

Modul V

Array dan Koleksi

Membuat contoh program untuk penjumlahan matriks.

```

Contoh-1
Jumlah Baris Matriks = 3
Jumlah Kolom Matriks = 2

Input Matriks A :
Nilai A [0, 0] = 1
Nilai A [0, 1] = 3
Nilai A [1, 0] = 5
Nilai A [1, 1] = 7
Nilai A [2, 0] = 9
Nilai A [2, 1] = 11

Input Matriks B :
Nilai B [0, 0] = 2
Nilai B [0, 1] = 4
Nilai B [1, 0] = 6
Nilai B [1, 1] = 8
Nilai B [2, 0] = 10
Nilai B [2, 1] = 12

Hasil Matriks C = A + B :
      3  7
      11 15
      19 23
  
```

Langkah-langkah untuk membuatnya :

1. Buat sebuah *Solution* dan *Project* baru dengan nama :
 Name = "Contoh_1"
 Location = <sesuai instruksi Dosen / Asisten Dosen>
 Solution Name = "Modul V"
2. Ketikkan kode program berikut ke dalam blok **Main** dari **Program.cs** pada *Project* "Contoh_1" :


```

int[,] A, B, C;
byte brs, klm;
Console.WriteLine("Jumlah Baris Matriks = ");
brs = Convert.ToByte(Console.ReadLine());
Console.WriteLine("Jumlah Kolom Matriks = ");
klm = Convert.ToByte(Console.ReadLine());
A = new int[brs, klm];
      
```

```

B = new int[brs, klm];
C = new int[brs, klm];
Console.WriteLine("\nInput Matriks A :");
for (int i = 0; i <= A.GetUpperBound(0); i++)
    for (int j = 0; j <= A.GetUpperBound(1); j++)
    {
        Console.Write("Nilai A [{0}, {1}] = ", i, j);
        A[i, j] = Convert.ToInt32(Console.ReadLine());
    }
Console.WriteLine("\nInput Matriks B :");
for (int i = 0; i <= B.GetUpperBound(0); i++)
    for (int j = 0; j <= B.GetUpperBound(1); j++)
    {
        Console.Write("Nilai B [{0}, {1}] = ", i, j);
        B[i, j] = Convert.ToInt32(Console.ReadLine());
    }
for (int i = 0; i <= B.GetUpperBound(0); i++)
    for (int j = 0; j <= B.GetUpperBound(1); j++)
        C[i, j] = A[i, j] + B[i, j];
Console.Clear();
Console.WriteLine("Hasil Matriks C = A + B :");
for (int i = 0; i <= B.GetUpperBound(0); i++)
{
    Console.Write("\t|");
    for (int j = 0; j <= B.GetUpperBound(1); j++)
        Console.Write("{0,5:D}", C[i, j]);
    Console.WriteLine(" |");
}
Console.ReadKey();

```

Catatan :

Semua fungsi dari kode program di atas dapat dilihat pada slide modul perkuliahan “Sesi 6 – Array dan Koleksi.pptx”.

3. Untuk melihat hasil eksekusi dari program ini, dapat melalui menu **Debug / Start Debugging**, atau dengan shortcut **F5**

Membuat contoh program sederhana untuk menginput sejumlah angka.

```

Contoh-2
Input Sembarang Angka = 10
Mau Input Data Lagi? (Y/T) = y

Input Sembarang Angka = 90
Mau Input Data Lagi? (Y/T) = y

Input Sembarang Angka = 30
Mau Input Data Lagi? (Y/T) = y

Input Sembarang Angka = -70
Mau Input Data Lagi? (Y/T) = y

Input Sembarang Angka = 50
Mau Input Data Lagi? (Y/T) = y

Input Sembarang Angka = 200
Mau Input Data Lagi? (Y/T) = t

Anda telah Menginput Sejumlah Angka Yakni :
10 90 30 -70 50 200

```

Langkah-langkah untuk membuatnya :

1. Buat sebuah *Solution* dan *Project* baru dengan nama :
 Name = "Contoh_2"
 Location = <sesuai instruksi Dosen / Asisten Dosen>
 Solution Name = "Modul V"
2. Ketikkan kode program berikut pada bagian teratas di area kode :

```
using System.Collections;
```
3. Ketikkan kode program berikut ke dalam blok **Main** dari **Program.cs** pada *Project "Contoh_2"* :

```

List<int> nilai = new List<int>();
ConsoleKeyInfo lagi;
int input;
do
{
    Console.Write("Input Sembarang Angka = ");
    input = Convert.ToInt32(Console.ReadLine());
    nilai.Add(input);
    Console.Write("Mau Input Data Lagi? (Y/T) = ");
    lagi = Console.ReadKey();
    Console.WriteLine("\n");
} while (lagi.KeyChar == 'y' || lagi.KeyChar == 'Y');
Console.WriteLine("\nAnda telah Menginput Sejumlah Angka Yakni :");
foreach (int i in nilai)
    Console.Write("{0} ", i);
Console.ReadKey();

```

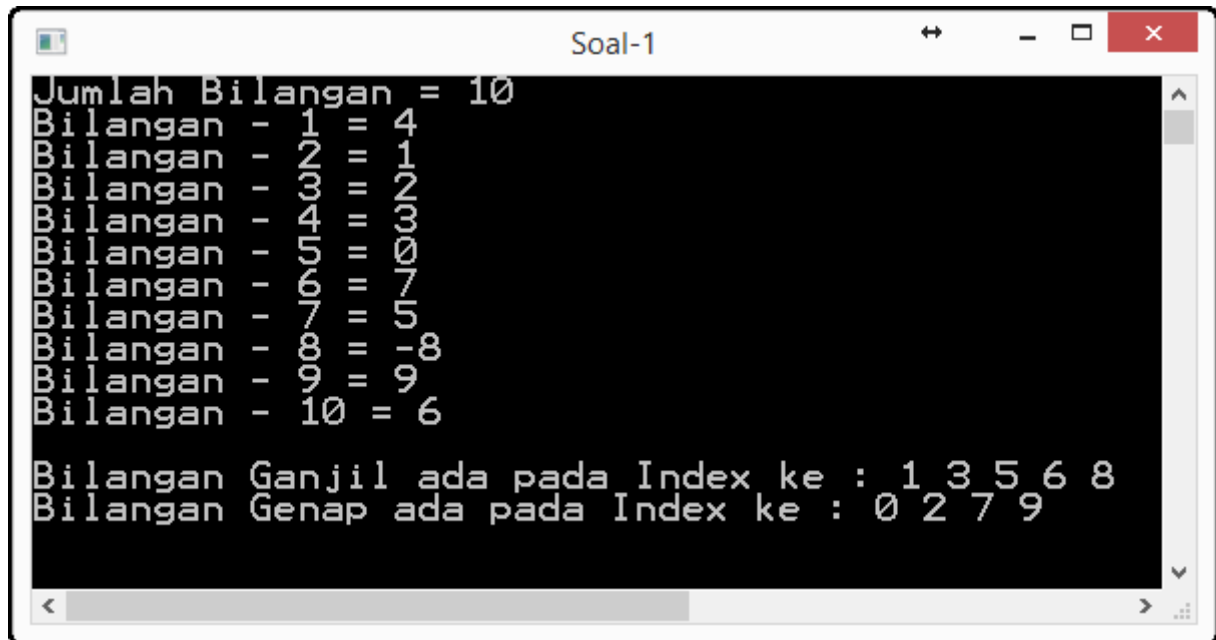
Catatan :

Semua fungsi dari kode program di atas dapat dilihat pada slide modul perkuliahan "Sesi 6 – Array dan Koleksi.pptx".

4. Untuk melihat hasil eksekusi dari program ini, dapat melalui menu **Debug / Start Debugging**, atau dengan shortcut **F5**

Soal Latihan

1. Buat program untuk menentukan jenis bilangan dengan tampilan sebagai berikut :



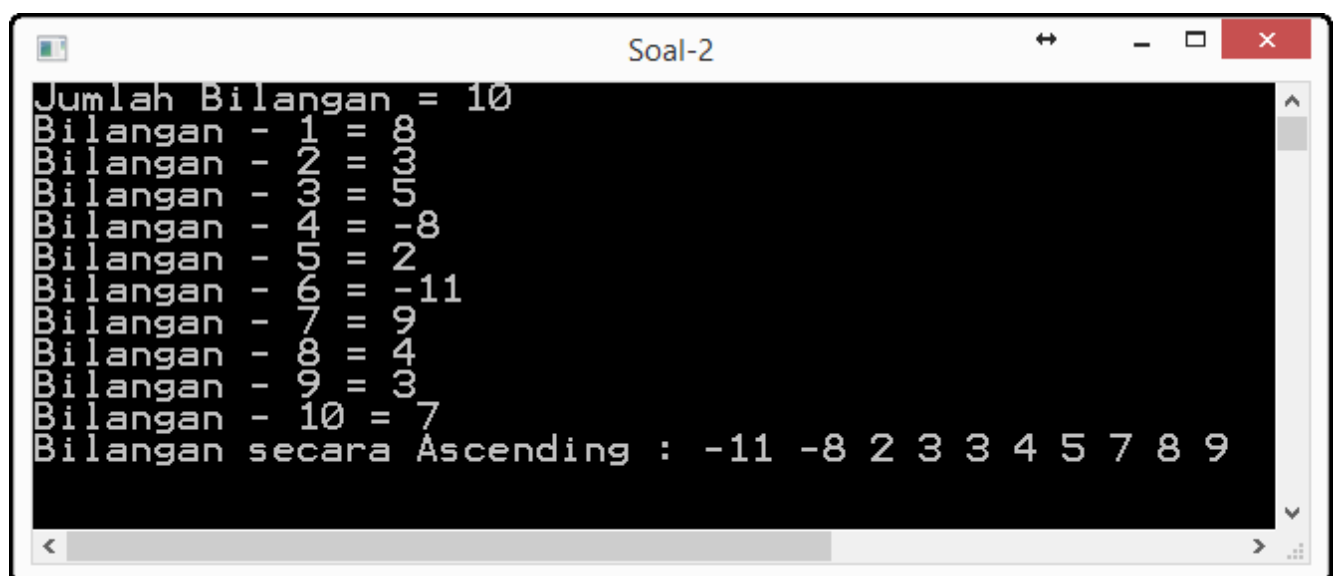
```
Jumlah Bilangan = 10
Bilangan - 1 = 4
Bilangan - 2 = 1
Bilangan - 3 = 2
Bilangan - 4 = 3
Bilangan - 5 = 0
Bilangan - 6 = 7
Bilangan - 7 = 5
Bilangan - 8 = -8
Bilangan - 9 = 9
Bilangan - 10 = 6

Bilangan Ganjil ada pada Index ke : 1 3 5 6 8
Bilangan Genap ada pada Index ke : 0 2 7 9
```

Dengan keterangan program :

- Input berupa : **Jumlah Bilangan** dan **masing-masing bilangan** sebanyak Jumlah Bilangan.
- Simpan nilai bilangan ke dalam Array.

2. Buat program untuk mengurutkan bilangan secara Bubble Sort dengan tampilan sebagai berikut :

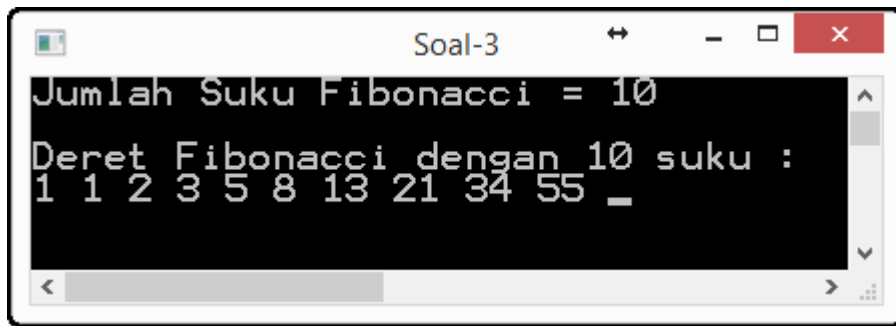


```
Jumlah Bilangan = 10
Bilangan - 1 = 8
Bilangan - 2 = 3
Bilangan - 3 = 5
Bilangan - 4 = -8
Bilangan - 5 = 2
Bilangan - 6 = -11
Bilangan - 7 = 9
Bilangan - 8 = 4
Bilangan - 9 = 3
Bilangan - 10 = 7
Bilangan secara Ascending : -11 -8 2 3 3 4 5 7 8 9
```

Dengan keterangan program :

- Input berupa : **Jumlah Bilangan** dan **masing-masing bilangan** sebanyak Jumlah Bilangan.
- Simpan nilai bilangan ke dalam Array.

3. Buat program untuk menentukan barisan Fibonacci dengan tampilan sebagai berikut :

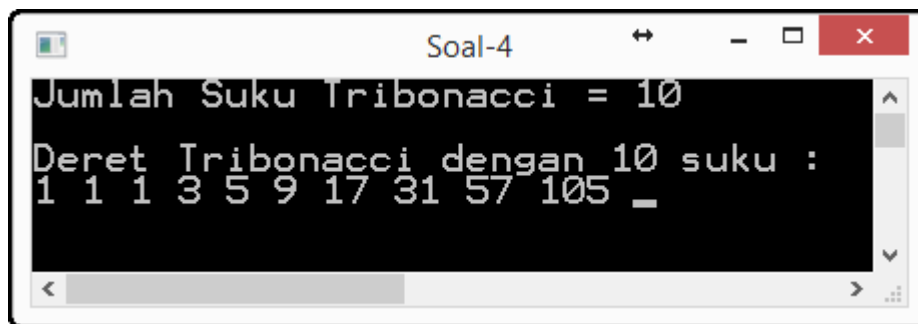


```
Soal-3
Jumlah Suku Fibonacci = 10
Deret Fibonacci dengan 10 suku :
1 1 2 3 5 8 13 21 34 55 _
```

Dengan keterangan program :

- Input berupa : **Jumlah Suku**.
- Simpan nilai setiap suku ke dalam Array.
- Rumus Fibonacci yang digunakan :
 $\text{Suku-1} = \text{suku-2} = 1$
 $\text{Fibo}(N) = \text{Fibo}(N - 2) + \text{Fibo}(N - 1)$

4. Buat program untuk menentukan barisan Tribonacci dengan tampilan sebagai berikut :

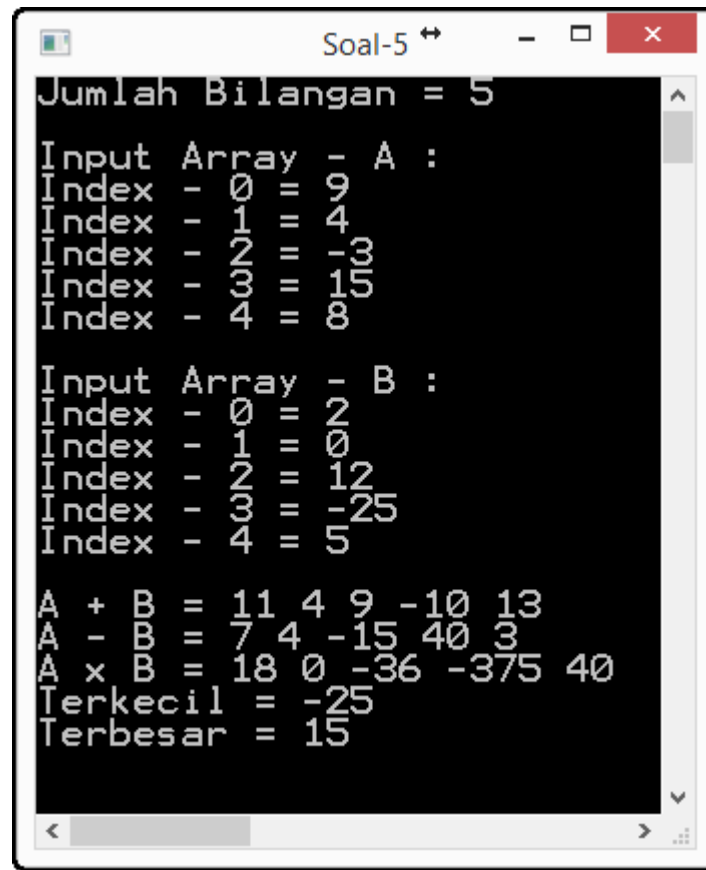


```
Soal-4
Jumlah Suku Tribonacci = 10
Deret Tribonacci dengan 10 suku :
1 1 1 3 5 9 17 31 57 105 _
```

Dengan keterangan program :

- Input berupa : **Jumlah Suku**.
- Setiap nilai suku akan disimpan ke dalam Array.
- Rumus Tribonacci yang digunakan :
 $\text{Suku-1} = \text{suku-2} = \text{suku-3} = 1$
 $\text{Tribo}(N) = \text{Tribo}(N - 3) + \text{Tribo}(N - 2) + \text{Tribo}(N - 1)$

5. Buat program untuk menghitung 2 buah Array dengan tampilan sebagai berikut :



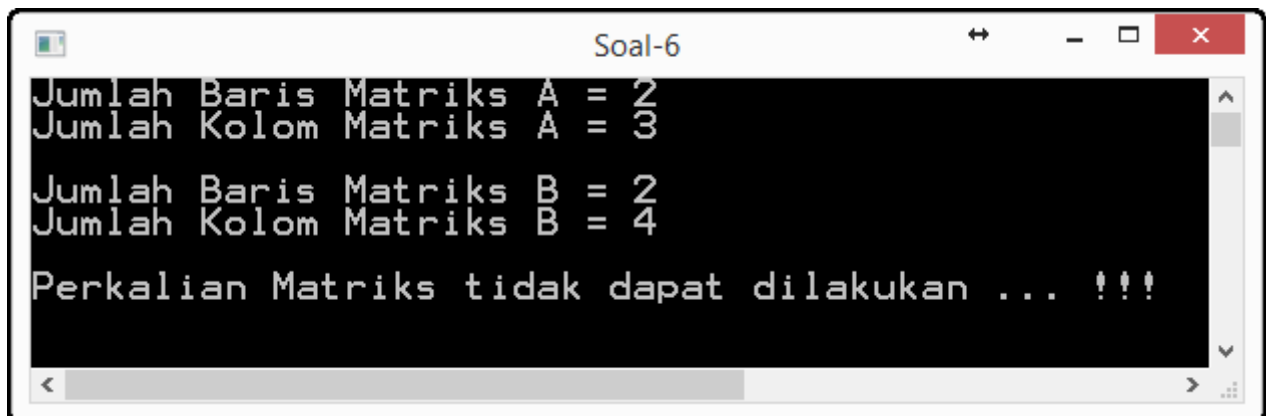
```
Jumlah Bilangan = 5
Input Array - A :
Index - 0 = 9
Index - 1 = 4
Index - 2 = -3
Index - 3 = 15
Index - 4 = 8

Input Array - B :
Index - 0 = 2
Index - 1 = 0
Index - 2 = 12
Index - 3 = -25
Index - 4 = 5

A + B = 11 4 9 -10 13
A - B = 7 4 -15 40 3
A x B = 18 0 -36 -375 40
Terkecil = -25
Terbesar = 15
```

Dengan input berupa : 2 buah Array dengan jumlah index yang sama.

6. Update program Contoh-1 di atas menjadi tampilan sebagai berikut :



```
Jumlah Baris Matriks A = 2
Jumlah Kolom Matriks A = 3

Jumlah Baris Matriks B = 2
Jumlah Kolom Matriks B = 4

Perkalian Matriks tidak dapat dilakukan ... !!!
```

```

Soal-6
Jumlah Baris Matriks A = 2
Jumlah Kolom Matriks A = 3

Jumlah Baris Matriks B = 3
Jumlah Kolom Matriks B = 4

Input Matriks A :
Nilai A [0, 0] = 2
Nilai A [0, 1] = 4
Nilai A [0, 2] = 6
Nilai A [1, 0] = 8
Nilai A [1, 1] = 10
Nilai A [1, 2] = 12

Input Matriks B :
Nilai B [0, 0] = 23
Nilai B [0, 1] = 21
Nilai B [0, 2] = 19
Nilai B [0, 3] = 17
Nilai B [1, 0] = 15
Nilai B [1, 1] = 13
Nilai B [1, 2] = 11
Nilai B [1, 3] = 9
Nilai B [2, 0] = 7
Nilai B [2, 1] = 5
Nilai B [2, 2] = 3
Nilai B [2, 3] = 1

```

```

Soal-6
Hasil Matriks C = A x B :
      148  124  100  76
      418  358  298  238

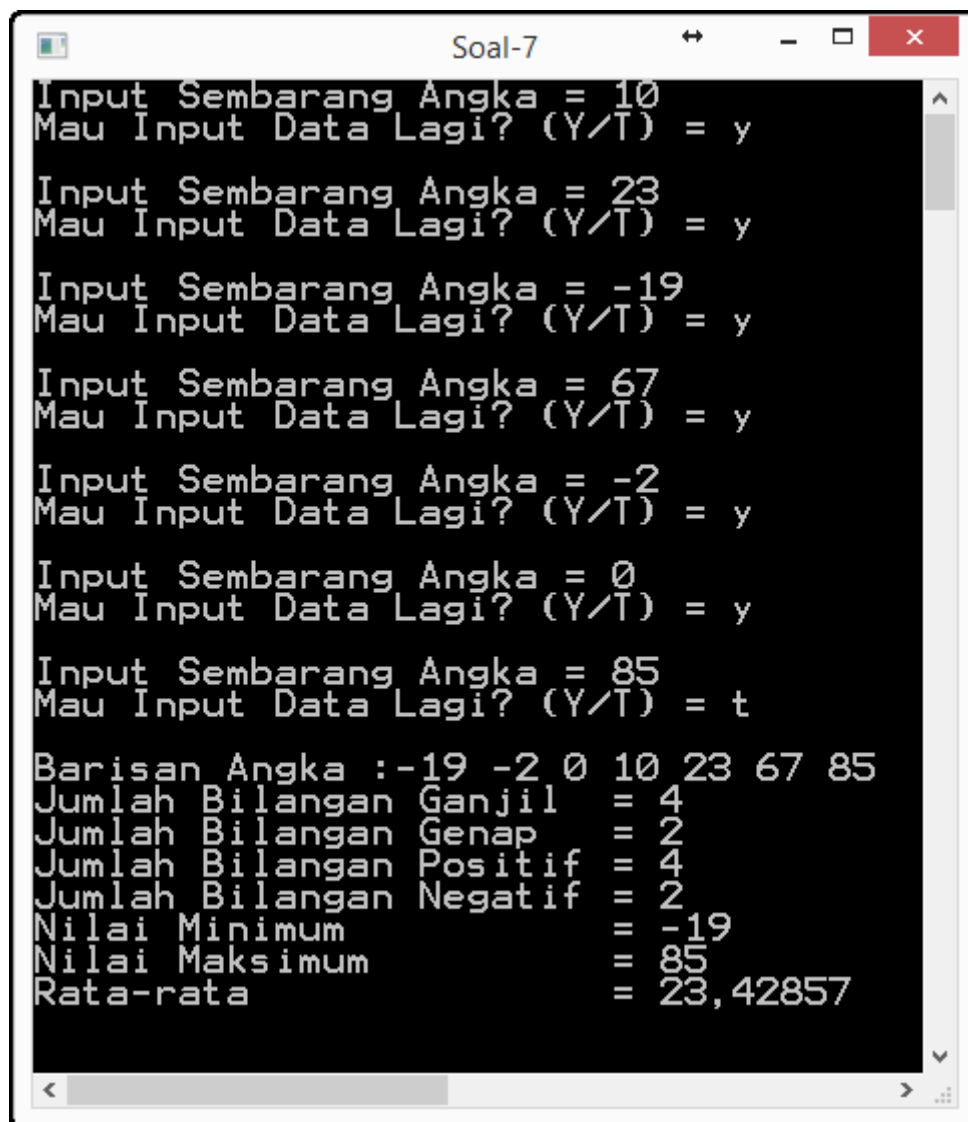
Matriks Transpose dari C :
      148  418
      124  358
      100  298
      76   238

```

Dengan keterangan program :

- Input berupa : 2 buah Matriks dengan ordo sesuai dengan keinginan user.
- Perkalian Matriks hanya dapat dilakukan apabila jumlah Kolom Matriks A = jumlah Baris Matriks B.

7. Buat program untuk menghitung statistik angka yang diinput dengan tampilan sebagai berikut :



```
Soal-7
Input Sembarang Angka = 10
Mau Input Data Lagi? (Y/T) = y
Input Sembarang Angka = 23
Mau Input Data Lagi? (Y/T) = y
Input Sembarang Angka = -19
Mau Input Data Lagi? (Y/T) = y
Input Sembarang Angka = 67
Mau Input Data Lagi? (Y/T) = y
Input Sembarang Angka = -2
Mau Input Data Lagi? (Y/T) = y
Input Sembarang Angka = 0
Mau Input Data Lagi? (Y/T) = y
Input Sembarang Angka = 85
Mau Input Data Lagi? (Y/T) = t
Barisan Angka : -19 -2 0 10 23 67 85
Jumlah Bilangan Ganjil = 4
Jumlah Bilangan Genap = 2
Jumlah Bilangan Positif = 4
Jumlah Bilangan Negatif = 2
Nilai Minimum = -19
Nilai Maksimum = 85
Rata-rata = 23,42857
```

Dengan keterangan program :

- Input berupa : **Angka**.
- Jumlah angka yang diinput tidak dibatasi, tergantung keinginan user.
- Setiap nilai yang diinput akan disimpan ke dalam Array atau Koleksi.

8. Buat program untuk menghitung IPK dengan tampilan sebagai berikut :

```

Kode Mata Kuliah      = IF0012
Nama Mata Kuliah      = PENGANTAR C#
Besar SKS             = 2
Nilai Huruf (A, A-, B+, B-, C+, C, D, E) = A-
Mau Input Data Lagi? (Y/T) = Y

Kode Mata Kuliah      = IF0023
Nama Mata Kuliah      = BASIS DATA
Besar SKS             = 4
Nilai Huruf (A, A-, B+, B-, C+, C, D, E) = B+
Mau Input Data Lagi? (Y/T) = Y

Kode Mata Kuliah      = UM0031
Nama Mata Kuliah      = KOMUNIKASI BISNIS
Besar SKS             = 2
Nilai Huruf (A, A-, B+, B-, C+, C, D, E) = C+
Mau Input Data Lagi? (Y/T) = _
  
```

```

Kode MK   Nama MK                               SKS   NH   Mutu
=====
IF0012    PENGANTAR C#                          2    A-   7,5
IF0023    BASIS DATA                          4    B+   14
UM0031    KOMUNIKASI BISNIS                         2    C+   5
=====
IPK = 3,3125
  
```

Dengan keterangan program :

- Input berupa : **Kode Mata Kuliah**, **Nama Mata Kuliah**, **Besar SKS**, dan **Nilai Huruf**.
- Jumlah mata kuliah yang diinput tidak dibatasi, tergantung keinginan user.
- Gunakan **Struct** berikut :


```

struct Nilai {
    public string kode, nama, NH;
    public byte SKS;
    public float mutu;
}
      
```
- Gunakan Array/Koleksi untuk menyimpan mata kuliah.

9. Buat program untuk menghitung statistik angka yang diinput dengan tampilan sebagai berikut :

```

Soal-9
Jumlah Bilangan Himpunan A = 4
Jumlah Bilangan Himpunan B = 5

Input Himpunan A :
    Nilai A-1 = 2
    Nilai A-2 = 9
    Nilai A-3 = 1
    Nilai A-4 = 0

Input Himpunan B :
    Nilai B-1 = 5
    Nilai B-2 = -8
    Nilai B-3 = 7
    Nilai B-4 = -2
    Nilai B-5 = 4

Himpunan A = { 2 9 1 0 }
Himpunan B = { 5 -8 7 -2 4 }
Himpunan A Gabung B = { -8 -2 0 1 2 4 5 7 9 }
  
```

Dengan keterangan program :

- Input berupa : **Jumlah Bilangan Himpunan A dan B**, serta **Nilai dari setiap Himpunan**.
- Setiap nilai dari himpunan yang diinput dan dihasilkan akan disimpan ke dalam Array.

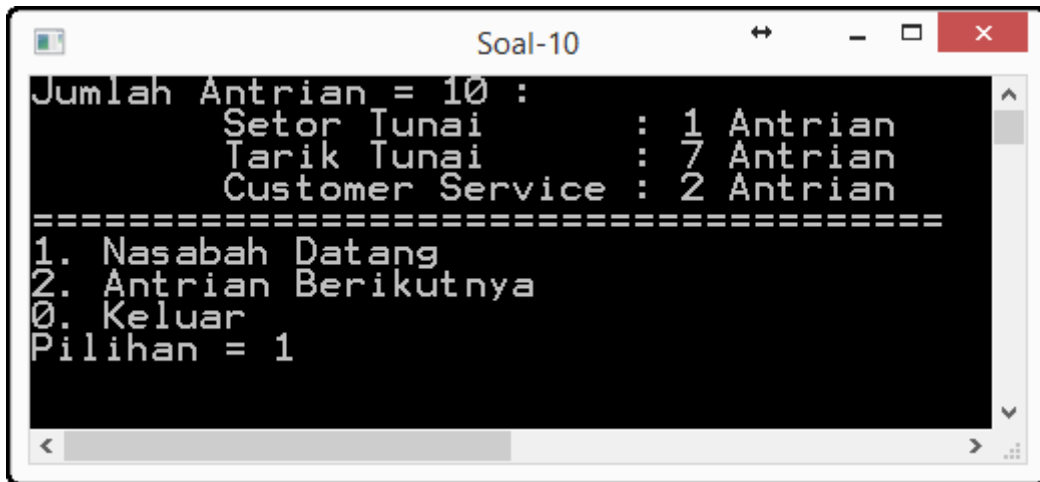
10. Buat program untuk mensimulasikan kedatangan dan pelayanan nasabah pada suatu bank dengan tampilan sebagai berikut :

```

Soal-10
Jumlah Antrian = 0 :
    Setor Tunai      : 0 Antrian
    Tarik Tunai      : 0 Antrian
    Customer Service : 0 Antrian
=====
1. Nasabah Datang
2. Antrian Berikutnya
0. Keluar
Pilihan = 2

Tidak ada Antrian ... !!!
  
```

*** Tampilan jika belum ada Antrian ***

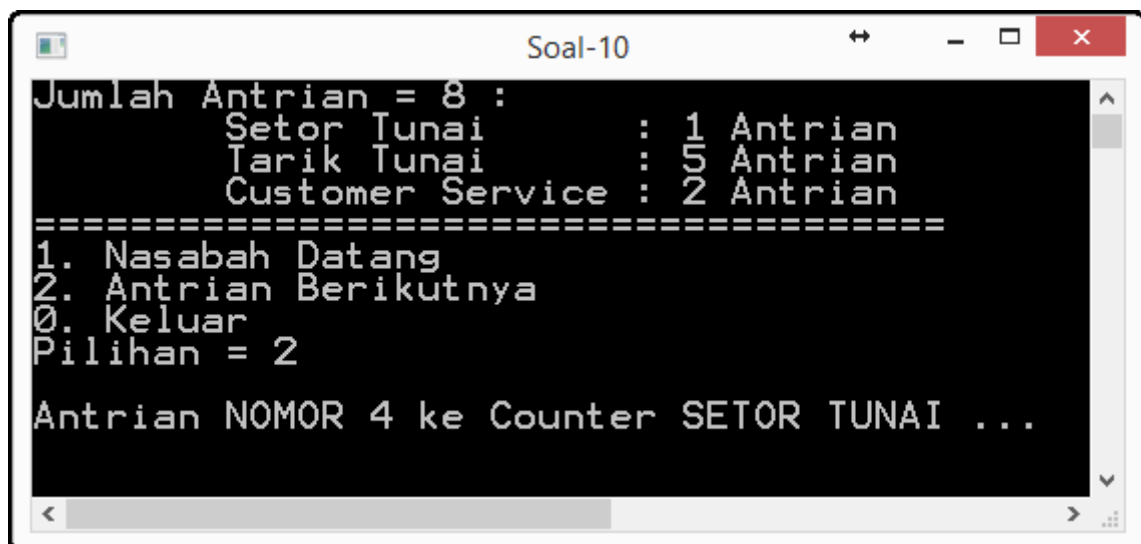


```
Soal-10
Jumlah Antrian = 10 :
    Setor Tunai      : 1 Antrian
    Tarik Tunai      : 7 Antrian
    Customer Service : 2 Antrian
=====
1. Nasabah Datang
2. Antrian Berikutnya
0. Keluar
Pilihan = 1
```

*** Tampilan jika nasabah datang ***



```
Soal-10
Jumlah Antrian = 11 :
    Setor Tunai      : 1 Antrian
    Tarik Tunai      : 7 Antrian
    Customer Service : 3 Antrian
=====
1. Nasabah Datang
2. Antrian Berikutnya
0. Keluar
Pilihan = 2
Antrian NOMOR 1 ke Counter TARIK TUNAI ...
_
```



```
Soal-10
Jumlah Antrian = 8 :
    Setor Tunai      : 1 Antrian
    Tarik Tunai      : 5 Antrian
    Customer Service : 2 Antrian
=====
1. Nasabah Datang
2. Antrian Berikutnya
0. Keluar
Pilihan = 2
Antrian NOMOR 4 ke Counter SETOR TUNAI ...
```

```

Jumlah Antrian = 6 :
    Setor Tunai      : 0 Antrian
    Tarik Tunai      : 4 Antrian
    Customer Service : 2 Antrian
=====
1. Nasabah Datang
2. Antrian Berikutnya
0. Keluar
Pilihan = 2
Antrian NOMOR 6 ke Counter CUSTOMER SERVICE ...

```

Dengan keterangan program :

- Nasabah yang datang terdiri dari keperluan layanan :

- Setor Tunai
- Tarik Tunai
- Customer Service

- Pelayanan nasabah secara FIFO.

- Gunakan **Enum** dan **Struct** berikut :

```

enum Layanan {
    Setor, Tarik, CS
}
struct Customer {
    public int no;
    public Layanan jenis;
}

```

- Gunakan Koleksi untuk menyimpan data.

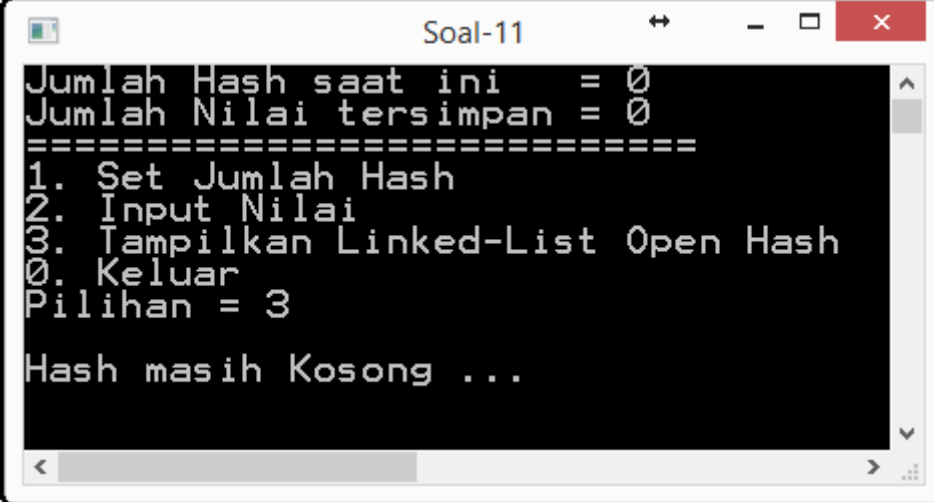
11. Buat program untuk menyimpan nilai ke dalam suatu Hash secara Linked-List Open Hash dengan tampilan sebagai berikut :

```

Jumlah Hash saat ini = 0
Jumlah Nilai tersimpan = 0
=====
1. Set Jumlah Hash
2. Input Nilai
3. Tampilkan Linked-List Open Hash
0. Keluar
Pilihan = 2
Harap set Jumlah Hash terlebih dahulu ...

```

*** Tampilan jika jumlah Hash belum di-set ***



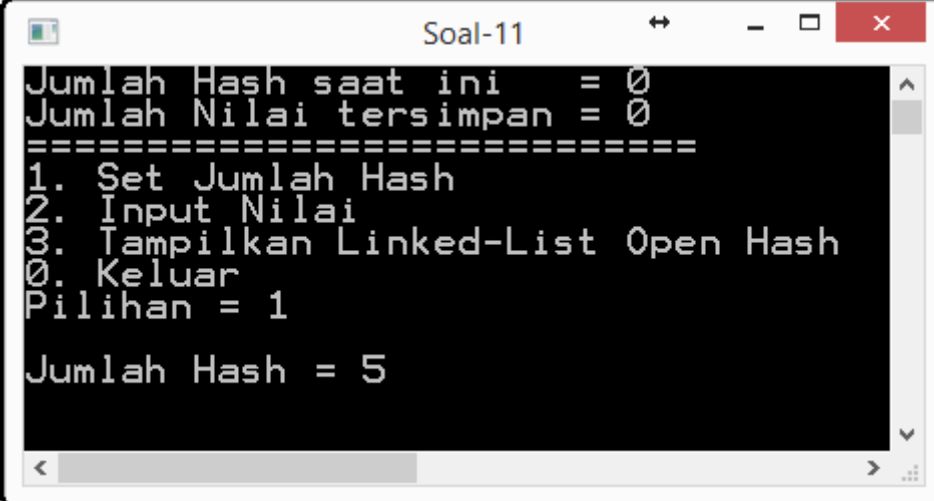
```

Jumlah Hash saat ini = 0
Jumlah Nilai tersimpan = 0
=====
1. Set Jumlah Hash
2. Input Nilai
3. Tampilkan Linked-List Open Hash
0. Keluar
Pilihan = 3

Hash masih Kosong ...

```

*** Tampilan jika menampilkan isi Hash yang belum di-set ***



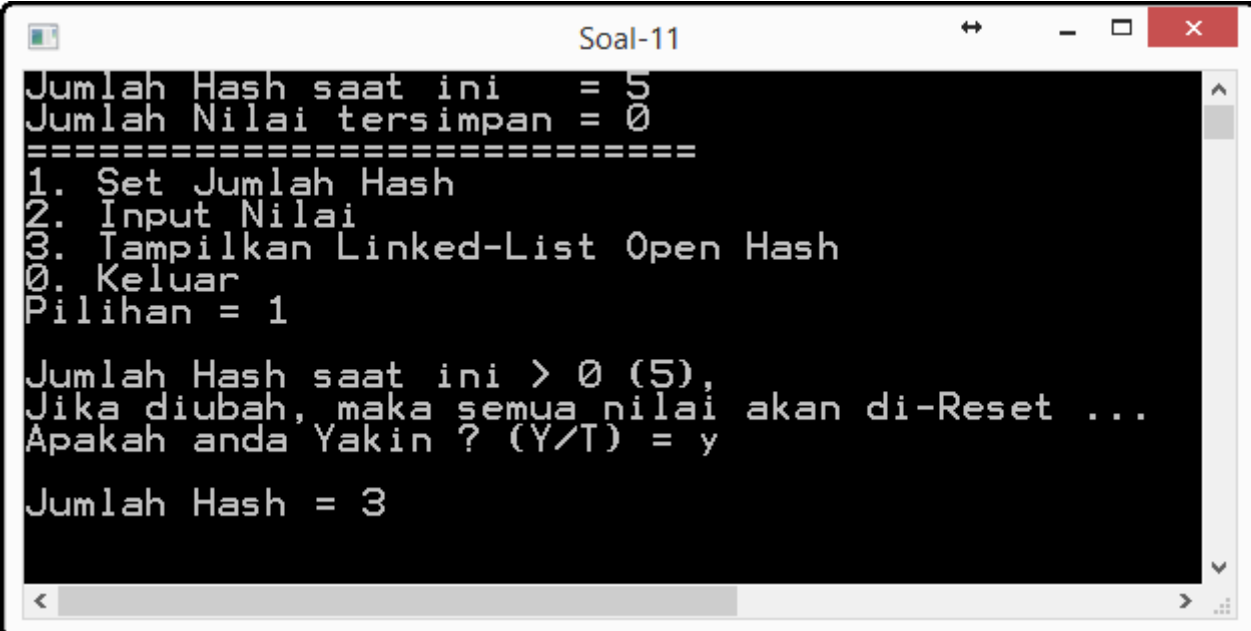
```

Jumlah Hash saat ini = 0
Jumlah Nilai tersimpan = 0
=====
1. Set Jumlah Hash
2. Input Nilai
3. Tampilkan Linked-List Open Hash
0. Keluar
Pilihan = 1

Jumlah Hash = 5

```

*** Tampilan saat set Jumlah Hash jika Hash sebelumnya = 0 ***



```

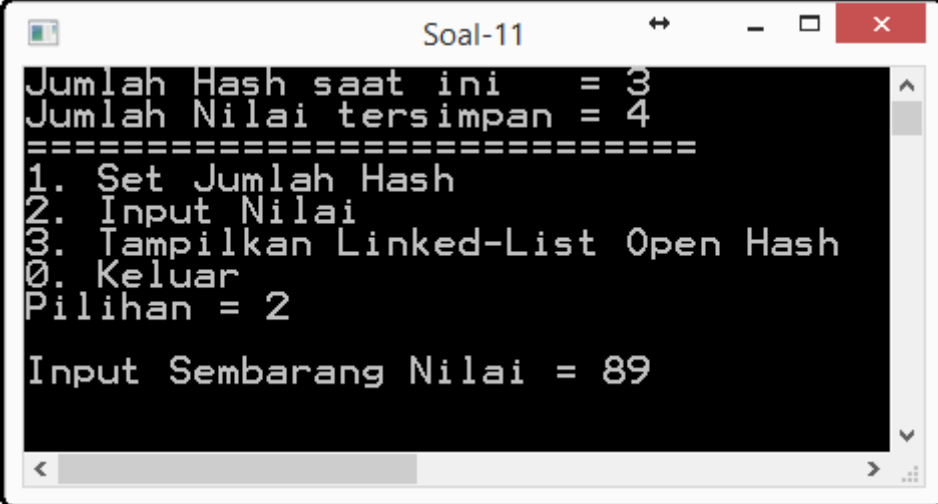
Jumlah Hash saat ini = 5
Jumlah Nilai tersimpan = 0
=====
1. Set Jumlah Hash
2. Input Nilai
3. Tampilkan Linked-List Open Hash
0. Keluar
Pilihan = 1

Jumlah Hash saat ini > 0 (5),
Jika diubah, maka semua nilai akan di-Reset ...
Apakah anda Yakin ? (Y/T) = y

Jumlah Hash = 3

```

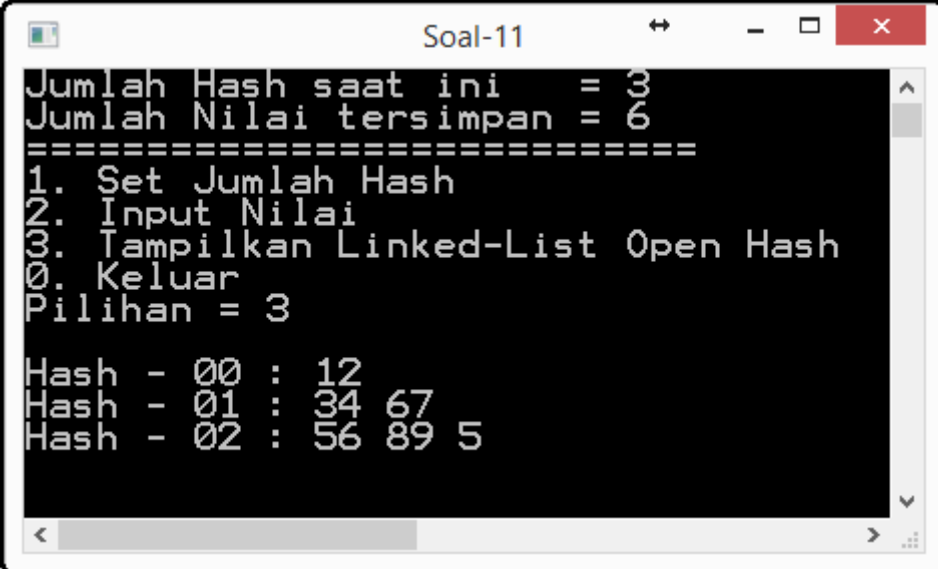
*** Tampilan saat set Jumlah Hash jika Hash sebelumnya > 0 ***



```
Soal-11
Jumlah Hash saat ini    = 3
Jumlah Nilai tersimpan = 4
=====
1. Set Jumlah Hash
2. Input Nilai
3. Tampilkan Linked-List Open Hash
0. Keluar
Pilihan = 2

Input Sembarang Nilai = 89
```

*** Tampilan saat input nilai ke dalam Hash ***



```
Soal-11
Jumlah Hash saat ini    = 3
Jumlah Nilai tersimpan = 6
=====
1. Set Jumlah Hash
2. Input Nilai
3. Tampilkan Linked-List Open Hash
0. Keluar
Pilihan = 3

Hash - 00 : 12
Hash - 01 : 34 67
Hash - 02 : 56 89 5
```

Dengan keterangan program : Penyisipan nilai ke dalam Linked-List adalah secara sisip belakang.