

TUGAS PRAKTIKUM KONSEP PEMROGRAMAN

JILID 10 part 1



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

Praktikum 6 (1/2)

ARRAY

1. Deklarasikan sebuah variabel array of int, selanjutnya isi array tsb kemudian tampilkan isi variabel tersebut menggunakan statement for(). Ingat apabila jumlah deklarasi index adalah n maka nilai index-nya adalah dimulai dari 0 sampai dengan n-1.

Jawab :

```
#include <stdio.h>
#include <stdlib.h>
#define maks 100

void input(int);
int main()
{
    int n;
    printf("Masukkan batas n: ");
    scanf("%d", &n);
    input(n);
    getch();
}
void input(int n)
{
    int a[maks],i;
    for(i=0;i<n;i++)
    {
        printf("Masukkan data ke-%d=", i+1);
        scanf("%d", &a[i]);
    }
}
```

Listing Program :

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #define maks 100
4
5 void input(int);
6 int main()
7 {
8     int n;
9     printf("Masukkan batas n: ");
10    scanf("%d", &n);
11    input(n);
12    getch();
13 }
14 void input(int n)
15 {
16     int a[maks], i;
17     for(i=0; i<n; i++)
18     {
19         printf("Masukkan data ke-%d=", i+1);
20         scanf("%d", &a[i]);
21     }
22 }
23
```

Output :

```
D:\Array1\bin\Debug\Array1.exe
Masukkan batas n: 5
Masukkan data ke-1=10
Masukkan data ke-2=20
Masukkan data ke-3=30
Masukkan data ke-4=40
Masukkan data ke-5=50

Process returned 0 (0x0)   execution time : 36.830 s
Press any key to continue.
```

2. Deret fibonacci adalah deret yang dimulai dengan dua angka yang bernilai 0 dan 1, kemudian deret ketiga ditentukan dari penjumlahan kedua angka tersebut, sedangkan deret keempat ditentukan dari dua angka sebelumnya begitu seterusnya. Sehingga didapatkan deret fibonanci sebagai berikut: 0 1 1 2 3 5 8 13 21 ... Buatlah program untuk meminta input dari user berupa sebuah bilangan, kemudian tampilkan deret fibonacci mulai dari 1 sampai dengan bilangan tsb.
Jawab :

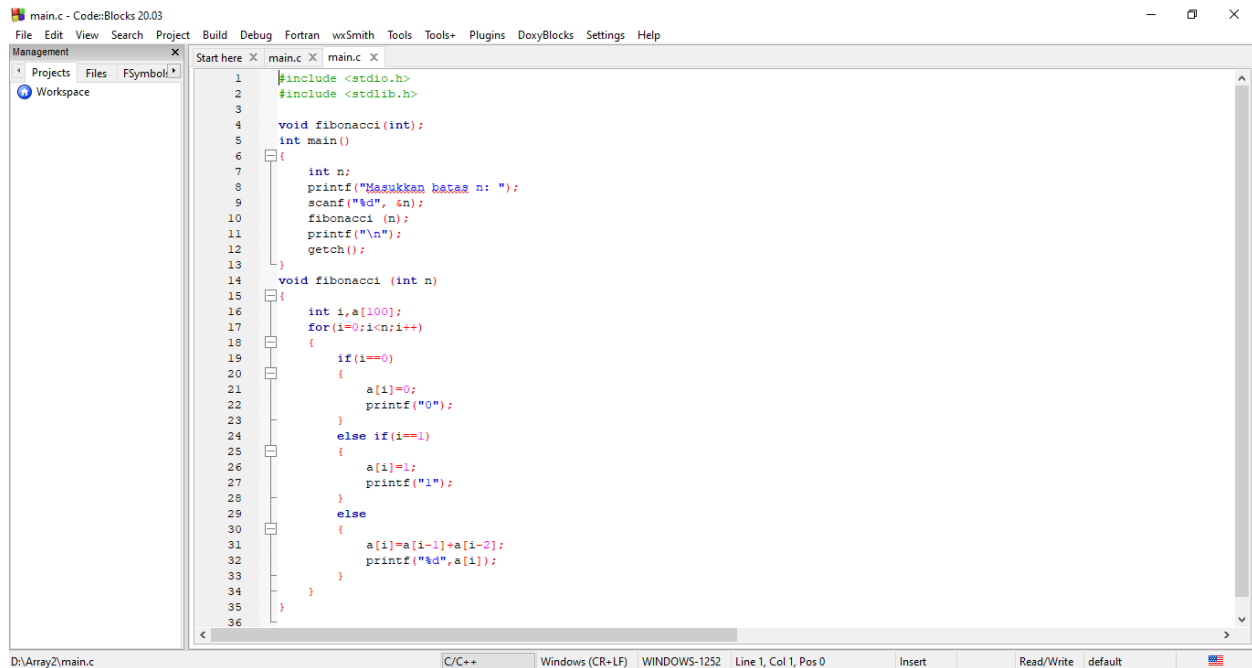
```

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

void fibonacci(int);
int main()
{
    int n;
    printf("Masukkan batas n: ");
    scanf("%d", &n);
    fibonacci (n);
    printf("\n");
    getch();
}
void fibonacci (int n)
{
    int i,a[100];
    for(i=0;i<n;i++)
    {
        if(i==0)
        {
            a[i]=0;
            printf("0");
        }
        else if(i==1)
        {
            a[i]=1;
            printf("1");
        }
        else
        {
            a[i]=a[i-1]+a[i-2];
            printf("%d",a[i]);
        }
    }
}

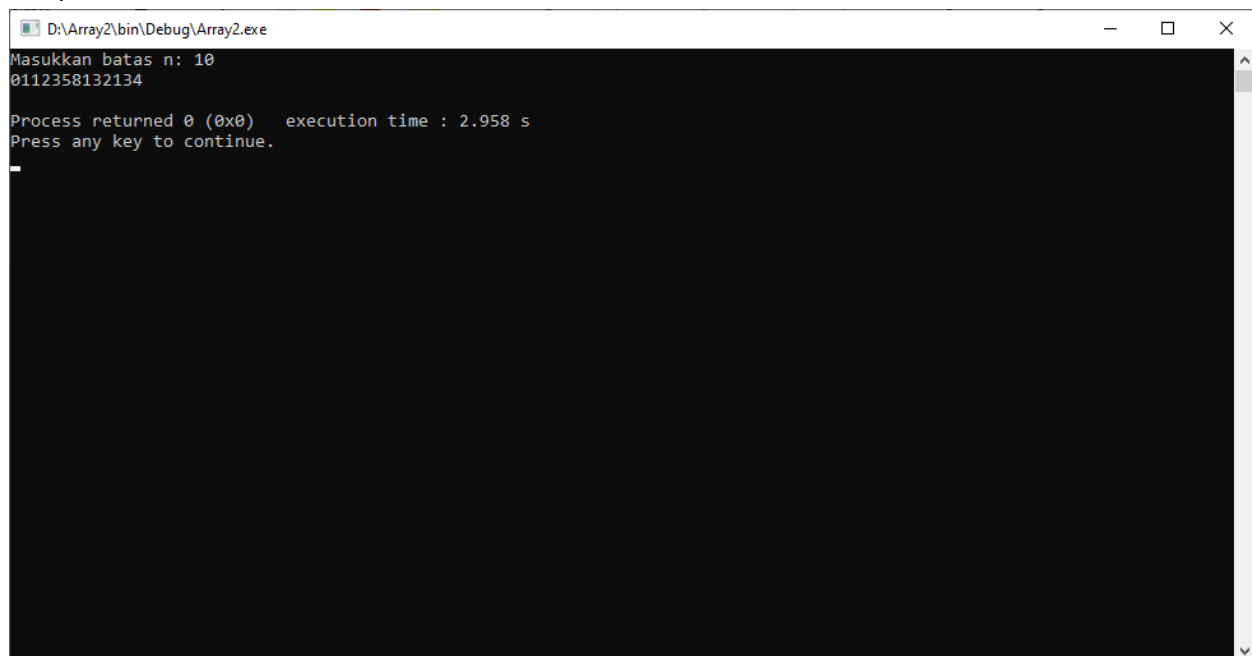
```

Listing Program :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 void fibonaccii(int);
5 int main()
6 {
7     int n;
8     printf("Masukkan batas n: ");
9     scanf("%d", &n);
10    fibonaccii(n);
11    printf("\n");
12    getch();
13 }
14 void fibonaccii (int n)
15 {
16     int i, a[100];
17     for(i=0; i<n; i++)
18     {
19         if(i==0)
20         {
21             a[i]=0;
22             printf("0");
23         }
24         else if(i==1)
25         {
26             a[i]=1;
27             printf("1");
28         }
29         else
30         {
31             a[i]=a[i-1]+a[i-2];
32             printf("%d", a[i]);
33         }
34     }
35 }
36
```

Output :



```
D:\Array2\bin\Debug\Array2.exe
Masukkan batas n: 10
0112358132134
Process returned 0 (0x0)   execution time : 2.958 s
Press any key to continue.
```

3. Buat penjumlahan dua matriks $A[2][2]$ dengan $B[2][2]$. Hasil penjumlahan tersebut disimpan dalam matrik C. Tampilkan di layar masing-masing isi dari matriks A, B dan C Catatan modifikasi (gunakan nested loop) :

- indeks MAKS dibuat konstanta
- variabel ordo matriks diinputkan
- memungkinkan perulangan

Jawab :

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

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

```
#define maxcol 10
```

```
#define maxrow 10
```

```
void main()
```

```
{
```

```
    int a,b,A[2][2],B[2][2], hasjum[2][2], baris, kolom;
```

```
    printf("Matriks A\n");
```

```
    for(a=0;a<2;a++)
```

```
    {
```

```
        for(b=0;b<2;b++)
```

```
        {
```

```
            printf("Masukkan Baris %d dan Kolom %d : ",a+1,b+1);
```

```
            scanf("%d",&A[a][b]);
```

```
        }
```

```
    }
```

```
    printf("\nMatriks B\n");
```

```
    for(a=0;a<2;a++)
```

```
    {
```

```
        for(b=0;b<2;b++)
```

```
        {
```

```
            printf("Masukkan Baris %d dan Kolom %d : ",a+1,b+1);
```

```
            scanf("%d",&B[a][b]);
```

```
        }
```

```
    }
```

```
    printf("\n");
```

```
    for(baris=0;baris<2;baris++)
```

```
    {
```

```
        for(kolom=0;kolom<2;kolom++)
```

```
        {
```

```
            hasjum[baris][kolom] = A[baris][kolom]+B[baris][kolom];
```

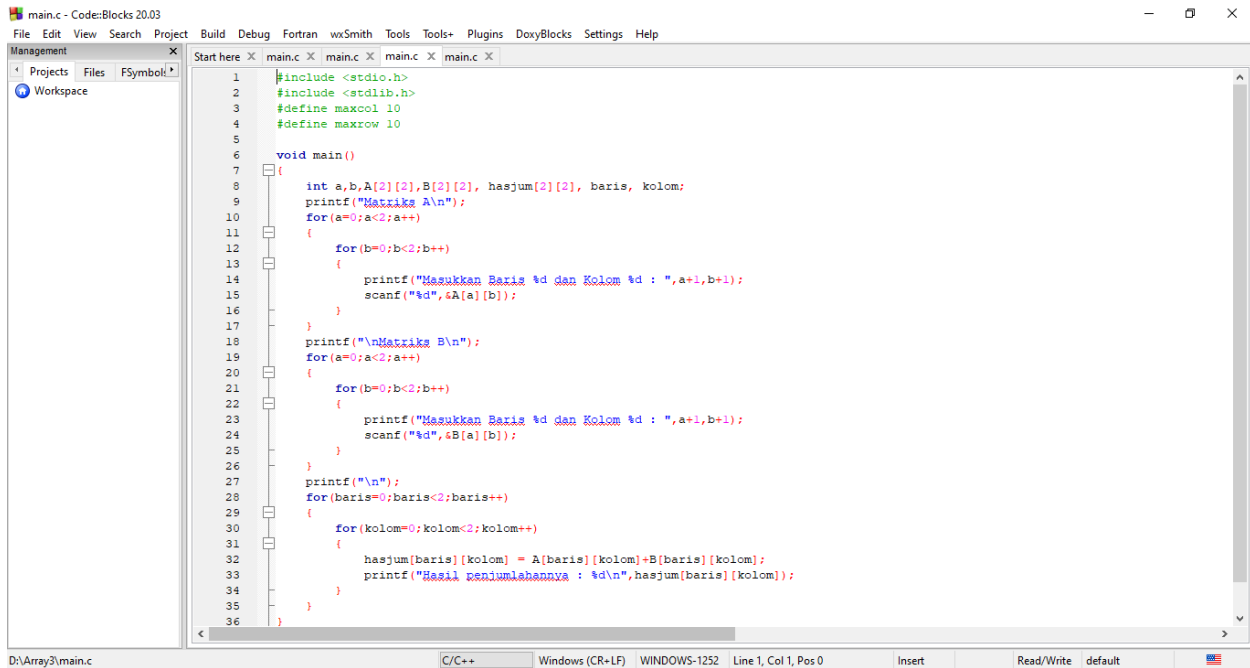
```
            printf("Hasil penjumlahannya : %d\n",hasjum[baris][kolom]);
```

```
        }
```

```
    }
```

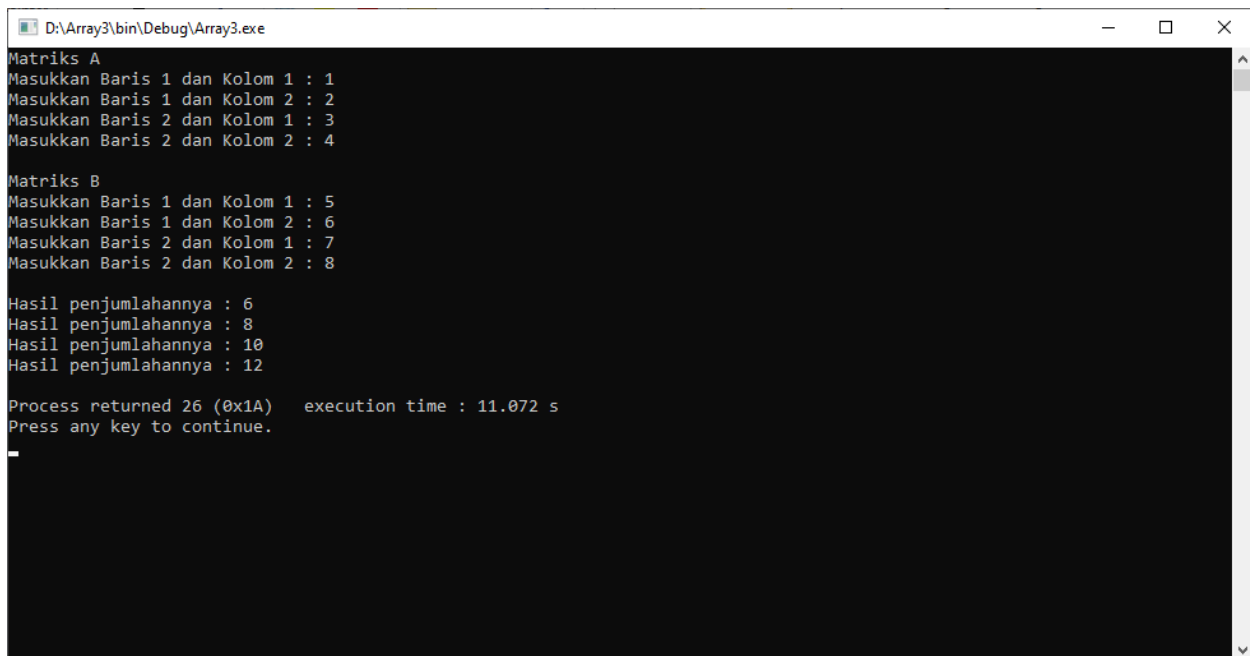
```
}
```

Listing Program :



```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #define maxcol 10
4 #define maxrow 10
5
6 void main()
7 {
8     int a,b,A[2][2],B[2][2], hasjum[2][2], baris, kolom;
9     printf("Matriks A\n");
10    for(a=0;a<2;a++)
11    {
12        for(b=0;b<2;b++)
13        {
14            printf("Masukkan Baris %d dan Kolom %d : ",a+1,b+1);
15            scanf("%d",&A[a][b]);
16        }
17    }
18    printf("\nMatriks B\n");
19    for(a=0;a<2;a++)
20    {
21        for(b=0;b<2;b++)
22        {
23            printf("Masukkan Baris %d dan Kolom %d : ",a+1,b+1);
24            scanf("%d",&B[a][b]);
25        }
26    }
27    printf("\n");
28    for(baris=0;baris<2;baris++)
29    {
30        for(kolom=0;kolom<2;kolom++)
31        {
32            hasjum[baris][kolom] = A[baris][kolom]+B[baris][kolom];
33            printf("Hasil penjumlahannya : %d\n",hasjum[baris][kolom]);
34        }
35    }
36 }
```

Output :



```
D:\Array3\bin\Debug\Array3.exe
Matriks A
Masukkan Baris 1 dan Kolom 1 : 1
Masukkan Baris 1 dan Kolom 2 : 2
Masukkan Baris 2 dan Kolom 1 : 3
Masukkan Baris 2 dan Kolom 2 : 4

Matriks B
Masukkan Baris 1 dan Kolom 1 : 5
Masukkan Baris 1 dan Kolom 2 : 6
Masukkan Baris 2 dan Kolom 1 : 7
Masukkan Baris 2 dan Kolom 2 : 8

Hasil penjumlahannya : 6
Hasil penjumlahannya : 8
Hasil penjumlahannya : 10
Hasil penjumlahannya : 12

Process returned 26 (0x1A)   execution time : 11.072 s
Press any key to continue.
```

4. Buatlah sebuah variabel array of int yang memiliki ukuran MAKS. Dengan menggunakan looping for(), masukkan sebanyak n data ke dalam array tsb kemudian carilah nilai terbesar yang ada dalam array tsb.

Jawab :

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

```

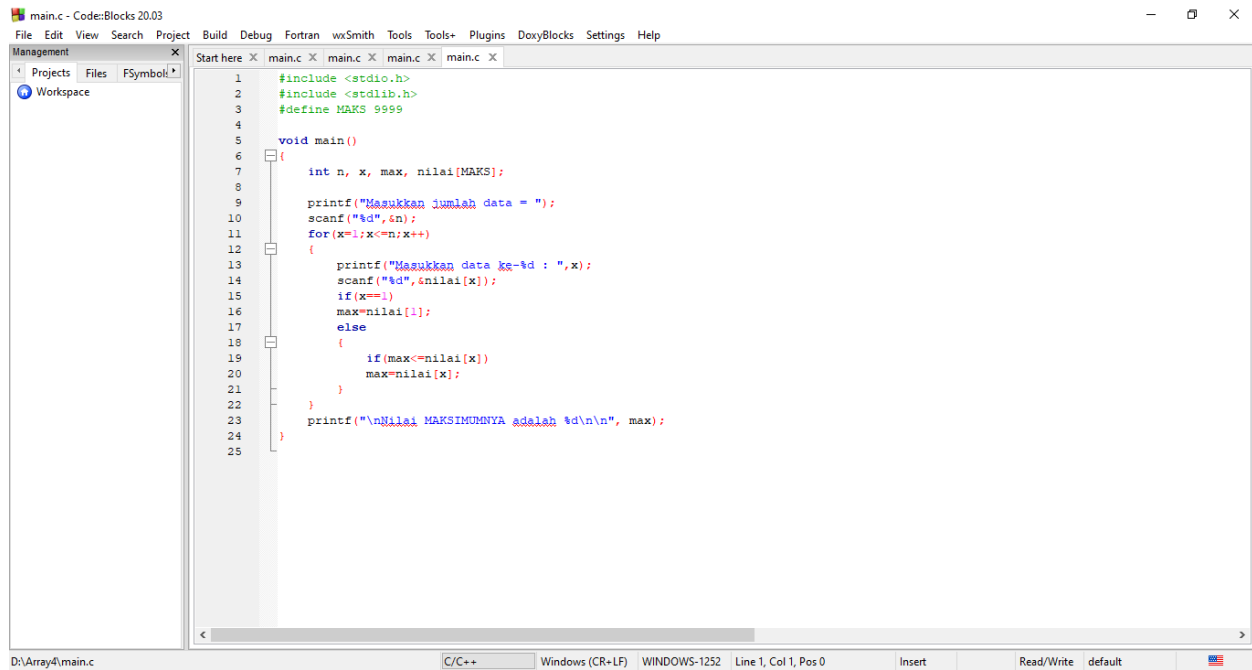
#include <stdlib.h>
#define MAKS 9999

void main()
{
    int n, x, max, nilai[MAKS];

    printf("Masukkan jumlah data = ");
    scanf("%d",&n);
    for(x=1;x<=n;x++)
    {
        printf("Masukkan data ke-%d : ",x);
        scanf("%d",&nilai[x]);
        if(x==1)
            max=nilai[1];
        else
        {
            if(max<=nilai[x])
                max=nilai[x];
        }
    }
    printf("\nNilai MAKSIMUMNYA adalah %d\n\n", max);
}

```

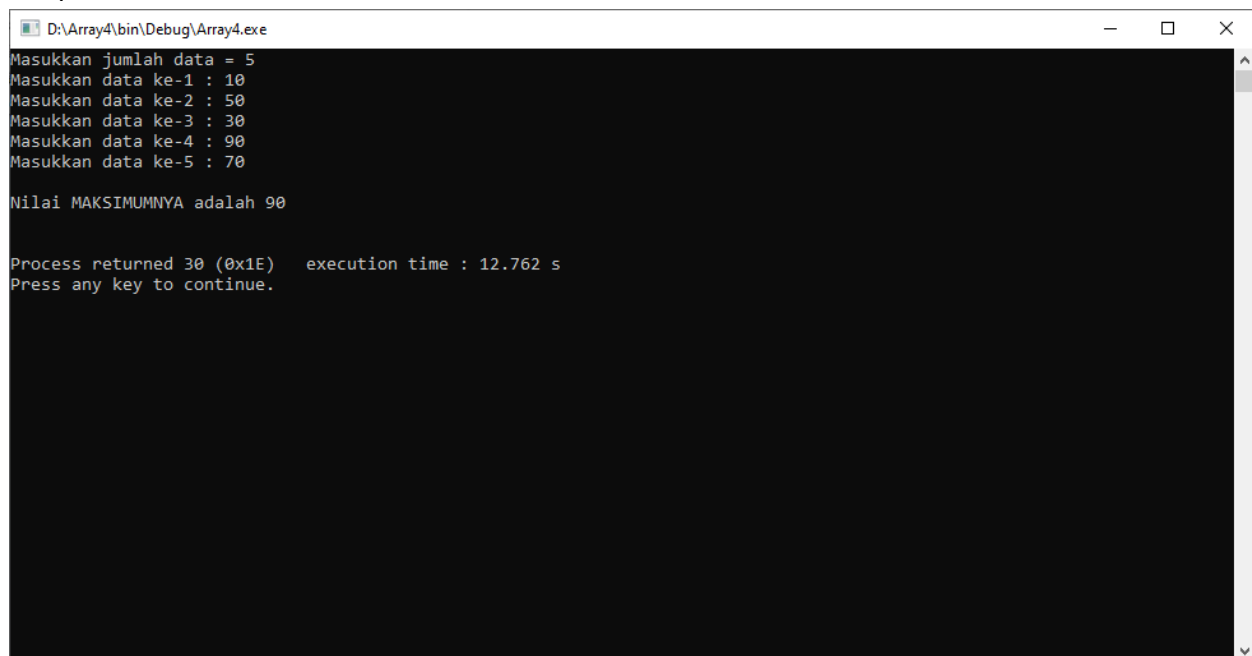
Listing Program :



The screenshot shows the Code::Blocks IDE with a C program open. The program is designed to find the maximum value in an array of size 9999. It prompts the user to enter the number of data points (n), then iteratively asks for each data point and updates the maximum value found so far. The final output is the maximum value.

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #define MAKS 9999
4
5 void main()
6 {
7     int n, x, max, nilai[MAKS];
8
9     printf("Masukkan jumlah data = ");
10    scanf("%d", &n);
11    for(x=1; x<=n; x++)
12    {
13        printf("Masukkan data ke-%d : ", x);
14        scanf("%d", &nilai[x]);
15        if(x==1)
16            max=nilai[1];
17        else
18        {
19            if(max<nilai[x])
20                max=nilai[x];
21        }
22    }
23    printf("\nNilai MAKSIMUMNYA adalah %d\n", max);
24 }
25
```

Output :



The screenshot shows a Windows command prompt window titled "D:\Array4\bin\Debug\Array4.exe". It displays the output of the program, which matches the input and output shown in the IDE screenshot. The program returns 30 (0x1E) and the execution time is 12.762 s.

```
D:\Array4\bin\Debug\Array4.exe
Masukkan jumlah data = 5
Masukkan data ke-1 : 10
Masukkan data ke-2 : 50
Masukkan data ke-3 : 30
Masukkan data ke-4 : 90
Masukkan data ke-5 : 70

Nilai MAKSIMUMNYA adalah 90

Process returned 30 (0x1E)   execution time : 12.762 s
Press any key to continue.
```