

Tugas Besar Laporan Prak.Algoritma dan Pemograman



Nama : Agil Deriansyah Hasan
Nim : 4522210125

Dosen:

Dra.SRI REZEKI CANDRA NURSARI,M.Kom
Prak. Algoritma dan Pemrograman - B

**S1-Teknik Informatika
Fakultas Teknik
Universitas Pancasila 2023/2024**

Kata Pengantar

Puji syukur kita panjatkan kehadiran Tuhan Yang Maha Esa, yang telah melimpahkan rahmat, hidayah, dan karunia-Nya sehingga penulis dapat menyelesaikan tugas besar ini. Shalawat serta salam senantiasa tercurah kepada junjungan kita, Nabi Muhammad SAW, yang menjadi tauladan dalam segala aspek kehidupan.

Laporan tugas besar ini disusun sebagai hasil dari dedikasi, kerja keras, dan ketekunan penulis dalam mengeksplorasi suatu topik yang memiliki dampak dan relevansi yang signifikan. Tugas besar ini tidak hanya merupakan bagian integral dari penyelesaian tugas akademis, tetapi juga menjadi wadah untuk mengasah kemampuan analisis, penelitian, dan sintesis penulis dalam memahami suatu masalah secara lebih mendalam.

Wassalamu'alaikum Wr. Wb.

Daftar Isi

Kata Pengantar.....	1
Daftar Isi.....	2
BAB 1.....	1
Pendahuluan.....	1
1.1 Latar Belakang.....	1
1.2 Tujuan.....	1
1.3 Pembahasan Teori.....	2
1.3.1 Sequence.....	2
1.3.2 Selection/Seleksi : IF, IF...Else.....	6
1.3.3 Selection/Seleksi : Nested IF,Switch.....	8
1.3.4 Repetition/Pengulangan : While, Do...While.....	10
1.3.5 Repetition/Pengulangan : For, Nested For.....	12
1.3.6 Rekrusif, Fibonacci.....	14
1.3.7 Array 1 Dimensi.....	15
1.3.8 Array 2 Dimensi.....	16
1.3.9 Function/Fungsi.....	21
1.3.10 String.....	26
1.3.11 Sort : Insertion, Selection.....	28
1.3.12 Pointer.....	30
1.3.13 File.....	32
1.4 Tugas Praktikum.....	36
BAB II.....	62
Tugas Besar.....	62
2.1 Pembahasan Tugas Besar.....	62
2.1.1 Judul Dan Materi.....	62
2.1.2 Source Code Program.....	62
2.1.3 Capture Program.....	68
2.1.4 Pseudocode.....	70
2.1.5 Algoritma.....	73
BAB III.....	76
Penutup.....	76
3.1 Kesimpulan.....	76
3.2 Kritik dan Saran.....	76
Daftar Pustaka.....	77
Video Presentasi.....	77

BAB 1

Pendahuluan

1.1 Latar Belakang

Dalam era di mana teknologi semakin berkembang pesat, pemahaman dan penguasaan terhadap algoritma dan pemrograman menjadi sangat penting. Algoritma merupakan langkah-langkah sistematis untuk menyelesaikan masalah atau mencapai tujuan tertentu. Pemrograman, di sisi lain, melibatkan implementasi algoritma ke dalam bahasa pemrograman tertentu untuk menciptakan program komputer.

Oleh karena itu, memahami dan menguasai algoritma dan pemrograman bukan hanya menjadi kebutuhan bagi mereka yang berkecimpung dalam dunia teknologi, tetapi juga menjadi keterampilan yang sangat berharga dalam menghadapi tantangan di masa depan yang semakin didominasi oleh perkembangan teknologi informasi. Dengan latar belakang tersebut, pemahaman mendalam terhadap materi algoritma dan pemrograman menjadi suatu hal yang esensial bagi setiap individu yang ingin bersaing dan berkontribusi dalam era digital ini.

1.2 Tujuan

Tujuan dari dibuat atau disusunnya laporan ini adalah :

1. Untuk memenuhi tugas besar mata kuliah Algoritma dan Pemrograman.
2. Memberikan pemahaman terhadap materi-materi yang sudah dipelajari dengan cara mengaplikasikannya menjadi sebuah program yang isinya terdapat materi tersebut.
3. Program bertujuan untuk mempermudah proses penginputan penilaian nilai mahasiswa bagi pengajar

1.3 Pembahasan Teori

1.3.1 Sequence

Sequence dalam bahasa pemrograman C++ merujuk pada urutan pengerjaan dari perintah atau statement pertama hingga terakhir. Hal ini merupakan struktur dasar algoritma yang berisi rangkaian instruksi yang diproses secara satu per satu, mulai dari instruksi pertama hingga terakhir sesuai dengan urutan penulisannya. Sequence sangat penting dalam algoritma karena memungkinkan untuk menjalankan suatu algoritma dengan efektif dan teratur. Dalam C++, escape sequences juga digunakan untuk memasukkan karakter khusus dalam string, seperti ganti baris, tab, dan tanda kutip.

Contoh :

Sequence1



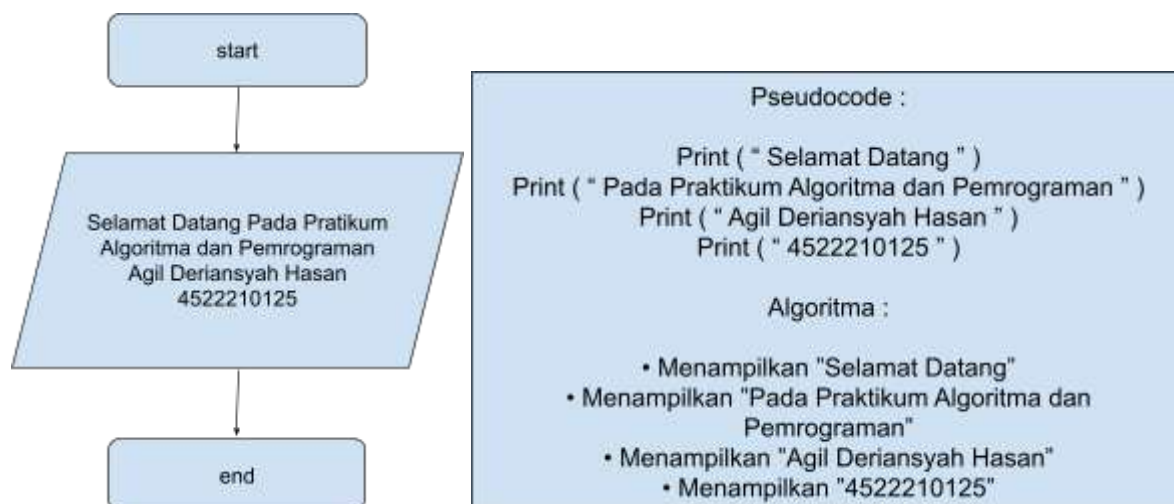
The image shows a screenshot of a C++ program in a code editor and its execution output. The code in the editor is as follows:

```
1 // Selamat Datang
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     cout << "Selamat Datang\n";
8     cout << "Pada Praktikum Algoritma dan Pemrograman\n";
9     cout << "Agil Deriansyah Hasan\n";
10    cout << "4522210125\n";
11    return 0;
12 }
```

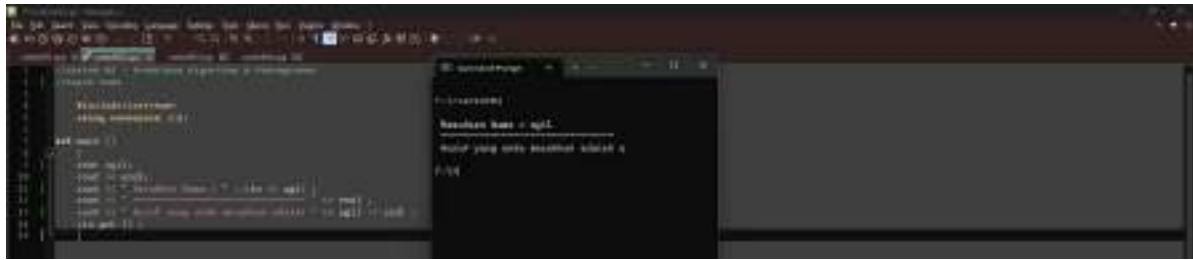
The output window shows the following text:

```
P: g++ -std=c++11 -c 01_01_hello.cpp
P: ./01_01_hello.exe
Selamat Datang
Pada Praktikum Algoritma dan Pemrograman
Agil Deriansyah Hasan
4522210125
```

Flowchart :

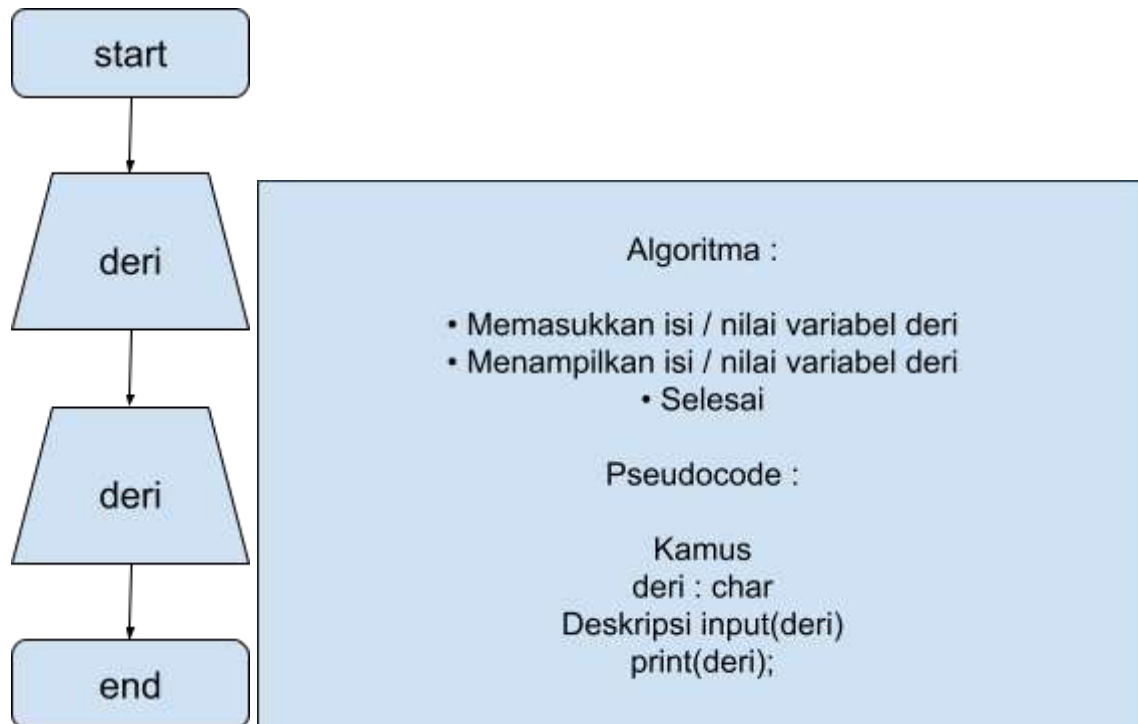


Sequence2



```
1 // Sequence2.cpp : This file contains the main function of the application.
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     char deri;
8     cout << "Masukkan isi / nilai variabel deri : ";
9     cin >> deri;
10    cout << "Menampilkan isi / nilai variabel deri : " << deri << endl;
11    return 0;
12 }
```

Flowchart :



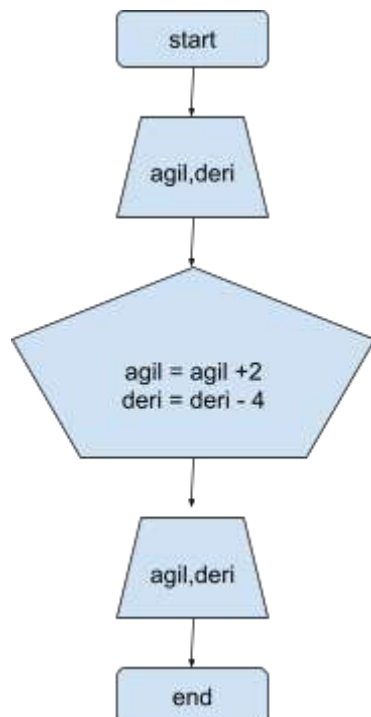
Sequence3

```

1 // Sequence3.cpp
2 //
3 // Created by: [Name]
4 // Date: [Date]
5
6 #include <iostream>
7 using namespace std;
8
9 int main()
10 {
11     int agil, deri;
12     cout << "Masukkan agil dan deri : ";
13     cin >> agil;
14     cin >> deri;
15
16     agil = agil + 2;
17     deri = deri - 4;
18
19     cout << "Nilai agil setelah : ";
20     cout << agil << endl;
21     cout << "Nilai deri setelah : ";
22     cout << deri << endl;
23
24     return 0;
25 }

```

Flowchart :



Algoritma :

- Memasukkan isi/nilai variabel agil
- Memasukkan isi/nilai variabel deri
 - $agil = agil + 2$
 - $deri = deri - 4$
- Menampilkan isi/nilai variabel agil
- Menampilkan isi/nilai variabel deri
 - Selesai

Pseudocode :

Kamus

agil : int
deri : int

Deskripsi input(agil, deri)

$agil = agil + 2$
 $deri = deri - 4$
 print(agil , deri);

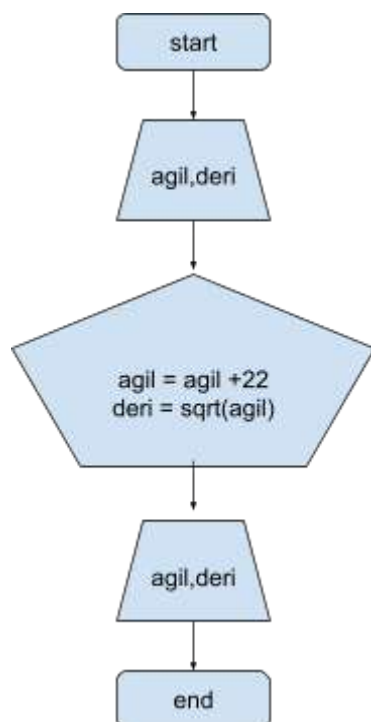
Sequence4

```

1 //Sequence4.cpp
2 #include <iostream>
3 #include <math>
4 using namespace std;
5
6 int main()
7 {
8     int agil;
9     float deri;
10
11     //Input
12     cout << "Masukkan isi/nilai variabel agil : ";
13     cin >> agil;
14
15     //Proses
16     agil = agil + 22;
17     deri = sqrt(agil);
18
19     //Output
20     cout << "Nilai variabel agil : " << agil << endl;
21     cout << "Nilai variabel deri : " << deri << endl;
22
23     return 0;
24 }

```

Flowchart :



Algoritma :

- Memasukkan isi/nilai variabel agil
 - $agil = agil + 22$
 - $deri = \sqrt{agil}$
- Menampilkan isi/nilai variabel agil
- Menampilkan isi/nilai variabel deri
- Selesai

Pseudocode :

Kamus

agil : int
deri : float

Deskripsi

Input (agil)
 $agil = agil + 22$
 $deri = \sqrt{agil}$
 print(agil, deri);

1.3.2 Selection/Seleksi : IF, IF...Else

Dalam bahasa pemrograman C++, seleksi if-else digunakan untuk melakukan percabangan kondisional. Pernyataan dalam blok if dijalankan hanya jika kondisi dievaluasi ke nilai bukan nol (atau true). Jika kondisi benar, pernyataan berikutnya dijalankan, dan pernyataan setelah else dilewati. Jika kondisi salah, pernyataan setelah else akan dijalankan.

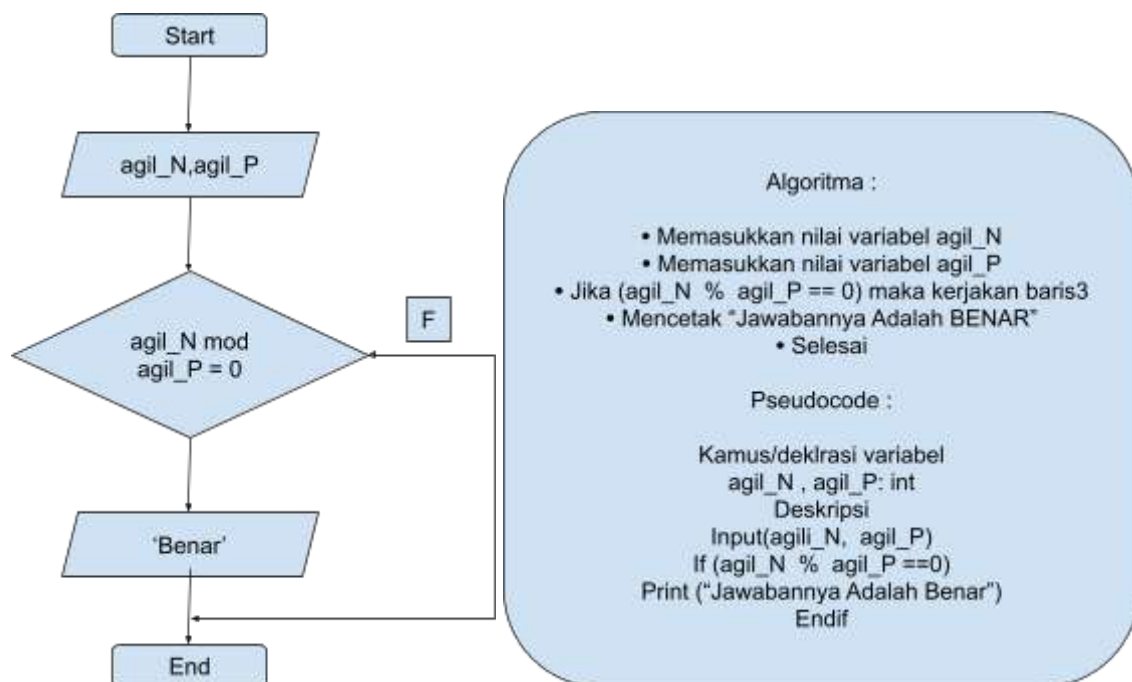
Contoh :

IF1



```
1 // Nama : Agil, NIM : 202101120
2 // No. : 202101120
3 //
4 //
5 //
6 //
7 //
8 //
9 //
10 //
11 //
12 //
13 //
14 //
15 //
16 //
17 //
18 //
19 //
20 //
21 //
22 //
23 //
24 //
25 //
26 //
27 //
28 //
29 //
30 //
31 //
32 //
33 //
34 //
35 //
36 //
37 //
38 //
39 //
40 //
41 //
42 //
43 //
44 //
45 //
46 //
47 //
48 //
49 //
50 //
51 //
52 //
53 //
54 //
55 //
56 //
57 //
58 //
59 //
60 //
61 //
62 //
63 //
64 //
65 //
66 //
67 //
68 //
69 //
70 //
71 //
72 //
73 //
74 //
75 //
76 //
77 //
78 //
79 //
80 //
81 //
82 //
83 //
84 //
85 //
86 //
87 //
88 //
89 //
90 //
91 //
92 //
93 //
94 //
95 //
96 //
97 //
98 //
99 //
100 //
```

Flowchart :



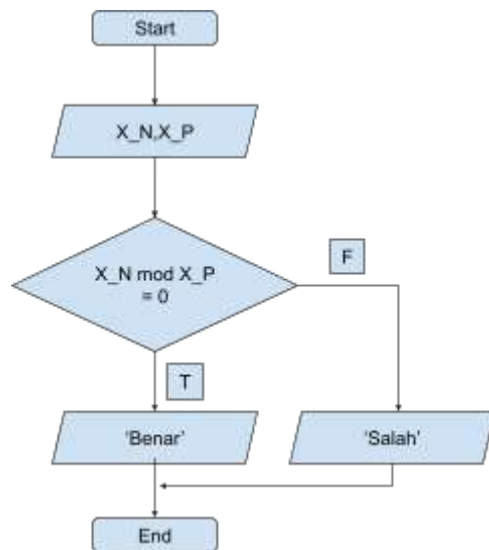
IF2

```

1 // Nama : Agil Santoso
2 // NPM : 2002201010
3
4 // Deskripsi :
5 // Memeriksa apakah satu bilangan habis dibagi oleh bilangan lainnya
6
7 #include <iostream>
8 using namespace std;
9
10 int main()
11 {
12     // Deklarasi variabel
13     int X, Y;
14     int X_P;
15
16     // Input
17     cout << "Masukkan bilangan pertama (X) : "; cin >> X;
18     cout << "Masukkan bilangan kedua (Y) : "; cin >> Y;
19     cout << endl;
20
21     // Proses
22     if (X % Y == 0)
23     {
24         cout << "Jawabannya adalah Benar" << endl;
25     }
26     else
27     {
28         cout << "Jawabannya adalah Salah" << endl;
29     }
30
31     return 0;
32 }

```

Flowchart :



Algoritma :

- Memasukkan nilai variabel X_N
- Memasukkan nilai variabel X_P
- Jika (X_N % X_P == 0) maka kerjakan baris4 kalau tidak 5
 - Mencetak "Jawabannya Adalah BENAR"
 - Mencetak "Jawabannya Adalah SALAH"
 - Selesai

Pseudocode :

Kamus/deklrasi variabel :

X_N, X_P; int

Deskripsi :

Input(X_N, X_P)

If (X_N % X_P == 0)

Print ("Jawabannya Adalah Benar")

Else

Print ("Jawabannya Adalah Salah")

Endif

1.3.3 Selection/Seleksi : Nested IF, Switch

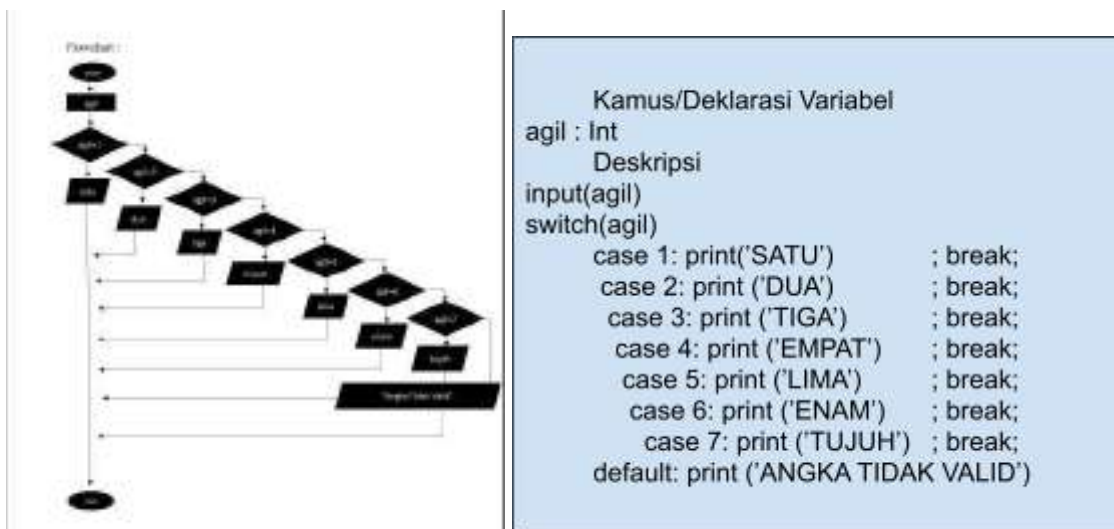
Selection nested if dan switch adalah bentuk seleksi kondisional yang lebih kompleks dibandingkan dengan seleksi if-else. Nested if adalah penggunaan if statement di dalam if statement lainnya. Sedangkan nested switch adalah penggunaan switch statement di dalam switch statement lainnya. Kedua bentuk seleksi ini digunakan ketika terdapat beberapa kondisi yang harus dipenuhi untuk menjalankan suatu perintah.

Contoh :

Nestedif2



```
1 // Nested if-else statement
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     int agil;
8     cout << "Masukkan angka 1-8 : ";
9     cin >> agil;
10
11     if (agil % 2 == 0)
12     {
13         cout << "Angka " << agil << " adalah genap." << endl;
14         if (agil % 2 == 0)
15         {
16             cout << "Angka " << agil << " adalah genap." << endl;
17             if (agil % 2 == 0)
18             {
19                 cout << "Angka " << agil << " adalah genap." << endl;
20                 if (agil % 2 == 0)
21                 {
22                     cout << "Angka " << agil << " adalah genap." << endl;
23                     if (agil % 2 == 0)
24                     {
25                         cout << "Angka " << agil << " adalah genap." << endl;
26                     }
27                 }
28             }
29         }
30     }
31     else
32     {
33         cout << "Angka " << agil << " adalah ganjil." << endl;
34     }
35 }
```



Nestedif1

```

1 // Program Memasukkan Angka dari 1 s.d. 7
2 #include <stdio.h>
3
4 int main()
5 {
6     int agil;
7     printf("Masukkan angka yang sesuai (1 s.d. 7) : ");
8     scanf("%d", &agil);
9
10    if (agil == 1)
11    {
12        printf("SATU\n");
13    }
14    else if (agil == 2)
15    {
16        printf("DUA\n");
17    }
18    else if (agil == 3)
19    {
20        printf("TIGA\n");
21    }
22    else if (agil == 4)
23    {
24        printf("EMPAT\n");
25    }
26    else if (agil == 5)
27    {
28        printf("LIMA\n");
29    }
30    else if (agil == 6)
31    {
32        printf("ENAM\n");
33    }
34    else if (agil == 7)
35    {
36        printf("TUJUH\n");
37    }
38    else
39    {
40        printf("ANGKA TIDAK VALID\n");
41    }
42
43    return 0;
44 }

```

```

C:\Users\user>gcc nestedif1.c -o nestedif1.exe
C:\Users\user>. nestedif1.exe
Masukkan angka yang sesuai : 1
SATU
Masukkan angka yang sesuai : 2
DUA
Masukkan angka yang sesuai : 3
TIGA
Masukkan angka yang sesuai : 4
EMPAT
Masukkan angka yang sesuai : 5
LIMA
Masukkan angka yang sesuai : 6
ENAM
Masukkan angka yang sesuai : 7
TUJUH
Masukkan angka yang sesuai : 8
ANGKA TIDAK VALID

```

Pseudocode

Program Memasukkan Angka dari 1 s.d. 7
Kamus

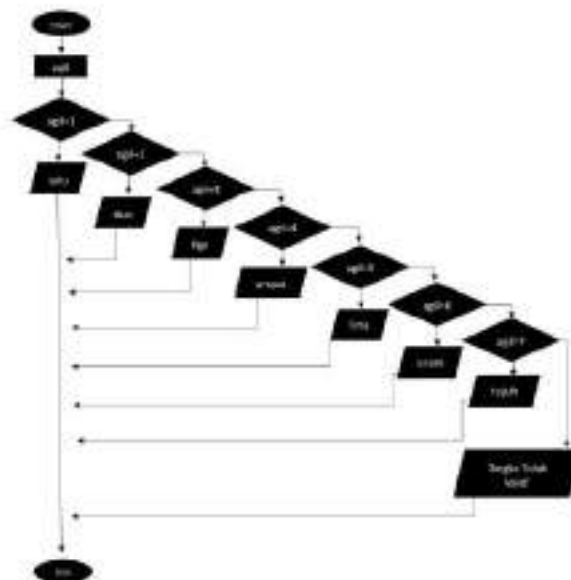
agil : Int

Deskripsi

```

input (agil)
    if (agil == 1) then
        print("SATU")
    else
        if (agil == 2) then
            print("DUA")
        else
            if (agil == 3) then
                print("TIGA")
            else
                if (agil == 4) then
                    print("EMPAT")
                else
                    if (agil == 5) then
                        print("LIMA")
                    else
                        if (agil == 6) then
                            print("ENAM")
                        else
                            if (agil == 7) then
                                print("TUJUH")
                            else
                                print("ANGKA TIDAK VALID")
                            endif
                        endif
                    endif
                endif
            endif
        endif
    endif
endif

```

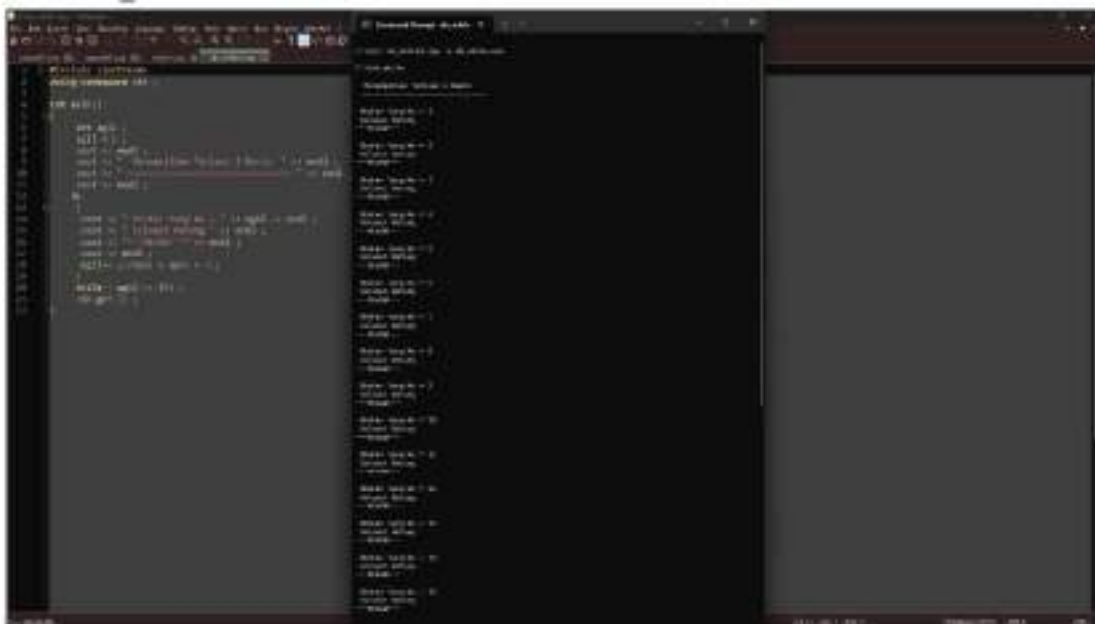


1.3.4 Repetition/Pengulangan : While, Do...While

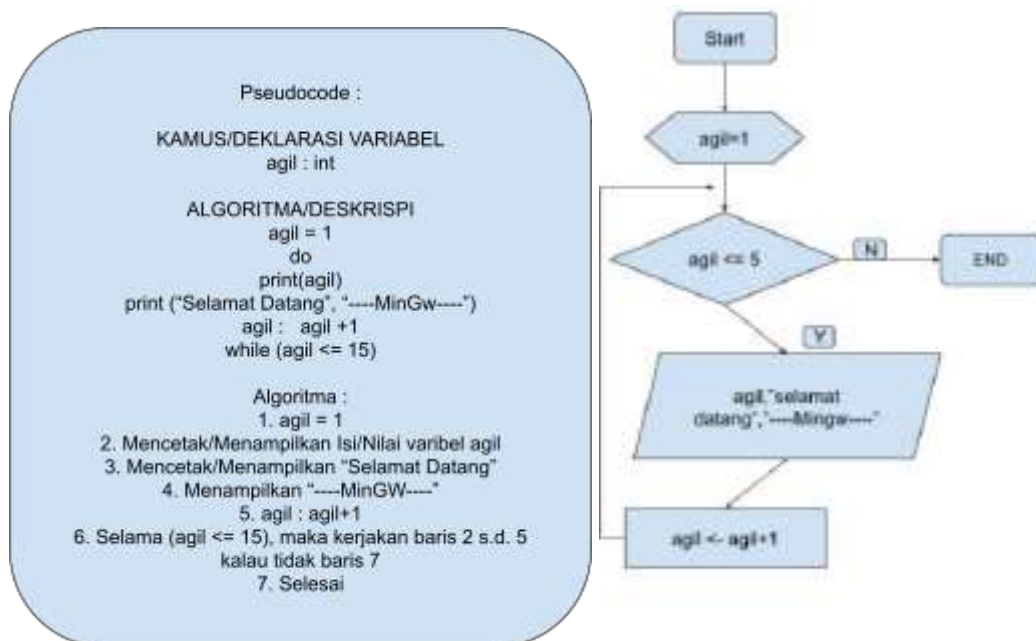
Nested repetition, atau pengulangan bersarang, adalah konsep pemrograman di mana sebuah struktur kontrol ditempatkan di dalam tubuh struktur kontrol lainnya. Dalam bahasa pemrograman C++, nested while loop digunakan untuk berbagai tujuan, seperti memproses array multidimensi atau membuat output berbasis pola. Loop while dalam nested repetition dijalankan untuk setiap iterasi dari loop while luar. Nested repetition adalah teknik yang kuat untuk menyelesaikan masalah yang memerlukan tindakan berulang dalam tindakan berulang.

Contoh

Do_While1



```
1 // Do_While1.cpp : This file contains the source code for the Do_While1 program.
2 //
3 // Created by: [Your Name]
4 // Date: [Your Date]
5 //
6 // Description: This program prints a pattern of asterisks using nested while loops.
7 //
8 // The pattern is as follows:
9 //
10 // *
11 // **
12 // ***
13 // ****
14 // *****
15 //
16 // The program uses a while loop to iterate over the number of rows (1 to 5).
17 // Inside the while loop, another while loop is used to print the asterisks for each row.
18 // The number of asterisks in each row is equal to the row number.
19 //
20 // Example output:
21 // *
22 // **
23 // ***
24 // ****
25 // *****
26 //
27 // The program ends with a return statement.
```



While1

```

1  # Program untuk menampilkan nilai variabel agil
2  # menggunakan while loop
3
4  # Deklarasi variabel
5  agil = 1
6
7  # Loop while
8  while (agil <= 15):
9      # Menampilkan nilai variabel agil
10     print(agil)
11     # Menampilkan "Nilai Agil"
12     print("Nilai Agil")
13     # Increment variabel agil
14     agil = agil + 1
15
16 # End of program
17
18 # Output
19 1
20 Nilai Agil
21 2
22 Nilai Agil
23 3
24 Nilai Agil
25 4
26 Nilai Agil
27 5
28 Nilai Agil
29 6
30 Nilai Agil
31 7
32 Nilai Agil
33 8
34 Nilai Agil
35 9
36 Nilai Agil
37 10
38 Nilai Agil
39 11
40 Nilai Agil
41 12
42 Nilai Agil
43 13
44 Nilai Agil
45 14
46 Nilai Agil
47 15
48 Nilai Agil
49
50

```

Pseudocode :

Kamus

agil = int

Deskripsi

agil = 15

do

print(agil)

print("Nilai Agil")

while (agil >= 1)

endwhile

Alogaritma :

agil = 15

menampilkan nilai variabel agil

menampilkan "Nilai Agil"

Selama (agil >= 1)

Selesai

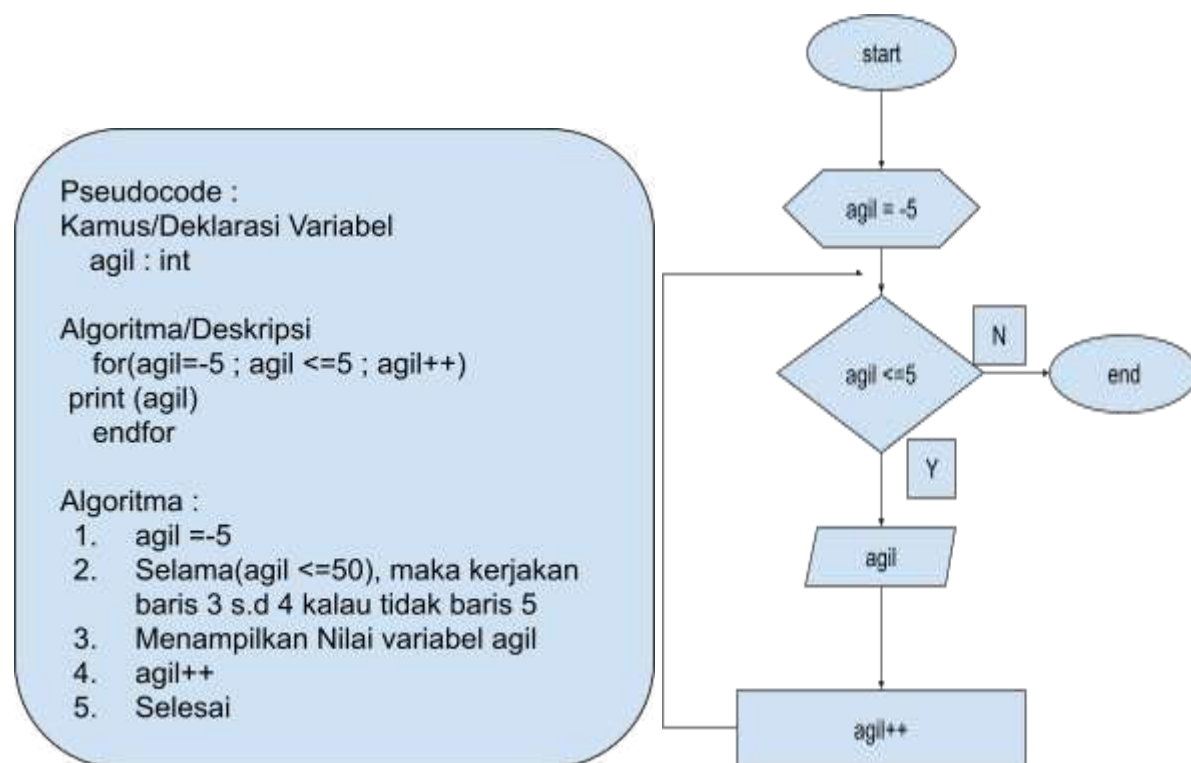
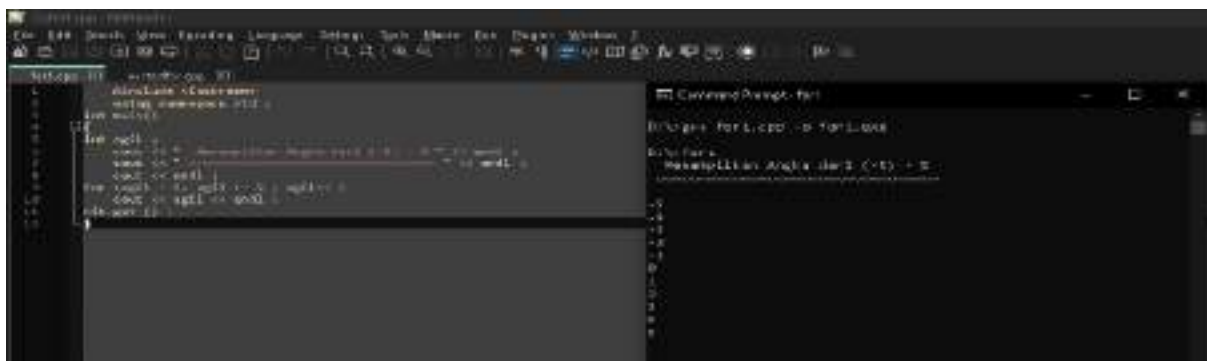


1.3.5 Repetition/Pengulangan : For, Nested For

Pengulangan (repetition) dapat dilakukan dengan menggunakan beberapa struktur pengulangan, seperti while dan do-while. Struktur pengulangan while akan mengevaluasi kondisi sebelum menjalankan blok kode, sedangkan struktur pengulangan do-while akan menjalankan blok kode setidaknya sekali sebelum mengevaluasi kondisi. Penggunaan kedua struktur pengulangan ini memungkinkan untuk melakukan tindakan berulang berdasarkan kondisi yang diberikan. Selain itu, pengulangan juga dapat dilakukan dengan struktur for yang umumnya digunakan ketika jumlah pengulangan sudah diketahui sebelumnya. Hal ini memungkinkan untuk melakukan pengulangan dengan jumlah iterasi yang sudah diketahui sebelumnya

Contoh :

For



NestedFor

The screenshot shows a C++ program in a code editor and its execution output in a terminal window.

```

1 //Nested For Loop
2 using namespace std;
3
4 int main ()
5 {
6     int agilk, agilm, agiln;
7     cout << endl;
8     cout << "Menampilkan Perulangan 3D (agilk) dari 00 (agilm) = 00 agiln" << endl;
9     cout << "===== " << endl;
10    agiln = 1;
11    for (agilk = 1; agilk <= 4; agilk++)
12    {
13        for (agilm = agilk; agilm <= 6; agilm++)
14        {
15            cout << "Nilai dari 0 agiln " << agiln << endl;
16        }
17    }
18 }
  
```

The terminal output shows the execution of the program, displaying the nested loops and the resulting output for each iteration of the innermost loop.

```

D:\C++> g++ nestedfor.cpp -o nestedfor.exe
D:\C++> nestedfor
Menampilkan Perulangan 3D (agilk) dari 00 (agilm) = 00 agiln
=====
00
01
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
  
```

Pseudocode :

Kamus/Deklarasi Variabel

agilk, : 0 ,agilm,agiln = int

Algoritma/Deskripsi

```

for(agilm=1 ; agilm <=4 ; agilm++)
    for(agiln=agilm ; agiln <=6 ; agiln++)
        agiln=agiln+5
    endfor
endfor
print(agiln)
  
```

Algoritma :

1. agiln = 1
2. agilm = 1
3. agilm = agilk
4. Selama(agilk <=4), maka kerjakan baris 5 s.d 8
5. Selama(agilm <=6), maka kerjakan baris 6 s.d 7
6. agiln = agiln+5
7. agilm = agilm+1
8. agilk = agilk+1
9. Menampilkan Nilai variabel agiln
10. Selesai

1.3.6 Rekrusif, Fibonacci

Rekursi dalam pemrograman adalah sebuah konsep di mana sebuah fungsi memanggil dirinya sendiri. Hal ini memungkinkan penyelesaian masalah secara iteratif dengan memecah masalah menjadi submasalah yang lebih kecil. Fungsi rekursif umumnya memiliki kondisi dasar (base case) yang menghentikan pemanggilan diri sendiri. Jika kondisi dasar tidak terpenuhi, fungsi akan terus memanggil dirinya sendiri. Contoh penerapan rekursi dalam bahasa pemrograman C++ antara lain untuk menghitung nilai faktorial suatu bilangan. Meskipun rekursi dapat memudahkan penyelesaian masalah tertentu, namun penggunaannya perlu hati-hati untuk menghindari infinite loop dan memastikan efisiensi penggunaan memori.

Contoh :

Fibonacci

```

#include <iostream>
using namespace std;

int fibonacci(int n) {
    if (n == 1 || n == 2) return 1;
    return fibonacci(n-1) + fibonacci(n-2);
}

int main() {
    for (int i = 1; i <= 10; i++) {
        cout << "Fibonacci(" << i << ") = " << fibonacci(i) << endl;
    }
    return 0;
}

```

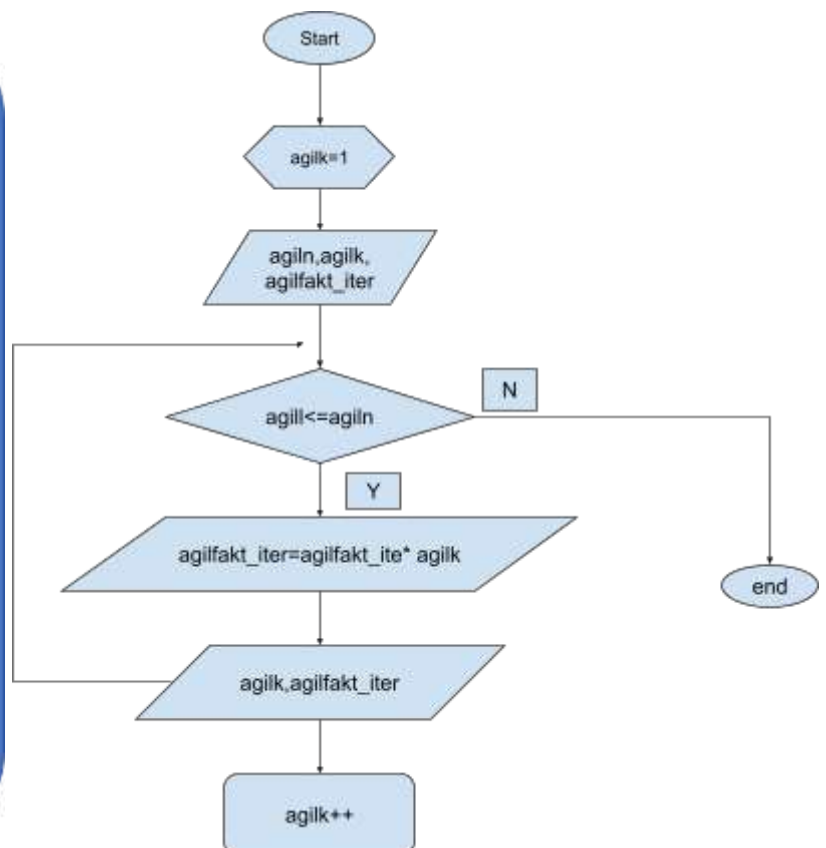
• Pseudocode

KAMUS/DEKLARASI VARIABEL
 agilk, agilkfakt_iter = 1 : long int

ALOGARITMA/DESKRIPSI
 for (agilk = 1; agilk <= agilkfakt_iter; agilk++)
 agilkfakt_iter = agilkfakt_iter * agilk
 print (agilk, agilkfakt_iter)
endfor

• Alogaritma/Bahasa Alami

- const agilkfakt_iter = 1
- const agilk = 1
- const agilk <= 5
- Selama (agilk <= agilkfakt_iter), maka kerjakan baris ke 4 s.d 5 kalau tidak baris 6
- Agilkfakt_iter = agilkfakt_iter * agilk
- mencetak isi/nilai variabel agilk dan agilkfakt_iter
- selesai



1.3.7 Array 1 Dimensi

Array satu dimensi adalah kumpulan nilai data dengan tipe yang sama dan urutan tertentu yang memakai nama. Array satu dimensi dapat diakses melalui indeks yang ada di dalamnya. Array satu dimensi sering digunakan dalam pemrograman untuk menyimpan data yang banyak dan memudahkan pengaksesan data tersebut. Deklarasi array satu dimensi dilakukan dengan menentukan tipe data dan jumlah elemen yang akan disimpan dalam array tersebut. Setelah dideklarasikan, elemen-elemen dalam array dapat diakses dan dimanipulasi menggunakan indeks.

Contoh :

Array

```

1 // Array 1 Dimensi
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     const int agilDt = 3;
8     int Nilai[agilDt] = {22, 7, 1996};
9     int agil, Total = 0;
10    cout << endl;
11    cout << "Menghitung Array 1 Dimensi " << endl;
12    cout << " " << endl;
13    for(agil = 0; agil < agilDt; agil++)
14    {
15        cout << "Nilai elemen ke- " << agil + 1 << " = " << Nilai[agil] << endl;
16        Total = Total + Nilai[agil];
17    }
18    cout << endl;
19    cout << "Total dari Penjumlahan = " << Total << endl;
20    cin.get();
21 }
  
```

Output Console Program - Array:

```

Menghitung Array 1 Dimensi
Nilai elemen ke- 1 = 22
Nilai elemen ke- 2 = 7
Nilai elemen ke- 3 = 1996
Total dari Penjumlahan = 2025
  
```

PSEUDOCODE
KAMUS/DEKLARASI VARIABEL
 agil, agilDt, Nilai[agilDt], Total : int

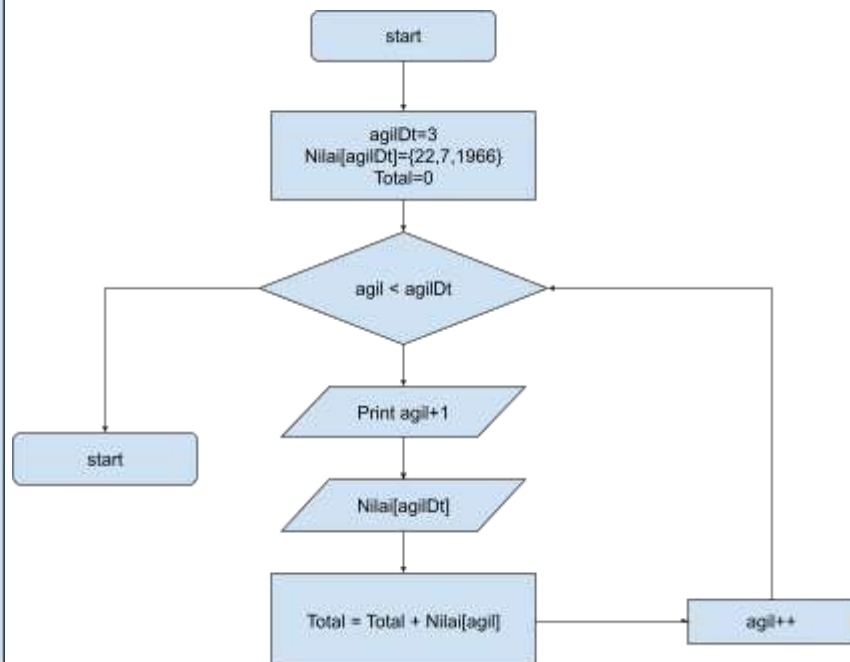
ALGORITMA/DESKRIPSI

```

    agilDt = 3
    Nilai[agilDt] = {22, 7, 1996}
    Total = 0
    for(agil = 0 ; agil < agilDt ; agil++)
        Total = Total + Nilai[agil]
        print(agil+1)
    end
  
```

Algoritma/Bahasa Alami:

1. const agilDt=3
2. Total=0
3. Nilai[agilDt] = {22,7,1966}
4. Selama (agil < agilDt),
5. Mencetak isi/nilai variabel (agil+1)
6. Mencetak isi/nilai variabel Nilai[agil]
7. agil++
8. selesai




1.3.8 Array 2 Dimensi

Array dua dimensi adalah kumpulan nilai data dengan tipe yang sama dan urutan tertentu yang memakai nama dan memiliki dua dimensi atau lebih. Array dua dimensi sering digunakan dalam pemrograman untuk menyimpan data yang terstruktur seperti matriks atau tabel. Deklarasi array dua dimensi dilakukan dengan menentukan tipe data, jumlah baris, dan jumlah kolom yang akan disimpan dalam array tersebut. Setelah dideklarasikan, elemen-elemen dalam array dapat diakses dan dimanipulasi menggunakan indeks baris dan kolom. Array dua dimensi juga dapat digunakan bersama dengan struktur pengulangan seperti for, while, dan do-while untuk melakukan pengolahan data secara berulang.

Contoh :

Array2



```
1 // Array 2 Dimensi
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     // Deklarasi Array 2 Dimensi
8     int nilai[3][4] = {{11,22,33,44},{63,11,66,7},{22,28,11,7}};
9
10    // Menampilkan Array 2 Dimensi
11    for (int i = 0; i < 3; i++)
12    {
13        for (int j = 0; j < 4; j++)
14        {
15            cout << nilai[i][j] << " ";
16        }
17        cout << endl;
18    }
19
20    return 0;
21 }
```

Pseudocode

KAMUS/DEKLARASI VARIABEL

Nilai[3][4] : int agil_i, agil_j: int

ALGORITMA/DESKRIPSI

Nilai[3][4] = {{11,22,33,44},{63,11,66,7},{22,28,11,7}}

for (agil_i=0; agil_i < 3; agil_i++)

for (agil_j=0; agil_j < 4; agil_j++)

print (nilai[i][j], " ")

end for

ALGORITMA :

1. agil_i 0

2. agil_j 0

3. Nilai[3][4] = {11, 22, 33, 44}; {63, 11, 66, 7}; {22, 28, 11, 7}

4. Selama (agil_i < 3), kerjakan baris 5 s.d. 9, kalau tidak baris 10

5. Selama (agil_j < 4), kerjakan baris 6 s.d. 8, kalau tidak baris 10

6. Mencetak Nilai[agil_i][agil_j]

7. Mencetak (" ")

8. agil_j = agil_j + 1

9. agil_i = agil_i + 1

10. Selesai

Array21

```

1 // Deklarasi Array
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     int Nilai[4][4];
7     int agil_i, agil_j;
8
9     // Input
10    for (agil_i = 0; agil_i < 4; agil_i++) {
11        for (agil_j = 0; agil_j < 4; agil_j++) {
12            cout << "Masukkan nilai array 2 dimensi ";
13            cin >> Nilai[agil_i][agil_j];
14        }
15    }
16
17    // Output
18    for (agil_i = 0; agil_i < 4; agil_i++) {
19        for (agil_j = 0; agil_j < 4; agil_j++) {
20            cout << Nilai[agil_i][agil_j] << " ";
21        }
22        cout << endl;
23    }
24
25    return 0;
26 }

```

```

Pilih> array21.cpp -o array21.exe
Pilih> array21
Masukkan nilai array 2 dimensi
1 2 3 4
5 6 7 8
9 10 11 12
13 14 15 16

```

Algoritma

1. $agil_i \leftarrow 0$
2. $agil_j \leftarrow 0$
3. Selama ($agil_i < 4$)
4. Selama ($agil_j < 4$)
5. Mencetak nilai variabel $agil_i+1$
6. Mencetak nilai variabel $agil_j+1$
7. Menginput nilai variabel $Nilai[agil_i][agil_j]$
8. $agil_i++$
9. $agil_j++$
10. Selama ($agil_i < 4$),
11. Selama ($agil_j < 4$),
12. Mencetak nilai variabel $Nilai[agil_i][agil_j]$
13. $agil_i++$
14. $agil_j++$
15. Selesai

PSEUDOCODE

DEKLARASI VARIABEL

Nilai[4][4], agil_i, agil_j: int

DESKRIPSI

agil_i = 0

agil_j = 0

for (agil_i = 0; agil_i < 4; agil_i++)

for (agil_j = 0; agil_j < 4; agil_j++)

print (agil_i+1, agil_j+1)

input (Nilai[agil_i][agil_j])

for (agil_i = 0; agil_i < 4; agil_i++)

for (agil_j = 0; agil_j < 4; agil_j++)

print (Nilai[agil_i][agil_j])

endfor


```

1 // main.cpp
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     int a[3][3], b[3];
7     int result[3];
8     for (int i = 0; i < 3; i++) {
9         for (int j = 0; j < 3; j++) {
10             cin >> a[i][j];
11         }
12     }
13     for (int i = 0; i < 3; i++) {
14         cin >> b[i];
15     }
16     for (int i = 0; i < 3; i++) {
17         result[i] = 0;
18         for (int j = 0; j < 3; j++) {
19             result[i] += a[i][j] * b[j];
20         }
21     }
22     for (int i = 0; i < 3; i++) {
23         cout << result[i] << " ";
24     }
25     return 0;
26 }
27
28 // main2.cpp
29 #include <iostream>
30 using namespace std;
31
32 int main() {
33     int a[3][3], b[3];
34     int result[3];
35     for (int i = 0; i < 3; i++) {
36         for (int j = 0; j < 3; j++) {
37             cin >> a[i][j];
38         }
39     }
40     for (int i = 0; i < 3; i++) {
41         cin >> b[i];
42     }
43     for (int i = 0; i < 3; i++) {
44         result[i] = 0;
45         for (int j = 0; j < 3; j++) {
46             result[i] += a[i][j] * b[j];
47         }
48     }
49     for (int i = 0; i < 3; i++) {
50         cout << result[i] << " ";
51     }
52     return 0;
53 }
54
55 // main3.cpp
56 #include <iostream>
57 using namespace std;
58
59 int main() {
60     int a[3][3], b[3];
61     int result[3];
62     for (int i = 0; i < 3; i++) {
63         for (int j = 0; j < 3; j++) {
64             cin >> a[i][j];
65         }
66     }
67     for (int i = 0; i < 3; i++) {
68         cin >> b[i];
69     }
70     for (int i = 0; i < 3; i++) {
71         result[i] = 0;
72         for (int j = 0; j < 3; j++) {
73             result[i] += a[i][j] * b[j];
74         }
75     }
76     for (int i = 0; i < 3; i++) {
77         cout << result[i] << " ";
78     }
79     return 0;
80 }
81
82 // main4.cpp
83 #include <iostream>
84 using namespace std;
85
86 int main() {
87     int a[3][3], b[3];
88     int result[3];
89     for (int i = 0; i < 3; i++) {
90         for (int j = 0; j < 3; j++) {
91             cin >> a[i][j];
92         }
93     }
94     for (int i = 0; i < 3; i++) {
95         cin >> b[i];
96     }
97     for (int i = 0; i < 3; i++) {
98         result[i] = 0;
99         for (int j = 0; j < 3; j++) {
100             result[i] += a[i][j] * b[j];
101         }
102     }
103     for (int i = 0; i < 3; i++) {
104         cout << result[i] << " ";
105     }
106     return 0;
107 }
108
109 // main5.cpp
110 #include <iostream>
111 using namespace std;
112
113 int main() {
114     int a[3][3], b[3];
115     int result[3];
116     for (int i = 0; i < 3; i++) {
117         for (int j = 0; j < 3; j++) {
118             cin >> a[i][j];
119         }
120     }
121     for (int i = 0; i < 3; i++) {
122         cin >> b[i];
123     }
124     for (int i = 0; i < 3; i++) {
125         result[i] = 0;
126         for (int j = 0; j < 3; j++) {
127             result[i] += a[i][j] * b[j];
128         }
129     }
130     for (int i = 0; i < 3; i++) {
131         cout << result[i] << " ";
132     }
133     return 0;
134 }
135
136 // main6.cpp
137 #include <iostream>
138 using namespace std;
139
140 int main() {
141     int a[3][3], b[3];
142     int result[3];
143     for (int i = 0; i < 3; i++) {
144         for (int j = 0; j < 3; j++) {
145             cin >> a[i][j];
146         }
147     }
148     for (int i = 0; i < 3; i++) {
149         cin >> b[i];
150     }
151     for (int i = 0; i < 3; i++) {
152         result[i] = 0;
153         for (int j = 0; j < 3; j++) {
154             result[i] += a[i][j] * b[j];
155         }
156     }
157     for (int i = 0; i < 3; i++) {
158         cout << result[i] << " ";
159     }
160     return 0;
161 }
162
163 // main7.cpp
164 #include <iostream>
165 using namespace std;
166
167 int main() {
168     int a[3][3], b[3];
169     int result[3];
170     for (int i = 0; i < 3; i++) {
171         for (int j = 0; j < 3; j++) {
172             cin >> a[i][j];
173         }
174     }
175     for (int i = 0; i < 3; i++) {
176         cin >> b[i];
177     }
178     for (int i = 0; i < 3; i++) {
179         result[i] = 0;
180         for (int j = 0; j < 3; j++) {
181             result[i] += a[i][j] * b[j];
182         }
183     }
184     for (int i = 0; i < 3; i++) {
185         cout << result[i] << " ";
186     }
187     return 0;
188 }
189
190 // main8.cpp
191 #include <iostream>
192 using namespace std;
193
194 int main() {
195     int a[3][3], b[3];
196     int result[3];
197     for (int i = 0; i < 3; i++) {
198         for (int j = 0; j < 3; j++) {
199             cin >> a[i][j];
200         }
201     }
202     for (int i = 0; i < 3; i++) {
203         cin >> b[i];
204     }
205     for (int i = 0; i < 3; i++) {
206         result[i] = 0;
207         for (int j = 0; j < 3; j++) {
208             result[i] += a[i][j] * b[j];
209         }
210     }
211     for (int i = 0; i < 3; i++) {
212         cout << result[i] << " ";
213     }
214     return 0;
215 }
216
217 // main9.cpp
218 #include <iostream>
219 using namespace std;
220
221 int main() {
222     int a[3][3], b[3];
223     int result[3];
224     for (int i = 0; i < 3; i++) {
225         for (int j = 0; j < 3; j++) {
226             cin >> a[i][j];
227         }
228     }
229     for (int i = 0; i < 3; i++) {
230         cin >> b[i];
231     }
232     for (int i = 0; i < 3; i++) {
233         result[i] = 0;
234         for (int j = 0; j < 3; j++) {
235             result[i] += a[i][j] * b[j];
236         }
237     }
238     for (int i = 0; i < 3; i++) {
239         cout << result[i] << " ";
240     }
241     return 0;
242 }
243
244 // main10.cpp
245 #include <iostream>
246 using namespace std;
247
248 int main() {
249     int a[3][3], b[3];
250     int result[3];
251     for (int i = 0; i < 3; i++) {
252         for (int j = 0; j < 3; j++) {
253             cin >> a[i][j];
254         }
255     }
256     for (int i = 0; i < 3; i++) {
257         cin >> b[i];
258     }
259     for (int i = 0; i < 3; i++) {
260         result[i] = 0;
261         for (int j = 0; j < 3; j++) {
262             result[i] += a[i][j] * b[j];
263         }
264     }
265     for (int i = 0; i < 3; i++) {
266         cout << result[i] << " ";
267     }
268     return 0;
269 }
270
271 // main11.cpp
272 #include <iostream>
273 using namespace std;
274
275 int main() {
276     int a[3][3], b[3];
277     int result[3];
278     for (int i = 0; i < 3; i++) {
279         for (int j = 0; j < 3; j++) {
280             cin >> a[i][j];
281         }
282     }
283     for (int i = 0; i < 3; i++) {
284         cin >> b[i];
285     }
286     for (int i = 0; i < 3; i++) {
287         result[i] = 0;
288         for (int j = 0; j < 3; j++) {
289             result[i] += a[i][j] * b[j];
290         }
291     }
292     for (int i = 0; i < 3; i++) {
293         cout << result[i] << " ";
294     }
295     return 0;
296 }
297
298 // main12.cpp
299 #include <iostream>
300 using namespace std;
301
302 int main() {
303     int a[3][3], b[3];
304     int result[3];
305     for (int i = 0; i < 3; i++) {
306         for (int j = 0; j < 3; j++) {
307             cin >> a[i][j];
308         }
309     }
310     for (int i = 0; i < 3; i++) {
311         cin >> b[i];
312     }
313     for (int i = 0; i < 3; i++) {
314         result[i] = 0;
315         for (int j = 0; j < 3; j++) {
316             result[i] += a[i][j] * b[j];
317         }
318     }
319     for (int i = 0; i < 3; i++) {
320         cout << result[i] << " ";
321     }
322     return 0;
323 }
324
325
```

1. $agil_i \leftarrow 0$
2. $Agil_j \leftarrow 0$
3. Selama ($agil_i < 4$)
4. Selama ($agil_j < agil_i+1$)
5. Mencetak nilai variabel $agil_i+1$
6. Mencetak nilai variabel $agil_j+1$
7. Menginput nilai variabel $Nilai[agil_i][agil_j]$
8. $Agil_i++$
9. $Agil_j++$
10. Selama ($agil_i < 4$),
11. Selama ($agil_j < agil_i+1$),
12. Mencetak nilai variabel $Nilai[agil_i][agil_j]$
13. $Agil_i++$
14. $Agil_bj++$
15. Selesai

endfor

Array23

[illegible]

```

17 cout << endl;
18 cout << "Transfer the Diagonal Array Elements A" << endl;
19 cout << "-----" << endl;
20
21 cout << endl;
22 for (agil_1=0; agil_1<agil_1A; agil_1++) {
23     for (agil_2=0; agil_2<agil_2A; agil_2++)
24         cout << agil_1[agil_1][agil_2] << " ";
25     cout << endl;
26 }
27 cout << endl;
28 cout << endl;
29 cout << "Transfer the Diagonal Array Elements B" << endl;
30 cout << "-----" << endl;
31
32 cout << endl;
33 for (agil_1=0; agil_1<agil_1B; agil_1++) {
34     for (agil_2=0; agil_2<agil_2B; agil_2++)
35         cout << agil_2[agil_1][agil_2] << " ";
36     cout << endl;
37 }
38 //Permutation Matrix
39 for (agil_1=0; agil_1<agil_1A; agil_1++)
40     for (agil_2=0; agil_2<agil_2A; agil_2++)
41         for (agil_3=0; agil_3<agil_3A; agil_3++)
42             {
43                 agil_3[agil_1][agil_2] = agil_1[agil_1][agil_2] + agil_2[agil_1][agil_3] * agil_3[agil_1][agil_2];
44             }
45 cout << endl;
46 cout << endl;
47 cout << "Transfer Result All the Matrix " << endl;
48 cout << "-----" << endl;
49
50 for (agil_1=0; agil_1<agil_1A; agil_1++)
51 {
52     for (agil_2=0; agil_2<agil_2A; agil_2++)
53     {
54         cout << agil_3[agil_1][agil_2] << " ";
55     }
56     cout << endl;
57 }
58 }

```

```
Command Prompt
E:\>g++ array23.cpp -o array23.exe
E:\>array23
Menampilkan Array/larik multi dimensi
Menampilkan Besar Baris dan Kolom Matriks A
~~~~~
Masukkan Jumlah Baris Matriks A = 3
Masukkan Jumlah Kolom Matriks A = 3
~~~~~
Masukkan Jumlah Baris Matriks B = 3
Masukkan Jumlah Kolom Matriks B = 2
~~~~~
Menginput Isi Elemen Array Matriks A
Masukkan Isi Elemen [1][1] : 3
Masukkan Isi Elemen [1][2] : 2
Masukkan Isi Elemen [1][3] : 1
Masukkan Isi Elemen [2][1] : 4
Masukkan Isi Elemen [2][2] : 3
Masukkan Isi Elemen [2][3] : 2
Masukkan Isi Elemen [3][1] : 5
Masukkan Isi Elemen [3][2] : 4
Masukkan Isi Elemen [3][3] : 2

Menginput isi elemen array matriks b
Masukkan Isi Elemen{1}{1} : 3
Masukkan Isi Elemen{1}{2} : 5
Masukkan Isi Elemen{2}{1} : 2
Masukkan Isi Elemen{2}{2} : 3
Masukkan Isi Elemen{3}{1} : 5
Masukkan Isi Elemen{3}{2} : 4

Tampilan Isi Elemen Array Matriks A
~~~~~
3      2      1
4      3      2
5      4      2
```

```
Tampilan Isi Elemen Array Matriks A
~~~~~
3      2      1
4      3      2
5      4      2

3      5
2      3
5      4

Tampilan Hasil Kali Dua Matriks
~~~~~
18      25
```

1.3.9 Function/Fungsi

Fungsi (function) dalam bahasa pemrograman adalah sebuah blok kode yang dirancang untuk melakukan tugas tertentu dan dapat dipanggil dari bagian lain dalam program. Fungsi dapat membantu memecah program menjadi bagian-bagian yang lebih kecil dan lebih mudah dikelola. Fungsi juga dapat digunakan untuk menghindari duplikasi kode dan meningkatkan efisiensi program. Fungsi dapat memiliki parameter sebagai input dan dapat mengembalikan nilai sebagai output.

Contoh :

Function

```
1 //main.cpp 2 //main.cpp 3 #include <iostream> 4 using namespace std; 5 6 int input(); 7 int hasil(); 8 int tampil(int hasil [10] [10]); 9 int agil_a[10] [10], agil_b[10] [10]; 10 int agil_c, agil_d, agil_e, agil_f; 11 12 int main() 13 { 14     int pilih; 15     input(); 16     do 17     { 18         cout << "Masukkan AKADEMI/SEKOLAH/UMUM: " << endl; 19         cout << "Masa Operasi Matriks: " << endl; 20         cout << "===== " << endl; 21         cout << "1. Penjumlahan Matriks" << endl; 22         cout << "2. Kalian" << endl; 23         cout << "===== " << endl; 24         cout << "Wilayah Area: " << endl; 25         cin >> pilih; 26         switch(pilih) 27         { 28             case 1: 29             { 30                 hasil(); 31                 break; 32             } 33             default: 34             { 35                 while(pilih != 2); 36             } 37         } 38     } while(pilih != 2); 39     return 0; 40 }
```

```
1 //main.cpp 2 //main.cpp 3 #include <iostream> 4 using namespace std; 5 6 int input(); 7 int hasil(); 8 int tampil(int hasil [10] [10]); 9 int agil_a[10] [10], agil_b[10] [10]; 10 int agil_c, agil_d, agil_e, agil_f; 11 12 int main() 13 { 14     int pilih; 15     input(); 16     do 17     { 18         cout << "Masukkan AKADEMI/SEKOLAH/UMUM: " << endl; 19         cout << "Masa Operasi Matriks: " << endl; 20         cout << "===== " << endl; 21         cout << "1. Penjumlahan Matriks" << endl; 22         cout << "2. Kalian" << endl; 23         cout << "===== " << endl; 24         cout << "Wilayah Area: " << endl; 25         cin >> pilih; 26         switch(pilih) 27         { 28             case 1: 29             { 30                 hasil(); 31                 break; 32             } 33             default: 34             { 35                 while(pilih != 2); 36             } 37         } 38     } while(pilih != 2); 39     return 0; 40 }
```

```
1 //main.cpp 2 //main.cpp 3 #include <iostream> 4 using namespace std; 5 6 int input(); 7 int hasil(); 8 int tampil(int hasil [10] [10]); 9 int agil_a[10] [10], agil_b[10] [10]; 10 int agil_c, agil_d, agil_e, agil_f; 11 12 int main() 13 { 14     int pilih; 15     input(); 16     do 17     { 18         cout << "Masukkan AKADEMI/SEKOLAH/UMUM: " << endl; 19         cout << "Masa Operasi Matriks: " << endl; 20         cout << "===== " << endl; 21         cout << "1. Penjumlahan Matriks" << endl; 22         cout << "2. Kalian" << endl; 23         cout << "===== " << endl; 24         cout << "Wilayah Area: " << endl; 25         cin >> pilih; 26         switch(pilih) 27         { 28             case 1: 29             { 30                 hasil(); 31                 break; 32             } 33             default: 34             { 35                 while(pilih != 2); 36             } 37         } 38     } while(pilih != 2); 39     return 0; 40 }
```

```
1 //main.cpp 2 //main.cpp 3 #include <iostream> 4 using namespace std; 5 6 int input(); 7 int hasil(); 8 int tampil(int hasil [10] [10]); 9 int agil_a[10] [10], agil_b[10] [10]; 10 int agil_c, agil_d, agil_e, agil_f; 11 12 int main() 13 { 14     int pilih; 15     input(); 16     do 17     { 18         cout << "Masukkan AKADEMI/SEKOLAH/UMUM: " << endl; 19         cout << "Masa Operasi Matriks: " << endl; 20         cout << "===== " << endl; 21         cout << "1. Penjumlahan Matriks" << endl; 22         cout << "2. Kalian" << endl; 23         cout << "===== " << endl; 24         cout << "Wilayah Area: " << endl; 25         cin >> pilih; 26         switch(pilih) 27         { 28             case 1: 29             { 30                 hasil(); 31                 break; 32             } 33             default: 34             { 35                 while(pilih != 2); 36             } 37         } 38     } while(pilih != 2); 39     return 0; 40 }
```



```

D:\>function1
Masukan Ordo Matriks A :
Jumlah Baris : 4
Jumlah Kolom : 4
Masukan Ordo Matriks B :
Jumlah Baris : 4
Jumlah Kolom : 4
Matriks A :
Masukan Isi Elemen [1][1] : 2
Masukan Isi Elemen [1][2] : 3
Masukan Isi Elemen [1][3] : 4
Masukan Isi Elemen [1][4] : 3
Masukan Isi Elemen [2][1] : 4
Masukan Isi Elemen [2][2] : 2
Masukan Isi Elemen [2][3] : 1
Masukan Isi Elemen [2][4] : 2
Masukan Isi Elemen [3][1] : 3
Masukan Isi Elemen [3][2] : 6
Masukan Isi Elemen [3][3] : 5
Masukan Isi Elemen [3][4] : 3
Masukan Isi Elemen [4][1] : 4
Masukan Isi Elemen [4][2] : 2
Masukan Isi Elemen [4][3] : 3
Masukan Isi Elemen [4][4] : 1
Matriks B :
Masukan Isi Elemen [1] [1] : 2
Masukan Isi Elemen [1] [2] : 3
Masukan Isi Elemen [1] [3] : 4
Masukan Isi Elemen [1] [4] : 6
Masukan Isi Elemen [2] [1] : 7
Masukan Isi Elemen [2] [2] : 8
Masukan Isi Elemen [2] [3] : 4
Masukan Isi Elemen [2] [4] : 5
Masukan Isi Elemen [3] [1] : 2
Masukan Isi Elemen [3] [2] : 3
Masukan Isi Elemen [3] [3] : 4
Masukan Isi Elemen [3] [4] : 1
Masukan Isi Elemen [4] [1] : 2
Masukan Isi Elemen [4] [2] : 4
Masukan Isi Elemen [4] [3] : 5
Masukan Isi Elemen [4] [4] : 6
Perintah ARRAY/LARIK Multi Dimensi
Menu Operasi Matriks
<=====>
1. Perkalian Matriks
2. Keluar
<=====>
Pilihan Anda :
1
Hasil Operasi Matriks :
39 54 51 49
28 39 38 47
64 84 71 71
30 41 41 43
Perintah ARRAY/LARIK Multi Dimensi
Menu Operasi Matriks
<=====>
1. Perkalian Matriks
2. Keluar
<=====>
Pilihan Anda :
2
D:\>

```

```

Perintah ARRAY/LARIK Multi Dimensi
Menu Operasi Matriks
<=====>
1. Perkalian Matriks
2. Keluar
<=====>
Pilihan Anda :
1
Hasil Operasi Matriks :
39 54 51 49
28 39 38 47
64 84 71 71
30 41 41 43
Perintah ARRAY/LARIK Multi Dimensi
Menu Operasi Matriks
<=====>
1. Perkalian Matriks
2. Keluar
<=====>
Pilihan Anda :
2
D:\>

```

Deklarasi Variabel Fungsi agil_kali

```
agil_kalip[10][10], agil_c, agil_d, agil_e : int
```

Deskripsi fungsi agil_kali

```
agil_kalip[10][10]={0}
```

```
if (agil_j == agil_k )
```

```
for (agil_c=0; agil_c < agil_i; agil_c++)
```

```
for(agil_d=0; agil_d < agil_l, agil_d++)
```

```
for(agil_e=0; agil_e < agil_c; agil_e++)
```

```
agil_kalip[agil_c][ agil_d] = agil_kalip [agil_c][agil_d] +  
agil_a [agil_c][ agil_e]* agil_b [agil_e][ agil_d]
```

```
endfor
```

```
agil_tampil(agil_kali)
```

```
endfor
```

```
endfor
```

```
else
```

```
print (" Ordo Matriks Tidak Sesuai")
```

```
endif
```

Deklarasi Variabel Fungsi agil_input

```
agil_i, agil_j, agil_k, agil_l, agil_c, agil_k = int
```

Deskripsi Fungsi dhandi input

```
Do
```

```
input(agil_i),
```

```
while (agil_i >= 11)
```

```
do
```

```
input (agil_j)
```

```
while(agil_j >= 11)
```

```
do
```

```
input (agil_k)
```

```
while (agil_k >= 11)
```

```
do
```

```
input(agil_l)
```

```
while(agil_l >= 1
```

```
1)
```

```
for (agil_c=0; agil_c< agil_i; agil_c++)
```

```
for (agil_d=0; agil_d< agil_j; agil_d++)
```

```
print (agil_c+1, agil_d+1)
```

```
input (agil_a[agil_c][ agil_d]
```

```
endfor
```

```
endfor
```

```
for (agil_c=0; agil_c< agil_k; agil_c++)
```

```
for (agil_d=0; agil_d< agil_l; agil_d++)
```

```
print (agil_c+1, agil_d+1)
```

```
input (agil_b[agil_c][ agil_d]
```

```
endfor
```

```
endfor
```

Deklarasi Variabel

```
agil_a[10][10], agil_b[10][10], agil_i,  
agil_j, agil_k, agil_l, pilih : int
```

Deksripsi/Algoritma

```
agil_input()
```

```
do
```

```
input(agil_pilih)
```

```
switch(agil_pilih)
```

```
case 1 :
```

```
agil_kali()
```

```
break;
```

```
default();
```

```
while (agil_pilih != 2)
```

Deklarasi Variabel Fungsi agil_tampil

```
hasil[10][10], agil_c, agil_d : int
```

Deksripsi Fungsi agil_tampil

```
for(agil_c=0; agil_c < agil_i; agil_c++)
```

```
for(agil_d=0; agil_d < agil_l; agil_d++)
```

```
print(hasil[agil_c][agil_d])
```

```
endfor
```

```
endfor
```

● ALOGARITMA

- Membuat fungsi agil_input()
- Selama (agil_i >= 11) maka kerjakan baris 3 kalau tidak baris 4
- Memasukan isi variable (agil_i)
- Selama (agil_j >= 11) maka kerjakan baris 5 kalau tidak baris 6
- Memasukan isi variabel agil_j
- Selama (agil_k >= 11) maka kerjakan baris 7 kalau tidak baris 8
- Menginput/memasukan agil_k
- Selama (agil_l >= 11) maka kerjakan baris 9 kalau tidak baris 10
- Memasukan isi variabel agil_l
- selama (agil_c < agil_i) maka kerjakan baris 11 s.d.16
- selama (agil_d < agil_c) kerjakan baris 12 s.d. 15
- Mencetak/menampilkan agil_c + 1
- Mencetak/Menampilkan agil_d + 1
- Memasukan isi variabel agil_a[agil_c][agil_d]
- agil_d++
- agil_c++
- Selama (agil_c < agil_k) kerjakan baris 18 s.d. 23
- Selama (agil_d < agil_l) kerjakan baris 19 s.d. 22
- Mencetak/menampilkan agil_c + 1
- Mencetak/menampilkan agil_d + 1

- Memasukan/menginput agil_b[agil_c][agil_d]
- agil_d++
- agil_c++
- Membuat fungsi agil_kali()
- Jika (agil_j = agil_k) kerjakan baris 28 s.d. 35 kalau tidak baris 36
- Selama (agil_c < agil_i) maka kerjakan baris 29 s.d.35 kalau tidak baris 36
- Selama (agil_d < agil_l) maka kerjakan baris 30 s.d. 34 kalau tidak baris 36
- Selama (agil_e < agil_j) maka kerjakan baris 31 s.d. 33 kalau tidak baris 36
- agil_kalip[agil_c][agil_d] = agil_kalip[agil_c][agil_d] + agil_a[agil_c][agil_c] * agil_b[agil_c][agil_d]
- agil_tampil(agil_kalip)
- agil_c++
- agil_d++
- agil_e++
- Mencetak/menampilkan “ Ordo Matriks Tidak Sesuai “
- Memanggil fungsi agil_tampil
- Selama (agil_c < agil_i) maka kerjakan baris 39 s.d.42 kalau tidak baris 43
- Selama (agil_d < agil_l) maka kerjakan baris 40 s.d. 41 kalau tidak baris 43
- Mencetak /menampilkan (hasil[agil_c][agil_d]
- agil_d++
- agil_c++
- Selesai

1.3.10 String

Tipe data string dalam bahasa pemrograman digunakan untuk menyimpan dan memanipulasi teks. String sebenarnya merupakan sebuah array satu dimensi yang berisi kumpulan dari huruf atau karakter dan diakhiri dengan karakter kosong \0 (null). Tipe data string memungkinkan programmer untuk mengolah teks dalam program mereka dan melakukan berbagai operasi manipulasi teks seperti menggabungkan, memotong, atau mencari karakter tertentu dalam teks. Beberapa fungsi umum pada operasi string meliputi strcpy() untuk menyalin nilai dari satu string ke string lain, strcat() untuk menggabungkan dua string, strcmp() untuk membandingkan dua string, strlen() untuk mengetahui panjang sebuah string, dan strchr() untuk mencari karakter tertentu dalam sebuah string.

Contoh :

String



- **PSEUDOCODE :**

Algoritma pergantian karakter pertama "A"
Menjadi "S"

KAMUS/DEKLARASI VARIABEL

nama[] : char

ALGORITMA/DESKRIPSI

nama [] = "Agil Deriansyah Hasan"

print (nama)

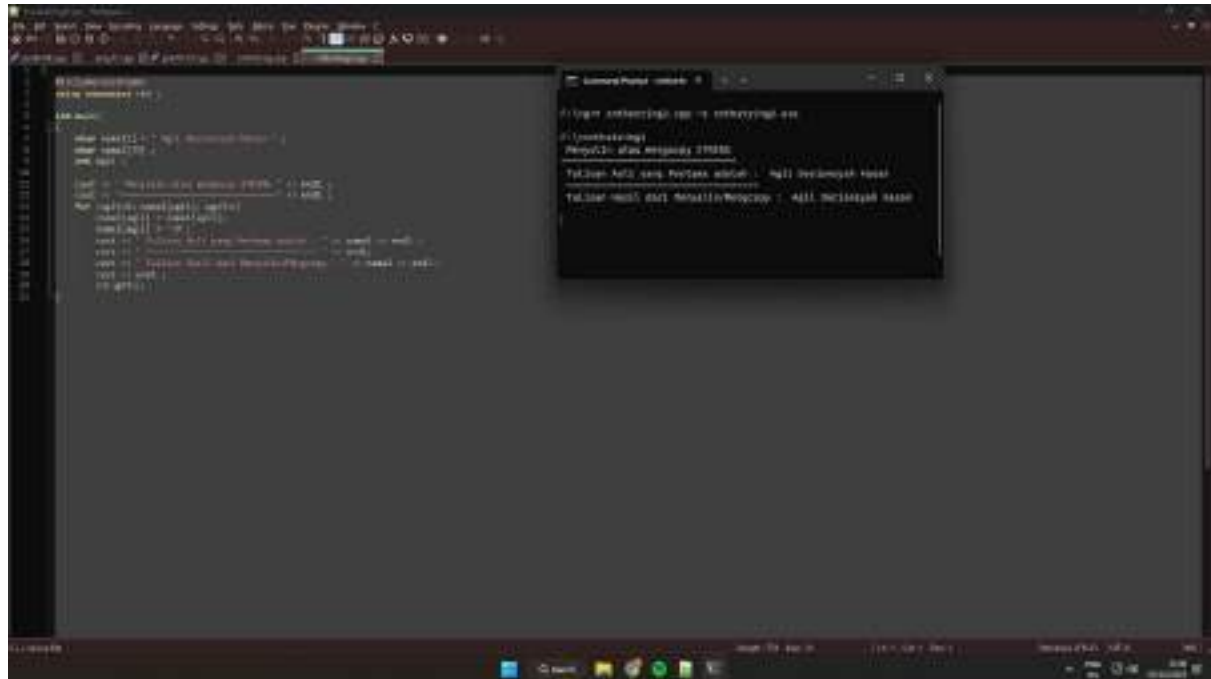
nama [0] = "S"

print (nama)

- **ALGORITMA :**

1. nama [] = "Agil Deriansyah H"
2. Mencetak/menampilkan nama
3. nama [0] = "S"
4. Mencetak/menampilkan nama
5. Selesai

String2



```
#include <iostream>
using namespace std;

int main()
{
    char nama1[20] = "Agil Deriansyah Hasan";
    char nama2[20];
    int agil = 0;

    while (nama1[agil] != '\0')
    {
        nama2[agil] = nama1[agil];
        agil++;
    }
    nama2[agil] = '\0';

    cout << "Nama Asli yang pertama adalah : " << nama1 << endl;
    cout << "Nama Asli yang kedua adalah : " << nama2 << endl;

    return 0;
}
```

- PSEUDOCODE :

Algoritma menyalin string

KAMUS/DEKLARASI VARIABEL

nama1 [], nama02 [30] = char

agil : Int

ALGORITMA/DESKRIPSI

nama01 [] = "Agil Deriansyah Hasan"

for(agil = 0; nama1 [agil]; agil++)

nama2 [agil] = nama01 [agil]

nama2 [agil] = "\0"

endfor

print(nama2)

Algoritma

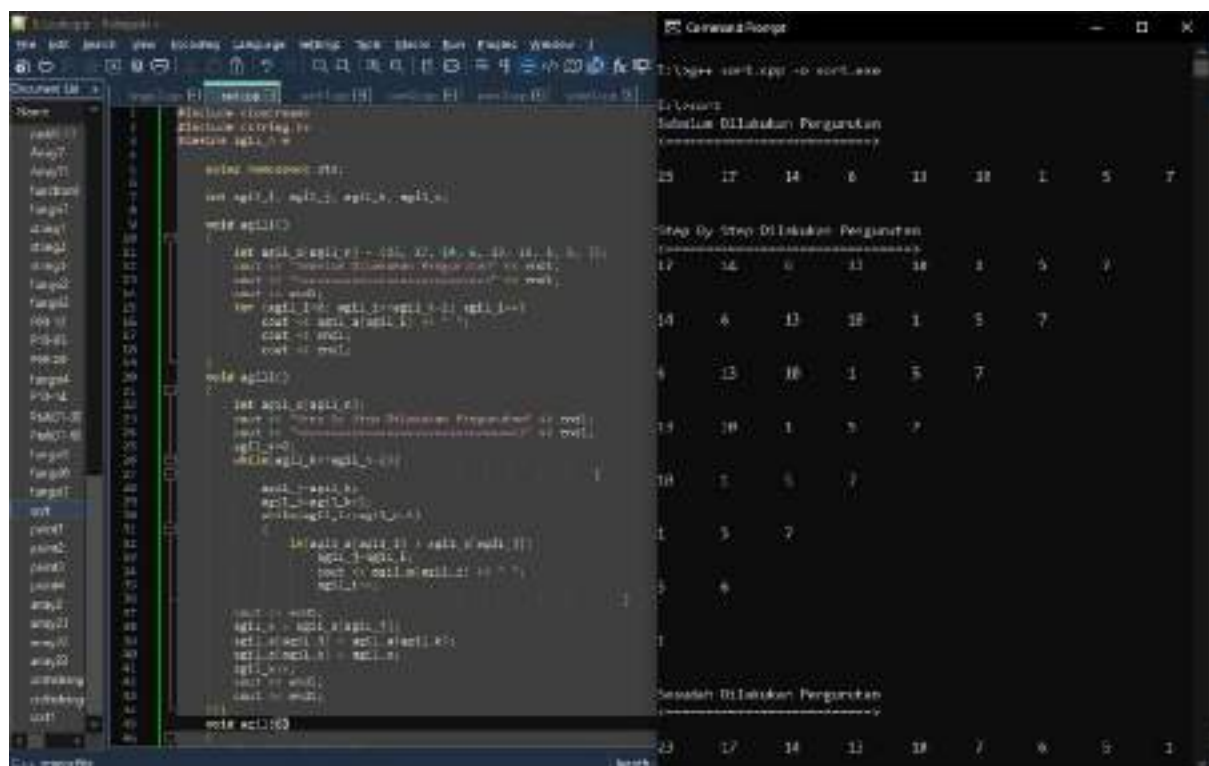
- Nama1[] = " Agil Deriansyah Hasan"
- Agil <- 0
- Selama nama1[agil], maka kerjakan baris s.d 7 kalau tidak baris 8
- Nama2[agil]=nama1[agil]
- Nama2[agil] ='\0'
- Mencetak/menampilkan nama2
- Agil<- agil+1
- selesai

1.3.11 Sort : Insertion, Selection

Dalam bahasa pemrograman, terdapat beberapa algoritma pengurutan data yang umum digunakan, seperti insertion sort, selection sort, dan bubble sort. Algoritma pengurutan data ini digunakan untuk mengurutkan data dalam suatu array atau struktur data lainnya. Algoritma insertion sort bekerja dengan cara membandingkan setiap elemen dalam array dan memindahkan elemen tersebut ke posisi yang tepat dalam array. Algoritma selection sort bekerja dengan cara mencari elemen terkecil dalam array dan memindahkan elemen tersebut ke posisi pertama dalam array, kemudian mencari elemen terkecil kedua dan menempatkannya di posisi array berikutnya.

Contoh :

Sort



```
1 // Selection Sort
2 #include <iostream>
3 #include <string.h>
4 #include <conio.h>
5 using namespace std;
6
7 int agil1, agil2, agil3, agil4;
8
9 void agil1()
10 {
11     int a[agil1] = {25, 17, 14, 6, 11, 18, 1, 5, 7};
12     cout << "Sebelum Dilakukan Pengurutan" << endl;
13     cout << "=====>" << endl;
14     for (int i=0; i<agil1; i++)
15         cout << a[agil1-i] << " ";
16     cout << endl;
17 }
18
19 void agil2()
20 {
21     int a[agil1], agil2;
22     cout << "Data Di Swap Diindex Pengurutan" << endl;
23     cout << "=====>" << endl;
24     for (int i=0; i<agil1; i++)
25         cout << a[agil1-i] << " ";
26     cout << endl;
27 }
28
29 void agil3()
30 {
31     int a[agil1], agil2;
32     cout << "Data Di Swap Diindex Pengurutan" << endl;
33     cout << "=====>" << endl;
34     for (int i=0; i<agil1; i++)
35         cout << a[agil1-i] << " ";
36     cout << endl;
37 }
38
39 int main()
40 {
41     agil1();
42     agil2();
43     agil3();
44     cout << endl;
45     return 0;
46 }
```

```
45 void agil3()
46 {
47     int agil_a[agil_n];
48     cout << "Sesudah Dilakukan Pengurutan" << endl;
49     cout << "=====>" << endl;
50     cout << endl;
51     for(agil_i=0; agil_i<=agil_n-1; agil_i++)
52         cout << agil_a[agil_i] << " ";
53 }
54
55 int main()
56 {
57     agil1();
58     agil2();
59     agil3();
60     cout << endl;
61     return 0; }
```

PSEUDOCODE

Mengurutkan data dengan metode Insertion Sort

KAMUS/DEKLARASI VARIABEL

agil_a[agil_n], agil_i, agil_k, agil_x, agil_n : int

Algoritma/Deskripsi

agil_n=9

agil_A[agil_n] = {23, 17, 14, 6, 13, 10, 1, 5, 7}

for(agil_i=0; agil_i<= agil_n-1; agil_i++)

 print(agil_A[agil_i], " ")

endfor

for(agil_k=0; agil_k<= agil_n; agil_k++)

 agil_i= agil_k

 agil_x= agil_A[agil_i]

 while(agil_i> 0 && agil_A[agil_i-1] > agil_x)

 agil_A[agil_i] = agil_A[agil_i-1]

 agil_i--

 endwhile

 agil_A[agil_i]= agil_x

endfor

for(agil_i= 0; agil_i<= agil_n-1; agil_i++)

 print(agil_A[agil_i], " ")

endfor

Algoritma/Bahasa Alami:

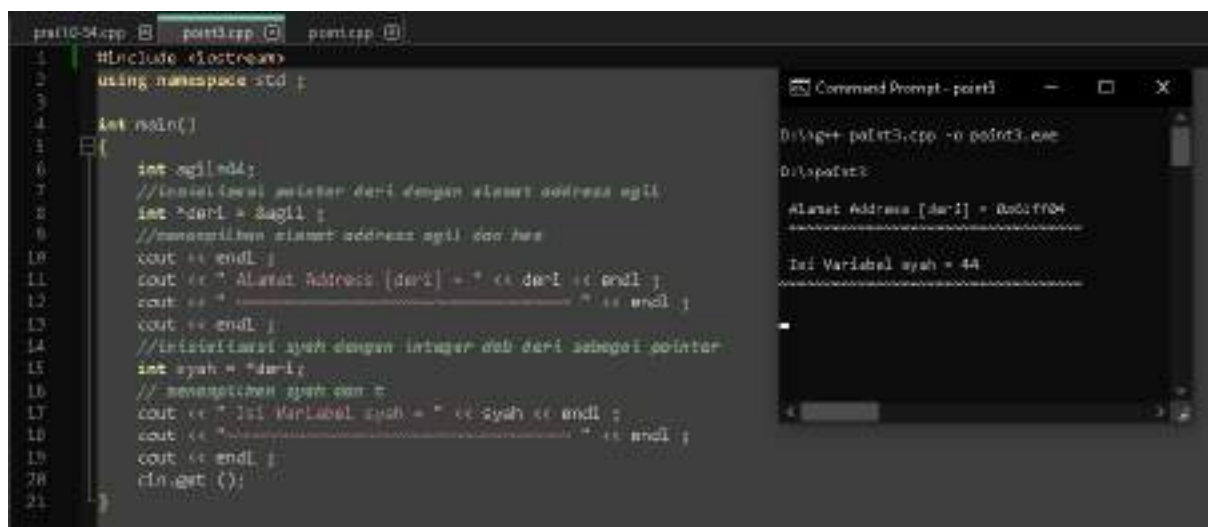
- agil_n=9
- Const agil_A[agil_n] = {23, 17, 14, 6, 13, 10, 1, 5, 7}
- agil_i=0
- Selama(agil_i <= agil_n-1) kerjakan baris 5 s.d.7 kalau tidak baris 8
- Mencetak/menampilkan isi variable agil_A[agil_i]
- Mencetak/menampilkan isi variable (" ")
- agil_i++
- agil_k=0
- Selama(agil_k <= agil_n), maka kerjakan baris 10 s.d. 16 kalau tidak baris 17
- agil_i = agil_k
- agil_x= agil_A[agil_i]
- Selama(agil_i>=0 dan agil_A[agil_i-1] > agil_x), maka kerjakan baris 13 s.d.14 kalau tidak baris 15
- agil_A[agil_i]= dhendi_A[dhendi_i-1]
- agil_i--
- agil_A[agil_i]= agil_x
- agil_k++
- Selama(agil_i <= agil_n-1) kerjakan baris 18 s.d.20 kalau tidak baris 21
- Mencetak/menampilkan isi variable agil_A[agil_i]
- Mencetak/menampilkan isi variable (" ")
- agil_i++
- Selesai

1.3.12 Pointer

Pointer dalam bahasa pemrograman C++ adalah tipe data yang nilainya mengacu pada nilai lain yang disimpan di lokasi memori komputer menggunakan alamatnya. Pointer digambarkan dengan simbol '*' dan memungkinkan program untuk bekerja secara lebih efisien dengan menggunakan alamat memori variabel. Dengan menggunakan pointer, program dapat mengakses dan memanipulasi variabel secara langsung di lokasi memori tertentu. Pointer juga digunakan untuk alokasi memori dinamis dan untuk mentransfer alamat memori ke fungsi lain.

Contoh :

Point3



The screenshot shows a C++ program named 'point3.cpp' in a code editor. The program includes `<iostream>` and uses the `std` namespace. It defines a `main` function where it declares an integer `agil` and a pointer `deri` of type `int`. `agil` is assigned the value 44. `deri` is assigned the address of `agil` using `&agil`. The program then prints the address stored in `deri` and the value stored at that address (which is 44). It also declares a pointer `syah` of type `int` and assigns it the value of `*deri` (44). Finally, it prints the value of `syah`.

```
1 #include <iostream>
2 using namespace std;
3
4 int main()
5 {
6     int agil=44;
7     //inisialisasi pointer deri dengan alamat address agil
8     int *deri = &agil;
9     //memasukkan alamat address agil ke deri
10    cout << endl;
11    cout << "Alamat Address [deri] = " << deri << endl;
12    cout << " " << endl;
13    cout << endl;
14    //inisialisasi syah dengan integer dan deri sebagai pointer
15    int syah = *deri;
16    //memasukkan syah dengan *
17    cout << "Isi Variabel syah = " << syah << endl;
18    cout << " " << endl;
19    cout << endl;
20    cin.get();
21 }
```

The output window shows the following text:

```
D:\g++ point3.cpp -o point3.exe
D:\g++>
Alamat Address [deri] = 00000004
Isi Variabel syah = 44
```

Pseudocode

Kamus/Deklarasi Variabel

Agil, syah, *deri : int

Algoritma/Deskripsi

Agil=44

*deri = &agil

Print(deri)

Syah = *deri

Print(syah)

Algoritma/Bahasa Alami :

Agil <- 44

*deri = &agil

Mencetak Nilai deri

Syah <- *deri

Mencetak nilai syah

Selesai

Point

```

1 //include library
2 #include <iostream>
3 using namespace std;
4 int main()
5 {
6     int agil, *deri, *syah;
7     agil = 44;
8     //mendeklarasikan pointer dari tempat variabel address agil
9     deri = &agil;
10    //mendeklarasikan pointer agar tempat isi variabel dalam kode
11    syah = &deri;
12    //mendeklarikan isi variabel, address dan isi variabel dalam kode agil
13    cout << endl;
14    cout << "Isi Variabel [agil + deri + *syah] = " << agil + deri + *syah << endl;
15    cout << endl;
16    cout << "Isi Variabel [agil - deri - *syah] = " << agil - deri - *syah << endl;
17    cout << endl;
18    cout << "Isi Variabel [agil * deri * syah] = " << agil * deri * syah << endl;
19    cout << endl;
20    cout << "Isi Variabel [agil / deri / syah] = " << agil / deri / syah << endl;
21    cout << endl;
22    cout << "Isi Variabel [agil % deri % syah] = " << agil % deri % syah << endl;
23    cout << endl;
24    return 0;
25 }

```

Pseudocode

Kamus/Deklarasi Variabel

Agil, *deri, *syah= int

Algoritma/Deskripsi

Agil = 44

Deri = &agil

Syah = &deri

Print(agil,*deri,*syah)

Print(&agil,&deri,&syah)

Print(agil,deri,syah)

end

Algoritma/Bahasa Alami

Agil <- 44

Deri = &agil

Syah = &deri

Mencetak/Menampilkan nilai agil,*deri,*syah

Mencetak/Menampilkan nilai &agil,&deri,&syah

Mencetak/Menampilkan nilai agil,deri,syah

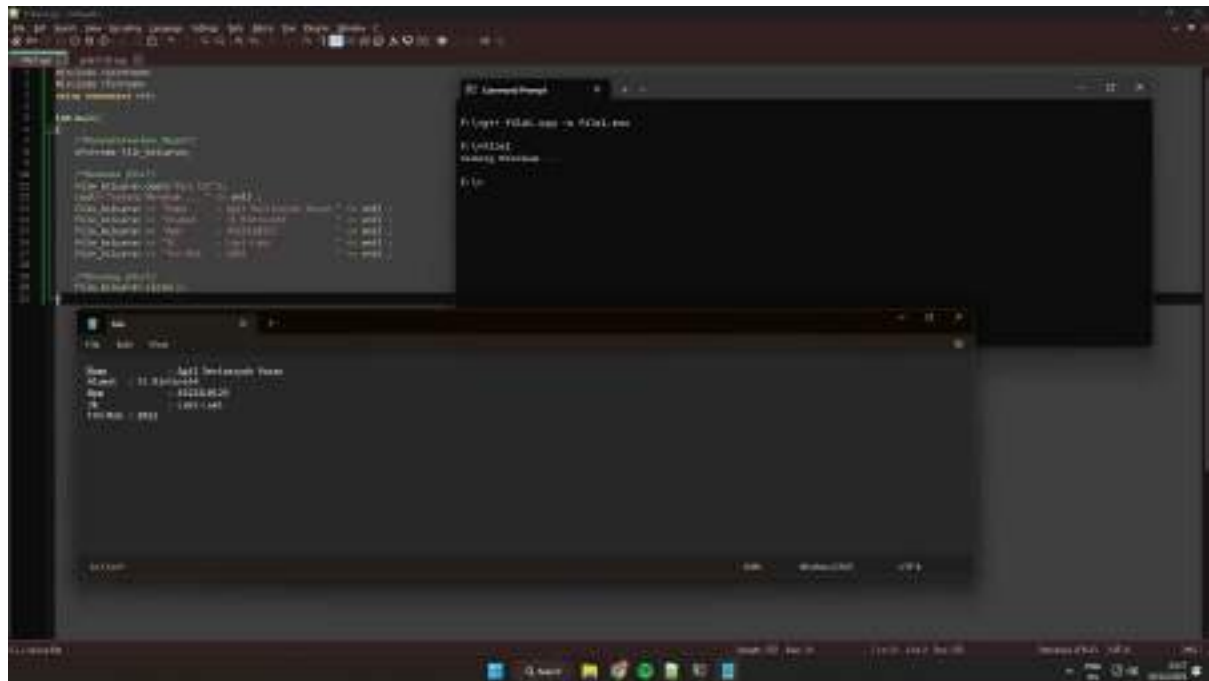
Selesai

1.3.13 File

File dalam bahasa pemrograman adalah kumpulan data yang disimpan dalam suatu lokasi memori atau penyimpanan. File dapat berupa file teks atau file biner, dan dapat dibuka, dibaca, ditulis, dan ditutup oleh program. Dalam bahasa pemrograman, terdapat beberapa fungsi dan kelas yang digunakan untuk melakukan operasi pada file, seperti ifstream, ofstream, dan fstream pada bahasa C++. Fungsi-fungsi ini memungkinkan program untuk membaca dan menulis data ke dalam file, membuka dan menutup file, serta melakukan operasi lainnya pada file.

Contoh :

File1



PSEUDOCODE :

Algoritma merekam File

KAMUS/DEKLARASI VARIABEL

-

ALGORITMA/DESKRIPSI ofstream file_keluaran

```
file_keluaran.open(BIODTKU.TXT)
file_keluaran("Nama : Agil Deriansyah Hasan")
file_keluaran("Alamat : Jl. Bintara14")
file_keluaran("Npm : 452221025")
file_keluaran("JK : Laki-Laki")
file_keluaran("Thn Msk : 2022")
file_keluaran.close()
```

Algoritma

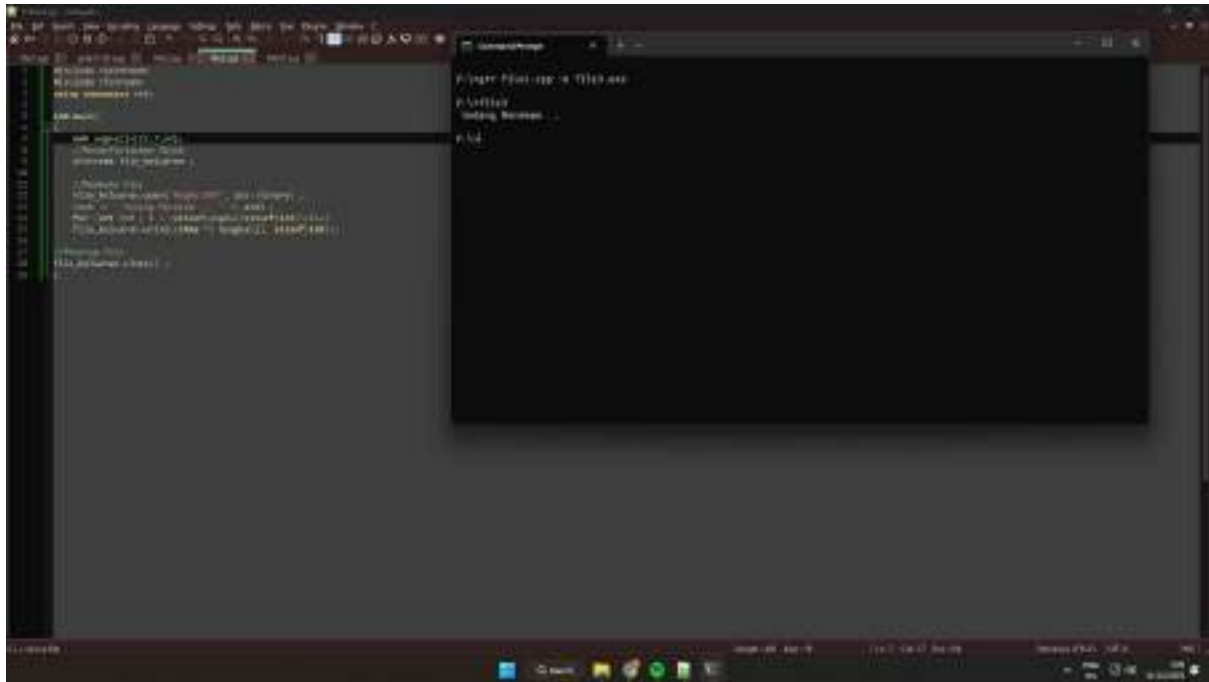
1. Mendefinisikan objek file_keluaran
2. Membuka file objek (file_keluaran.open("BIODTKU.TXT"))
3. Menulis/merekam ke file objek ("Nama : Agil Deriansyah Hasan")
4. Menulis/merekam ke file objek ("Alamat : Jl. Bintara14")
5. Menulis/merekam ke file objek ("Npm : 452221025 ")
6. Menulis/merekam ke file objek ("JK : Laki-Laki")
7. Menulis/merekam ke file objek (" Thn Msk: 2022")
8. Menutup file (file_keluaran.close())
9. Selesai

```
MAKS = 80
ifstream file_masukan(BIODTKU.TXT)
while (file_masukan)
file_masukan.getline(penyangga, MAKS)
print(penyangga)
endwhile file_masukan.close()
```

Algoritma

1. MAKS = 80 (Definisikan variabel untuk Σ karakter yang diinginkan)
2. Definisikan buffer/penyangganya
3. Memanggil file teks yang telah dibuat (ifstream file_masukan("BIODTKU.TXT"))
4. Jika (file_masukan) kerjakan baris 5 s.d. 6
5. Membaca sejumlah karakter sebanyak MAKS (file_masukan.getline(penyangga, MAKS))
6. Menampilkan/mencetak penyangga
7. Menutup file (file_masukan.close())
8. Selesai

File3



PSEUDOCODE :

Algoritma merekam File

KAMUS/DEKLARASI VARIABEL

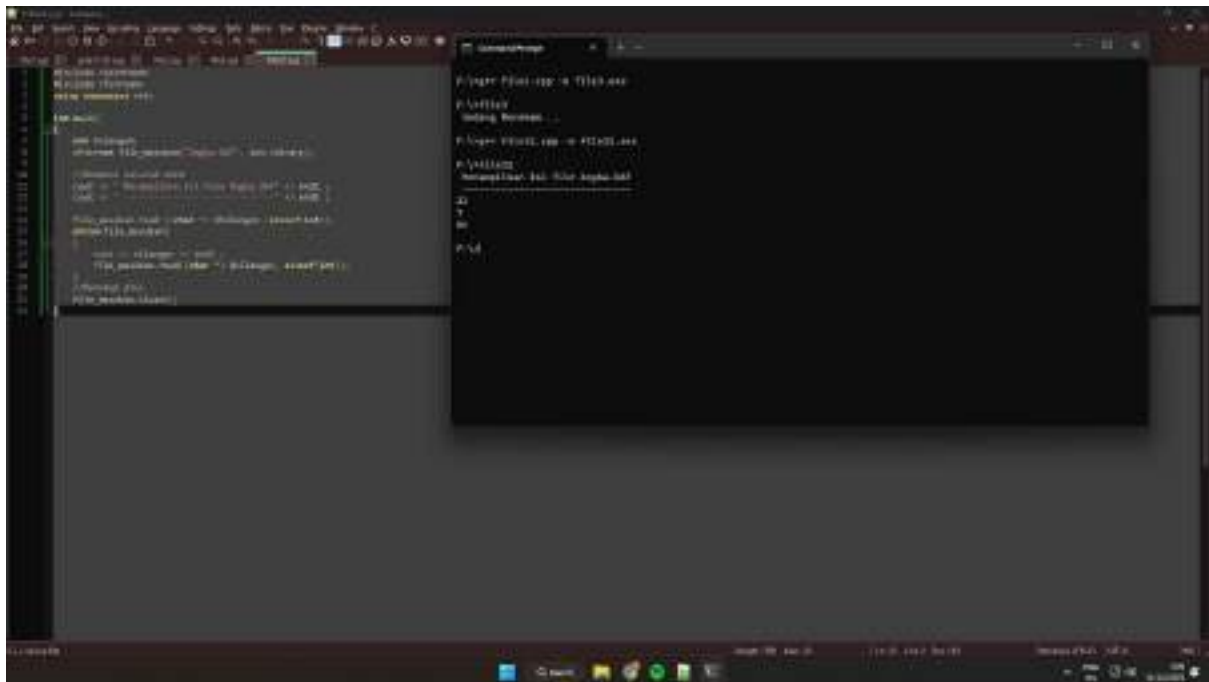
angka[]={22,7,66} : int

ALGORITMA/DESKRIPSI

```
ofstream file_keluaran  
for (int i=0 ; i < (sizeof(angka)/sizeof(int));i++)  
file_keluaran.write((char *) &angka[i], sizeof(int));  
file_keluaran.close()
```

Algoritma

1. Mendefinisikan objek file_keluaran
2. Membuka file objek (file_keluaran.open("Angka.TXT"))
3. selama (i=0) kerjakan baris 5 s.d 7
4. Membaca file_keluaran(file_keluaran.write
5. Menutup file (file_keluaran.close())
6. Selesai



PSEUDOCODE :

Algoritma membaca File Angka.TXT

KAMUS/DEKLARASI VARIABEL

bilangan= int

ALGORITMA/DESKRIPSI

```

ifstream file_masukan(Angka.TXT, ios::binary)
while (file_masukan)
file_masukan.read ((char *) &bilangan, sizeof(int));
while(file_masukan)
    file_masukan.read((char *) &bilangan,
        sizeof(int));
print bilangan
endwhile
file_masukan.close()

```

Algoritma

1. Membaca Seluruh Data
2. Memanggil file teks yang telah dibuat (ifstream file_masukan("Angka.TXT"))
3. Membaca file_masukan
4. Jika (file_masukan) kerjakan baris 5 s.d. 7
5. Menampilkan/mencetak bilangan
6. Membaca file_masukan
7. Menutup file (file_masukan.close())
8. Selesai

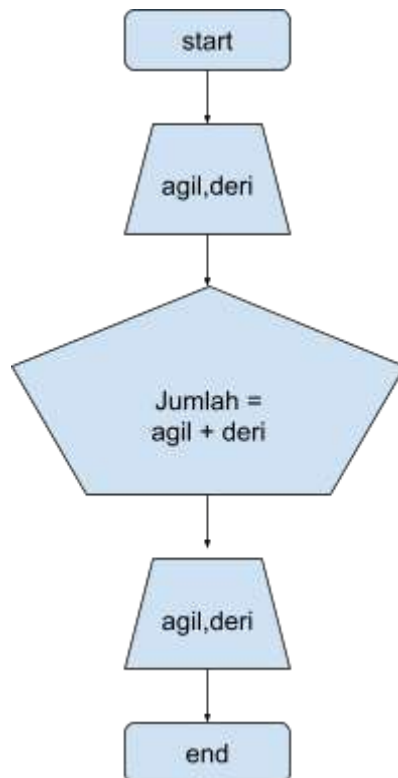
1.4 Tugas Praktikum

Praktikum 01

Prak1-07

```
1 // Agil dan Deri  
2 #include <iostream>  
3 using namespace std;  
4  
5 int main()  
6 {  
7     int agil, deri, jumlah;  
8     cout << "Masukkan isi variabel agil : ";  
9     cin >> agil;  
10    cout << "Masukkan isi variabel deri : ";  
11    cin >> deri;  
12  
13    jumlah = agil + deri;  
14    cout << "Jumlah dari agil + deri = " << jumlah << endl;  
15    return 0;  
16 }
```

```
Microsoft Windows [Versi 10.0.17134.110]  
(c) Microsoft Corporation. All rights reserved.  
C:\Users\agil>g++ 12345-07.cpp -o 12345-07.exe  
C:\Users\agil>. 12345-07.exe  
Masukkan isi variabel agil : 10  
Masukkan isi variabel deri : 7  
Jumlah dari agil + deri = 17
```



Algoritma :

- Memasukkan isi/nilai variabel agil
- Memasukkan isi/nilai variabel deri
 - Jumlah = agil + deri
- Menampilkan isi/nilai variabel agil + deri
 - Selesai

Pseudocode :

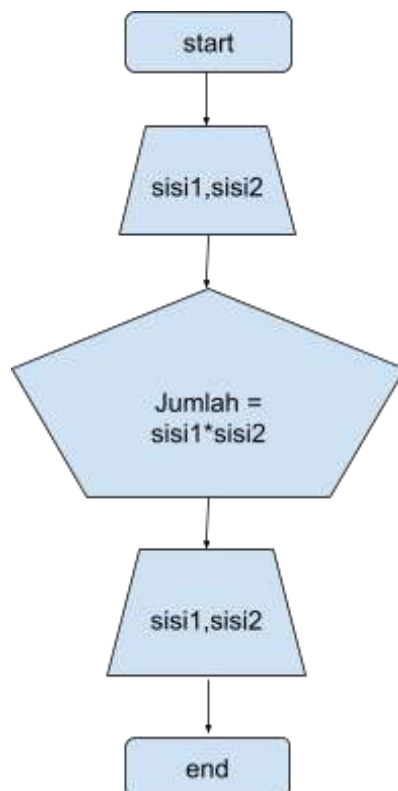
Kamus

int : agil, deri, jumlah

Deskripsi

input(agil, deri)
jumlah = agil + deri
print(agil, deri);

Prak1-21

[illegible]

Algoritma :

- Memasukkan isi/nilai variabel sisi1
- Memasukkan isi/nilai variabel sisi2
 - Jumlah = sisi1*sisi2
- Menampilkan isi/nilai variabel sisi1 * sisi2
 - Selesai

Pseudocode :

Kamus

```
int : sisi1,sisi2,jumlah
```

Deskripsi

```
input(sisi1, sisi2)
jumlah = sisi1 * sisi2
print(sisi1 , sisi2);
```


Praktikum 02

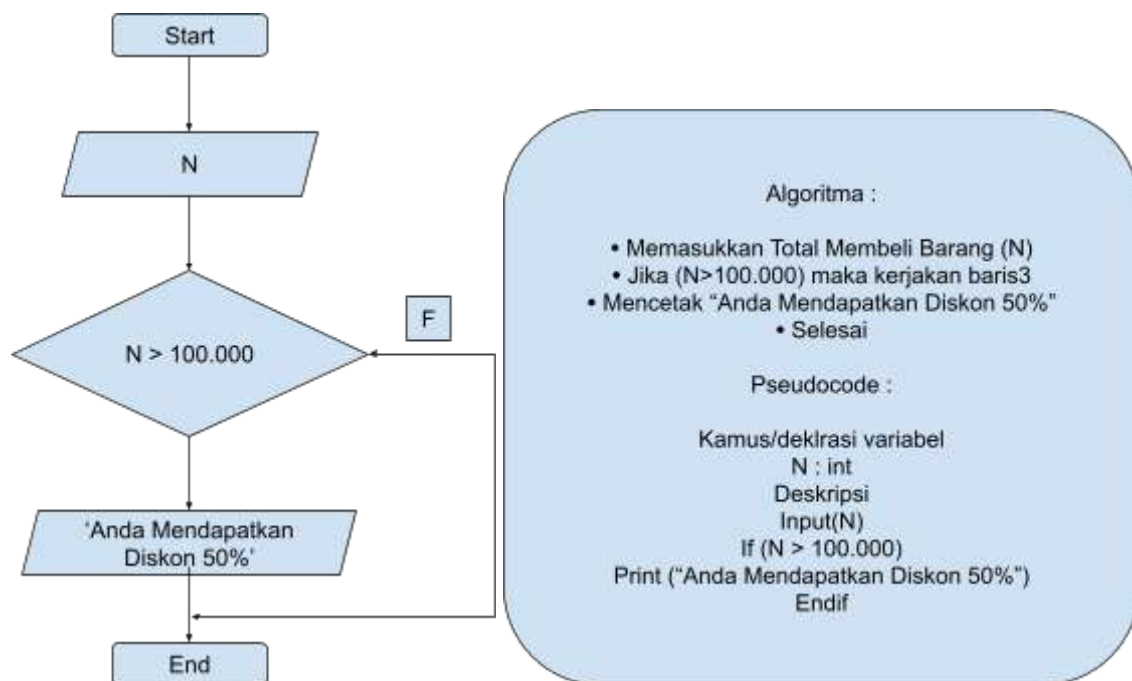
Prak2-15

```

1 // Praktikum 02
2 // Nama : ...
3 // NIM : ...
4
5 int main()
6 {
7     // Deklarasi variabel
8     int N;
9
10    // Input
11    cout << "Masukkan total barang (N) : "; cin >> N;
12    cout << endl;
13
14    // Logika
15    if (N > 100000)
16    {
17        cout << "Anda Mendapatkan Diskon 50%";
18    }
19
20    return 0;
21 }
    
```

```

P:\gpr> g++ 02.cpp -o 02.exe
P:\gpr> .\02.exe
Masukkan total barang (N) : 100000
Anda Mendapatkan Diskon 50%
P>
    
```



Prak2-45

```

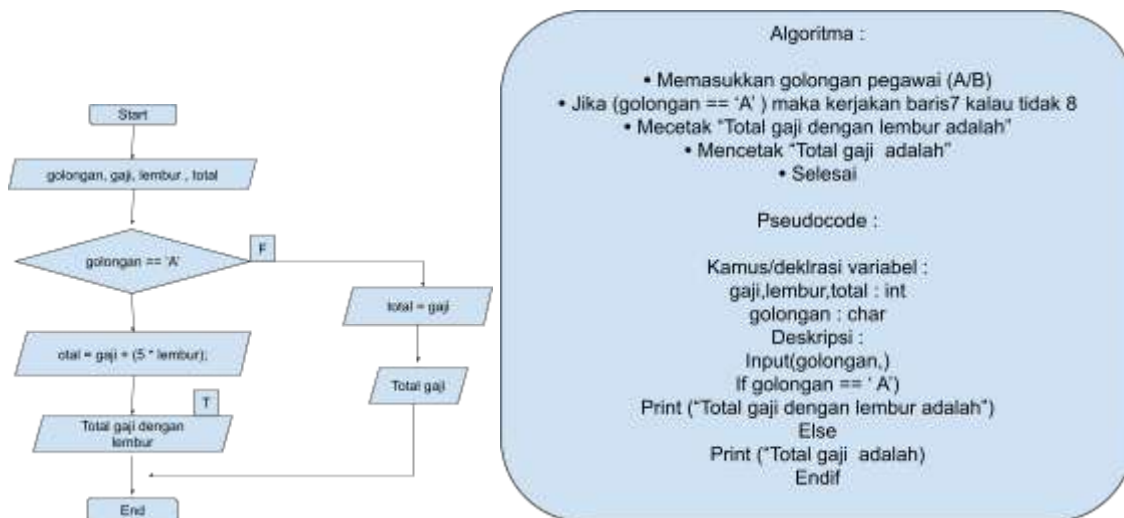
1 // Program 1: Menghitung total gaji dengan lembur
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     char golongan;
7     int gaji, lembur, total;
8
9     // Input data
10    cout << "Masukkan golongan pegawai (A/B) : ";
11    cin >> golongan;
12
13    if (golongan == 'A') {
14        // Jika golongan A, maka kerjakan baris 7
15        total = gaji + (5 * lembur);
16        cout << "Total gaji dan lembur adalah Rp. " << total << endl;
17    } else {
18        // Jika golongan B, maka kerjakan baris 8
19        total = gaji;
20        cout << "Total gaji dan lembur adalah Rp. " << total << endl;
21    }
22
23    return 0;
24 }

```

```

F:\Open Visual-Studio -> Praktek2-45
F:\Praktek2-45>
Masukkan golongan pegawai (A/B) : A
Total gaji dan lembur adalah Rp. 100000
F:\Praktek2-45>
Masukkan golongan pegawai (A/B) : B
Total gaji dan lembur adalah Rp. 50000
F:\Praktek2-45>

```



Praktikum 03

Prak3_15

The screenshot shows a C++ program in a code editor and its execution output in a terminal window. The program calculates a discount based on the total price of an order. If the total price is greater than 150,000, a 10% discount is applied. If it is greater than 100,000, a 5% discount is applied. Otherwise, no discount is applied. The program prints the total price after the discount.

```
1 // Program 15: Menghitung total harga pesanan setelah diskon
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     int agil;
7     int diskon;
8     int total;
9     int total_harga;
10
11     cout << "Masukkan total harga pesanan: ";
12     cin >> agil;
13
14     if (agil > 150000) {
15         diskon = 0.10 * agil;
16         total_harga = agil - diskon;
17     }
18     else if (agil > 100000) {
19         diskon = 0.05 * agil;
20         total_harga = agil - diskon;
21     }
22     else {
23         total_harga = agil;
24     }
25
26     cout << "Total harga setelah diskon: " << total_harga << endl;
27
28     return 0;
29 }
```

Output:

```
Program 15: Menghitung total harga pesanan setelah diskon
Masukkan total harga pesanan: 200000
Total harga setelah diskon: 180000
Program 15: Menghitung total harga pesanan setelah diskon
Masukkan total harga pesanan: 100000
Total harga setelah diskon: 95000
Program 15: Menghitung total harga pesanan setelah diskon
Masukkan total harga pesanan: 50000
Total harga setelah diskon: 50000
```

Pseudocode

kamus/deskripsi variabel

agil, diskon : Int

Deskripsi:

input (agil)

if (agil > 150000)

diskon = 0.10 * agil

Print ("Selamat Anda Mendapatkan souvenir !")

if (agil > 100000)

Print ("Total Harga setelah diskon")

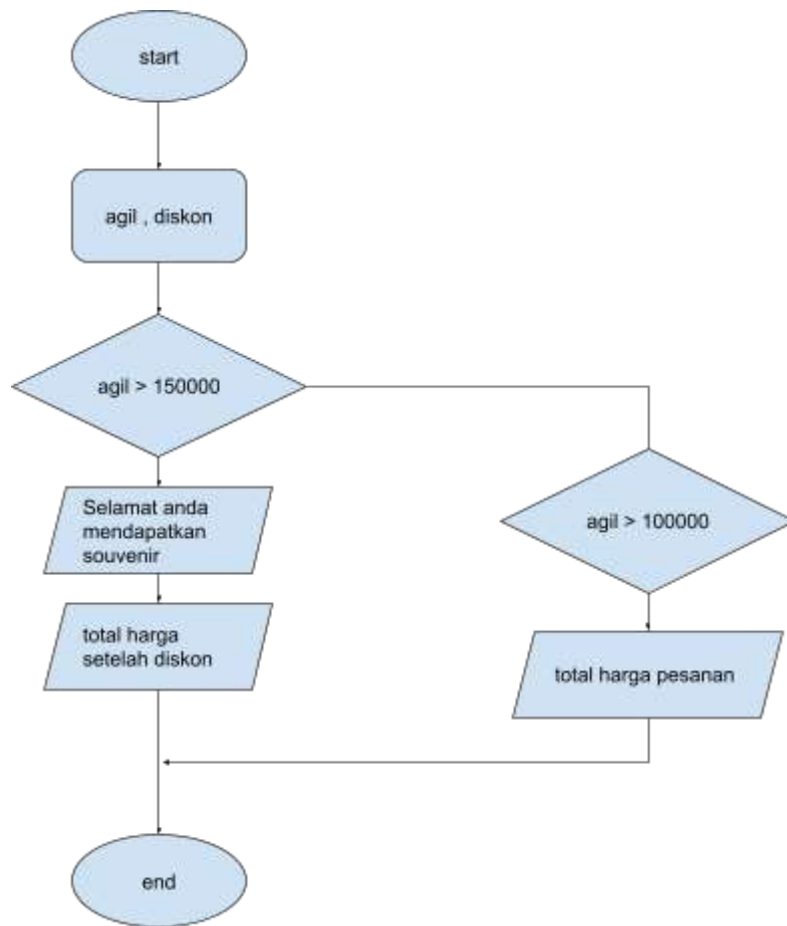
else

Print " total harga pesanan "

endif

Algoritma

- Memasukkan Total harga pesanan
- jika (agil > 150000) maka kerjakan baris 8 kalau tidak 9
- Menampilkan total harga pesanan = 200000
- Mencetak " Selamat anda mendapatkan souvenir"
- Mencetak " Total harga setelah diskon : 20000"
- Selesai



Prak3_45

```

// Program untuk menghitung total nilai RGB dalam Seri dan Paralel
// Input: pilihan (S/P), nilai r1, r2, r3

// Fungsi untuk menghitung total RGB dalam Seri
int hitungSeri(int r1, int r2, int r3) {
    return r1 + r2 + r3;
}

// Fungsi untuk menghitung total RGB dalam Paralel
int hitungParalel(int r1, int r2, int r3) {
    return r1 * r2 * r3;
}

int main() {
    char pilihan;
    int r1, r2, r3;

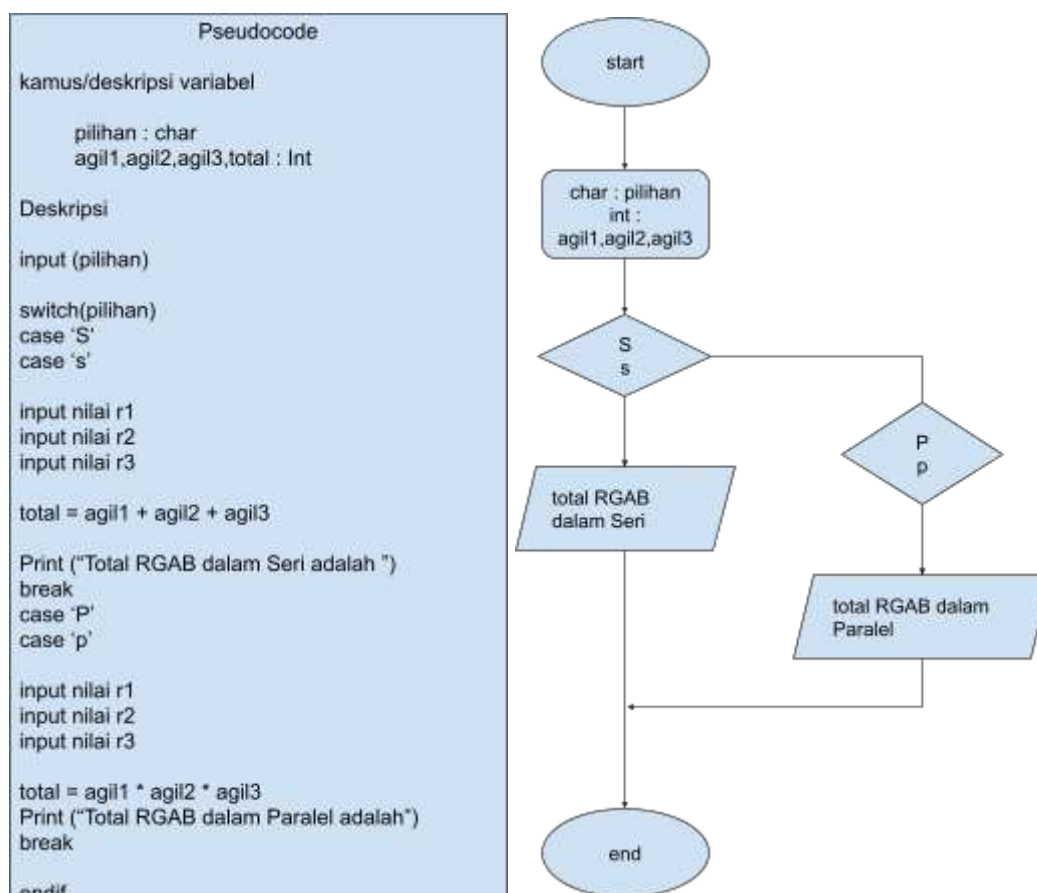
    // Input pilihan
    cout << "Pilih konfigurasi (S/P): ";
    cin >> pilihan;

    // Input nilai RGB
    cout << "Masukkan nilai r1: ";
    cin >> r1;
    cout << "Masukkan nilai r2: ";
    cin >> r2;
    cout << "Masukkan nilai r3: ";
    cin >> r3;

    // Hitung total RGB
    int total;
    if (pilihan == 'S' || pilihan == 's') {
        total = hitungSeri(r1, r2, r3);
        cout << "Total RGB dalam Seri adalah: " << total << endl;
    } else if (pilihan == 'P' || pilihan == 'p') {
        total = hitungParalel(r1, r2, r3);
        cout << "Total RGB dalam Paralel adalah: " << total << endl;
    }

    return 0;
}

```



Praktikum 04

Prak4-03

```

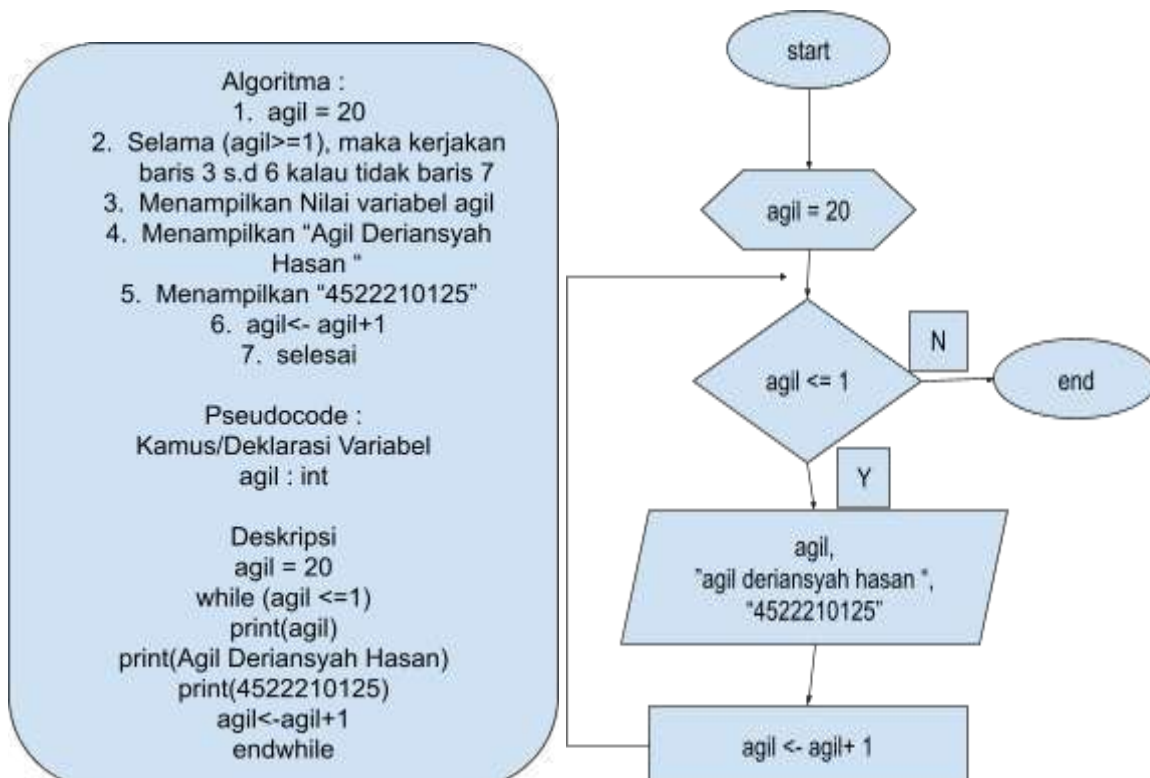
1 // Praktikum 04
2 #include <iostream>
3 using namespace std;
4
5 int agil = 20;
6
7 while (agil >= 1)
8 {
9     cout << "Agil Deriansyah Hasan " << agil << endl;
10    cout << "4522210125" << endl;
11    agil--;
12 }
13
14 return 0;
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

```

F:\praktikum\Prak4-03>g++ prak4-03.cpp
F:\praktikum\Prak4-03>.\\praktikum-03.exe
Agil Deriansyah Hasan 20
4522210125
Agil Deriansyah Hasan 19
4522210125
Agil Deriansyah Hasan 18
4522210125
Agil Deriansyah Hasan 17
4522210125
Agil Deriansyah Hasan 16
4522210125
Agil Deriansyah Hasan 15
4522210125
Agil Deriansyah Hasan 14
4522210125
Agil Deriansyah Hasan 13
4522210125
Agil Deriansyah Hasan 12
4522210125
Agil Deriansyah Hasan 11
4522210125
Agil Deriansyah Hasan 10
4522210125
Agil Deriansyah Hasan 9
4522210125
Agil Deriansyah Hasan 8
4522210125
Agil Deriansyah Hasan 7
4522210125
Agil Deriansyah Hasan 6
4522210125
Agil Deriansyah Hasan 5
4522210125
Agil Deriansyah Hasan 4
4522210125
Agil Deriansyah Hasan 3
4522210125
Agil Deriansyah Hasan 2
4522210125
Agil Deriansyah Hasan 1
4522210125

```



Prak4-16

```

1 // Program untuk mencari bilangan terbesar
2 #include <iostream>
3 using namespace std;
4
5 int main()
6 {
7     int agilBil;
8     int agilting;
9     int agil = 1;
10    while (agil <= agilBil)
11    {
12        cout << "Masukkan Bilangan ke- " << agil << endl;
13        int bilangan;
14        cin >> bilangan;
15        if (bilangan > agilting)
16        {
17            agilting = bilangan;
18        }
19        agil++;
20    }
21    cout << "Bilangan tertinggi adalah : " << agilting << endl;
22    return 0;
23 }

```

```

C:\Users\user> g++-12.exe -std=c++11 -c praktik4-16.cpp
C:\Users\user> g++-12.exe -std=c++11 -o praktik4-16.exe praktik4-16.o
C:\Users\user> .\praktik4-16.exe
Masukkan Bilangan ke- 1:
Masukkan Bilangan ke- 2: 12
Masukkan Bilangan ke- 3: 99
Masukkan Bilangan ke- 4: 87
Masukkan Bilangan ke- 5: 8
Bilangan tertinggi adalah: 99

```

- Algoritma :
1. agil = 1
 2. Menampilkan Nilai Variabel agilBil
 3. Menampilkan Masukkan Bilangan ke-
 4. agil==1
 5. selama (agil <= agilBil) kerjakan baris 10 s.d 12
 6. selesai

Pseudocode :

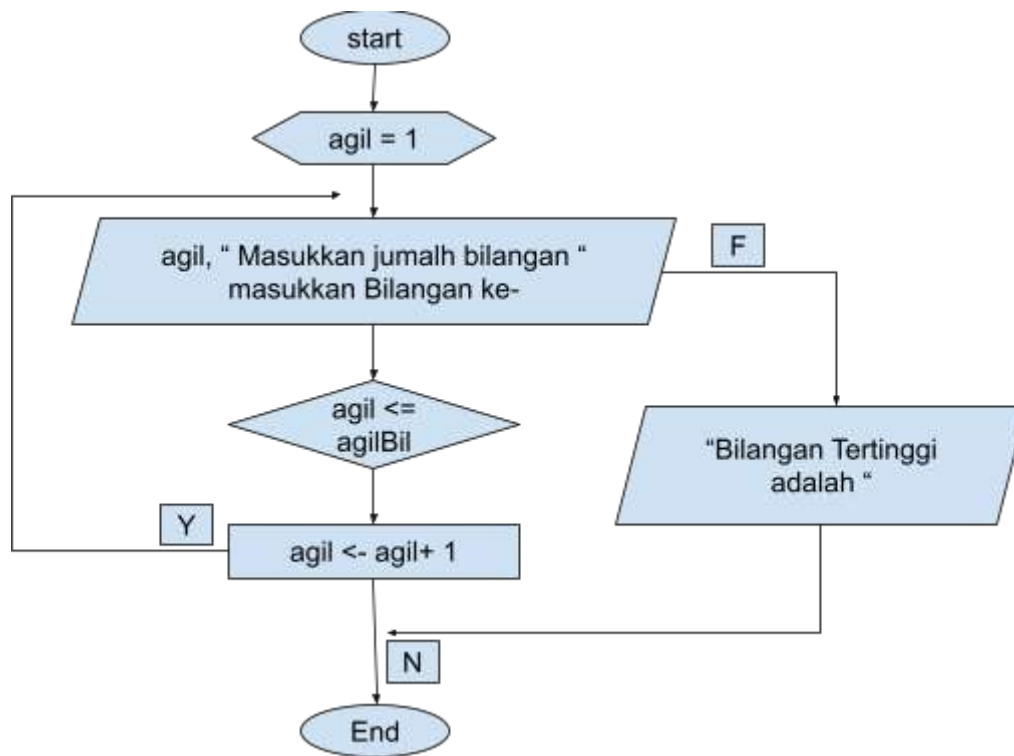
Kamus/Deklarasi Variabel
 agilbil,bilangan,agilting,agil : int

Deskripsi

```

agil = 1
Input agilBil
do
    if(agil==1)
        Print (" Masukkan bilangan ke- ")
        agilting = bilangan
    else
        if(bilangan>agilting)
            agilting=bilangan
            agil++
        agil <= agilBil
        Print ("Bilangan tertinggi adalah ")
    endwhile

```



Praktikum 05

Prak5-01

```
for1.cpp | nestedfor.cpp | prak5-01.cpp
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int agil;
6     cout << " Menampilkan Angka dari 10 s.d 1 " << endl;
7     cout << " " << endl;
8     cout << endl;
9     for (agil = 20; agil >= 1; agil--)
10         cout << agil << endl;
11     cin.get();
12 }
```

```
Command Prompt - prak5-01
D:\>g++ prak5-01.cpp -o prak5-01.exe
D:\>prak5-01
Menampilkan Angka dari 20 s.d 1
20
19
18
17
16
15
14
13
12
11
10
9
8
7
6
5
4
3
2
1
```

Pseudocode :

Kamus/Deklarasi Variabel

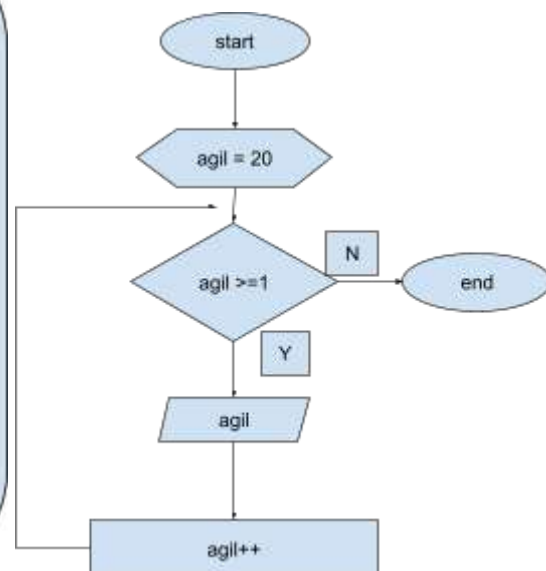
agil : int

Algoritma/Deskripsi

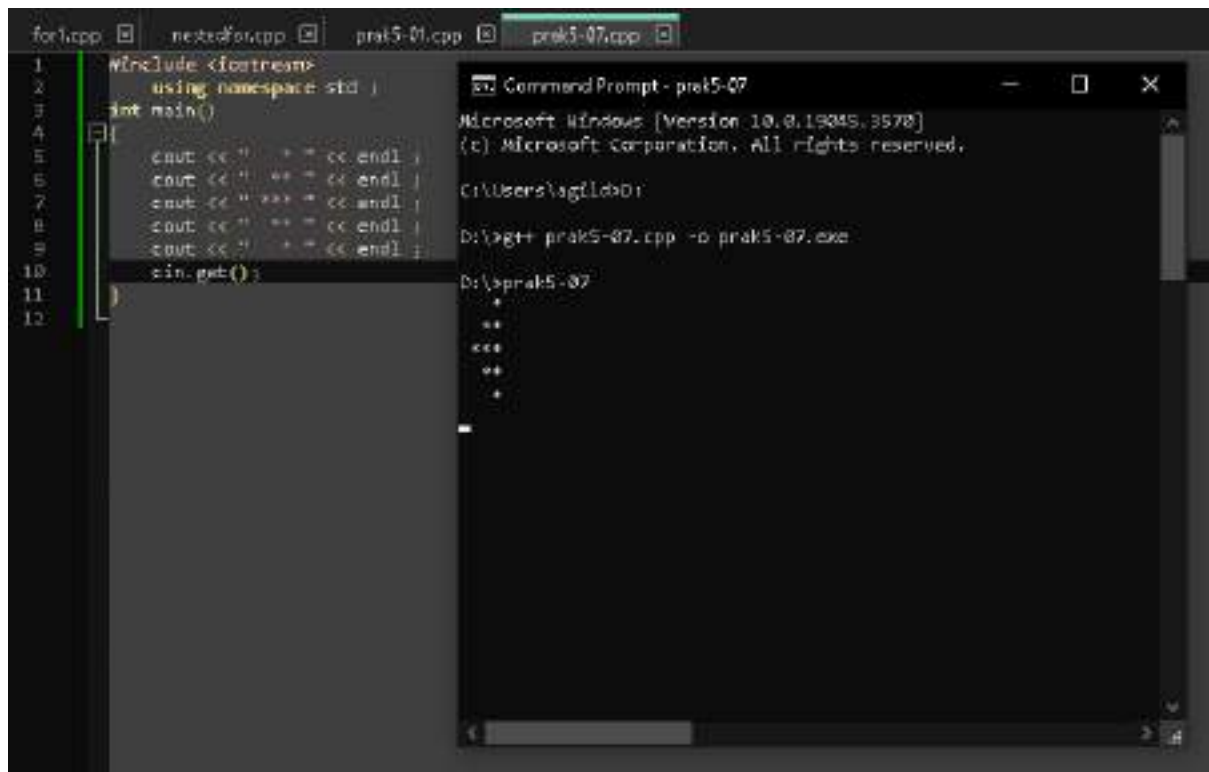
```
for(agil=20 ; agil >=1 ; agil++)
print (agil)
endfor
```

Algoritma :

1. agil =20
2. Selama(agil >=1), maka kerjakan baris 3 s.d 4 kalau tidak baris 5
3. Menampilkan Nilai variabel agil
4. agil++
5. Selesai



Prak5-07



```
for1.cpp | nested5.cpp | prak5-01.cpp | prak5-07.cpp
1  #include <iostream>
2  using namespace std;
3  int main()
4  {
5      cout << "  *  " << endl;
6      cout << " ** " << endl;
7      cout << " *** " << endl;
8      cout << " ** " << endl;
9      cout << "  *  " << endl;
10     cin.get();
11 }
12
```

Command Prompt - prak5-07

Microsoft Windows [Version 10.0.19045.3578]
(c) Microsoft Corporation. All rights reserved.

C:\Users\agild>D:

D:\>g++ prak5-07.cpp -o prak5-07.exe

D:\>prak5-07

```
  *
 **
 ***
 **
  *
```

Praktikum 06

Prak6-3

```

1 // Faktorial
2 #include <iostream>
3 using namespace std;
4
5 int faktorial(int n) {
6     if (n == 1) return 1;
7     return n * faktorial(n - 1);
8 }
9
10 int main() {
11     int n;
12     cout << "Masukkan bilangan: ";
13     cin >> n;
14     cout << "Faktorial dari " << n << " adalah: " << faktorial(n) << endl;
15     return 0;
16 }
    
```

```

$ g++ praktik6-3.cpp -std=c++11
$ ./praktik6-3
Masukkan bilangan: 5
Faktorial dari 5 adalah: 120
    
```

• Pseudocode

KAMUS/DEKLARASI VARIABEL
 agilk, agilkfakt_iter = 1 : long int

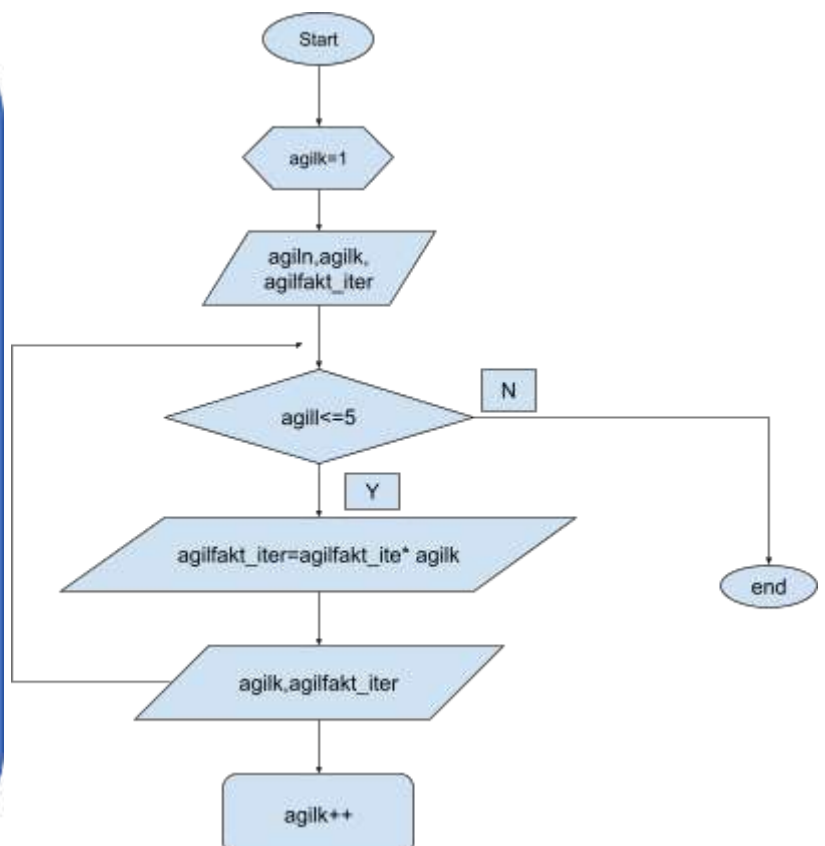
ALOGARITMA/DESKRIPSI

```

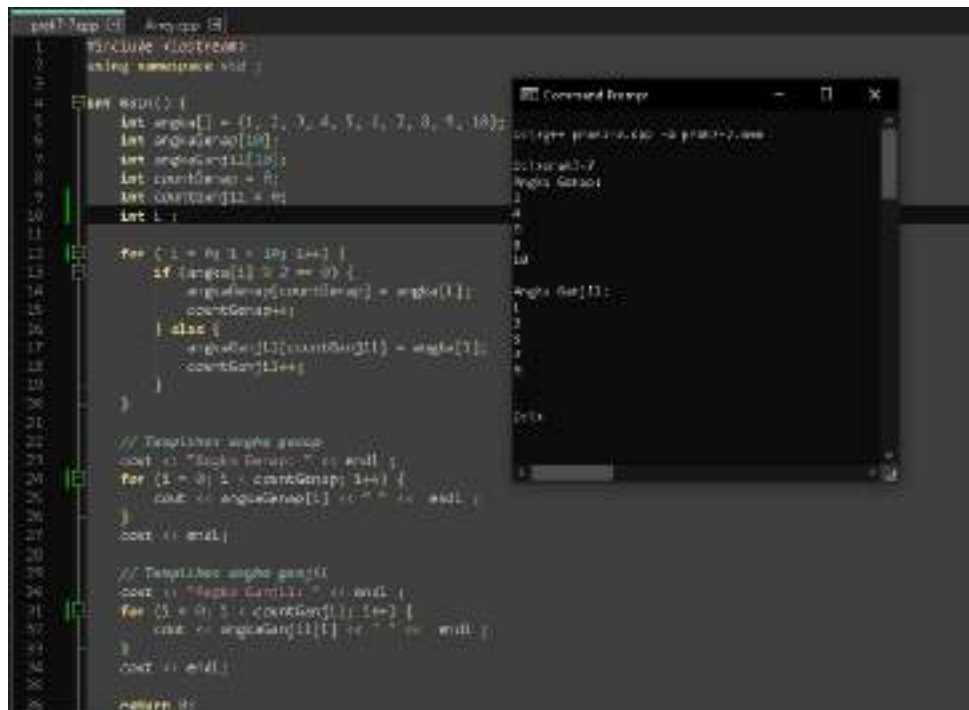
for ( agilk = 1; agilk <= 5; agilk++ )
    agilkfakt_iter = agilkfakt_iter * agilk
    print ( agilk, agilkfakt_iter )
endfor
    
```

• Algoritma/Bahasa Alami

- const agilkfakt_iter = 1
- const agilk = 1
- const agilk <= 5
- Selama (agilk <= 5), maka kerjakan baris ke 4 s.d 5 kalau tidak baris 6
- Agilkfakt_iter = agilkfakt_iter * agilk
- mencetak isi/nilai variabel agilk dan agilkfakt_iter
- selesai



Prak7-7



Pseudocode :

Kamus/Deklarasi Variabel

```
angka[], angkaGenap, angkaGanjil,  
countGenap, countGanjil, i : int
```

Algoritma/Deskripsi

```
angka[]={1,2,3,4,5,6,7,8,9,10}  
angkaGenap[10]  
angkaGanjil[10]
```

```
for(i = 0 ; i < 10 ; i++)
    if(angka[i] % 2 == 0)
        angkaGenap[countGenap] = angka[i]
        counGenap++
else
    angkaGanjil[countGanjil]=angka[i]
    countGanjil++
```

```
Print Angka Genap
for( i = 0 ; i < countGenap ; i++)
    angkaGenap[i]
end!
```

```
Print Angka Ganjil
  for(i = 0 ; i < countGanjil ; i++)
    angkaGanjil[i]
  endl
```

endif

Algoritma :

1. angka[] : {1,2,3,4,5,6,7,8,9,10}
2. angkaGenap [10]
3. angkaGanjil[10]
4. countGenap,coutGanjil : 0
5. Menampilkan Angka Genap
6. Selama (i < countGanjil) Kerjakan baris 16 s.d 18
7. Mencetak Isi variabel angkaGenap[i]
8. Menampilkan Angka Ganjil
9. Selama (i < countGanjil) kerjakan baris 13 s.d 15
10. Mencetak Isi Variabel angkaGanjil[i]
11. Selesai

Prak8-08

```

1 // Sum of Squares of Numbers
2 #include <iostream>
3 using namespace std;
4
5 int sum(int n)
6 {
7     if (n == 1)
8         return 1;
9     else
10        return n*n + sum(n-1);
11 }
12
13 int main()
14 {
15     int n;
16     cin >> n;
17     cout << sum(n) << endl;
18     return 0;
19 }

```

Output: 55

PSEUDOCODE

KAMUS/DEKLARASI VARIABEL

```
baris,kolom,i,j = int
```

```
matrix[baris][kolom] = int
```

jumlah = int

DESKRIPSI

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

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

```
print (i+1, j+1)
```

```
input matrix[i][j]
```

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

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

endl

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

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

```
jumlah += matrix[i][j];
```

```
print (j+1)
```

```
print (jumlah)
```

Algoritma :

1. $i \leftarrow 0$
2. $j \leftarrow 0$
3. Masukkan Nilai Variabel baris
4. Masukkan Nilai Variabel kolom
5. selama ($i < \text{baris}$) kerjakan baris 12 s.d 16
6. selama ($j < \text{kolom}$) kerjakan baris 12 s.d 16
7. Mencetak nilai variabel $i+1$
8. Mencetak nilai variabel $j+1$
9. Menampilkan matriks $[i][j]$
10. $i++$
11. $j++$
12. selama ($i < \text{baris}$) kerjakan baris 23 s.d 26
13. selama ($j < \text{kolom}$) kerjakan baris 23 s.d 26
14. Menampilkan Matriks $[i][j]$
15. $i++$
16. $j++$
17. selama ($i < \text{baris}$) kerjakan baris 29 s.d 33
18. selama ($j < \text{kolom}$) kerjakan baris 29 s.d 33
19. Menampilkan variabel Jumlah
20. $i++$
21. $j++$
22. Selesai


```

1 // Pascal's Triangle
2 #include <iostream>
3 using namespace std;
4
5 int main() {
6     int i, j, k;
7     int size;
8     cout << "Masukkan ukuran segitiga: ";
9     cin >> size;
10
11     // Validasi ukuran
12     if (size < 1) {
13         cout << "Ukuran harus lebih dari 0. Ulangi input." << endl;
14         return 1;
15     }
16
17     // Membuat segitiga
18     for (i = 0; i < size; i++) {
19         // Menghitung jumlah spasi
20         int spaces = size - i - 1;
21         for (k = 0; k < spaces; k++) {
22             cout << " ";
23         }
24
25         // Menghitung nilai segitiga
26         int val = 1;
27         for (j = 0; j < i + 1; j++) {
28             cout << val << " ";
29             val = val * (i - j) / (j + 1);
30         }
31         cout << endl;
32     }
33
34     return 0;
35 }

```

```

Pascal's Triangle
Masukkan ukuran segitiga: 5
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1

```

Algoritma

1. $i \leftarrow 0$
2. $j \leftarrow 0$
3. $Size = 5$
4. Selama ($i < size$) kerjakan baris 9 s.d 14
5. Selama ($< 2 * size - 1$) kerjakan baris 9 s.d 14
6. Mencetak nilai variabel $i+1$
7. Mencetak nilai variabel $j+1$
8. Menginput nilai variabel Segitiga[i][j]
9. $i++$
10. $j++$
11. Selama ($i < size$),
12. Selama ($j < 2 * size - 1$),
13. Mencetak nilai variabel Segitiga[i][j]
14. $i++$
15. $j++$
16. Selesai

PSEUDOCODE

KAMUS/DEKLARASI VARIABEL

size = const int

i, j = int

segitiga[size] = char

DESKRIPSI

size = 5

segitiga[size][2 * size - 1]

for (i = 0; i < size; i++)

for (j = 0; j < 2 * size - 1; j++)

if (j >= i && j < 2 * size - 1 - i)

print (i + 1, j + 1)

input (segitiga[i][j])

else

segitiga[i][j] = ' ';

for (i = 0; i < size; ++i)

for (j = 0; j < 2 * size - 1; ++j)

segitiga[i][j]

endfor

Prak9-28

[illegible]

```

11 int input()
12 {
13     int a[10], b[10];
14     cout << "Maximum Grade Matrix: A = ?" << endl;
15     do
16     {
17         cout << "Enter Marks : "; cin >> a[0];
18     }
19     while(a[0] >= 10);
20     do
21     {
22         cout << "Enter Marks : "; cin >> b[0];
23     }
24     while(b[0] >= 10);
25     cout << "Maximum Grade Matrix B : "; cin >> b[0];
26     do
27     {
28         cout << "Enter Marks : "; cin >> a[1];
29     }
30     while(a[1] >= 10);
31     do
32     {
33         cout << "Enter Marks : "; cin >> b[1];
34     }
35     while(b[1] >= 10);
36     cout << "Matrix A = ";
37     for(int i = 0; i < 10; i++)
38     {
39         cout << a[i] << " ";
40     }
41     cout << "Matrix B : ";
42     for(int i = 0; i < 10; i++)
43     {
44         cout << b[i] << " ";
45     }
46 }

```

[illegible]

```

94
95     int tampil(int hasil [10] [10])
96     {
97         int agil_c, agil_d;
98         cout << "Masih Querasi Matrics : " << endl;
99         for(agil_c=0; agil_c<agil_l; agil_c++)
100         {
101             for(agil_d=0; agil_d<agil_l; agil_d++)
102             {
103                 cout << hasil [agil_c][agil_d] << " ";
104             }
105             cout << endl;
106         }
107     }

```

```

D:\>prak9-28
Masukan Ordo Matriks A :
Jumlah Baris : 4
Jumlah Kolom : 2
Masukan Ordo Matriks B :
Jumlah Baris : 2
Jumlah Kolom : 3
Matriks A :
Masukan Isi Elemen [1][1] : 2
Masukan Isi Elemen [1][2] : 3
Masukan Isi Elemen [2][1] : 4
Masukan Isi Elemen [2][2] : 5
Masukan Isi Elemen [3][1] : 6
Masukan Isi Elemen [3][2] : 7
Masukan Isi Elemen [4][1] : 2
Masukan Isi Elemen [4][2] : 1
Matriks B :
Masukan Isi Elemen [1] [1] : 2
Masukan Isi Elemen [1] [2] : 3
Masukan Isi Elemen [2] [1] : 4
Masukan Isi Elemen [2] [2] : 5
Masukan Isi Elemen [3] [1] : 2
Masukan Isi Elemen [3] [2] : 3
Masukan Isi Elemen [4] [1] : 4
Masukan Isi Elemen [4] [2] : 5
Perintah ARRAY/LARIK Multi Dimensi
Menu Operasi Matriks
<=====>
1. Perkalian Matriks
2. Keluar
<=====>
Pilihan Anda :
1
Hasil Operasi Matriks :
16 21 0
28 37 0
40 53 0
8 11 0
Perintah ARRAY/LARIK Multi Dimensi
Menu Operasi Matriks
<=====>
1. Perkalian Matriks
2. Keluar
<=====>
Pilihan Anda :
2

```

Deklarasi Variabel Fungsi agil_kali

```
agil_kalip[10][10], agil_c, agil_d, agil_e : int
```

Deskripsi fungsi agil_kali

```
agil_kalip[10][10]={0}
```

```
if (agil_j == agil_k)
```

```
for (agil_c=0; agil_c < agil_i; agil_c++)
```

```
for(agil_d=0; agil_d < agil_l, agil_d++)
```

```
for(agil_e=0; agil_e < agil_c; agil_e++)
```

```
agil_kalip[agil_c][ agil_d] = agil_kalip [agil_c][agil_d] +  
agil_a [agil_c][ agil_e]* agil_b [agil_e][ agil_d]
```

```
endfor
```

```
agil_tampil(agil_kali)
```

```
endfor
```

```
endfor
```

```
else
```

```
print (" Ordo Matriks Tidak Sesuai")
```

```
endif
```

Deklarasi Variabel Fungsi agil_input

```
agil_i, agil_j, agil_k, agil_l, agil_c, agil_k = int
```

Deskripsi Fungsi dhandi input

```
Do
```

```
input(agil_i),
```

```
while (agil_i >= 11)
```

```
do
```

```
input (agil_j)
```

```
while(agil_j >= 11)
```

```
do
```

```
input (agil_k)
```

```
while (agil_k >= 11)
```

```
do
```

```
input(agil_l)
```

```
while(agil_l >= 1
```

```
1)
```

```
for (agil_c=0; agil_c< agil_i; agil_c++)
```

```
for (agil_d=0; agil_d< agil_j; agil_d++)
```

```
print (agil_c+1, agil_d+1)
```

```
input (agil_a[agil_c][ agil_d]
```

```
endfor
```

```
endfor
```

```
for (agil_c=0; agil_c< agil_k; agil_c++)
```

```
for (agil_d=0; agil_d< agil_l; agil_d++)
```

```
print (agil_c+1, agil_d+1)
```

```
input (agil_b[agil_c][ agil_d]
```

```
endfor
```

```
endfor
```

Deklarasi Variabel

```
agil_a[10][10], agil_b[10][10], agil_i,  
agil_j, agil_k, agil_l, pilih : int
```

Deksripsi/Algoritma

```
agil_input()
```

```
do
```

```
input(agil_pilih)
```

```
switch(agil_pilih)
```

```
case 1 :
```

```
agil_kali()
```

```
break;
```

```
default();
```

```
while (agil_pilih != 2)
```

Deklarasi Variabel Fungsi agil_tampil

```
hasil[10][10], agil_c, agil_d : int
```

Deksripsi Fungsi agil_tampil

```
for(agil_c=0; agil_c < agil_i; agil_c++)
```

```
for(agil_d=0; agil_d < agil_l; agil_d++)
```

```
print(hasil[agil_c][agil_d])
```

```
endfor
```

```
endfor
```

• ALOGARITMA

```

Membuat fungsi agil_input()
Selama (agil_i >= 11 ) maka kerjakan baris 3 kalau tidak
baris 4
Memasukan isi variable ( agil_i )
Selama ( agil_j >= 11 ) maka kerjakan baris 5 kalau tidak
baris 6
Memasukan isi variabel agil_j
Selama (agil_k >= 11 ) maka kerjakan baris 7 kalau tidak
baris 8
Menginput/memasukan agil_k
Selama ( agil_l >= 11 ) maka kerjakan baris 9 kalau tidak
baris 10
Memasukan isi variabel agil_l
selama (agil_c < agil_i ) maka kerjakan baris 11 s.d.16
selama ( agil_d < agil_c ) kerjakan baris 12 s.d. 15
Mencetak/menampilkan agil_c + 1
Mencetak/Menampilkan agil_d + 1
Memasukan isi variabel agil_a[agil_c][agil_d]
agil_d++
agil_c++
Selama ( agil_c < agil_k ) kerjakan baris 18 s.d. 23
Selama ( agil_d < agil_l ) kerjakan baris 19 s.d. 22
Mencetak/menampilkan agil_c + 1
Mencetak/menampilkan agil_d + 1

```

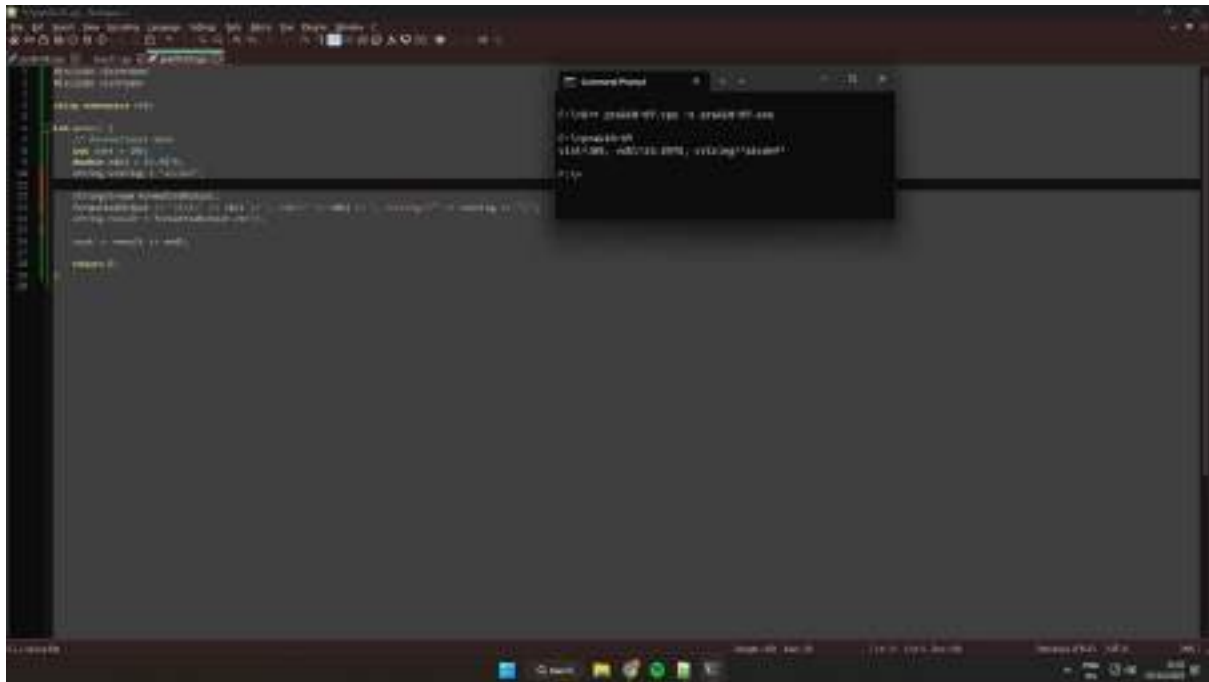
```

Memasukan/menginput agil_b[agil_c][ agil_d]
agil_d++
agil_c++
Membuat fungsi agil_kali()
Jika ( agil_j = agil_k ) kerjakan baris 28 s.d. 35 kalau tidak baris
36
Selama ( agil_c < agil_i ) maka kerjakan baris 29 s.d.35 kalau
tidak baris 36
Selama ( agil_d < agil_l ) maka kerjakan baris 30 s.d. 34 kalau
tidak baris 36
Selama ( agil_e < agil_j ) maka kerjakan baris 31 s.d. 33 kalau
tidak baris 36
agil_kalip[agil_c][ agil_d] = agil_kalip[agil_c][ agil_d] + agil
_a[agil_c][ agil_e] * agil_b[agil_c][ agil_d]
agil_tampil( agil_kalip )
agil_c++
agil_d++
agil_e++
Mencetak/menampilkan “ Ordo Matriks Tidak Sesuai “
Memanggil fungsi agil_tampil
Selama ( agil_c < agil_i ) maka kerjakan baris 39 s.d.42 kalau
tidak baris 43
Selama ( agil_d < agil_l ) maka kerjakan baris 40 s.d. 41 kalau
tidak baris 43
Mencetak /menampilkan (hasil[agil_c][ agil_d]
agil_d++
agil_c++
Selesai

```

Praktikum 10

Prak10-07



```
praktikum10.cpp
// Sum of squares of first n natural numbers
// Author: [Your Name]
// Date: [Date]

#include <iostream>
using namespace std;

int main() {
    int n;
    cout << "Enter a number: ";
    cin >> n;

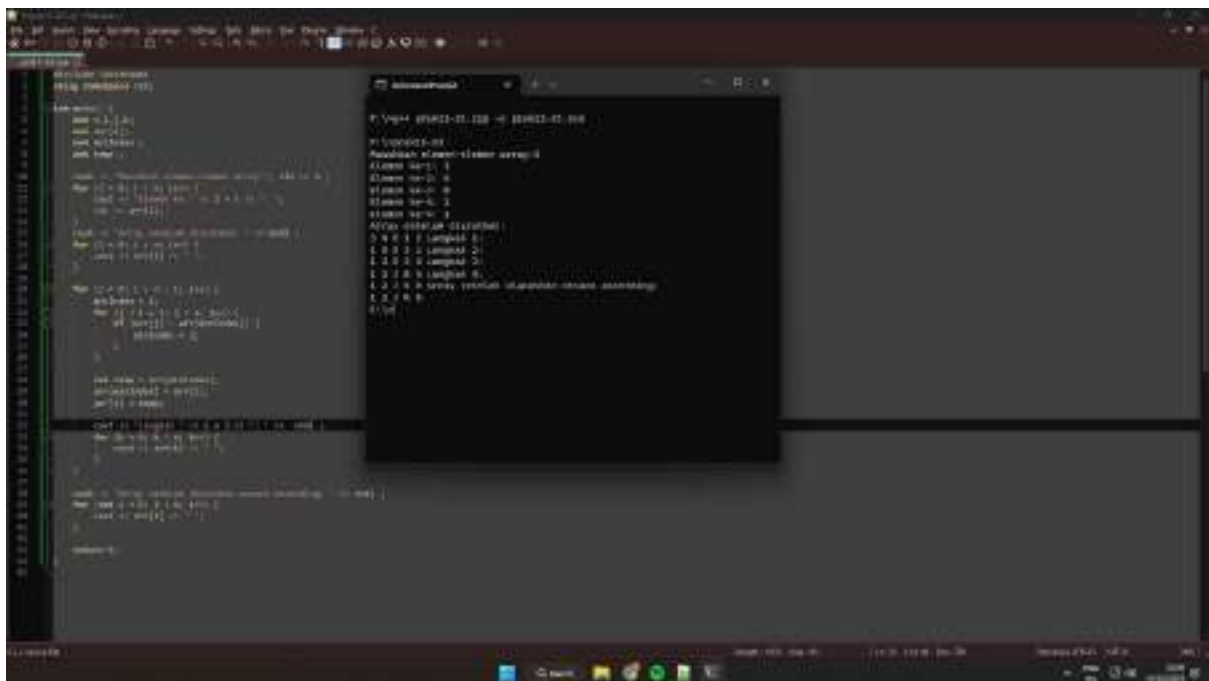
    int sum = 0;
    for (int i = 1; i <= n; i++) {
        sum += i * i;
    }

    cout << "Sum of squares of first " << n << " natural numbers is: " << sum << endl;
    return 0;
}
```

```
Terminal
C:\Users\praktikum10> g++ praktikum10.cpp -o praktikum10.exe
C:\Users\praktikum10> .\praktikum10.exe
Enter a number: 5
Sum of squares of first 5 natural numbers is: 55
```

Praktikum 11

Prak11-23



```
praktikum11.cpp
// Sum of first n natural numbers
// Author: [Your Name]
// Date: [Date]

#include <iostream>
using namespace std;

int main() {
    int n;
    cout << "Enter a number: ";
    cin >> n;

    int sum = 0;
    for (int i = 1; i <= n; i++) {
        sum += i;
    }

    cout << "Sum of first " << n << " natural numbers is: " << sum << endl;
    return 0;
}
```

```
Terminal
C:\Users\praktikum11> g++ praktikum11.cpp -o praktikum11.exe
C:\Users\praktikum11> .\praktikum11.exe
Enter a number: 5
Sum of first 5 natural numbers is: 15
```


PSEUDOCODE

Mengurutkan data dengan metode Insertion Sort

KAMUS/DEKLARASI VARIABEL

n, [arr_n], j, k, i, minIndex, temp : int

Algoritma/Deskripsi

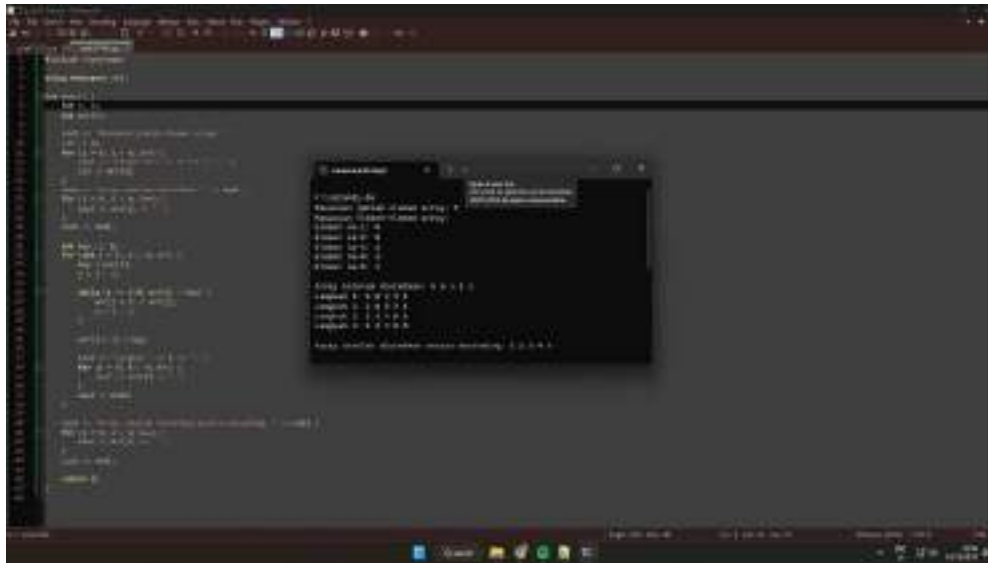
```
input n
for(i=0; i<n ; i++)
    input arr(i)
endfor

for (i =0; i<n;i++)
    arr(i) " "
for (i =0; i<n-1;i++)
    minIndex = i
    for( i =0; i<n;i++)
        if(arr(j) < arr(minIndex))
            minIndex = j
temp = arr(minIndex)
arr(minIndex) = arr(i)
arr(i) = temp
    for (k =0; k<n-1;i++)
    endfor
for (i =0; i<n;i++)
    arr(i) " "
```

Algoritma/Bahasa Alami:

- i=0
- menginput nilai variabel n
- Selama(i < n) kerjakan baris 11 s.d.13
- Menginput variabel arr[i]
- Selama(i < n) kerjakan baris 15 s.d.17
- Mencetak/menampilkan isi variable (" ")
- i++
- Selama(i < n- 1), maka kerjakan baris 21 s.d 22
- minIndex = i
- Selama(j < n), maka kerjakan baris 23 s.d 25
- minIndex = j
- Mencetak/menampilkan isi variabel i+1
- Selama(k < n), maka kerjakan baris 34 s.d 35
- Mencetak/menampilkan isi variable (" ")
- k++
- Selama(i <= n) kerjakan baris 40 s.d.41
- Mencetak/menampilkan isi variable (" ")
- i++
- Selesai

Prak11-54



PSEUDOCODE

Mengurutkan data dengan metode Insertion Sort

KAMUS/DEKLARASI VARIABEL

n, [arr_n], j, k, i, key: int

Algoritma/Deskripsi

input n

for(i=0; i<n; i++)

 input arr(i)

endfor

for (i =0; i< n;i++)

 arr(i) " "

for (i =0; i< n;i++)

 key = arr[i]

 j = i-1

 while(j>=0 && arr[j] > key)

 arr[j+1] = arr[j]

 j = j-1

arr[j+1]=key

for (k=0;k<n;k++)

 endfor

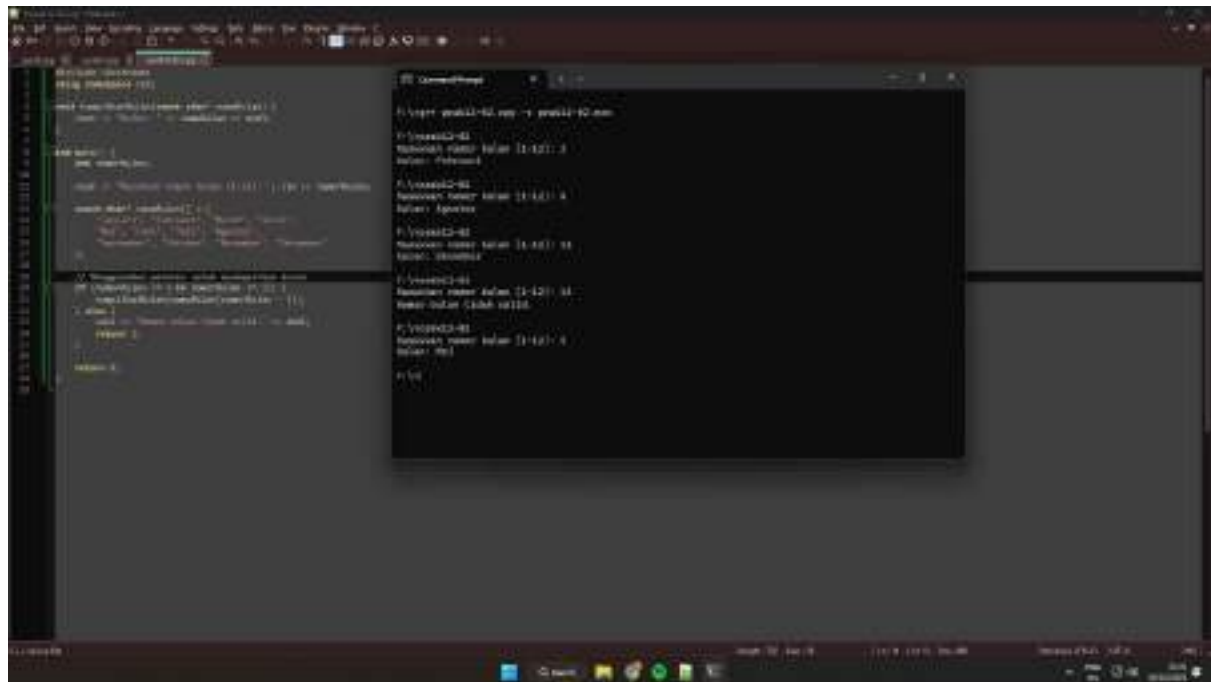
for (i =0; i< n;i++)

 arr(i) " "

endfor

Praktikum 12

Prak12-02



PSEUDOCODE

KAMUS/DEKLARASI VARIABEL

void : tampilkanBulan

(const char* namabulan)

nomor bulan : int

Deskripsi

input nomorBulan

const char* namaBulan[] =

"Januari", "Februari", "Maret", "April",

"Mei", "Juni", "Juli", "Agustus",

"September", "Oktober", "November", "Desember"

if (nomorBulan >= 1 && nomorBulan <= 12) {

tampilkanBulan(namaBulan[nomorBulan - 1]);

} else {

cout << "Nomor bulan tidak valid." << endl;

return 1;

}

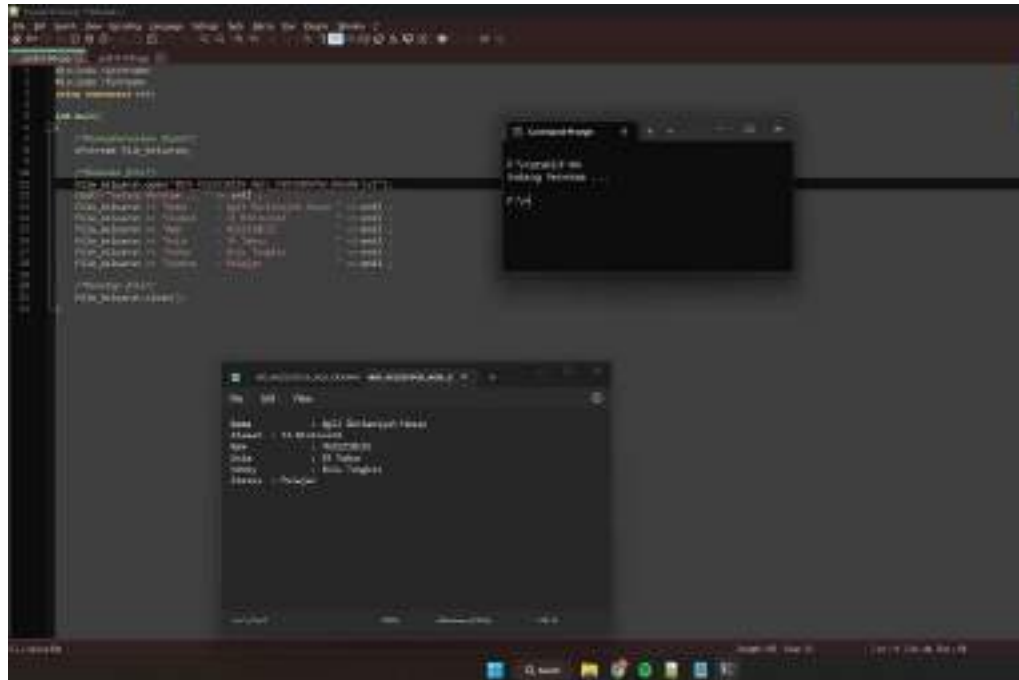
return 0

Algoritma/Bahasa Alami

- Memasukkan nilai variabel nomorBulan
- namaBulan[] = ("Januari", "Februari", "Maret", "April", "Mei", "Juni", "Juli", "Agustus", "September", "Oktober", "November", "Desember")
- Jika (nomorBulan >= 1 && nomorBulan <= 12) maka kerjakan baris 11
- Mencetak namaBulan
- Mencetak Nomor bulan tidak valid.
- selesai

Praktikum 13

Prak13-04



PSEUDOCODE :
Algoritma merekam File

KAMUS/DEKLARASI VARIABEL

-

ALGORITMA/DESKRIPSI
ofstream file_keluaran

```
file_keluaran.open("BIO_4522210125_AGIL_DERIANSYAH_HASAN.txt")
file_keluaran ( "Nama : Agil Deriansyah Hasan " )
file_keluaran ( "Alamat : Jl. Bintara14 " )
file_keluaran ( "Npm : 4522210125 " )
file_keluaran ( "Usia : 19 Tahun " )
file_keluaran ( "Hobby : Bulu Tangkis " )
file_keluaran ( "Status : Pelajar " )
file_keluaran.close()
```

Algoritma

1. Mendefinisikan objek file_keluaran
2. Membuka file objek
(file_keluaran.open("BIO_4522210125_AGIL_DERIANSYAH_HASAN.txt"))
3. Menulis/merekam ke file objek ("Nama : Agil Deriansyah Hasan")
4. Menulis/merekam ke file objek ("Alamat : Jl. Bintara14")
5. Menulis/merekam ke file objek ("Npm : 452221025 ")
6. Menulis/merekam ke file objek ("Usia : 19 Tahun")
7. Menulis/merekam ke file objek (" Hobby : Bulu Tangkis")
8. Menulis/merekam ke file objek ("Status : Pelajar")
9. Menutup file (file_keluaran.close())
10. Selesai

BAB II

Tugas Besar

2.1 Pembahasan Tugas Besar

2.1.1 Judul Dan Materi

Judul : Si Hotel Dengan Inputan

Materi:

1. Selection/Seleksi
2. Repetition/Perulangan
3. Array
4. Function
5. String/File
6. Selection Sort Descending

2.1.2 Source Code Program

```
/*
    Nama      : Agil Deriansyah Hasan
    NPM       : 4522210125
    Mata Kuliah : Prak. Algoritma dan Pemrograman
    Kelas     : B

Si Hotel
Menggunakan Perintah Inputan

Materi :
-Selections/Seleksi
-Repetation/Pengulangan
-Array
-Function
-String/File
-Selection Sort - Descending */

#include <iostream> //Akses untuk Fungsi-Fungsi input/output
#include <string.h> //Akses Untuk Fungsi Manipulasi String
using namespace std; //Untuk Mensederhanakan Penulisan

//variabel Global
//Array
char Nama[30], No_Telp[15], Email[30], Jumlah_Tamu[2],
Tipe[10], Tipe_Kasur[20], Id_Kamar[5], pembayaran[20], menginap[20], hargakmr[10],
harga[30], menu[20] ;
```

```

void Menu(); // Menu Utama Program
//Function : Menampilkan displayMenu Utama
void displayMenu() {
    cout << endl;
    cout <<
    "=====
===== " << endl;
    cout << "|                Selamat Datang Di Hotel Deriansyah                |" << endl;
    cout << "|                Jln.Bintara14, No 25 Telp 087833160661                |" <<
endl;
    cout << "|                Jl. Kp.Bojong, Kec. Bekasi Bar., Kota Bekasi, Jawa Barat, 17134
|" << endl;
    cout <<
    "=====
===== " << endl;
    Menu(); //Memanggil Fungsi Menu untuk kembali Ke Menu Utama
}

//Function : Memamnggil Fungsi dari Hasil Pemesanan Kamar
void Hasil_Pilih(){
    char NamaK[30], TipeK[20] ; //Variabel Lokal untuk menyimpan duplikat nama
dan tipe
    cout << endl;
    cout << "=====
<< endl;
    cout << "|                Data Pemesanan Kamar                |" << endl;
    cout << "=====
<< endl;
    //String : Menduplikat nama ke NamaK, lalu mengubah NamaK menjadi huruf
besar semua
    strcpy(NamaK,Nama);
    //String : Menduplikat nama ke TipeK, lalu mengubah TipeK menjadi huruf besar
semua
    strcpy(TipeK,Tipe);
    strupr(TipeK);
    strupr(NamaK);
    cout << " Nama          : " << NamaK << endl;
    cout << " No Telp       : " << No_Telp << endl;
    cout << " Email        : " << Email << endl;
    cout << " Jumlah Tamu   : " << Jumlah_Tamu << endl;
    cout << " Tipe Kamar    : " << TipeK << endl;
    cout << " ID Kamar      : " << Id_Kamar << endl;
    cout << " Tipe Kasur    : " << Tipe_Kasur << endl;
    cout << " Menginap     : " << menginap << endl;
    cout << " Total Harga   : " << hargakmr << endl;
    cout << endl;
    Menu();}

```



```

//Fungcion : Memanggil Fungsi Untuk Melihat dan Memilih Tipe Kamar
void Pemilihan(){
    cout << "-----" << endl;
    cout << "| No | Tipe Kamar || Harga |" << endl;
    cout << "-----" << endl;
    cout << "| 1 | Anggrek || Rp. 500.000/Mlm |" << endl;
    cout << "| 2 | Mawar || Rp. 300.000/Mlm |" << endl;
    cout << "| 3 | Melati || Rp. 400.000/Mlm |" << endl;
    cout << "| 4 | Matahari || Rp. 200.000/Mlm |" << endl;
    cout << "| 5 | Lavender || Rp. 150.000/Mlm |" << endl;
    cout << "-----" << endl;
    cout << endl ;
    cout << " Tipe Kamar : "; fflush(stdin); cin.getline(Tipe, sizeof(Tipe));
    cout << " Tipe Kasur : "; fflush(stdin); cin.getline(Tipe_Kasur,
sizeof(Tipe_Kasur));
    cout << " ID Kamar : "; fflush(stdin); cin.getline(Id_Kamar,
sizeof(Id_Kamar));
    cout << " Menginap Berapa Hari : "; fflush(stdin); cin.getline (menginap,
sizeof(menginap));
    cout << " Total Harga Kamar : "; fflush(stdin); cin.getline (hargakmr,
sizeof(hargakmr));
    cout << endl;
    Hasil_Pilih(); }

//Fungcion : Memanggil Fungsi Untuk Mengisi Form Biodata
void Form_Kamar(){
    cout << "=====
<< endl;
    cout << "| Pengisian Form Biodata Diri |" << endl;
    cout << "=====
<< endl;
    cout << " Nama : "; fflush(stdin); cin.getline(Nama, sizeof(Nama));
    cout << " No Telp : "; fflush(stdin); cin.getline(No_Telp, sizeof(No_Telp));
    cout << " Email : "; fflush(stdin); cin.getline(Email, sizeof(Email));
    cout << " Jumlah Tamu : "; fflush(stdin); cin.getline(Jumlah_Tamu,
sizeof(Jumlah_Tamu));
    cout << "=====
<< endl;
    cout << endl;
    Pemilihan();}

//Function : Memanggil Fungsi dari Hasil Pemesanan Makanan
void Hasil_Makan(){
    char NamaK[30], menuK[20];
    cout << endl ;
    cout << "=====
<< endl;
    cout << "| Data Pemesanan Makanan |" << endl;

```

```

        cout << "===== "
<< endl;
        //String : Menduplikat nama ke NamaK, lalu mengubah NamaK menjadi huruf
        besar semua
        strcpy(NamaK,Nama);
        //String : Menduplikat menu ke MenuK, lalu mengubah MenuK menjadi huruf
        besar semua
        strcpy(menuK,menu);
        strupr(menuK);
        strupr(NamaK);
        cout << " Nama      : " << NamaK << endl;
        cout << " No Telp   : " << No_Telp << endl;
        cout << " Makanan   : " << menuK << endl;
        cout << " Tipe Kamar : " << Tipe << endl;
        cout << " ID Kamar   : " << Id_Kamar << endl;
        cout << endl;
        Menu();}

//Funcion : Memanggil Fungsi Untuk Mengisi Form Memesan Makanan
void pesan_makanan(){
    cout << "===== "
<< endl;
    cout << " | Silahkan Mengisi Form Untuk Memesan Makanan | " << endl;
    cout << "===== "
<< endl;
    cout << " Nama      : "; fflush(stdin); cin.getline(Nama, sizeof(Nama));
    cout << " No Telp   : "; fflush(stdin); cin.getline(No_Telp, sizeof(No_Telp));
    cout << " Makanan   : "; fflush(stdin); cin.getline(menu, sizeof(menu));
    cout << " Tipe Kamar : "; fflush(stdin); cin.getline(Tipe, sizeof(Tipe));
    cout << " ID Kamar   : "; fflush(stdin); cin.getline(Id_Kamar, sizeof(Id_Kamar));
    Hasil_Makan();}

//Fungcion : Memanggil Fungsi Untuk Melihat dan Memilih Makanan dari Menu
void Makanan(){
    int jlmdt = 5 ;
    int a,b,c,p;
    int deri[jlmdt] = {20000, 25000, 30000, 50000, 40000}; //Array
    cout << endl;
    cout << "===== " <<
endl;
    cout << "          Menu Makanan          " << endl;
    cout << "===== " <<
endl;
    cout << " | No | Makanan || Harga | " << endl;
    cout << "===== " <<
endl;
    cout << " | 1 | Nasi Goreng || Rp. 20.000 | " << endl;
    cout << " | 2 | Satay || Rp. 25.000 | " << endl;
    cout << " | 3 | Soto Betawi || Rp. 30.000 | " << endl;

```

```

cout << "| 4 | Sop Buntut || Rp. 50.000 |" << endl;
cout << "| 5 | Ayam Bakar Madu || Rp. 40.000 |" << endl;
cout << "===== " <<
endl;
cout << endl;
cout << "===== "
<< endl;
cout << "| Sebelum Pengurutan Harga Makanan |" << endl;
cout << "===== "
<< endl;
//Repetition (for) : Mencetak nilai akhir sebelum dilakukan pengurutan
for (a = 0; a <= jlmdt-1; a++)
    cout << deri[a] << " ";
    cout << endl;
for(b = 1; b <= jlmdt-1; b++) {
    a=b ;
    c=deri[a];
    while (a >= 0 && deri [a-1] < c) {
        deri[a] = deri [a-1];
        a--; }
    deri[a] = c; }
    cout << "
===== " << endl;
    cout << "| Pengurutan Descending Harga Makanan |" << endl;
    cout << "
===== " << endl;
    //Repetition (for) : Mencetak nilai akhir setelah dilakukan pengurutan
    for(a = 0; a <= jlmdt-1; a++) //Selection Sort Descending
        cout << deri[a] << " ";
        cout << endl ;
    cout << endl;
    pesan_makanan(); }

//Function : Menu utama
void Menu() {
    int pilih;
    cout << "===== "
<< endl;
    cout << "| Menu Pilihan |" << endl;
    cout << "===== "
<< endl;
    cout << "| 1. Pesan Kamar |" << endl;
    cout << "| 2. Pesan Makanan Hotel |" << endl;
    cout << "| 3. Keluar Dari Hotel |" << endl;
    cout << "===== "
<< endl;
    cout << " Masukkan angka yang dipilih [1/2/3] : "; cin >> pilih;
    cout << endl;

```

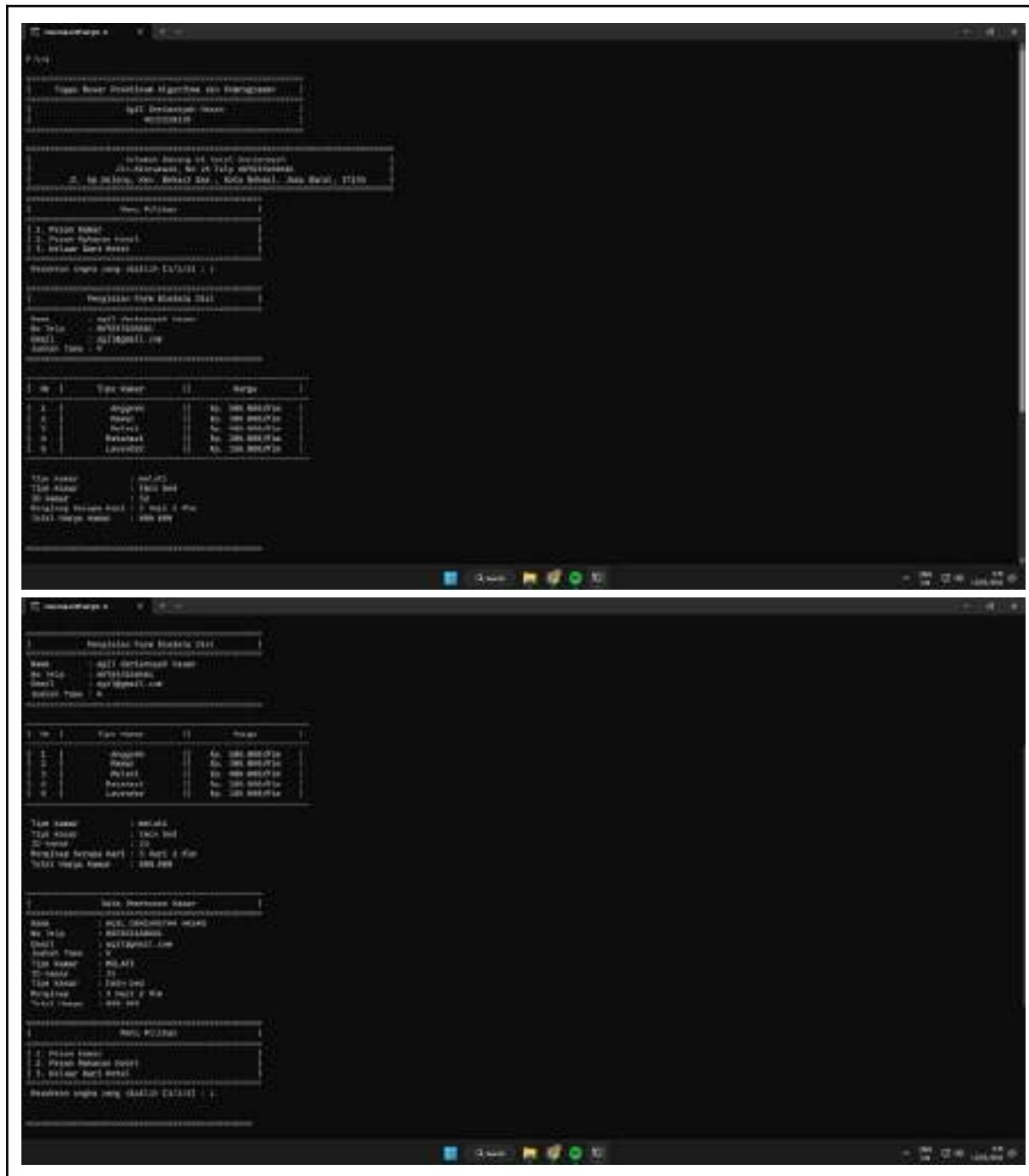
```

//Selection (switch) : Memilih angka untuk proses function selanjutnya
switch(pilih){
    case 1: Form_Kamar(); break;
    case 2: Makanan(); break;
    default: cout << "        Keluar Dari Hotel        " << endl ;
            cout << " Terima Kasih Telah Menginap Di Hotel Deriansyah" <<
endl;
    cout << endl; } }

int main() {
    cout << endl;
    cout <<
    "===== " <<
endl;
    cout << "|    Tugas Besar Praktikum Algoritma dan Pemrograman    |" << endl;
    cout <<
    "===== " <<
endl;
    cout << "|                Agil Deriansyah Hasan                |" << endl;
    cout << "|                4522210125                |" << endl;
    cout <<
    "===== " <<
endl;
    displayMenu();//Memanggil Fungsi displayMenu untuk menuju ke proses
selanjutnya
    cin.get();}

```

2.1.3 Capture Program



2.1.4 Pseudocode

ALGORITMA/DESKRIPSI FUNCTION displayMenu

```
print(" Selamat Datang Di Hotel Deriansyah")
```

```
print("Jln.Bintara14, No 25 Telp 087833160661")
```

```
print("Jl. Kp.Bojong, Kec. Bekasi Bar., Kota Bekasi, Jawa Barat, 17134")
```

KAMUS/DEKLARASI VARIABEL FUNCTION Form_Kamar

Nama[30], No_Telp[15], Email[30], Jumlah_Tamu[2] : char

ALGORITMA/DESKRIPSI FUNCTION Form_Kamar

```
print("Pengisian Form Biodata Diri")
```

```
print("Nama :")
```

```
input>Nama)
```

```
print("No Telp :")
```

```
input>No_Telp)
```

```
print("Email : ")
```

```
input>Email)
```

```
print("Jumlah Tamu :")
```

```
input(Jumlah_Tamu)
```

KAMUS/DEKLARASI VARIABEL FUNCTION Pemilihan

Tipe[10],Tipe_Kasur[20],Id_Kamar[5],menginap[20],hargakmr[10], harga[30] : char

ALGORITMA/DESKRIPSI FUNCTION Pemilihan

```
print("| No | Tipe Kamar || Harga |")
```

```
print("| 1 | Anggrek || Rp. 500.000/Mlm |")
```

```
print("| 2 | Mawar || Rp. 300.000/Mlm |")
```

```
print("| 3 | Melati || Rp. 400.000/Mlm |")
```

```
print("| 4 | Matahari || Rp. 200.000/Mlm |")
```

```
print("| 5 | Lavender || Rp. 150.000/Mlm |")
```

```
print("Tipe Kamar :")
```

```
input(Tipe)
```

```
print("Tipe Kasur :")
```

```
input(Tipe_Kasur)
```

```
print("ID Kamar : ")
```

```
input(id_Kamar)
```

```
print("Menginap Berapa Hari :")
```

```
input(menginap)
```

```
print("Total Harga Kamari :")
```

```
input(hargakmr)
```

KAMUS/DEKLARASI VARIABEL FUNCTION Hasil_Pilih
char NamaK[30], TipeK[20] : char

ALGORITMA/DESKRIPSI FUNCTION Hasil_Pilih

```
print(" Data Pemesanan Kamar")
strcpy>NamaK>Nama)
strcpy>TipeK>Tipe)
strupr>TipeK)
strupr>NamaK)
print>Nama, namaK)
print>No Telp, no_telp)
print>Email, email)
print>Jumlah Tamu, jumlah_tamu)
print>Tipe Kamar, TipeK)
print>Tipe Kasur, tipe_Kasur)
print>ID Kamar, id_kamar)
print>Menginap Berapa Hari, menginap)
print>Total Harga Kamari, hargakmr)
```

KAMUS/DEKLARASI VARIABEL FUNCTION Makanan

```
jlmdt = 5 : int
a,b,c,p : int
deri[jlmdt] = {20000, 25000, 30000, 50000, 40000} : int
```

ALGORITMA/DESKRIPSI FUNCTION Makanan

```
print"          Menu Makanan          "
print("| No | Makanan      || Harga      |")
print("| 1 | Nasi Goreng    || Rp. 20.000  |")
print("| 2 | Satay          || Rp. 25.000  |")
print("| 3 | Soto Betawi    || Rp. 30.000  |")
print("| 4 | Sop Buntut     || Rp. 50.000  |")
print("| 5 | Ayam Bakar Madu || Rp. 40.000  |")

print (" Sebelum Pengurutan Harga Makanan ")
for (a = 0; a <= jlmdt-1; a++)
    deri[a] " " ;
endl;
for(b = 1; b <= jlmdt-1; b++) {
    a=b ;
    c=deri[a];
    while (a >= 0 && deri [a-1] < c) {
        deri[a] = deri [a-1];
        a--; }
    deri[a] = c; }
print(" Pengurutan Descending Harga Makanan")
for(a = 0; a <= jlmdt-1; a++) //Selection Sort Descending
    deri[a]
endl
```

KAMUS/DEKLARASI VARIABEL FUNCTION pesan_makanan
Nama[30], No_Telp[15], menu[20], Id_Kamar[5], Tipe[10] : char

```

ALGORITMA/DESKRIPSI FUNCTION pesan_makanan
print("Silahkan Mengisi Form Untuk Memesan Makanan")
print("Nama :")
input>Nama)
print("No Telp :")
input>No_Telp)
print("Makanan : ")
input(makanan)
print("Tipe Kamar :")
input(tipe_Kamar)

```

KAMUS/DEKLARASI VARIABEL FUNCTION hasil_makan

NamaK[30], menuK[20] : char

```

ALGORITMA/DESKRIPSI FUNCTION hasil_makan
strcpy>NamaK,>Nama);
strcpy(menuK,menu);
strupr(menuK);
strupr>NamaK);
print>Nama, namaK)
print>No Telp, no_telp)
print>Makanan, MenuKl)
print>Tipe Kamar, TipeK)
print>ID Kamar, id_kamar)

```

KAMUS/DEKLARASI VARIABEL FUNCTION menu

pilih : int

```

ALGORITMA/DESKRIPSI FUNCTION menu
print(" Menu Pilihan")
print("1. Pesan Makanan")
print("2. Pesan Makanan Hotel")
print("3. Keluar Dari Hotel")
print("Masukkan angka yang dipilih [1/2/3] :")
input(pilih)
switch(pilih)
case 1: Form_Kamar(); break;
case 2: Makanan(); break;
default:
print(" Keluar Dari Hotel ")
print(" Terima Kasih Telah Menginap Di Hotel Deriansyah")

```

KAMUS/DEKLARASI VARIABEL

-

```

ALGORITMA/DESKRIPSI
print(" Tugas Besar Praktikum Algoritma dan Pemrograman  |")
print(" | Agil Deriansyah Hasan |")
print(" | 4522210125 |")

```

2.1.5 Algoritma

```
1.  Membuar function displayMenu()
2.  Mencetak/Menampilkan "Selamat Datang Di Hotel Deriansyah"
3.  Mencetak/Menampilkan "Jln.Bintara14, No 25 Telp 087833160661"
4.  Mencetak/Menampilkan "Jl. Kp.Bojong, Kec. Bekasi Bar., Kota Bekasi, Jawa Barat, 17134"
5.  Memanggil function Menu()
6.  Membuat function form_kamar()
7.  Mencetak/Menampilkan "Pengisian Form Biodata Diri"
8.  Mencetak/Menampilkan "Nama : "
9.  Memasukkan/menginput isi/nilai variabel Nama
10. Mencetak/Menampilkan "No Telp : "
11. Memasukkan/menginput isi/nilai variabel No_Telp
12. Mencetak/Menampilkan "Email : "
13. Memasukkan/menginput isi/nilai variabel Email
14. Mencetak/Menampilkan "Jumlah Tamu : "
15. Memasukkan/menginput isi/nilai variabel Jumlah_Tamu
16. Memanggil function Pemilihan
17. Membuat function Pemilihan()
18. Mencetak/Menampilkan" | No | Tipe Kamar || Harga |"
19. Mencetak/Menampilkan" | 1 | Anggrek || Rp. 500.000/Mlm |"
20. Mencetak/Menampilkan" | 2 | Mawar || Rp. 300.000/Mlm |"
21. Mencetak/Menampilkan" | 3 | Melati || Rp. 400.000/Mlm |"
22. Mencetak/Menampilkan" | 4 | Matahari || Rp. 200.000/Mlm |"
23. Mencetak/Menampilkan" | 5 | Lavender || Rp. 150.000/Mlm |"
24. Mencetak/Menampilkan "Tipe Kamar : "
25. Memasukkan/menginput isi/nilai variabel Tipe
26. Mencetak/Menampilkan "Tipe Kasur : "
27. Memasukkan/menginput isi/nilai variabel Tipe_Kasur
28. Mencetak/Menampilkan "ID Kamar : "
29. Memasukkan/menginput isi/nilai variabel id_Kamar
30. Mencetak/Menampilkan "Menginap Berapa Hari : "
31. Memasukkan/menginput isi/nilai variabel menginap
32. Mencetak/Menampilkan "Total Harga Kamari : "
33. Memasukkan/menginput isi/nilai variabel hargakmr
34. Memanggil function Hasil_Pilih
35. Membuat function Hasil_Pilih
36. strcpy>NamaK>Nama)
37. strcpy(TipeK,Tipe)
38.strupr(TipeK)
39.strupr>NamaK)
40. Mencetak/Menampilkan " Data Pemesanan Kamar"
41. Mencetak/Menampilkan "Nama"
42. Mencetak/menampilkan isi/nilai variabel namaK
43. Mencetak/Menampilkan "No Telp"
44. Mencetak/menampilkan isi/nilai variabel no_telp
45. Mencetak/Menampilkan "Email"
46. Mencetak/menampilkan isi/nilai variabel email
47. Mencetak/Menampilkan "Jumlah Tamu"
48. Mencetak/menampilkan isi/nilai variabel jumlah_tamu
49. Mencetak/Menampilkan "Tipe Kamar"
50. Mencetak/menampilkan isi/nilai variabel TipeK
51. Mencetak/Menampilkan "Tipe Kasur"
52. Mencetak/menampilkan isi/nilai variabel tipe_Kasur
```

```

53. Mencetak/Menampilkan "ID Kamar"
54. Mencetak/menampilkan isi/nilai variabel id_kamar
55. Mencetak/Menampilkan "Menginap Berapa Hari"
56. Mencetak/menampilkan isi/nilai variabel menginap
57. Mencetak/Menampilkan "Total Harga Kamari"
58. Mencetak/menampilkan isi/nilai variabel hargakmr
59. Memanggil function Menu()
60. Membuat function Makanan()
61. Mencetak/menampilkan"           Menu Makanan           "
62. Mencetak/menampilkan"| No |   Makanan   ||   Harga   |"
63. Mencetak/menampilkan"| 1 |   Nasi Goreng   || Rp. 20.000   |"
64. Mencetak/menampilkan"| 2 |   Satay           || Rp. 25.000   |"
65. Mencetak/menampilkan"| 3 |   Soto Betawi    || Rp. 30.000   |"
66. Mencetak/menampilkan"| 4 |   Sop Buntut     || Rp. 50.000   |"
67. Mencetak/menampilkan"| 5 |   Ayam Bakar Madu || Rp. 40.000   |"
68. Mencetak/menampilkan "Sebelum Pengurutan Harga Makanan"
69. a = 0
70. selama (a<=jmltdt-1), kerjakan baris 71 s.d 72 kalau tidak baris 73
71. deri[a]
72. Mencetak/menampilkan " "
73. selama (b<=jmltdt-1), kerjakan baris 74 s.d 75 kalau tidak baris 76
74. a=b
75. c=deri[a]
76. selama (a>= 0 && deri[a-1] < c) kerjakan baris 77 s.d 79 kalau tidak baris 80
77. deri[a] = deri[a-1]
78. a-
79. deri[a] = c
80. Mencetak/menampilkan " Pengurutan Descending Harga Makanan"
81. deri[a]
82. Mencetak/menampilkan " "
83. selama (a<=jmltdt-1) kerjakan baris 84
84. Memanggil function pesan_makanan
85. Membuat function pesan_makanan
86. Mencetak/Menampilkan "Silahkan Mengisi Form Untuk Memesan Makanan"
87. Mencetak/Menampilkan "Nama : "
88. Memasukkan/menginput isi/nilai variabel Nama
89. Mencetak/Menampilkan "No Telp : "
90. Memasukkan/menginput isi/nilai variabel No_Telp
91. Mencetak/Menampilkan "Tipe Kamar : "
92. Memasukkan/menginput isi/nilai variabel Tipe
93. Mencetak/Menampilkan "Makanan : "
94. Memasukkan/menginput isi/nilai variabel menu
95. Mencetak/Menampilkan "ID Kamar : "
96. Memasukkan/menginput isi/nilai variabel id_Kamar
97. Memanggil function hasil_makan()
98. Membuat function hasil_makan
99. strcpy>NamaK>Nama)
100. strcpy(menuK,Tipe)
101. strupr(menuK)
102. strupr>NamaK)
103. Mencetak/Menampilkan "Nama"
104. Mencetak/menampilkan isi/nilai variabel namaK
105. Mencetak/Menampilkan "No Telp"
106. Mencetak/menampilkan isi/nilai variabel no_telp

```

```

107. Mencetak/Menampilkan "Makanan"
108. Mencetak/menampilkan isi/nilai variabel menuK
109. Mencetak/Menampilkan "Tipe Kamar"
110. Mencetak/menampilkan isi/nilai variabel tipe
111. Mencetak/Menampilkan "id kamar"
112. Mencetak/menampilkan isi/nilai variabel id_kamar
113. Memanggil function menu()
114. Membuat function Menu
115. Mencetak/menampilkan "Menu Pilihan"
116. Mencetak/menampilkan "1. Pesan Kamar"
117. Mencetak/menampilkan "2. Pesan Makanan Hotel"
118. Mencetak/menampilkan "3. Keluar Dari Hotel"
119. Mencetak/menampilkan "Masukkan angka yang dipilih [1/2/3]"
120. Memasukkan/menginput isi/nilai variabel pilih
121. Jika (pilih == 1), maka kerjakan baris 121 kalau tidak baris 122
122. Memanggil function form kamar()
123. Jika (pilih == 2), maka kerjakan baris 123 kalau tidak baris 124
124. Memanggil function form makanan()
125. Mencetak/menampilkan " Keluar Dari Hotel "
126. Mencetak/menampilkan "Terima Kasih Telah Menginap Di Hotel Deriansyah"
127. Mencetak/menampilkan "| Tugas Besar Praktikum Algoritma dan Pemrograman |"
128. Mencetak/menampilkan "| Agil Deriansyah Hasan |"
129. Mencetak/menampilkan "| 4522210125 |"
130. Memanggil function displayMenu()
131. Selesai

```


BAB III

Penutup

3.1 Kesimpulan

Dari paparan di atas, dapat disimpulkan bahwa pemahaman dan penguasaan terhadap materi algoritma dan pemrograman memiliki peran krusial dalam era teknologi informasi. Kemampuan merancang algoritma, mengimplementasikan kode dengan efisiensi, dan memilih struktur data yang tepat menjadi fondasi utama dalam menghadapi tantangan di dunia digital.

Pembelajaran algoritma dan pemrograman bukan hanya tentang aspek teknis semata, tetapi juga melibatkan kemampuan analisis, pemecahan masalah, dan kreativitas. Para pembelajar diharapkan dapat menguasai konsep dasar, memahami efisiensi algoritma, serta memiliki kemampuan beradaptasi dengan perkembangan paradigma pemrograman.

3.2 Kritik dan Saran

Penulis menyadari bahwa program yang dibuat ini masih terdapat banyak kekurangan karena keterbatasan pengetahuan dan kurangnya pengalaman penulis. Program masih bisa dikembangkan menjadi lebih luas lagi, dan memiliki lebih banyak fungsi lagi sehingga program dapat menjadi jauh lebih efisien.

Daftar Pustaka

Dra.SRI REZEKI CANDRA NURSARI,M.Kom. 2022. Algoritma dan Pemrograman.
<https://lms.univpancasila.ac.id/course/view.php?id=13380>. Diakses pada 12 Januari 2023

Video Presentasi

https://youtu.be/UT2CUmf_8lE