- | --- |
| a | hasil | b |
| 5 | 1 | 5 |
| 6 | 0 | 6 |
| 7 | 1 | 7 |
| 8 | 0 | 8 |
| 9 | 1 | 9 |
| 10 | 0 | 10 |

C.Analisa

Dari pengamatan listing diatas dapat dilihat bahwa untuk nilai variable a nanti akan dibawa ke fungsi diubah menjadi variable b dan di moduluskan 2 lalu dikembalikan menggunakan return value untuk nantinya ditampilkan di int main.

1.4 B. Listing

#include <stdio.h>

#include <stdlib.h>

void demo(void);

main()

{

int i=0;

while(i < 3) {

demo();

i++;

}

}

void demo(void)

{

auto int var\_auto = 0;

static int var\_static = 0;

printf("auto = %d, static = %d\n",

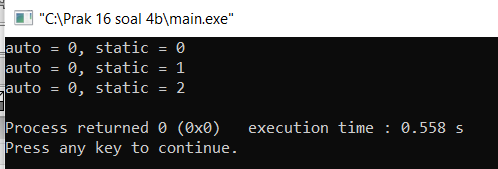
var\_auto, var\_static);

++var\_auto;

++var\_static;

}

B. Output



|  |  |  |
| --- | --- | --- |
| i | var\_auto | var\_static |
| 1 | 0 | 0 |
| 2 | 0 | 1 |
| 3 | 0 | 2 |

C.Analisa

Dari pengamatan listing diatas dapat dilihat bahwa keluaran hasilnya dapat dilihat di dalam table, yang dimana I awal 0 dan sedangkan syarat loopnya antara urutan 1 sampai 3 kali jadi nanti fungsinya dilakukan sebanyak 3 kali. Sedangkan didalam fungsi terdapat int auto dan static, untuk auto int nilai nya setelah program fungsi selesai maka nilainya tetap jika nantinya dilakukan perubahan maka jika dilakukan pemanggilan fungsi lagi nilainya sama dengan yang diawalnya ditetapkan, sedangkan static nilai variabelnya berubah setelah dilakukan perubahan di variabelnya.

1.4 C. Listing

#include <stdio.h>

#include <stdlib.h>

void fung\_a(void);

void fung\_b(void);

int x = 20;

main()

{

x += 2;

fung\_a();

fung\_a();

printf("\nNilai x dalam main() = %d\n\n",x);

}

void fung\_a(void)

{

static x = 5;

x++;

printf("Nilai x dalam fung\_a() = %d\n", x);

fung\_b();

}

void fung\_b(void)

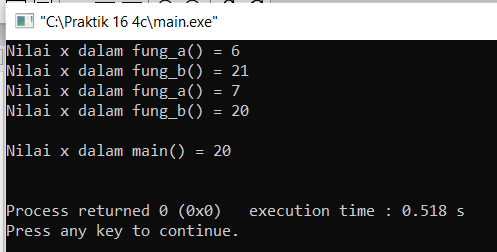
{

x--;

printf("Nilai x dalam fung\_b() = %d\n", x);

}

B. Output



|  |
| --- |
| x |
| 20 |
| 22 |
| 6 |
| 21 |
| 7 |
| 20 |
| 20 |

C.Analisa

Dari pengamatan listing diatas hasil tampilan program jika di tampilkan secara manual terdapat pada table. Dapat dilihat bahwa yang awalnya x = 20 nanti ditambah 2 di main lalu dimasukan fungsi\_a, lalu di fungsi\_a terdapat variable local yang hanya dapat ditampilkan di fungsinya saja, lalu dimasukan ke fungsi\_b dan di fungsi b dilakukan pengurangan dari variable global yang sudah ditambah tdi. Dan dilakukan kerja program di main yang selanjutnya seperti fungsi\_a lagi lalu Nilai x dalam main.

* 1. Listing

#include <stdio.h>

#include <stdlib.h>

float masukan(int);

float average(int, float);

main()

{

int hasil;

float nilai, rata;

printf("Masukkan hasil yang akan dibagi :");

scanf("%d", &hasil);

puts(" ");

nilai = masukan(hasil);

rata = average(hasil, nilai);

printf("\nHasil nilai = %.2f", nilai);

printf("\nRata-rata = %.2f", rata);

}

float masukan(int nilai\_data)

{

int x;

float total = 0, input;

for(x = 1 ; x <= nilai\_data ; x++)

{

printf("Input nilai hasil ke-%d : ", x);

scanf("%f", &input);

fflush(stdin);

total += input;

}

return(total);

}

float average(int i, float f)

{

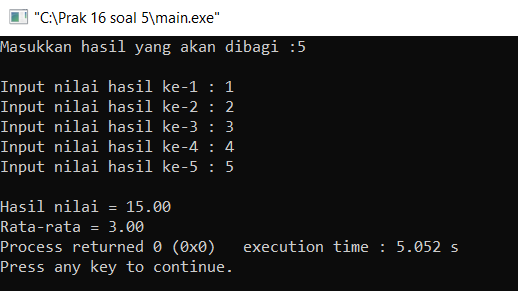
float fl;

fl = f / i;

return(fl);

}

B. Output



C.Analisa

Dari percobaan program diatas yaitu mencari sebuah nilai rata rata dari seluruh data dengan menggunakan fungsi masukan() sebagai berapa banyak nilai yang nantinya kita dapat memasukan sebuah nilai dan fungsi average() yang nantinya menghitung rata rata dari seluruh data yang dimasukan, lalu semua nanti di tampilkan di int main.

* 1. Listing

#include <stdio.h>

#include <stdlib.h>

float f\_to\_i(float);

float i\_to\_cm(float);

float cm\_to\_m(float);

main()

{

float feet, inchi, cm, meter;

printf("Input panjang dalam feet : ");

scanf("%f", &feet);

inchi = f\_to\_i(feet);

cm = i\_to\_cm(inchi);

meter = cm\_to\_m(cm);

printf("%.2f feet = %.2f inchi\n", feet, inchi);

printf("%.2f inchi = %.2f cm\n", inchi, cm);

printf("%.2f cm = %.2f meter\n", cm, meter);

}

float f\_to\_i(float f)

{

float hasil;

hasil = f \* 12;

return(hasil);

}

float i\_to\_cm(float i)

{

float hasil;

hasil = i \* 2.54f;

return(hasil);

}

float cm\_to\_m(float c)

{

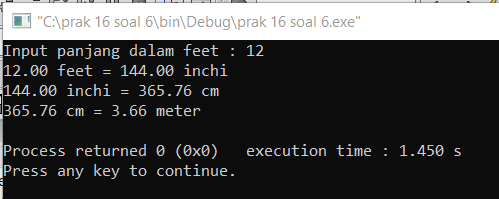
float hasil;

hasil = c / 100;

return(hasil);

}

B. Output



C.Analisa

Dari percobaan program diatas yaitu sebuah program yang dapat menghitung dari satuan feet ke inchi , inchi ke centimeter, dan centi meter ke meter. Dengan menggunakan 3 fungsi yang terdapat fungsi untuk menghitung inci, centimeter dan meter, jadi setelah fungsi menghitung inchi selesai nantinya langsung hasil inchi dilanjutkan ke fungsi centimeter dan lanjut ke fungsi selanjutnya. Lalu ditampilkan hasilnya di int main.

1. Kesimpulan :

Fungsi main() terdiri dari fungsi\_a() sampai dengan fungsi\_n(), menegaskan

bahwa dalam program fungsi main() akan memanggil fungsi\_a() sampai dengan

fungsi\_n(). Adapun fungsi-fungsi yang dipanggil oleh fungsi main() juga bisa

memanggil fungsi-fungsi yang lain. Variabel statis diperoleh dengan menambahkan kata kunci static di depan tipe data vriabel.