

TUGAS TEORI KONSEP PEMROGRAMAN

JILID 5



Oleh :

Nama : Rosi Arif Mulyadi

NRP : 3121522021

Prodi : D3 Teknik Informatika PENS PSDKU Sumenep

Kelas : 1 ITA D3 Sumenep

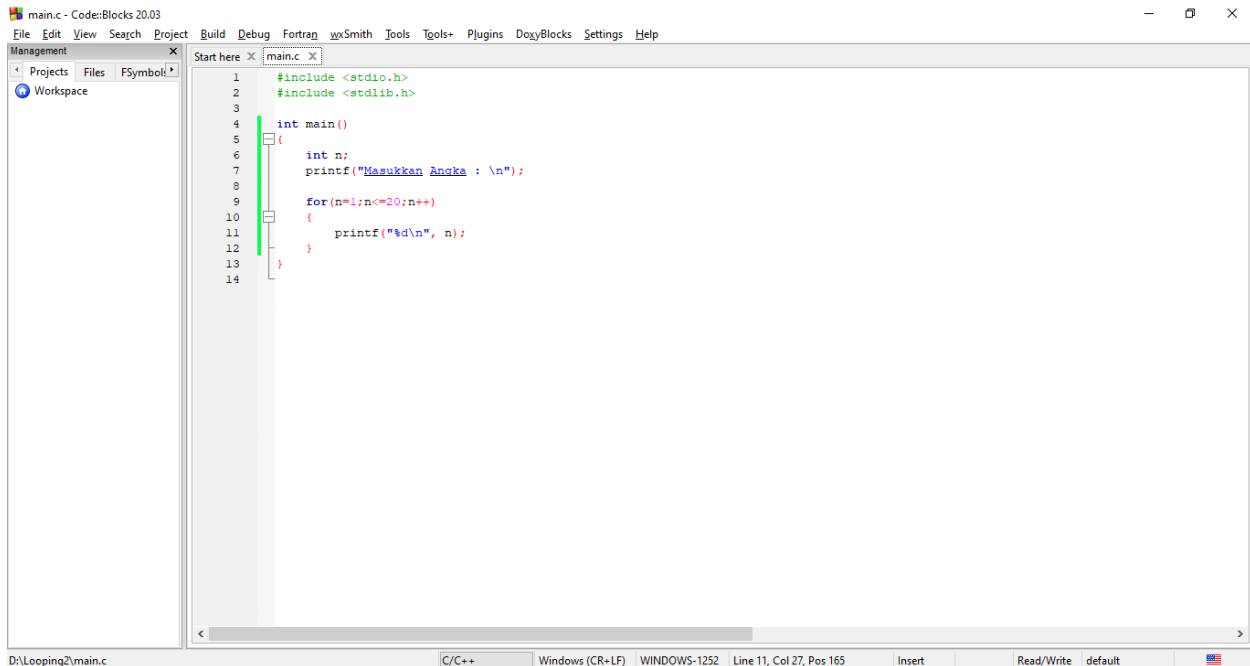
Dosen :

Lusiana Agustien M.Kom

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

1. Gunakan loop *for* untuk menampilkan nilai 1 sampai dengan 20 dalam baris-baris yang terpisah.

Jawab :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     int n;
7     printf("Masukkan Angka : \n");
8
9     for(n=1;n<=20;n++)
10     {
11         printf("%d\n", n);
12     }
13 }
14
```

Output :

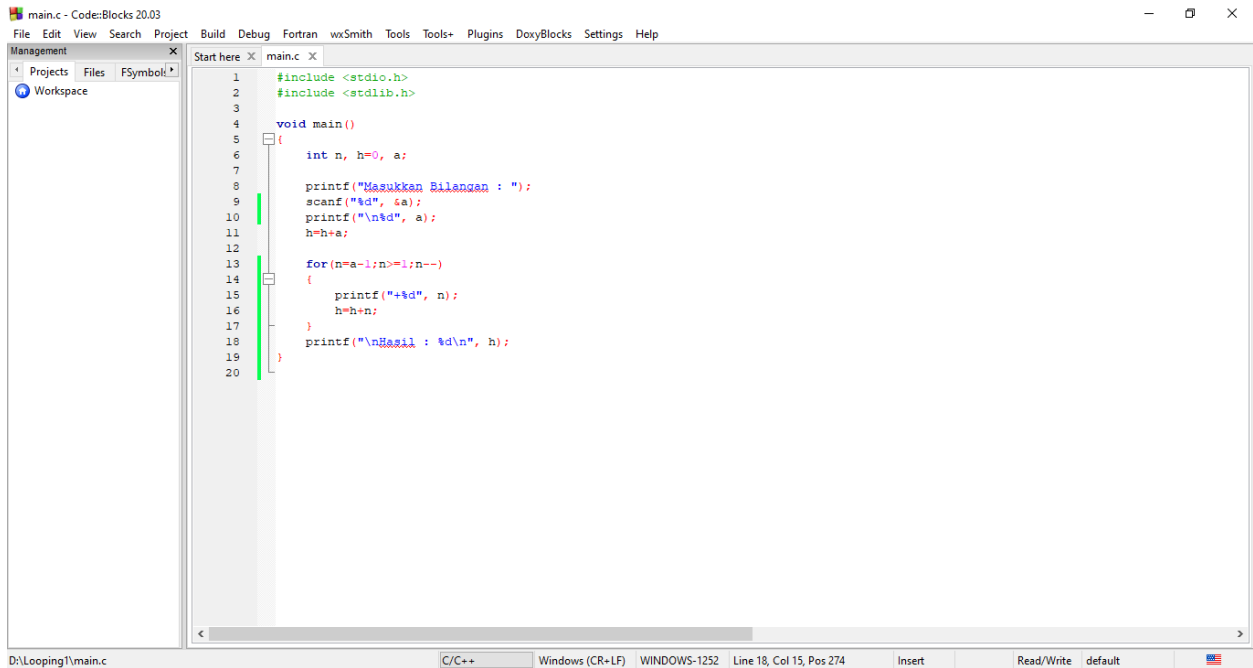


```
D:\Looping2\main.exe
Masukkan Angka :
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20

Process returned 0 (0x0)   execution time : 0.061 s
Press any key to continue.
```

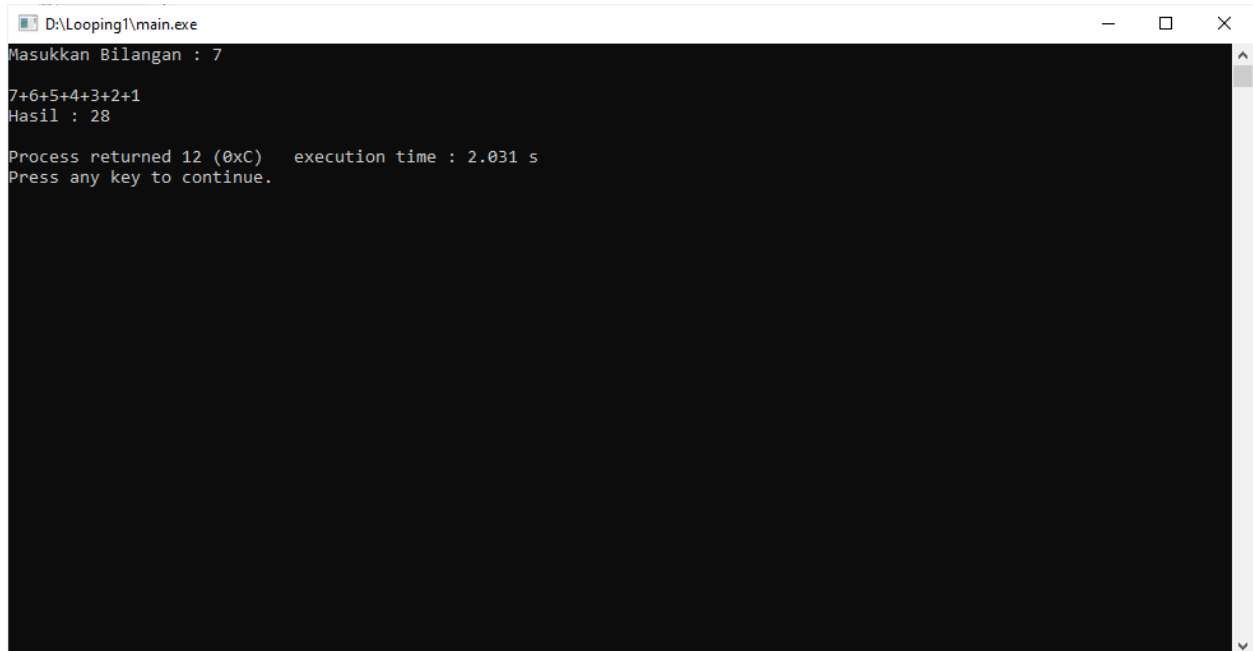
2. Hitunglah bilangan triangular dari masukan pengguna, yang dibaca dari keyboard dengan menggunakan *scanf()*. Bilangan triangular adalah penjumlahan dari bilangan masukan dengan seluruh bilangan sebelumnya, sehingga bilangan triangular dari 7 adalah : $7 + 6 + 5 + 4 + 3 + 2 + 1$

input : Masukkan sebuah bilangan : 7
output : Bilangan triangular 7 adalah 28
Jawab :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 void main()
5 {
6     int n, h=0, a;
7
8     printf("Masukkan Bilangan : ");
9     scanf("%d", &a);
10    printf("\n%d", a);
11    h=h+a;
12
13    for(n=a-1;n>=1;n--)
14    {
15        printf("+%d", n);
16        h=h+n;
17    }
18    printf("\nHasil : %d\n", h);
19 }
20
```

Output :

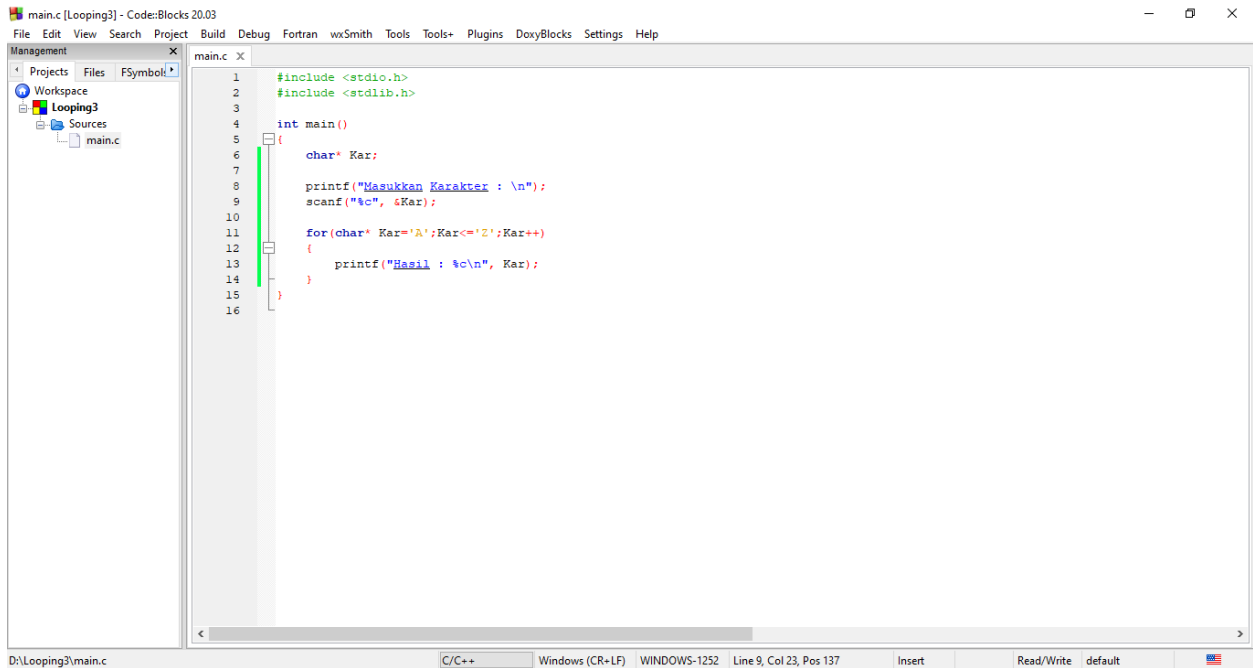


```
D:\Looping1\main.exe
Masukkan Bilangan : 7
7+6+5+4+3+2+1
Hasil : 28

Process returned 12 (0xC)   execution time : 2.031 s
Press any key to continue.
```

3. Gunakan loop *for* untuk menampilkan seluruh karakter dari A sampai dengan Z dalam baris-baris yang terpisah.

Jawab :

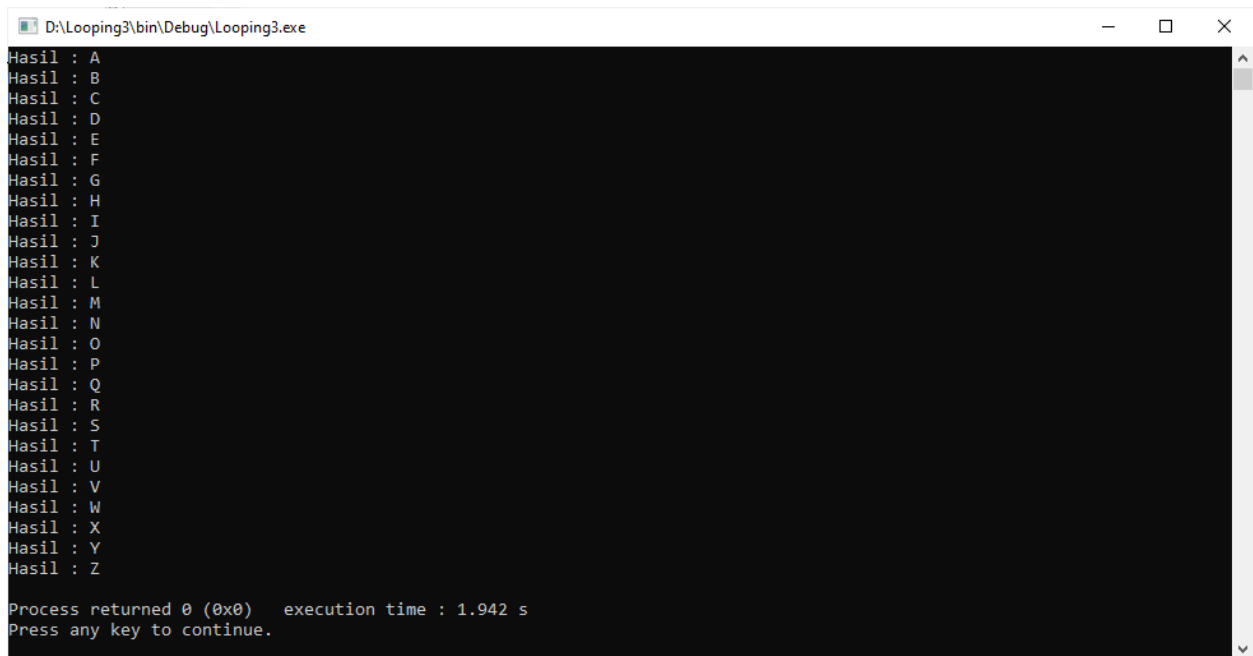


The screenshot shows the Code::Blocks IDE with a project named 'Looping3'. The main.c file contains the following C code:

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     char* Kar;
7
8     printf("Masukkan Karakter : \n");
9     scanf("%c", &Kar);
10
11     for(char* Kar='A'; Kar<='Z'; Kar++)
12     {
13         printf("Hasil : %c\n", Kar);
14     }
15 }
16
```

The status bar at the bottom indicates the file is 'D:\Looping3\main.c', the compiler is 'C/C++', the encoding is 'Windows (CR+LF)', the font is 'WINDOWS-1252', and the cursor is at 'Line 9, Col 23, Pos 137'.

Output :



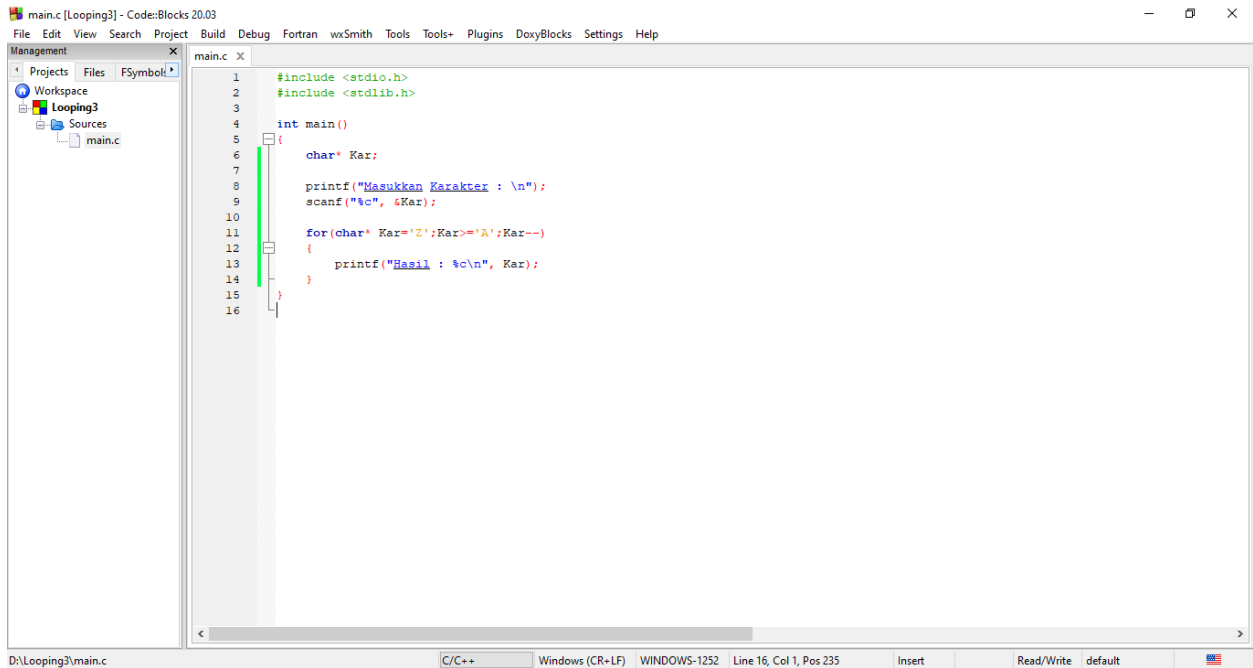
The screenshot shows a command prompt window titled 'D:\Looping3\bin\Debug\Looping3.exe'. The output of the program is as follows:

```
Hasil : A
Hasil : B
Hasil : C
Hasil : D
Hasil : E
Hasil : F
Hasil : G
Hasil : H
Hasil : I
Hasil : J
Hasil : K
Hasil : L
Hasil : M
Hasil : N
Hasil : O
Hasil : P
Hasil : Q
Hasil : R
Hasil : S
Hasil : T
Hasil : U
Hasil : V
Hasil : W
Hasil : X
Hasil : Y
Hasil : Z

Process returned 0 (0x0)   execution time : 1.942 s
Press any key to continue.
```

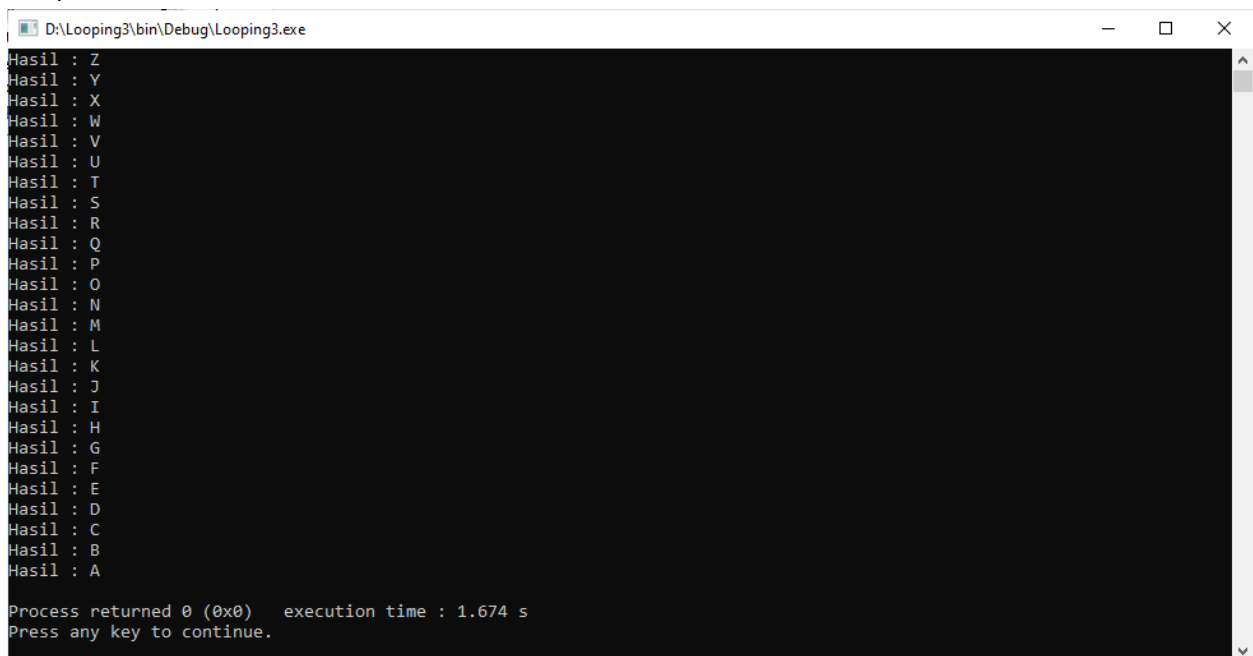
4. Gunakan loop *for* dengan kenaikan varibel negatif untuk menampilkan seluruh karakter dari Z sampai dengan A dalam baris-baris yang terpisah.

Jawab :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     char* Kar;
7
8     printf("Masukkan Karakter : \n");
9     scanf("%c", &Kar);
10
11     for(char* Kar='Z'; Kar>='A'; Kar--)
12     {
13         printf("Hasil : %c\n", Kar);
14     }
15 }
16
```

Output :



```
D:\Looping3\bin\Debug\Looping3.exe
Hasil : Z
Hasil : Y
Hasil : X
Hasil : W
Hasil : V
Hasil : U
Hasil : T
Hasil : S
Hasil : R
Hasil : Q
Hasil : P
Hasil : O
Hasil : N
Hasil : M
Hasil : L
Hasil : K
Hasil : J
Hasil : I
Hasil : H
Hasil : G
Hasil : F
Hasil : E
Hasil : D
Hasil : C
Hasil : B
Hasil : A

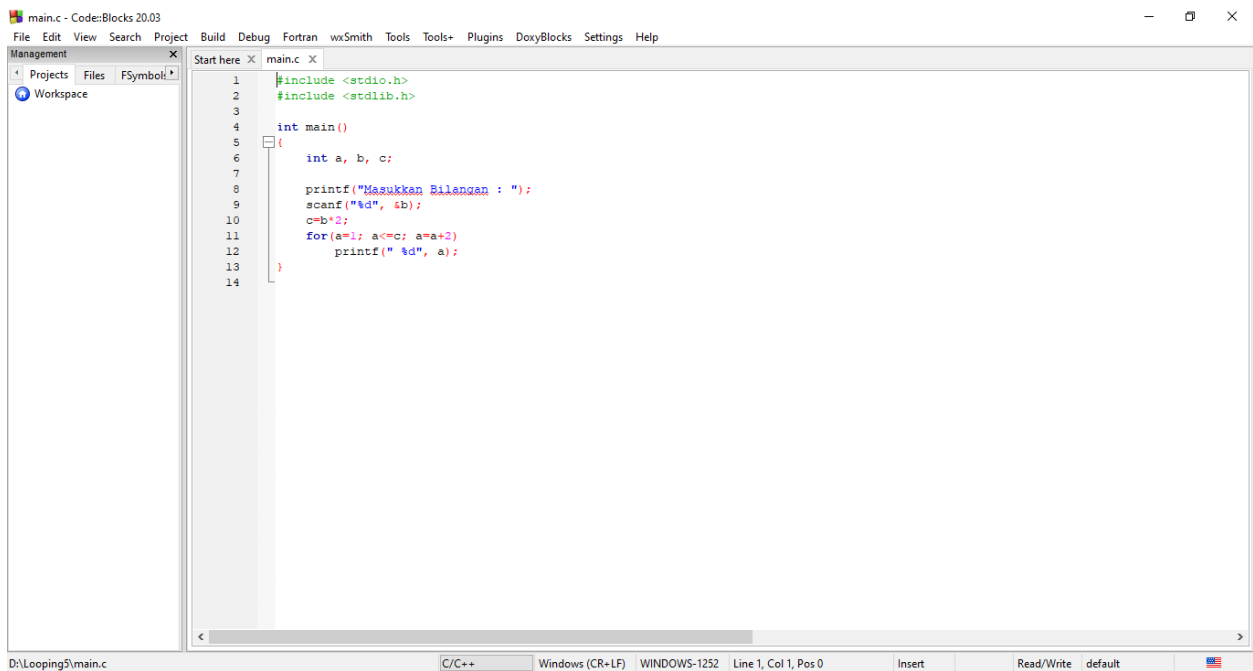
Process returned 0 (0x0)   execution time : 1.674 s
Press any key to continue.
```

5. Gunakan loop *for* untuk membuat program sebagai berikut:

input : n

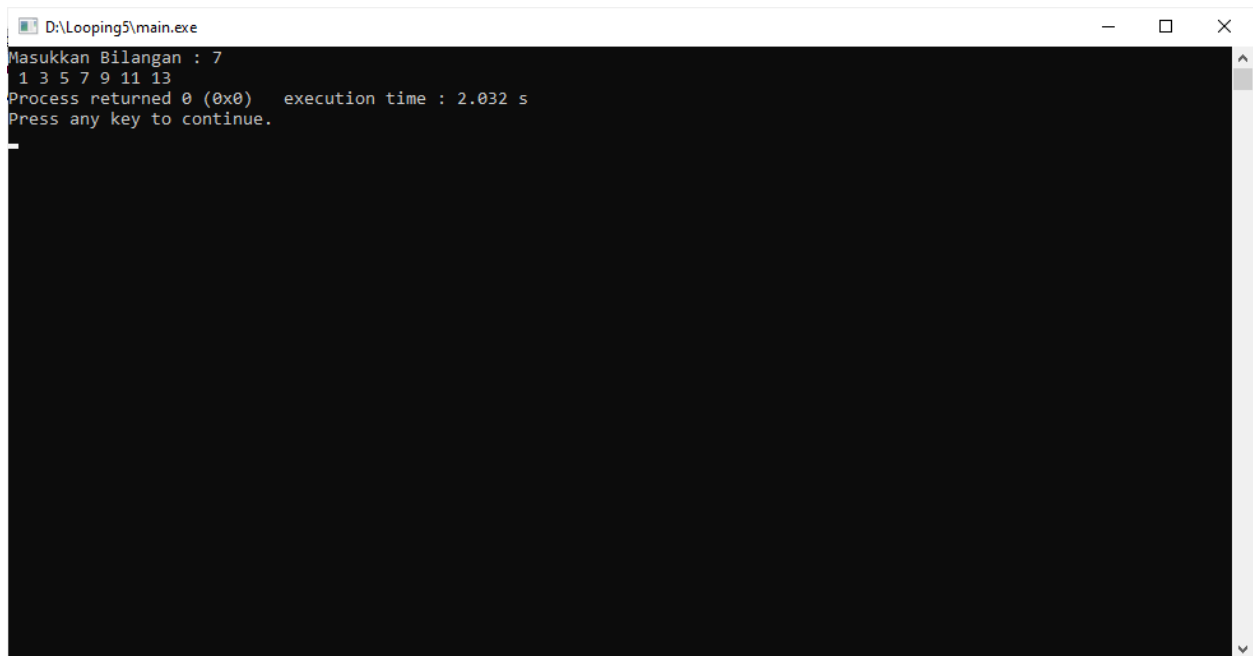
output : 1 3 5 7 ... m (m = bilangan ganjil ke n)

Jawab :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     int a, b, c;
7
8     printf("Masukkan Bilangan : ");
9     scanf("%d", &b);
10    c=b*2;
11    for(a=1; a<=c; a=a+2)
12        printf("%d", a);
13 }
14
```

Output :



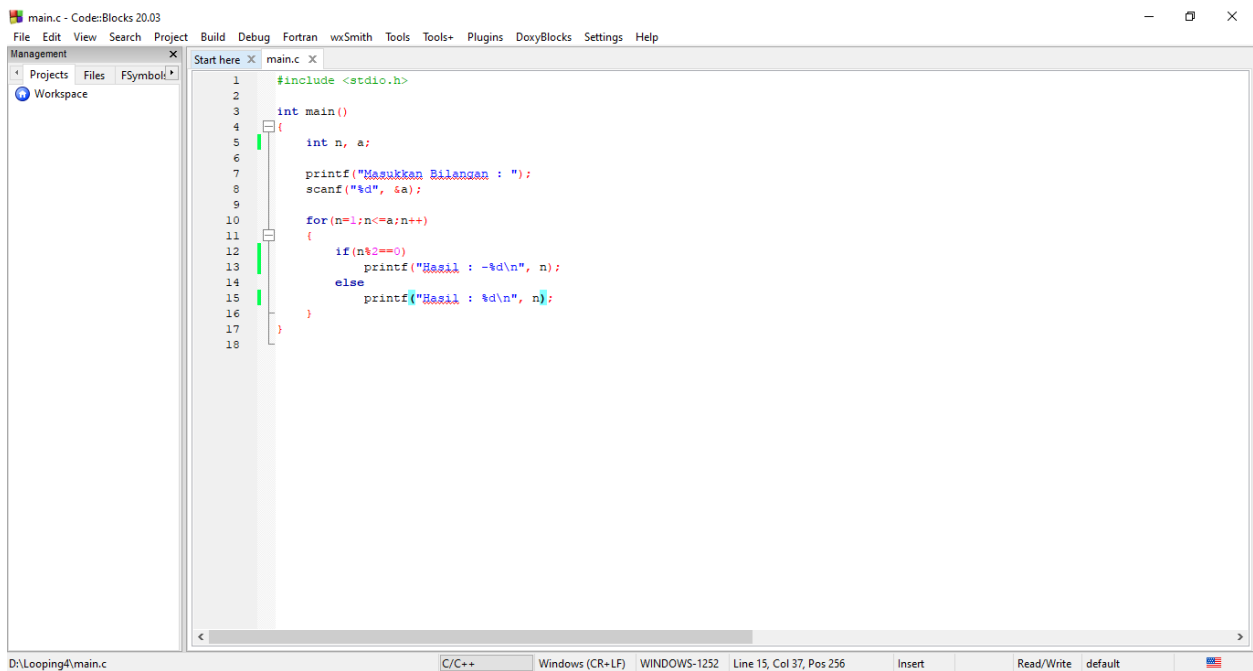
```
D:\Looping5\main.exe
Masukkan Bilangan : 7
1 3 5 7 9 11 13
Process returned 0 (0x0) execution time : 2.032 s
Press any key to continue.
```

6. Gunakan loop *for* untuk membuat program sebagai berikut:

input : n

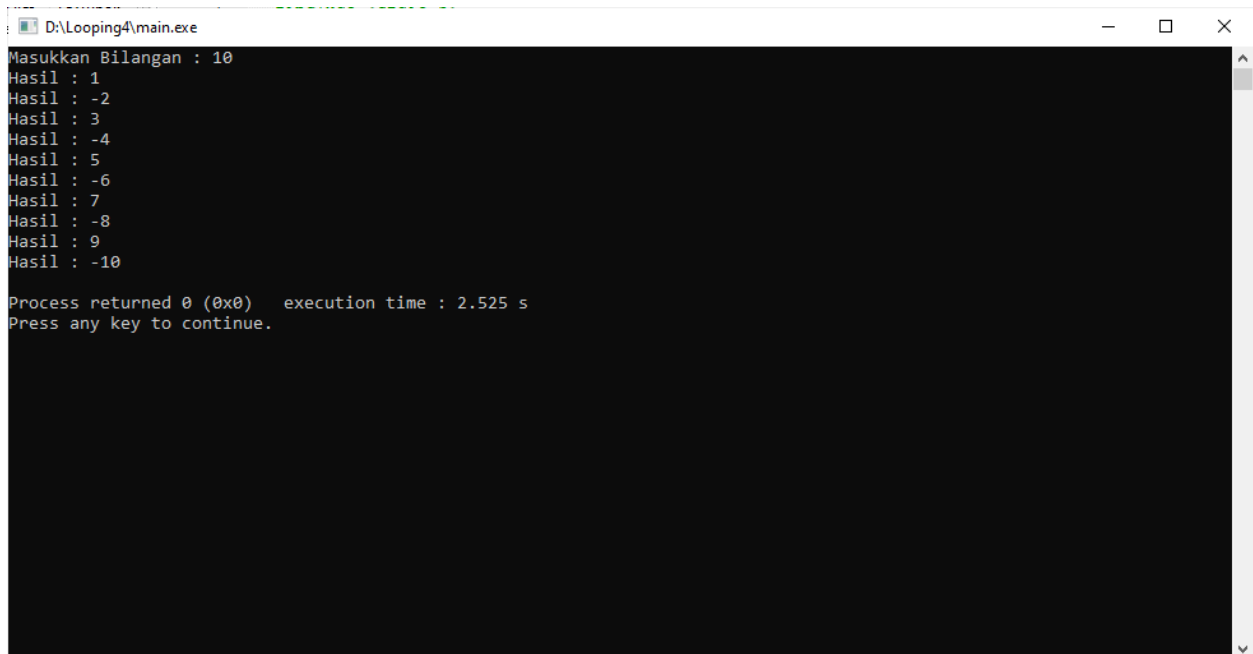
output : 1 -2 3 -4 5 -6 7 -8 ... n

Jawab :



```
1 #include <stdio.h>
2
3 int main()
4 {
5     int n, a;
6
7     printf("Masukkan Bilangan : ");
8     scanf("%d", &a);
9
10    for(n=1; n<=a; n++)
11    {
12        if(n%2==0)
13            printf("Hasil : -%d\n", n);
14        else
15            printf("Hasil : %d\n", n);
16    }
17 }
18
```

Output :



```
D:\Looping4\main.exe
Masukkan Bilangan : 10
Hasil : 1
Hasil : -2
Hasil : 3
Hasil : -4
Hasil : 5
Hasil : -6
Hasil : 7
Hasil : -8
Hasil : 9
Hasil : -10

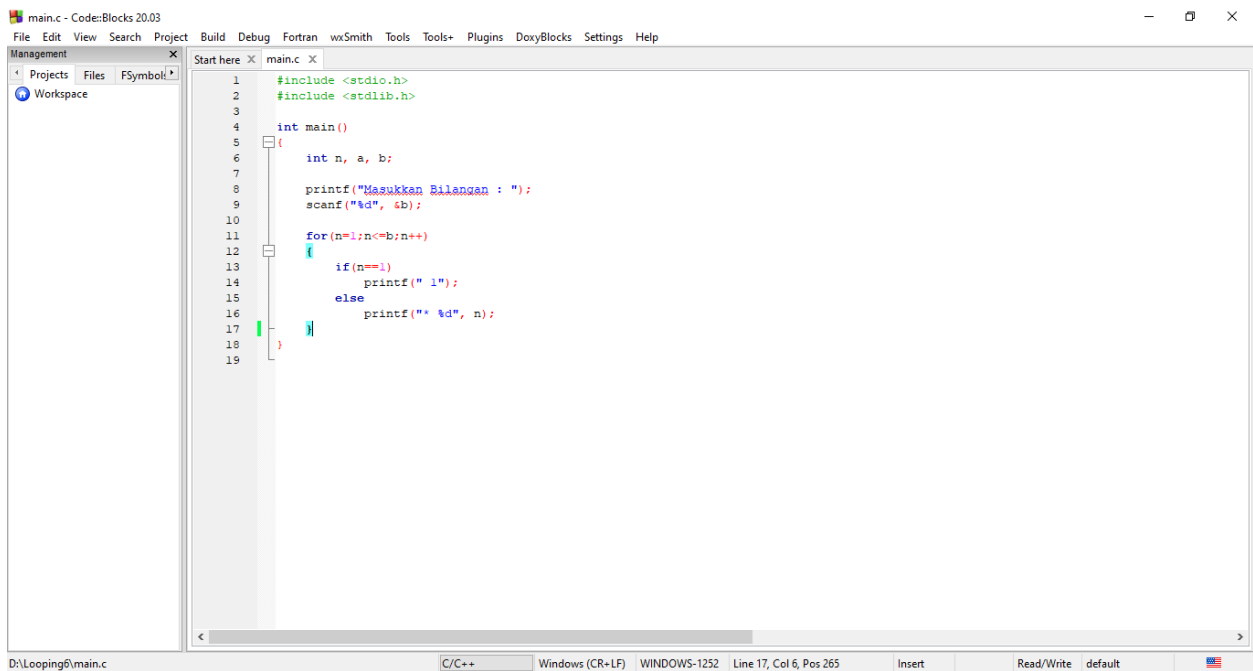
Process returned 0 (0x0)   execution time : 2.525 s
Press any key to continue.
```

7. Gunakan loop *for* untuk membuat program sebagai berikut:

input : n

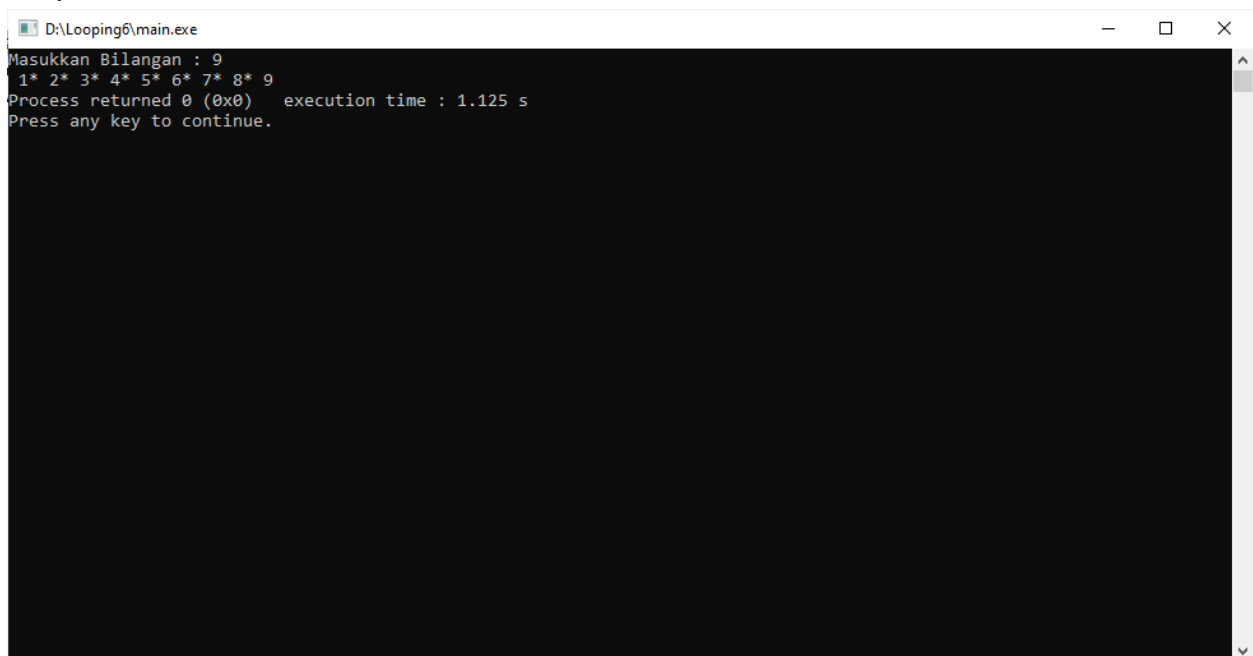
output : 1*2*3*4*5*... *n (faktorial)

Jawab :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     int n, a, b;
7
8     printf("Masukkan Bilangan : ");
9     scanf("%d", &b);
10
11     for(n=1; n<=b; n++)
12     {
13         if(n==1)
14             printf(" 1");
15         else
16             printf(" * %d", n);
17     }
18
19 }
```

Output :



```
D:\Looping6\main.exe
Masukkan Bilangan : 9
1* 2* 3* 4* 5* 6* 7* 8* 9
Process returned 0 (0x0)   execution time : 1.125 s
Press any key to continue.
```

8. Gunakan loop *while* untuk membuat program yang dapat mencari total angka yang dimasukkan dengan tampilan sebagai berikut :\

Masukkan bilangan ke-1 : 5

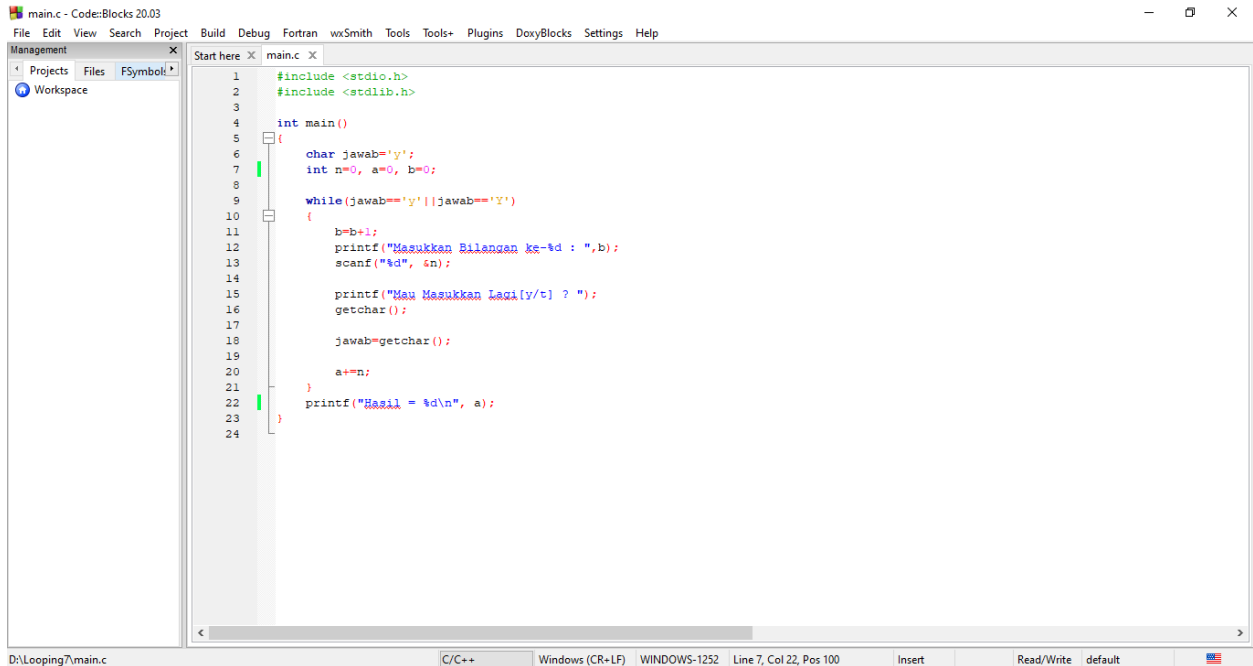
Mau memasukkan data lagi [y/t] ? y

Masukkan bilangan ke-2 : 3

Mau memasukkan data lagi [y/t] ? t

Total bilangan = 8

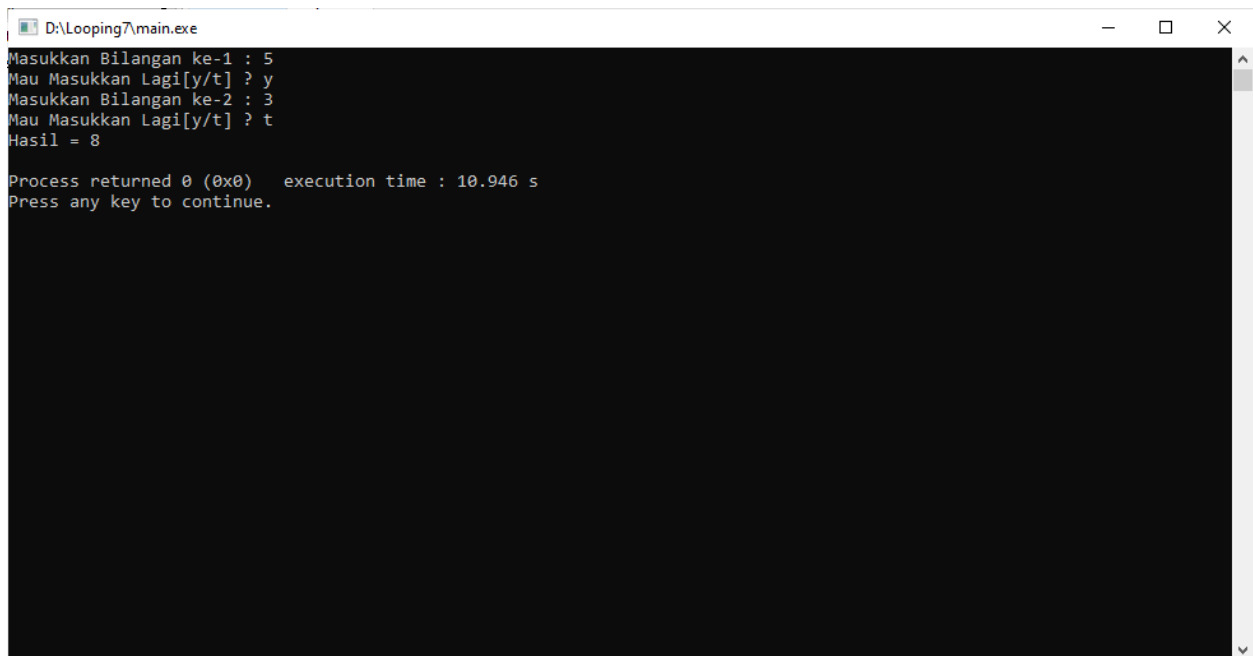
Jawab :



The screenshot shows the Code::Blocks IDE with a C program open. The program is designed to sum a series of numbers entered by the user. It includes headers for `stdio.h` and `stdlib.h`. The `main` function initializes a character variable `jawab` to 'y', and integer variables `n`, `a`, and `b` to 0. A `while` loop continues as long as `jawab` is 'y' or 'Y'. Inside the loop, `b` is incremented by 1, the user is prompted to enter a number, it is read using `scanf`, and the user is asked if they want to continue. The input number is added to `a`. After the loop, the total sum `a` is printed.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     char jawab='y';
7     int n=0, a=0, b=0;
8
9     while(jawab=='y' || jawab=='Y')
10    {
11        b=b+1;
12        printf("Masukkan Bilangan ke-%d : ",b);
13        scanf("%d", &n);
14
15        printf("Mau Masukkan Lagi[y/t] ? ");
16        getchar();
17
18        jawab=getchar();
19
20        a+=n;
21    }
22    printf("Hasil = %d\n", a);
23 }
24
```

Output :



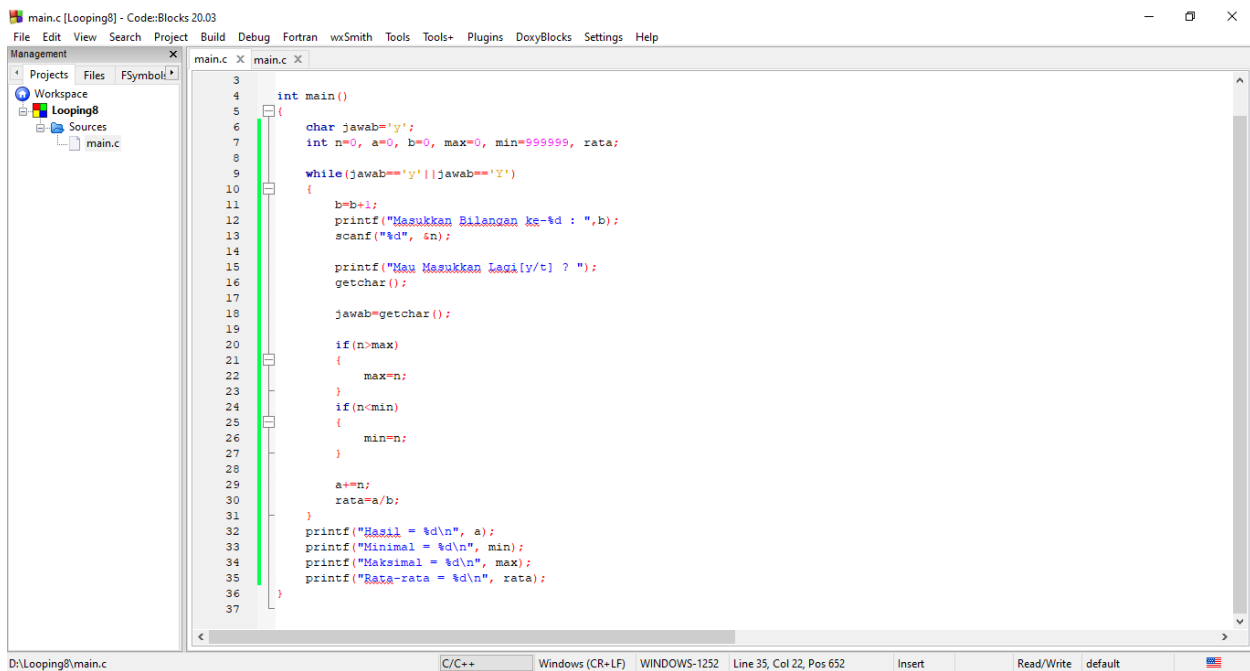
The screenshot shows a Windows command prompt window titled "D:\Looping7\main.exe". It displays the output of the program, which prompts the user to enter numbers and asks if they want to continue. The user enters 5, then y, then 3, then t. The program outputs "Hasil = 8". At the bottom, it shows "Process returned 0 (0x0) execution time : 10.946 s" and "Press any key to continue."

```
D:\Looping7\main.exe
Masukkan Bilangan ke-1 : 5
Mau Masukkan Lagi[y/t] ? y
Masukkan Bilangan ke-2 : 3
Mau Masukkan Lagi[y/t] ? t
Hasil = 8

Process returned 0 (0x0)   execution time : 10.946 s
Press any key to continue.
```

9. Pada program no 8 tambahkan penghitungan rata-rata, maksimum dan minimum dari angka yang dimasukkan.

Jawab :

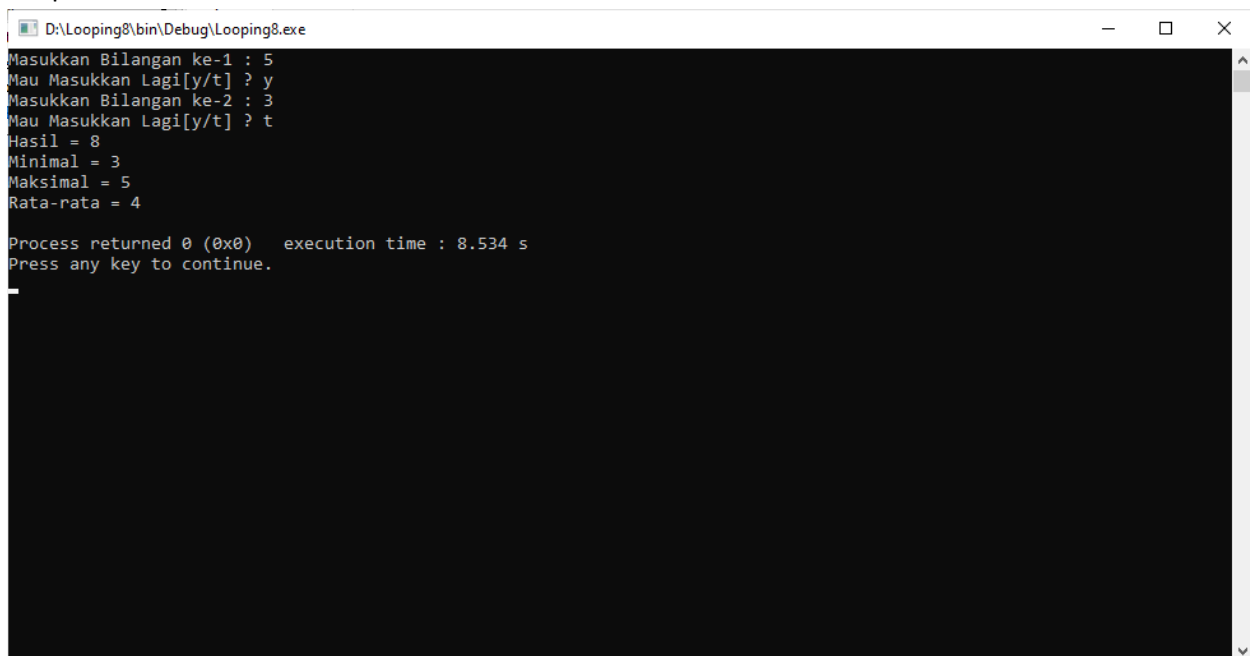


```
main.c [Looping8] - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

Management
  Projects Files FSymbol
  Workspace
  Looping8
    Sources
      main.c

main.c
3
4 int main()
5 {
6     char jawab='y';
7     int n=0, a=0, b=0, max=0, min=999999, rata;
8
9     while(jawab=='y' || jawab=='Y')
10    {
11        b=b+1;
12        printf("Masukkan Bilangan ke-%d : ",b);
13        scanf("%d", &n);
14
15        printf("Mau Masukkan Lagi[y/t] ? ");
16        getchar();
17
18        jawab=getchar();
19
20        if(n>max)
21        {
22            max=n;
23        }
24        if(n<min)
25        {
26            min=n;
27        }
28
29        a+=n;
30        rata=a/b;
31    }
32    printf("Hasil = %d\n", a);
33    printf("Minimal = %d\n", min);
34    printf("Maksimal = %d\n", max);
35    printf("Rata-rata = %d\n", rata);
36 }
37
```

Output :



```
D:\Looping8\bin\Debug\Looping8.exe
Masukkan Bilangan ke-1 : 5
Mau Masukkan Lagi[y/t] ? y
Masukkan Bilangan ke-2 : 3
Mau Masukkan Lagi[y/t] ? t
Hasil = 8
Minimal = 3
Maksimal = 5
Rata-rata = 4

Process returned 0 (0x0)   execution time : 8.534 s
Press any key to continue.
```

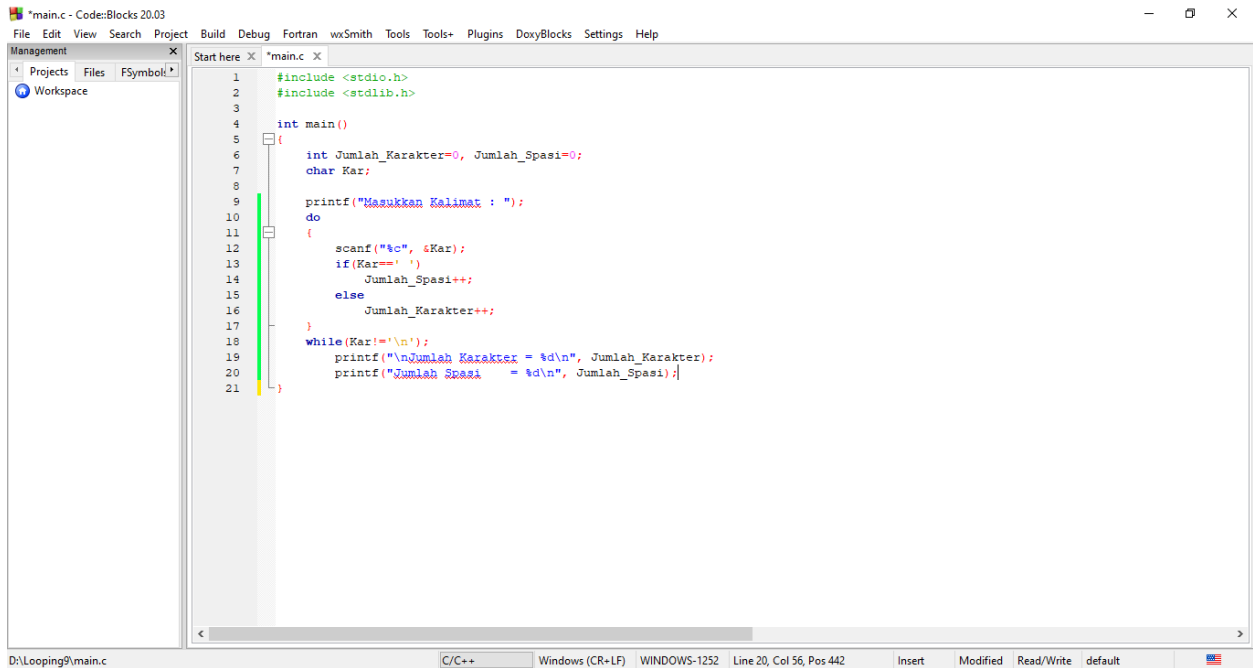
10. Gunakan `while` pada program yang digunakan untuk menghitung banyaknya karakter dari kalimat yang dimasukkan melalui keyboard (termasuk karakter spasi). Untuk mengakhiri pemasukan kalimat, tombol ENTER (`\n`) harus ditekan

Input : Ketikkan sembarang kalimat, akhiri dengan enter

Output: jumlah karakter = m

jumlah spasi = n

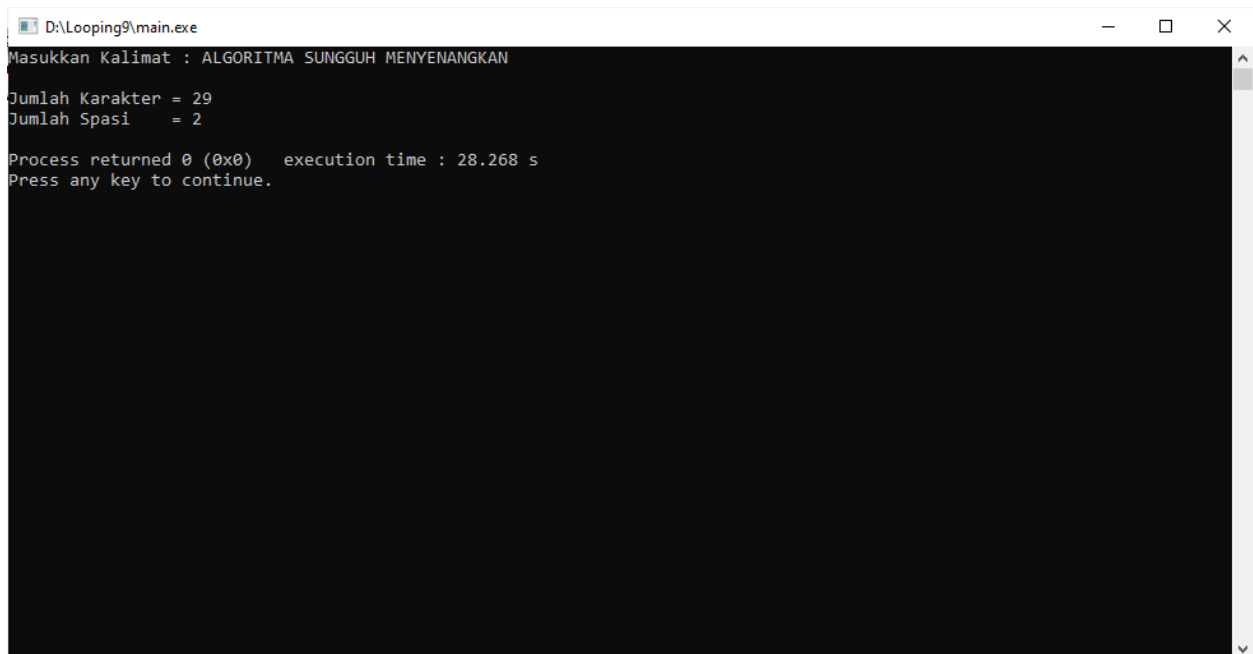
Jawab :



The screenshot shows the Code::Blocks IDE with a C program open. The program is designed to count the number of characters and spaces in a user-input string. It includes headers for `stdio.h` and `stdlib.h`. The `main` function initializes two counters, `Jumlah_Karakter` and `Jumlah_Spasi`, to zero. It prompts the user to enter a string using `printf`. A `do-while` loop reads the input character by character using `scanf`. Inside the loop, it checks if the character is a space. If it is, it increments `Jumlah_Spasi`; otherwise, it increments `Jumlah_Karakter`. After the loop ends, it prints the final counts using `printf`.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main()
5 {
6     int Jumlah_Karakter=0, Jumlah_Spasi=0;
7     char Kar;
8
9     printf("Masukkan Kalimat : ");
10    do
11    {
12        scanf("%c", &Kar);
13        if(Kar==' ')
14            Jumlah_Spasi++;
15        else
16            Jumlah_Karakter++;
17    }
18    while(Kar!='\n');
19    printf("\nJumlah Karakter = %d\n", Jumlah_Karakter);
20    printf("Jumlah Spasi = %d\n", Jumlah_Spasi);
21 }
```

Output :



The screenshot shows a Windows command prompt window titled "D:\Looping9\main.exe". It displays the output of the program after running. The user input "ALGORITMA SUNGGUH MENYENANGKAN" is shown. The program outputs the character count as 29 and the space count as 2. It also shows the process return code as 0 (0x0) and the execution time as 28.268 seconds. The prompt "Press any key to continue." is visible at the bottom.

```
D:\Looping9\main.exe
Masukkan Kalimat : ALGORITMA SUNGGUH MENYENANGKAN

Jumlah Karakter = 29
Jumlah Spasi    = 2

Process returned 0 (0x0)   execution time : 28.268 s
Press any key to continue.
```