

**LAPORAN PRAKTIKUM ALGORITMA DAN PEMPROGRAMAN**  
**“SI PEMBAYARAN PAM”**

Disusun Guna Memenuhi Tugas Mata Kuliah Algoritma dan Pemograman

Dosen Pengampu :

**Dra. Sri Rezeki Candra Nursari, M.KOM**



**Disusun Oleh :**

Zahra Tsabitah

NPM 4523210145

**ALGORITMA DAN PEMOGRAMAN**  
**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**

**2023**

## KATA PENGANTAR

Puji dan syukur saya panjatkan kepada Allah swt, Tuhan Yang Maha Esa atas limpahan rahmatnya yang telah diberikan sehingga penulis dapat menyelesaikan Laporan Praktikum dengan baik dan tepat waktu tanpa terkendala halangan apapun.

Laporan praktikum ini disusun sebagai tugas akhir pada mata kuliah Algoritma dan Pemograman, penulis menyajikan laporan praktikum sebagai bentuk dokumentasi atas segala implementasi materi atau teori pada saat praktikum. Praktikum ini dilaksanakan dengan tujuan untuk mengembangkan keterampilan praktis, meningkatkan pemahaman konsep pada materi, dan mengevaluasi kemampuan saya sebagai seorang mahasiswa. Terimakasih kepada Dra. Sri Rezeki Candra Nursari, M.Kom selaku dosen pengampu pada mata kuliah Algoritma dan Pemograman atas bimbingan, arahan dan ilmu yang beliau sampaikan.

Dalam laporan praktikum mengambil hasil praktikum dari materi pertama yaitu “Sequence” sampai akhir “File Teks/Biner” serta tugas besar praktik “Sistem Pembayaran PAM” pada hasil praktikum mengambil beberapa sumber referensi agar program yang dihasilkan sesuai dan relevan pada masa kini. Penulis berharap agar laporan praktikum ini dapat menjadi bahan referensi yang berguna untuk kedepannya dan juga sebagai panduan pembuatan program-program lainnya.

Semoga laporan praktikum ini memberikan kontribusi positif dan menjadi referensi yang bermanfaat bagi pembaca. Atas waktu dan perhatiannya saya ucapkan terimakasih.

Jakarta, 26 Desember 2023

Penulis :

Zahra Tsabitah

NPM 4523210145

## DAFTAR ISI

<b>KATA PENGANTAR .....</b>	<b>i</b>
<b>DAFTAR ISI .....</b>	<b>ii</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1    Latar Belakang .....	1
1.2    Tujuan .....	1
1.3    Pembahasan Teori .....	2
1.3.1    Sequence/Urutan.....	2
1.3.2    Selection/Seleksi.....	2
1.3.3    Repetition/Perulangan .....	3
1.3.4    Rekursif .....	3
1.3.5    Array .....	4
1.3.6    Function/Fungsi .....	4
1.3.7    String .....	4
1.3.8    Sort/Pengurutan .....	5
1.3.9    Pointer.....	5
1.3.10    File(Biner/Teks) .....	6
1.4    Tugas Praktikum .....	7
1.4.1    Prak 1 – Sequence .....	7
1.4.2    Prak 2 – Selection (If & If-Else).....	7
1.4.3    Prak 3 – Selection (Nested If & Switch ) .....	8
1.4.4    Prak 4 – Repetition (While & do-While).....	9
1.4.5    Prak 5 – Repetition (For & Nested for) .....	10
1.4.6    Prak 6 – Rekursif .....	11
1.4.7    Prak 7 – Array (1 Dimensi) .....	12
1.4.8    Prak 8 – Array (2 MultiDimensi) .....	12
1.4.9    Prak 9 – Function.....	14

1.4.10	Prak 10 – String .....	15
1.4.11	Prak 11 – Sort (SelectionSort & InsertionSort) .....	16
1.4.12	Prak 12 – Pointer .....	20
1.4.13	Prak 13 – File (Biner/Teks) .....	24
<b>BAB II PEMBAHASAN</b>	<b>.....</b>	<b>25</b>
2.1	Tugas Besar.....	25
2.2	Pembahasan Tugas Besar.....	25
2.2.1	Program/Source Code.....	25
2.2.2	Hasil Compile/Run Code.....	28
2.2.3	Pseudocode .....	30
2.2.4	Algoritma.....	34
<b>BAB III PENUTUP</b>	<b>.....</b>	<b>37</b>
3.1	Kesimpulan .....	37
3.2	Kritik dan Saran .....	37
3.2.1	Kritik.....	37
3.2.2	Saran .....	38
<b>DAFTAR PUSTAKA</b>	<b>.....</b>	<b>39</b>

## **BAB I**

### **PENDAHULUAN**

#### **1.1 Latar Belakang**

Di zaman sekarang kemajuan dan perkembangan teknologi sudah cukup canggih dan modern, banyak hal hal yang tadinya susah dilakukan ataupun tidak bisa dilakukan oleh umat manusia menjadi suatu hal yang mudah digapai. Perkembangan dan kemajuan tersebut pun tidak bisa lepas dari kehidupan sehari – hari umat manusia dalam melakukan kegiatan maupun pekerjaan.

Maka dengan adanya perkembangan dan kemajuan teknologi, umat manusia dituntut harus bisa beradaptasi dengan hal tersebut, seperti dalam melakukan sebuah transaksi atau pembayaran yang tadinya menggunakan buku dicatat dengan pena berubah menjadi komputer menggunakan perangkat keras (Hardware) dengan didukung perangkat lunak (Software).

Dengan tersedianya berbagai bahasa pemrograman komputer, penulis pun terdorong untuk membuat sebuah program sederhana dengan bahasa pemrograman C++. Seperti membuat program Pembayaran PAM, sekaligus untuk memenuhi tugas besar praktikum algoritma dan pemograman. Diharapkan dengan penulis membuat program ini dapat memudahkan dalam transaksi dan melihat data pembayaran serta mengurangi waktu tunggu dan antrian, karena sudah dialihkan agar menjadi Online atau dapat diakses dimana saja dengan mudah.

#### **1.2 Tujuan**

Berdasarkan latar belakang tujuan pembuatan laporan praktikum dan tujuan pembelajaran praktikum sebagai berikut :

1. Untuk memenuhi tugas besar Praktikum Algoritma dan Pemograman yang diberikan kepada mahasiswa semester 1 sebagai bentuk Laporan Praktikum atas pembelajaran praktik selama ini.
2. Sebagai bahan bacaan referensi yang bermanfaat dan bisa menjadi dasar ide untuk pengembangkan sistem selanjutnya.
3. Melatih kemampuan dan mengembangkannya melalui pembelajaran praktik sesuai dengan materi yang diajarkan sebagaimana implementasinya pada saat memprograman suatu sistem.
4. Sebagai media pengasahan skill, melalui pembelajaran praktikum kemampuan dapat berkembang sesuai dengan kebutuhan tenaga kerja.

5. Guna mempermudah atau memberikan referensi ide Sistem Pembayaran PAM atau air minum atau pembayaran air, yang memudahkan para pengguna dalam melakukan transaksi secara online atau virtual dan dapat melihat bukti berupa rekap laporan pembayaran 1 tahun sebelumnya (tahun lalu) dan bukti transaksi pembayaran bulan ini.

## 1.3 Pembahasan Teori

### 1.3.1 Sequence/Urutan

Merupakan proses melakukan perintah atau pernyataan pada baris code secara berurutan dilakukan satu per-satu, saat program dijalankan atau di compile maka baris code paling atas atau pertama akan berjalan terlebih dahulu entah itu output ataupun inputan.

```
cout<<"\nSilahkan Pilih Aktivitas yang diinginkan sesuai dengan menu dibawah ini"<<endl;
cout<<"1. Lihat Rekap Tahun Lalu"<<endl;
cout<<"2. Melakukan Pembayaran Bulan Ini "<<endl;
cout<<"Masukan Pilihan : ...."<<endl;
    Sleep(2000);cout << "\x1b[1A\x1b[2K";
    cout<<"\nAnda memilih pilihan ke-"<<pilihan<<endl;
    cout<<endl;
cout<<"MOHON TUNGGU SEBENTAR SEDANG BERALIH....."<<endl;
    Sleep(2000);cout << "\x1b[1A\x1b[2K";
cout<<endl;
cout<<endl;
```

Gambar 1 : Contoh Sequence (TugasBesar)

### 1.3.2 Selection/Seleksi

Seperti namanya yaitu seleksi merupakan salah satu struktur kontrol untuk melakukan pemilihan untuk melakukan proses lanjutan sesuai dengan kondisi yang telah dibuat pada program atau code. Contoh sistem seleksi yaitu if, if-else-nested if dan switch.

```
//sistem pemilihan aktivitas dengan menampilkan
switch (pilihan) { //sesuai dengan inisialisasi
    case 1:
        lihatRekapTahunLalu(); //memanggil fungsi
        break;
    case 2:
        PembayaranBulanIni();
        break;
    default:
        cout << "Pilihan tidak valid." << endl;
}
```

Gambar 2.1 : Contoh Switch  
(TugasBesar)

Gambar 2.1 : Contoh If-Else  
(TugasBesar)

### 1.3.3 Repetition/Perulangan

Repetition atau perulangan pada sistem algoritma pemograman merupakan sistem yang akan melakukan perulangan atau looping sesuai kondisi atau baris code yang dibuat. Looping juga bisa terjadi terus menerus jika tidak diberikan kondisi untuk berhenti/stop contoh kondisi agar perulangan tidak menjadi infinite loop adalah dengan pernyataan kondisi seperti pada contoh code tugas besae “R < JUMLAH\_BULAN” dan post-increment “++”. Contoh sistem repetition adalah for, nested-for, while dan do-While.

```
/*Sistem proses menampilkan data berdasarkan array looping
variable z sebagai patokan*/
for (int z = 0; z < JUMLAH_BULAN; z++) {
    //sistem menghitung biaya
    if(pakaiAir[z] <= 35){
        tagihanAir[z] = 7450;
        tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] +
    } else {
        tagihanAir[z] = 8250;
        tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] +
    }
}
```

Gambar 3.1 : Contoh For (TugasBesar)

```
while (R < JUMLAH_BULAN) {
    if (bulan[R] < bulan[t])
        t = R;
    R++;
}
```

Gambar 3.2 : Contoh While  
(TugasBesar)

### 1.3.4 Rekursif

Rekursif berarti proses yang memanggil dirinya sendiri, terdiri dari prosedur atau fungsi. Perbedaannya rekursif dapat memanggil ke dirinya sendiri sedangkan prosedur atau fungsi harus dipanggil oleh prosedur/fungsi lainnya. Pemakaian sistem rekursif dipakai oleh pangkat, factorial, Fibonacci, dan proses deret.

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

int main(){
    long int z, t, fak_iterZ = 1;
    cout<<endl;
    cout<<"Menampilkan Rekursif - FAKTORIAL"<<endl;
    cout<<"-----"<<endl;
    cout<<"Masukan Sebuah Bilangan = ";cin>>z;
    cout<<endl;
    cout<<"-----"<<endl;
    for(t=1; t<=z; t++){
        fak_iterZ = fak_iterZ * t;
        cout<<"Faktorial dari "<<t<<"! adalah = "<<fak_iterZ<<endl;
    }
    cin.get();
}
```

Gambar 4 : Contoh Rekursif-faktorial (ContohPraktikum6)

### 1.3.5 Array

Array/Larik adalah suatu tipe data terstruktur yang terdapat dalam memori yang terdiri dari sejumlah elemen yang mempunyai tipe data sama. Array memiliki beberapa jenis yaitu 1 Dimensi, 2 Dimensi, 3 Dimensi, dan MultiDimensi. Pada array 1 dimensi memiliki format seperti ini “TipeData <NamaArray> [Ukuran]” sedangkan untuk array 2 dimensi “TipeData <NamaArray> [baris][kolom]”

```
char nama[50] = "Zahra Tsabitah", //Menyimpan d  
alamat[50] = "Jl. Pam Baru Raya No.7a",  
ID[50] = "45-23-2-101-45",  
jenis[50] = "Rumah Tinggal",  
tanggal[50] = "2 Januari 2024",
```

Gambar 5.1 : Contoh Array 1Dimensi (TugasBesar)

```
char bayar[JUMLAH_BULAN] [10] = {"CASH", "BCA", "BRI", "CASH",  
"BRI", "CASH", "BCA", "CASH",  
"BCA", "BCA", "BRI", "CASH"};
```

Gambar 5.2 : Contoh Array 2Dimensi (TugasBesar)

### 1.3.6 Function/Fungsi

Function atau fungsi tempat pengumpulan code dijadikan satu kelompok sesuai dengan nama fungsinya yang kemudian fungsi dipanggil pada menu main atau dapat dipanggil pada fungsi lainnya. Fungsi terdiri dari 2 komponen yaitu **Definisi Fungsi** berisi tipe data dari fungsi, nama fungsi, dan argument/paramenter yang digunakan kemudian **Tubuh Fungsi** berisi dengan perintah yang diberikan kepada fungsi.

```
void lihatRekapTahunLalu() {  
    lihatRekapTahunLalu(); //memanggil fungsi  
    break;  
}  
void PembayaranBulanIni() {  
    PembayaranBulanIni();  
    break;  
};
```

Gambar 6.1 : Contoh Membuat Function (TugasBesar)

Gambar 6.2 : Contoh Memanggil Function (TugasBesar)

### 1.3.7 String

String untuk penulisan kalimat atau karakter yang panjang seperti teks, tipe data dan fungsi yang ada pada string bisa digunakan jika memakai library string/string.h. Karakter konstanta string menempati memori sebesar 1 byte dan selalu diakhiri dengan karakter ‘NULL’ atau “0”. Deklarasi string terdapat 2 tipe “char Variable\_string[<panjang>]” dan “string Variable\_string”.

```
char nama[50] = "Zahra Tsabitah", //Menyimpulkan bahwa dia adalah seorang wanita  
alamat[50] = "Jl. Pan Baru Raya No.7a",  
ID[50] = "45-23-2-101-45",  
jenis[50] = "Rumah Tangga",  
tanggal[50] = "2 Januari 2024",  
  
string status[JUMLAH_BULAN] =  
{"TERLAMBAT (!)", "TERLAMBAT (!)", "TERLAMBAT (!)",  
"TERLAMBAT (!)", "TERLAMBAT (!)", "TERLAMBAT (!)",  
"TERLAMBAT (!)", "TERLAMBAT (!)", "TERLAMBAT (!)",  
"TERLAMBAT (!)", "TERLAMBAT (!)", "TERLAMBAT (!)"}
```

Gambar 7.1 : Contoh Deklarasi String Menggunakan Char (TugasBesar)

```
    strcpy(code, "4444 6793 8712 0853 4365");
} else{
    strcpy(code, "-");
}
    << strupr(nama) << endl;
    << strupr(alamat) << endl;
```

Gambar 7.2 : Contoh Deklarasi String Menggunakan string (TugasBesar)

Gambar 7.3 : Contoh Penggunaan Fungsi String-Copy dan UpperCase/Huruf Besar (TugasBesar)

### 1.3.8 Sort/Pengurutan

Sort merupakan proses pengurutan nilai atau data yang sebelumnya tersusun acak menjadi terurut menurut aturan tertentu, terdapat 2 jenis pengurutan secara Ascending(naik) dan Descending(turun). Ada beberapa metode sorting yaitu Selection Sort, Insertion Sort, Bubble/Exchange Sort, Shell Sort, Merge Sort, Heap Sort dan Quick Sort.

Tetapi pada peraktikum hanya membahas 2 metode yaitu Selection Sort mencari nilai terbesar/terkecil kemudian akan ditukarkan dengan nilai sekarang dan Insertion Sort membandingkan nilai sekarang dan nilai disebelahnya atau berikutnya kemudian memindahkannya ke posisi depan atau awal sesuai kondisi.

```
/*Sistem proses SelectionSort-Ascending untuk mengurutkan  
data berdasarkan variable array*/  
while (s < JUMLAH_BULAN - 1) {  
    t = s;  
    R = s + 1;  
    while (R < JUMLAH_BULAN) {  
        if (bulan[R] < bulan[t])  
            t = R;  
        R++;  
    }  
  
    h = bulan[t];  
    bulan[t] = bulan[s];  
    bulan[s] = h;  
    s++;  
}
```

Gambar 8 : Contoh SelectionSort-Ascending (TugasBesar)

### 1.3.9 Pointer

Pointer adalah variable yang nilainya merupakan address dari variable yang lain, suatu pointer dalam program dimaksudkan untuk menunjuk ke suatu Alamat memori. Pointer juga bisa digunakan sebagai alias untuk variable lain contoh “\*data = dataPembayaran”, definisi format pointer “tipe\_data \*nama\_variable”

```
const char *bayar[] = {"BRI", "BCA", "CASH"}; //deklarasi array dari pointer ke kontanta string
panjang_array = sizeof(bayar) / sizeof(bayar[0]); //menghitung panjang character array
metode = rand() % panjang_array; //menghasilkan kata random pada data yang ada di dalam variable bayar

if(bayar[metode] == "BRI"){
    tagihanPakaiAir += 2500;
    strcpy(code, "0097 8823 2791 1111 0273");
} else if (bayar[metode] == "BCA"){
    tagihanPakaiAir += 2500;
    strcpy(code, "4444 6793 8712 0853 4365");
} else{
    strcpy(code, "-");
}
```

Gambar 9 : Contoh Elemen Array menjadi Pointer ke char/string dan Elemen Array merunjuk karakter constanta (TugasBesar)

### 1.3.10 File(Biner/Teks)

File terdiri dari beberapa komponen secara satu kesatuan yaitu File → Record → Field → Byte → Bit. File terdiri dari 2 jenis yaitu File Teks dengan format “.txt” dan File Biner “.Dat“, file biner memiliki kecepatan akses lebih cepat dibandingkan file teks karena terdiri hanya dari 2 bilangan yaitu 0 dan 1. Dengan menggunakan library fstream fungsi-fungsi Sistem I/O File dapat digunakan bisa untuk membuat/memasukan data kedalam file dan menampilkan isi file.

```
ofstream file("TAGIHAN PEMBAYARAN.txt"); // membuka/membuat file "TAGIHAN PEMBAYARAN"
if (file.is_open()) { //nyatakan kondisi apakah file terbuka
    //Masukan seluruh data kedalam file
    file << "-----\n";
    file << "-----TAGIHAN PEMBAYARAN AIR-----\n";
    file << "-----\n";
    file << setw(25) << left << "Nama Pelanggan" << endl;
    file << setw(25) << left << "Alamat Pelanggan" << endl;
    file << setw(25) << left << "ID Pelanggan" << endl;
    file << setw(25) << left << "Jenis Golongan" << endl;
    file << setw(25) << left << "Tanggal" << endl;
    file << setw(25) << left << "Pemakaian Air (m3)" << endl;
    file << setw(25) << left << "Tagihan Air" << endl;
    file << setw(25) << left << "Biaya Admin" << endl;
    file << setw(25) << left << "Biaya Pajak" << endl;
    file << "-----\n";
    file << setw(25) << left << "TOTAL TAGIHAN PEMAKAIAN AIR" << endl;
    file << setw(25) << left << "SISTEM PEMBAYARAN" << endl;
    file << setw(25) << left << "CODE PEMBAYARAN" << endl;
    file << "-----\n";

    file.close(); //menutup file
    cout << "SEDANG MENCETAK TAGIHAN MOHON TUNGGU SEBENTAR.....\n";
    Sleep(5000);
    cout << "\x1b[1A\x1b[2K";
    cout << "\nBERHASIL MENCETAK TAGIHAN KE DALAM FILE | TAGIHAN PEMBAYARAN.txt |";
} else {
    cout << "GAGAL CETAK TAGIHAN" << endl; //kondisi jika file gagal dibuka
}
```

Gambar 10 : Contoh Membuat/Memasukan data ke File Teks (TugasBesar)

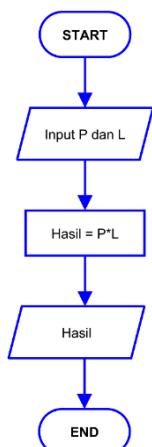
## 1.4 Tugas Praktikum

### 1.4.1 Prak 1 – Sequence

No. Topik Soal : Prak1-12

Tanggal Pengerjaan : 29/10/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     float P, L, Hasil;     cout &lt;&lt;"Menghitung Luas Kamar Adi (Persegi Panjang)"&lt;&lt;endl;     cout &lt;&lt;"-----";     cout &lt;&lt;"Masukan Panjang kamar = "; cin&gt;&gt;P;     cout &lt;&lt;"Masukan Lebar kamar = "; cin&gt;&gt;L;     Hasil = P*L;     cout &lt;&lt; endl;     cout &lt;&lt; "Luas Kamar Adi Adalah " &lt;&lt; Hasil &lt;&lt; endl;     cin.get(); }</pre>	C:\ProjectMinGW\TugasModul1-5>cd Modul1 C:\ProjectMinGW\TugasModul1-5\Modul1>Prak1 Menghitung Luas Kamar Adi (Persegi Panjang) ----- Masukan Panjang kamar = 34 Masukan lebar kamar = 5.2 Luas Kamar Adi Adalah 176.8 C:\ProjectMinGW\TugasModul1-5\Modul1>cd .. C:\ProjectMinGW\TugasModul1-5\Modul1



#### Pseudecode

##### Kamus/Deklarasi Variabel

P : Float

L : Float

Hasil : Float

##### Dekripsi

input (P,L)

    Hasil  $\leftarrow$  P\*L

print(Hasil);

#### Algoritma

1. Memasukan/Menginput Nilai Variable P
2. Memasukan/Menginput Nilai Variable L
3. Menghitung Hasil  $\leftarrow$  P\*L
4. Mencetak/Menampilkan Nilai Variable Hasil
5. Selesai

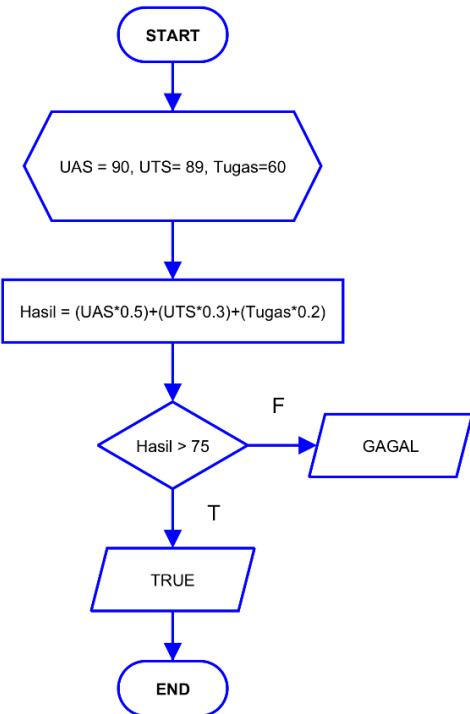
### 1.4.2 Prak 2 – Selection (If & If-Else)

No. Topik Soal : Prak4-06

Tanggal Pengerjaan : 29/10/2023

Source Code	Hasil Compile
-------------	---------------

<pre>#include &lt;iostream&gt; using namespace std;  int main(){     int UAS = 90, UTS = 89, Tugas = 60;     double Hasil;     cout &lt;&lt; " Menyatakan Pernyataan Untuk Nilai Akhir " &lt;&lt; endl;     cout &lt;&lt; "-----" &lt;&lt; endl;     Hasil = (UAS*0.5)+(UTS*0.3)+(Tugas*0.2);     if(Hasil &gt; 75)         cout &lt;&lt; " LULUS" &lt;&lt; endl;     else         cout &lt;&lt; " GAGAL" &lt;&lt; endl;     cin.get(); }</pre>	<pre>C:\ProjectMinGW\TugasModul1-5\Modul2&gt;Prak2-06 Menyatakan Pernyataan Untuk Nilai Akhir ----- LULUS ^C C:\ProjectMinGW\TugasModul1-5\Modul2&gt;cd ..</pre>
---	--



## Pseudecode

### Kamus/Deklarasi Variabel

UAS, UTS, Tugas : Intenger  
Hasil : Double

### Deksripsi

```

UAS = 90
UTS = 89
Tugas = 60
Hasil ← (UAS*0.5)+(UTS*0.3)+(Tugas*0.2)
If (Hasil > 75) then
    Print('LULUS')
Else
    Print('GAGAL')
Endif
  
```

## Algoritma

- 1.UAS = 90
- 2.UTS = 89
- 3.Tugas =60
- 4.Menghitung Hasil ← (UAS\*0.5)+(UTS\*0.3)+(Tugas\*0.2)
- 5.Mencetak/Menampilkan Nilai Variable Hasil
- 6.Jika Hasil > 75
- 7.Mencetak/Menampilkan "LULUS"
- 8.Jika lainnya/tidak
- 9.Mencetak/Menampilkan "Gagal"
- 10.Selesai

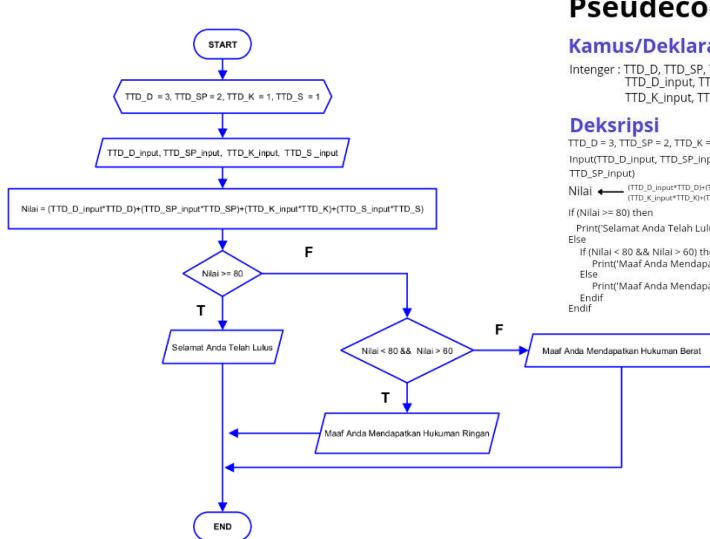
### 1.4.3 Prak 3 – Selection (Nested If & Switch )

No. Topik Soal : Prak3-05

Tanggal Pengerjaan : 29/10/2023

Source Code	Hasil Compile

<pre>#include &lt;iostream&gt; using namespace std;  int main() {     int TTD_D_input, TTD_SP_input, TTD_K_input, TTD_S_input, Nilai;     int TTD_D = 3; //tanda tangan dosen     int TTD_SP = 2; //tanda tangan senior panitia     int TTD_K = 1; //tanda tangan senior karyawan     int TTD_S = 1; //tanda tangan senior lainnya      cout &lt;&lt; "----- Menyatakan Pernyataan Untuk Nilai -----" &lt;&lt; endl;     cout &lt;&lt; "Masukan Jumlah Tanda Tangan Dosen = ";     cout &lt;&lt; cin &gt;&gt; TTD_D_input;     cout &lt;&lt; "Masukan Jumlah Tanda Tangan Senior Panitia = ";     cout &lt;&lt; cin &gt;&gt; TTD_SP_input;     cout &lt;&lt; "Masukan Jumlah Tanda Tangan Karyawan = ";     cout &lt;&lt; cin &gt;&gt; TTD_K_input;     cout &lt;&lt; "Masukan Jumlah Tanda Tangan Senior Lainnya = ";     cout &lt;&lt; cin &gt;&gt; TTD_S_input;      Nilai = (TTD_D_input * TTD_D) + (TTD_SP_input * TTD_SP) + (TTD_K_input * TTD_K) + (TTD_S_input * TTD_S);      if(Nilai &gt;= 80)         cout &lt;&lt; "Selamat Anda Telah Lulus" &lt;&lt; endl;     else         if(Nilai &lt; 80 &amp;&amp; Nilai &gt; 60)             cout &lt;&lt; "Maaf Anda Mendapatkan Hukuman Ringan" &lt;&lt; endl;         else             cout &lt;&lt; "Maaf Anda Mendapatkan Hukuman Berat" &lt;&lt; endl; }  cin.get();</pre>	<p>C:\ProjectMinGW\TugasModul1-5\Modul3&gt;Prak3-05 Menyatakan Pernyataan Untuk Nilai</p> <hr/> <p>Masukan Jumlah Tanda Tangan Dosen = 5 Masukan Jumlah Tanda Tangan Senior Panitia = 30 Masukan Jumlah Tanda Tangan Karyawan = 30 Masukan Jumlah Tanda Tangan Senior Lainnya = 15 Selamat Anda Telah Lulus</p> <p>C:\ProjectMinGW\TugasModul1-5\Modul3&gt;cd ..</p>
--	--



### Pseudecode

#### Kamus/Deklarasi Variabel

Intenger : TTD\_D, TTD\_SP, TTD\_K, TTD\_S  
TTD\_D\_input, TTD\_SP\_input,  
TTD\_K\_input, TTD\_S\_input, Nilai

#### Dekripsi

1. TTD\_D = 3, TTD\_SP = 2, TTD\_K = 1, TTD\_S = 1  
2. Memasukan/Menginput Nilai TTD\_SP\_input  
3. Memasukan/Menginput Nilai TTD\_K\_input  
4. Memasukan/Menginput Nilai TTD\_SP\_input  
5. Menghitung  
6. Mencetak/Menampilkan Nilai Variable Nilai  
7. Jika Nilai >= 80  
8. Mencetak/Menampilkan "Selamat Anda Telah Lulus"  
9. Jika Nilai < 80 && Nilai > 60  
10. Mencetak/Menampilkan "Maaf Anda Mendapatkan Hukuman Ringan"  
11. Jika lainnya/tidak  
12. Mencetak/Menampilkan "Maaf Anda Mendapatkan Hukuman Berat"  
13. Selesai

### Algoritma

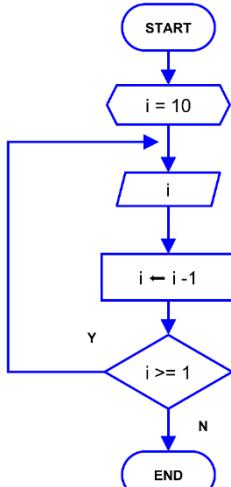
#### 1.4.4 Prak 4 – Repetition (While & do-While)

No. Topik Soal : Prak3-05

Tanggal Pengerjaan : 29/10/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     int i = 10;     do{         cout &lt;&lt; i &lt;&lt; " ";         i--;     }     while (i &gt;= 1);     cout &lt;&lt; endl;     cin.get(); }</pre>	<p>C:\ProjectMinGW\TugasModul1-5\Modul4&gt;Prak4-06 10 9 8 7 6 5 4 3 2 1</p> <p>C:\ProjectMinGW\TugasModul1-5\Modul4&gt;cd ..</p>

## Pseudecode



### Kamus/Deklarasi Variabel

Intenger : i

### Deksripsi

```

i = 20
do
    Print(i)
    i ← i - 1
while (i ≥ 1)

```

## Algoritma

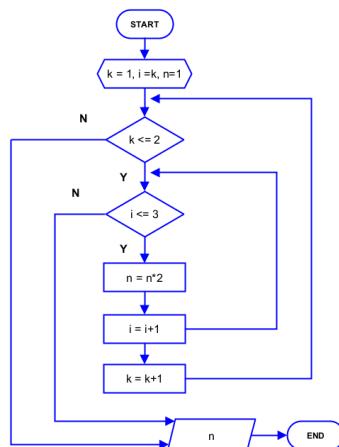
1.  $i = 20$
2. Mencetak/Menampilkan Nilai Variable i
3.  $i \leftarrow i - 1$
4. Selama ( $i \geq 1$ ), maka kerjakan bari 2 s.d 3, kalau tidak kerjain baris 5
5. Selesai

## 1.4.5 Prak 5 – Repetition (For & Nested for)

No. Topik Soal : Prak5-10

Tanggal Pengerjaan : 29/10/2023

Source Code	Hasil Compile
<pre> #include &lt;iostream&gt; using namespace std;  int main(){     int i,k,n;     cout &lt;&lt; " Menampilkan pengulangan 2x(k) dan 3x (i) " &lt;&lt; endl;     cout &lt;&lt; "-----" &lt;&lt; endl;     cout &lt;&lt; endl;     n = 1;     for (k = 1; k &lt;= 2; k++)         for(i = k; i &lt;= 3; i++)             n = n*2;         cout &lt;&lt; "Nilai Akhir dari N adalah " &lt;&lt; n &lt;&lt; endl;     cin.get(); } </pre>	<pre> C:\ProjectMinGW\TugasModul1-5\Modul5&gt;Prak5-10 Menampilkan pengulangan 2x(k) dan 3x (i) ----- Nilai Akhir dari N adalah 32  C:\ProjectMinGW\TugasModul1-5\Modul5&gt;_ </pre>



## Pseudecode

### Kamus/Deklarasi Variabel

Intenger : n=1, k, i

### Deksripsi

```

for(k = 1; k ≤ 2; k++)
    for(i = k; i ≤ 3; i++)
        n = n*2
endfor
endfor
print(n)

```

## Algoritma

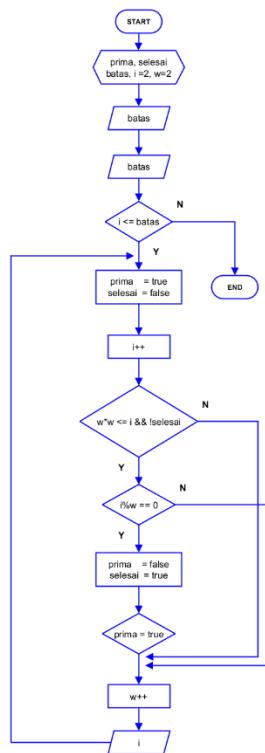
1.  $n \leftarrow 1$
2.  $k \leftarrow 1$
3.  $i \leftarrow k$
4. Selama ( $k \leq 2$ ), maka kerjakan baris 5 s.d 8
5. Selama ( $i \leq 3$ ), maka kerjakan baris 6 s.d 7
6.  $n \leftarrow n * 2$
7.  $i \leftarrow i + 1$
8.  $k \leftarrow k + 1$
9. Mencetak/Menampilkan Nilai Variable n
10. Selesai

## 1.4.6 Prak 6 – Rekursif

No. Topik Soal : Prak6-08

Tanggal Pengerjaan : 7/11/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     bool prima, selesai;     int batas,i,w;     cout&lt;&lt;"Masukan Batas Angka = ";cin&gt;&gt;batas;     cout&lt;&lt;"Menampilkan Bilangan Prima dari 2 sampai "&lt;&lt;batas&lt;&lt;endl;     cout&lt;&lt;"-----&lt;&lt;endl;      for(i = 2; i&lt;=batas; i++){         prima = true;         selesai = false;         for(w = 2; w * w &lt;= i &amp;&amp; !selesai; w++){             if(i%w == 0){                 prima = false;                 selesai = true;             }         }         if(prima){             cout&lt;&lt; i&lt;&lt;" Bilangan prima karena habis dibagi dirinya sendiri " &lt;&lt;i&lt;&lt;endl;         }     }     return 0; }</pre>	<pre>C:\ProjectMinGW\TugasAlgo\modul6&gt;Prak6-08 Masukan Batas Angka = 9 ----- Menampilkan Bilangan Prima dari 2 sampai 9 ----- 2 Bilangan prima karena habis dibagi dirinya sendiri 2 3 Bilangan prima karena habis dibagi dirinya sendiri 3 5 Bilangan prima karena habis dibagi dirinya sendiri 5 7 Bilangan prima karena habis dibagi dirinya sendiri 7 C:\ProjectMinGW\TugasAlgo\modul6&gt;s</pre>



## Pseudocode

### Kamus/Deklarsi Variable

prima, selesai : bool  
batas, i, w : int

### Algoritma/Deksripsi

```

input(batas)
print(batas)
for(i = z; i < batas; i++)
    prima = true
    selesai = false
    for(w = 2; w*w <= i; i++)
        if(i%w == 0)
            prima = false
            selesai = true
        endif
    endfor
    if(prima)
        print(i)
    endif
endfor

```

## Algoritma

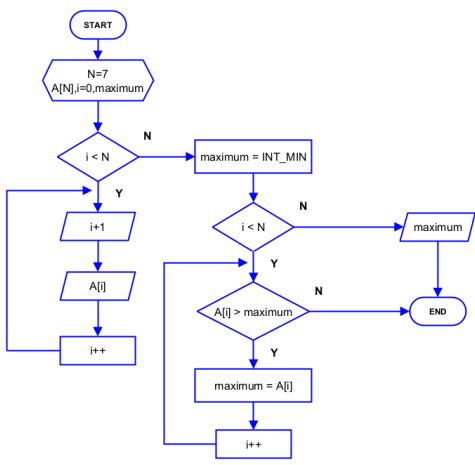
1. const bool prima, selesai
2. const int, batas, i = 2, w = 2
3. Masukan nilai variable batas
4. Mencetak variable batas
5. Selama ( $i \leq \text{batas}$ ), maka kerjakan baris
6. Mengubah nilai variable prima menjadi true
7. Mengubah nilai variable selesai menjadi false
8.  $i++$
9. Selama ( $w^2 \leq i \text{ & } \text{!selesai}$ ), maka kerjakan baris 10
10. Jika ( $i \% w = 0$ ), maka kerjakan baris 11
11. Mengubah nilai variable prima menjadi false
12. Mengubah nilai variable selesai menjadi true
13. jika (prima) bernilai true maka kerjakan baris 14 kalau tidak baris 15
14. Mencetak nilai variable i
15. Selesai

### 1.4.7 Prak 7 – Array (1 Dimensi)

No. Topik Soal : Prak7-10

Tanggal Pengerjaan : 13/11/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; #include &lt;climits&gt; using namespace std;  int main() {     const int N = 7;     int A[N], i, maximum;     for(i=0; i&lt;N; i++){         cout&lt;&lt; "Nilai tes ke-&lt;&lt; i+1 &lt;&lt; " = " ; cin&gt;&gt;A[i];     }     cout&lt;&lt;"======"&lt;&lt;endl;      maximum = INT_MIN;     for(i=0; i&lt;N; i++){         if(A[i] &gt; maximum){             maximum = A[i];         }     }     cout&lt;&lt;"Nilai Terbesar adalah "&lt;&lt;maximum;     return 0; }</pre>	<pre>C:\ProjectMinGW\PrakAlgo\Prak7&gt;Prak7-10 Nilai tes ke-1 = 66 Nilai tes ke-2 = 63 Nilai tes ke-3 = 70 Nilai tes ke-4 = 50 Nilai tes ke-5 = 66 Nilai tes ke-6 = 98 Nilai tes ke-7 = 45 ===== Nilai Terbesar adalah 98 C:\ProjectMinGW\PrakAlgo\Prak7&gt;S</pre>



#### Pseudocode

##### Kamus/Deklarsi Variabel

N = 7, A[N], i, i=0, maximum : int

##### Algoritma/Dekripsi

```

N = 7
for(i = 0; i < N; i++)
    print(i+1)
    input(A[i])
endfor
maximum = INT_MIN
for(i = 0; i < N; i++)
    if(A[i] > maximum)
        maximum = A[i]
    endif
endfor
print(maximum)

```

#### Algoritma

1. const N = 7
2. i = 0
3. Selama (i < N), maka kerjakan baris 4 s.d 6 jika tidak baris 7
4. Menampilkan variable i+1
5. Menampilkan variable array A[i]
6. i++
7. Menginisialisasi maximum = INT\_MIN
8. i = 0
9. Selama (i < N), maka kerjakan baris 9 s.d 12 jika tidak 13
10. Jika (A[i] > maximum), maka kerjakan baris 11 jika tidak baris 12
11. Menginisialisasi maximum = A[i]
12. i++
13. Menampilkan variable maximum
14. Selesai

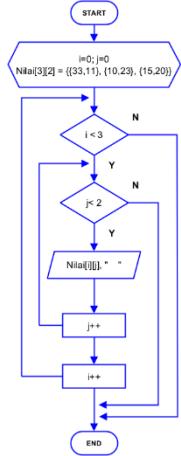
### 1.4.8 Prak 8 – Array (2 MultiDimensi)

No. Topik Soal : P8-25

Tanggal Pengerjaan : 20/11/2023

Source Code	Hasil Compile

<pre>#include &lt;iostream&gt; using namespace std;  int main(){     int Nilai[3][2] = {{33,11}, {10,23}, {15,20}};     int i, j;     cout&lt;&lt; endl;     cout&lt;&lt; "Menampilkan Array dengan ordo 3x2" &lt;&lt; endl;     cout&lt;&lt; "-----" &lt;&lt; endl;     for(i=0; i&lt;3; i++){         for(j=0; j&lt;2; j++)             cout&lt;&lt; Nilai[i][j] &lt;&lt; " ";         cout&lt;&lt; endl;     }     cin.get(); }</pre>	<p>C:\ProjectMinGW\PrakAlgo\Prak8&gt;P08-25</p> <p>Menampilkan Array dengan ordo 3x2</p> <p>-----</p> <p>33 11 10 23 15 20</p>
--	--



## Pseudocode

### Kamus/Deklarasi Variable

Nilai[3][2] : int

### Algoritma/Dekripsi

Nilai[3][2] = {{33,11}, {10,23}, {15,20}}

```
i = 0
j = 0

for(i=0; i<3; i++)
    for(j=0; j<2; j++)
        print(Nilai[i][j], " ")
    endfor
endfor
```

## Algoritma

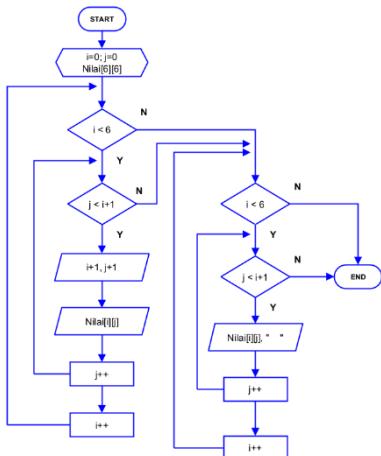
1. const i = 0
2. const j = 0
3. Nilai[3][2] = {33,11} ; {10,23} ; {15,20}
4. Selama (i < 3), maka kerjakan baris jika tidak 10
5. Selama (j < 2), maka kerjakan baris jika tidak 10
6. Mencetak variable Nilai[i][j]
7. Mencetak (" ")
8. j++
9. i++
10. Selesai

No. Topik Soal : P8-55

Tanggal Pengerjaan : 20/11/2023

Source Code	Hasil Compile
-------------	---------------

<pre>#include &lt;iostream&gt; using namespace std;  int main() {     char Nilai[6][6];     char i,j;     cout&lt;&lt; "Menampilkan Array ordo 6x6 dengan inputan" &lt;&lt; endl;     for(i=0; i&lt;6; i++){         for(j=0; j&lt;i+1; j++){             cout&lt;&lt; "Masukan Isi Eleme["&lt;&lt;i+1&lt;&lt;"] ["&lt;&lt;j+1&lt;&lt;"] : ";             cin &gt;&gt; Nilai[i][j];         }     }     cout&lt;&lt;endl;     cout&lt;&lt; "Tampilkan Isi Elemen Array" &lt;&lt; endl;     for(i=0; i&lt;6; i++){         for(j=0; j&lt;i+1; j++)             cout &lt;&lt; Nilai[i][j] &lt;&lt; " ";         cout&lt;&lt; endl;     }     cin.get(); }</pre>	C:\ProjectMinGW\PrakAlgo\Prak8>P8-55 Menampilkan Array ordo 6x6 dengan inputan Masukan Isi Eleme[1][1] : A Masukan Isi Eleme[2][1] : B Masukan Isi Eleme[2][2] : C Masukan Isi Eleme[3][1] : K Masukan Isi Eleme[3][2] : L Masukan Isi Eleme[3][3] : O Masukan Isi Eleme[4][1] : P Masukan Isi Eleme[4][2] : M Masukan Isi Eleme[4][3] : N Masukan Isi Eleme[4][4] : S Masukan Isi Eleme[5][1] : D Masukan Isi Eleme[5][2] : C Masukan Isi Eleme[5][3] : E Masukan Isi Eleme[5][4] : R Masukan Isi Eleme[5][5] : V Masukan Isi Eleme[6][1] : T Masukan Isi Eleme[6][2] : Y Masukan Isi Eleme[6][3] : U Masukan Isi Eleme[6][4] : N Masukan Isi Eleme[6][5] : J Masukan Isi Eleme[6][6] : F  Tampilkan Isi Elemen Array A B C K L O P M N S D C E R V T Y U N J F
---	--



## Pseudocode

**Kamus/Deklarsi Variable**  
Nilai[6][6], i, j : char

**Algoritma/Deksripsi**

```

for(i=0; i<6; i++)
    for(j=0; j<i+1; j++)
        print(i+1, j+1)
        input(Nilai[i][j])
    endfor
endfor
for(i=0; i<6; i++)
    for(j=0; j<i+1; j++)
        print(Nilai[i][j], " ")
    endfor
endfor

```

## Algoritma

- const i = 0
- const j = 0
- Selama (i < 6), maka kerjakan baris 4 s.d 9 jika tidak 10
- Selama (j < i+1), maka kerjakan baris 5 s.d 8 jika tidak 10
- Mencetak variable i+1
- Mencetak variable j+1
- Memasukan nilai variable Nilai[i][j]
- j++
- i++
- Selama (i < 6), maka kerjakan baris 10 s.d 15 jika tidak 16
- Selama (j < i+1), maka kerjakan baris 12 s.d 14 jika tidak 16
- Mencetak nilai variable Nilai[i][j]
- Mencetak (" ")
- j++
- i++
- Selesai

## 1.4.9 Prak 9 – Function

No. Topik Soal : P9-16

Tanggal Pengerjaan : 22/11/2023

Source Code	Hasil Compile
-------------	---------------

<pre>#include &lt;iostream&gt; using namespace std;  double opmat16(double d, double e);  int main(){     cout&lt;&lt;endl;     cout&lt;&lt;"Hasil Jumlah Kali Variable d dan e : "&lt;&lt;opmat16(0.15,0.18);     cin.get(); }  double opmat16(double d, double e){     cout&lt;&lt;endl;     return(d*e); }</pre>	C:\ProjectMinGW\PrakAlgo\Prak9>P09-16  Hasil Jumlah Kali Variable d dan e : 0.027
---	---

## Psuedocode

**Kamus/Deklarasi Variable Function opmat16**  
d,e : double

**Algoritma/Deksripsi Function opmat16**  
d\*e

**Kamus/Deklarasi Variable**

**Algoritma/Deksripsi**  
opmat16(0.15,0.18)

## Algoritma

1. Membuat Function opmat16()
2. Menghitung d\*e
3. Memanggil function opmat16(0.15, 0.18)
4. Selesai

## 1.4.10 Prak 10 – String

No. Topik Soal : P10-04

Tanggal Pengerjaan : 27/11/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; #include &lt;string.h&gt; using namespace std;  int main() {     char tekniki1[] = "TEKNIK INFORMATIKA FALKUTAS TEKNIK UNIVERSITAS PANCASILA";     cout&lt;&lt;"Kalimat Asli = "&lt;&lt;tekniki1&lt;&lt;endl;     cout&lt;&lt;"Mencari karakter C pada kalimat asli = "&lt;&lt;strchr(tekniki1,'C')&lt;&lt;endl;     cin.get(); }</pre>	C:\ProjectMinGW\PrakAlgo\Prak10>P10-04 Kalimat Asli = TEKNIK INFORMATIKA FALKUTAS TEKNIK UNIVERSITAS PANCASILA Mencari karakter C pada kalimat asli = CASILA

## Psuedocode

**Kamus/Deklarasi Variable**  
tekniki1[] : char

**Algoritma/Deksripsi**

```
tekniki1[] = "TEKNIK INFORMATIKA FALKUTAS
TEKNIK UNIVERSITAS PANCASILA"
print(tekniki1)
print(strchr(tekniki1,'C'))
```

## Algoritma

1. tekniki1[] = "TEKNIK INFORMATIKA FALKUTAS TEKNIK UNIVERSITAS PANCASILA"
2. Mencetak teknik1
3. Mencari character 'C' pada teknik1 atau strchr(teknik1,'C')
4. Mencetak hasil teknik1
5. Selesai

## 1.4.11 Prak 11 – Sort (SelectionSort & InsertionSort)

No. Topik Soal : P11-09

Tanggal Pengerjaan : 4/12/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; #include &lt;string.h&gt; #define h 9 using namespace std;  int main() {     int z,t,r,s;     int ra[h] = {8, 2, 3, 20, 0, 12, 4, 5, 7};      cout &lt;&lt; "Sebelum Dilakukan Pengurutan" &lt;&lt; endl;     cout &lt;&lt; "-----" &lt;&lt; endl;     for(z=0; z&lt;=h-1; z++)         cout &lt;&lt; ra[z] &lt;&lt; " ";     cout &lt;&lt; endl;     cout &lt;&lt; endl;      cout &lt;&lt; "Step By Step Dilakukan Pengurutan" &lt;&lt; endl;     cout &lt;&lt; "-----" &lt;&lt; endl;     z=0;     while(z&lt;=h-2) {         t=z;         z=z+1;         while(z&lt;=h-1)             if(ra[z] &lt; ra[t])                 t=z;         cout &lt;&lt; ra[z] &lt;&lt; " ";         z++;         cout &lt;&lt; endl;         s = ra[t];         ra[t] = ra[z];         ra[z] = s;         r++;         cout &lt;&lt; endl; cout &lt;&lt; endl;         cout &lt;&lt; endl; cout &lt;&lt; endl;     }     cout &lt;&lt; "Sesudah Dilakukan Pengurutan" &lt;&lt; endl;     cout &lt;&lt; "-----" &lt;&lt; endl;     for(z=0; z&lt;=h-1; z++)         cout &lt;&lt; ra[z] &lt;&lt; " ";     cin.get(); }</pre>	<pre>C:\ProjectMinGW\PrakAlgo\Prak11&gt;P11-09 Sebelum Dilakukan Pengurutan ----- 8 2 3 20 0 12 4 5 7  Step By Step Dilakukan Pengurutan ----- 2 3 20 0 12 4 5 7 3 20 8 12 4 5 7 20 8 12 4 5 7 8 12 4 5 7 12 20 5 7 20 8 7 8 12 12  Sesudah Dilakukan Pengurutan ----- 0 2 3 4 5 7 8 12 20</pre>

### Psuedocode

#### Kamus/Deklarasi Variable

ra[h], z, h, t, r, s : int

#### Algoritma/Dekripsi

```
ra[h] = {23,17,14,6,13,10,1,5,7}
for(z=0; z<=h-1; z++)
    print(ra[z]," ")
endfor
r=0
while(r <= h-1)
    t=r
    z=r+1
    while(z<=h-1)
        if(ra[z] < ra[t])
            t=z
        print( ra[z], " ")
    endif
    z++
endwhile
s = ra[t]
ra[t] = ra[r]
ra[r] = s
r++
endwhile
for(z=0; z<=h-1; z++)
    print(ra[z]," ")
endfor
```

### Algoritma

1. Const ra[h] = {8, 2, 3, 20, 0, 12, 4, 5, 7}
2. z=0
3. Selama (z <= h-1), maka kerjakan baris 4 s.d 6 kalau tidak baris 7
4. Mencetak Isi variable ra[h]
5. Mencetak (" ")
6. z++
7. r=0
8. Selama (r <= h-1), maka kerjakan baris 9 s.d 20 kalau tidak baris 21
9. t=r
10. z=r+1
11. Selama (z <= h-1), maka kerjakan baris 12 s.d 16 kalau tidak baris 17
12. jika (ra[z] < ra[t]), maka kerjakan baris 13 s.d 15 kalau tidak baris 16
13. t=z
14. Mencetak Isi variable ra[z]
15. Mencetak (" ")
16. z++
17. s = ra[t]
18. ra[t] = ra[r]
19. ra[r] = s
20. r++
21. z=0
22. Selama (z <= h-1), maka kerjakan baris 23 s.d 25 kalau tidak baris 26
23. Mencetak Isi variable ra[h]
24. Mencetak (" ")
25. z++
26. Selesai

**MANUAL : TUGAS\_P11-09 (Selection-Ascending)**

<b>Awal</b>	<b>8</b>	<b>2</b>	<b>3</b>	<b>20</b>	<b>0</b>	<b>12</b>	<b>4</b>	<b>5</b>	<b>7</b>
<b>z = 1</b>	<b>8</b>	<b>2</b>	<b>3</b>	<b>20</b>	<b>0</b>	<b>12</b>	<b>4</b>	<b>5</b>	<b>7</b>
<b>z = 2</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>20</b>	<b>8</b>	<b>12</b>	<b>4</b>	<b>5</b>	<b>7</b>
<b>z = 3</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>20</b>	<b>8</b>	<b>12</b>	<b>4</b>	<b>5</b>	<b>7</b>
<b>z = 4</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>20</b>	<b>8</b>	<b>12</b>	<b>4</b>	<b>5</b>	<b>7</b>
<b>z = 5</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>8</b>	<b>12</b>	<b>20</b>	<b>5</b>	<b>7</b>
<b>z = 6</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>12</b>	<b>20</b>	<b>8</b>	<b>7</b>
<b>z = 7</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>7</b>	<b>20</b>	<b>8</b>	<b>12</b>
<b>z = 8</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>7</b>	<b>8</b>	<b>20</b>	<b>12</b>
<b>Akhir</b>	<b>0</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>7</b>	<b>8</b>	<b>12</b>	<b>20</b>

No. Topik Soal : P11-33

Tanggal Pengerjaan : 4/12/2023

Source Code	Hasil Compile

```

#include <iostream>
#include <string.h>
#define h 9
using namespace std;

int main() {
    int z,t,r,i;
    char ra[h];

    cout<<"PENGURUTAN INSERTION SECARA ASCENDING"<<endl;
    cout << "-----" << endl;
    cout<<"Masukan Kumpulan Huruf Acak sampai 9x"<<endl;
    for(z=0; z<h; z++){
        cout<<"Masukan Huruf ke-<<z+1<<: "; cin>>ra[z];
    }

    cout << "\n-----" << endl;
    cout<<"Sebelum diurutkan"<<endl;
    cout << "-----" << endl;
    for(z=0; z<=h-1; z++){
        cout<<ra[z]<< " ";
    }

    cout<<endl;
    cout << "\n-----" << endl;
    cout<<"Proses Pengurutan"<<endl;
    cout << "-----" << endl;
    for(t=1; t<h; t++){
        z=t;
        r=ra[z];

        while(z>0 && ra[z-1]>r){
            ra[z] = ra[z-1];
            z--;
        }
        ra[z] = r;

        cout <<"sort ke-" << t << ": ";
        for (i = 0; i < h; i++)
            cout << ra[i] << " ";
        cout<<endl;
    }

    cout<<endl;
    cout << "-----" << endl;
    cout<<"Sesudah diurutkan"<<endl;
    cout << "-----" << endl;
    for(z=0; z<=h-1 ;z++)
        cout<<ra[z]<< " ";
    cout<<endl;
    cin.get();
}

```

```

C:\ProjectMinGW\PrakAlgo\Prak11>P11-09
PENGURUTAN INSERTION SECARA ASCENDING
-----
Masukan Kumpulan Huruf Acak sampai 9x
Masukan Huruf ke-1: H
Masukan Huruf ke-2: L
Masukan Huruf ke-3: U
Masukan Huruf ke-4: E
Masukan Huruf ke-5: W
Masukan Huruf ke-6: C
Masukan Huruf ke-7: A
Masukan Huruf ke-8: S
Masukan Huruf ke-9: M
-----
Sebelum diurutkan
-----
H L U E W C A S M
-----
Proses Pengurutan
-----
sort ke-1: H L U E W C A S M
sort ke-2: H L U E W C A S M
sort ke-3: E H L U W C A S M
sort ke-4: E H L U W C A S M
sort ke-5: C E H L U W A S M
sort ke-6: A C E H L U W S M
sort ke-7: A C E H L S U W M
sort ke-8: A C E H L M S U W
-----
Sesudah diurutkan
-----
A C E H L M S U W

```

## Psuedocode

### Kamus/Deklarasi Variable

ra[h], z, h, t, r, s : int

### Algoritma/Dekripsi

```

ra[h] = {23,17,14,6,13,10,1,5,7}
for(z=0; z<=h-1; z++)
    print(ra[z]," ")
endfor
r=0
while(r <= h-1)
    t=r
    z=r+1
    while(z<=h-1)
        if(ra[z] < ra[t])
            t=z
            print( ra[z], " ")
        endif
        z++
    endwhile
    s = ra[t]
    ra[t] = ra[r]
    ra[r] = s
    r++
endwhile
for(z=0; z<=h-1; z++)
    print(ra[z]," ")
endfor

```

## Algoritma

1. Const ra[h] = {8, 2, 3, 20, 0, 12, 4, 5, 7}
2. z=0
3. Selama (z <= h-1), maka kerjakan baris 4 s.d 6 kalau tidak  
baris 7
4. Mencetak Isi variable ra[h]
5. Mencetak (" ")
6. z++
7. r=0
8. Selama (r <= h-1), maka kerjakan baris 9 s.d 20 kalau tidak  
baris 21
9. t=r
10. z=r+1
11. Selama (z<=h-1), maka kerjakan baris 12 s.d 16 kalau tidak  
baris 17
12. Jika (ra[z] < ra[t]), maka kerjakan baris 13 s.d 15 kalau tidak  
baris 16
13. t=z
14. Mencetak Isi variable ra[z]
15. Mencetak (" ")
16. z++
17. s = ra[t]
18. ra[t] = ra[r]
19. ra[r] = s
20. r++
21. z=0
22. Selama (z <= h-1), maka kerjakan baris 23 s.d 25 kalau  
tidak baris 26
23. Mencetak Isi variable ra[h]
24. Mencetak (" ")
25. z++
26. Selesai

## MANUAL : TUGAS\_P11-33 (Insertion-Ascending)

Awal	H	L	U	E	W	C	A	S	M
<b>z = 2</b>	H	L	U	E	W	C	A	S	M
<b>z = 3</b>	H	L	U	E	W	C	A	S	M
<b>z = 4</b>	H	L	U	E	W	C	A	S	M
<b>z = 5</b>	E	H	L	U	W	C	A	S	M
<b>z = 6</b>	E	H	L	U	W	C	A	S	M
<b>z = 7</b>	C	E	H	L	U	W	A	S	M
<b>z = 8</b>	A	C	E	H	L	U	W	S	M
<b>z = 9</b>	A	C	E	H	L	S	U	W	M
<b>Akhir</b>	A	C	E	H	L	M	S	U	W

## 1.4.12 Prak 12 – Pointer

No. Topik Soal : P12-04

Tanggal Pengerjaan : 11/12/2023

### Source Code

```
#include <iostream>
#include <iomanip> //untuk setw
#include <string>
using namespace std;

struct Pelanggan {
    //menyimpan struktur data
    char namabar[50], code[10];
    float harga;
    int jumlah, qty;
};

int main(){
    //Deklarasi seluruh variable
    int z, t;
    int totalBarang = 0, totalHarga = 0;
    char nama[50], alamat[100], tlp[15];
    cout<<"-----"
    cout<<"                                     FROM PEMBELIAN BARANG"
    cout<<"-----"
    cout<<"Masukan Nama Anda                 = ";gets(nama);
    cout<<"Masukan Alamat Anda                = ";gets(alamat);
    cout<<"Masukan Nomor Telepon Anda        = ";cin>>tlp;
    cout<<"Masukan Jumlah Jenis Barang Yang Ingin Dibeli = ";cin>>z; //array inputan
    cout<<endl;

    Pelanggan pel[z]; //Deklarasi data Array
    Pelanggan *data = pel; /*data sebagai pointer
```

```

//Proses Input Data sesuai array inputan
for(t=0; t<z; t++){
    cout<<"\n#Data Barang ke-"<<t+1<<endl;
    cout<<"Masukan Kode Barang      = ";cin>>(data+t)->code;cin.ignore();
    cout<<"Masukan Nama Barang     = ";gets((data+t)->nabar);
    cout<<"Masukan Harga Barang    = ";cin>>(data+t)->harga;
    cout<<"Masukan Quantity Barang = ";cin>>(data+t)->qty;
    cout<<"Masukan Jumlah Barang   = ";cin>>(data+t)->jumlah;
}

cout<<-----
cout<<"                               BUKTI PEMESANAN BARANG
cout<<-----

cout<<"Nama Pelanggan          : "<<nama<<endl;
cout<<"Alamat Pelanggan        : "<<alamat<<endl;
cout<<"Nomor Telepon Pelanggan : "<<tlp<<endl;
cout<<"Barang Yang Dibeli       : "<<endl<<endl;
cout << setfill('-') << setw(92) << "-" << endl; // garis pembatas otomatis
cout << setfill(' '); // mengembalikan karakter pengisi ke spasi
//header judul diratakan dengan setw untuk merapihkan tabel
cout << "| " << setw(4) << left << "No"
    << "| " << setw(15) << left << "Kode"
    << "| " << setw(15) << left << "Nama Barang"
    << "| " << setw(15) << left << "Harga"
    << "| " << setw(15) << left << "Quantity"
    << "| " << setw(15) << left << "Jumlah" << "|" << endl;
cout << setfill('-') << setw(92) << "-" << endl; // garis pembatas otomatis
cout << setfill(' ');

//Proses Menampilkan Data yang tersimpan di struct Pelanggan sesuai dengan array
for (t = 0; t < z; t++) {
    cout << "| " << setw(4) << left << t+
        << "| " << setw(15) << left << (data + t)->code
        << "| " << setw(15) << left << (data + t)->nabar
        << "| " << setw(15) << left << (data + t)->harga
        << "| " << setw(15) << left << (data + t)->qty
        << "| " << setw(15) << left << (data + t)->jumlah << "|" << endl;
    totalBarang += (data+t)->jumlah; //Menjumlahkan seluruh barang
    totalHarga += (data+t)->jumlah * (data+t)->harga; //Proses mencari total harga
}

cout << setfill('-') << setw(92) << "-" << endl; // garis pembatas otomatis
cout << setfill(' ');
cout<<"Jumlah Barang Yang Dibeli = "<<totalBarang<<endl;
cout<<"Total Pembayaran         = "<<"Rp."<<totalHarga<<endl;
cout<<-----
cin.get();
}

```

## Hasil Compile

```
C:\ProjectMinGW\PrakAlgo\Prak12>P12-04
-----
          FROM PEMBELIAN BARANG
-----
Masukan Nama Anda      = Zahra Tsabitah
Masukan Alamat Anda    = Jl PerezUi
Masukan Nomor Telepon Anda = 0893973444
Masukan Jumlah Jenis Barang Yang Ingin Dibeli = 4

#Data Barang ke-1
Masukan Kode Barang    = gudguW
Masukan Nama Barang    = Pin Killua
Masukan Harga Barang   = 12000
Masukan Quantity Barang = 6
Masukan Jumlah Barang   = 2

#Data Barang ke-2
Masukan Kode Barang    = hu*ushi
Masukan Nama Barang    = Hiasan Kerang
Masukan Harga Barang   = 43000
Masukan Quantity Barang = 25
Masukan Jumlah Barang   = 4

#Data Barang ke-3
Masukan Kode Barang    = jhbu#uhii-H
Masukan Nama Barang    = Slayer Midnight
Masukan Harga Barang   = 75000
Masukan Quantity Barang = 4
Masukan Jumlah Barang   = 1

#Data Barang ke-4
Masukan Kode Barang    = Ojddej
Masukan Nama Barang    = PhotoCard
Masukan Harga Barang   = 8000
```

### BUKTI PEMESANAN BARANG

```
Nama Pelanggan : Zahra Tsabitah
Alamat Pelanggan : Jl PerezUi
Nomor Telepon Pelanggan : 0893973444
Barang Yang Dibeli :
```

No	Kode	Nama Barang	Harga	Quantity	Jumlah
1	gudguW	Pin Killua	12000	6	2
2	hu*ushi	Hiasan Kerang	43000	25	4
3	jhbu#uhii-H	Slayer Midnight	75000	4	1
4	Ojddej	PhotoCard	8000	56	8

```
Jumlah Barang Yang Dibeli = 15
Total Pembayaran = Rp.335000
```

## Psuedocode

### Kamus/Deklarasi Struct Pelanggan

```
namabar[50], code[10] : char  
harga : float  
jumlah, qty : int
```

### Algoritma/Deksripsi Struct Pelanggan

### Kamus/Deklarasi Variable

```
z, t : int  
totalBarang = 0, totalHarga = 0 : int  
nama[50], alamat[100], tlp[15] : char
```

### Algoritma/Deksripsi

```
input(nama)  
input(alamat)  
input(tlp)  
input(z)  
Pelanggan pel[z]  
Pelanggan *data = pel  
for(t=0; t<z; t++)  
    print(t+1)  
    input((data+t)->code)  
    input((data+t)->nabar)  
    input((data+t)->harga)  
    input((data+t)->qty)  
    input((data+t)->jumlah)  
endfor  
print(nama)  
print(alamat)  
print(tlp)  
for(t=0; t<z; t++)  
    print(t+1)  
    print((data+t)->code)  
    print((data+t)->nabar)  
    print((data+t)->harga)  
    print((data+t)->qty)  
    print((data+t)->jumlah)  
    totalBarang += (data+t)->jumlah  
    totalHarga += (data+t)->jumlah * (data+t)->harga  
endfor  
print(totalBarang)  
print(totalHarga)
```

## Algoritma

1. Memasukan nilai nama
2. Memasukan nilai alamat
3. Memasukan nilai tlp
4. Memasukan nilai z
5. Pelanggan pel[z]
6. Pelanggan \*data = pel
7. t=0
8. Jika (t<z), maka kerjakan baris 9 s.d 15 jika tidak baris 16
9. Mencetak nilai (t+1)
10. Memasukan nilai (data+t)->code
11. Memasukan nilai (data+t)->nabar
12. Memasukan nilai (data+t)->harga
13. Memasukan nilai (data+t)->qty
14. Memasukan nilai (data+t)->jumlah
15. t++
16. Mencetak nilai nama
17. Mencetak nilai alamat
18. Mencetak nilai tlp
19. t=0
20. Jika (t<z), maka kerjakan baris 21 s.d 29 jika tidak baris 30
21. Mencetak nilai (t+1)
22. Mencetak nilai (data+t)->code
23. Mencetak nilai (data+t)->nabar
24. Mencetak nilai (data+t)->harga
25. Mencetak nilai (data+t)->qty
26. Mencetak nilai (data+t)->jumlah
27. totalBarang += (data+t)->jumlah
28. totalHarga += (data+t)->jumlah \* (data+t)->harga
29. t++
30. Mencetak nilai totalBarang
31. Mencetak nilai totalHarga
32. Selesai

## 1.4.13 Prak 13 – File (Biner/Teks)

No. Topik Soal : P13-01

Tanggal Pengerjaan : 18/12/2023

Source Code	Hasil Compile
<pre>#include &lt;iostream&gt; #include &lt;fstream&gt; using namespace std;  int main(){     //Deklarasi variable buffer dengan isi array     const int MAX=25; //jumlah max karakter perbaris     char buffer[MAX+1]; //tempat tampung array      //Deklarasi Objek     ifstream file_output("NILAI1.txt"); //Membaca file      //Proses Menampilkan Isi File dengan CMD     while(file_output){         //masukan karakter baris ke array         file_output.getline(buffer,MAX);         //keluarkan penyimpanan         cout &lt;&lt; buffer &lt;&lt; endl;     }      file_output.close(); // Tutup File }</pre>	<pre>C:\ProjectMinGW\PrakAlgo\Prak13&gt;P13-01 Nilai A : 85-100 Nilai A- : 80-85 Nilai B+ : 76-80 Nilai B : 72-76 Nilai B- : 68-72 Nilai C+ : 64-68 Nilai C : 56-64 Nilai D : 45-56 Nilai E : 30-45  C:\ProjectMinGW\PrakAlgo\Prak13&gt;type NILAI1.txt Nilai A : 85-100 Nilai A- : 80-85 Nilai B+ : 76-80 Nilai B : 72-76 Nilai B- : 68-72 Nilai C+ : 64-68 Nilai C : 56-64 Nilai D : 45-56 Nilai E : 30-45</pre>

## Psuedocode

### Kamus/Deklarasi Variable

MAX : int  
buffer[MAX+1] : char

### Algoritma/Deksripsi

```
MAX = 25
ifstream file_output(NILAI1.txt)
while(file_output)
    file_output.getline(buffer, MAX)
    print(buffer)
endwhile
file_output.close()
```

## Algoritma

1. MAX = 25 (Definisi variable untuk karakter yang diinginkan)
2. Definisikan buffer atau penyangganya
3. Memanggil file teks yang telah dibuat (ifstream file\_output("NILAI1.txt"))
4. Jika (file\_output) kerjakan baris 5 s.d 6
5. Membaca sejumlah karakter sebanyak MAX (file\_output.getline(buffer, MAX))
6. Mencetak buffer
7. Menutup file (file\_output.close())
8. Selesai

## BAB II

## PEMBAHASAN

### 2.1 Tugas Besar

Soal : SI Pembayaran PAM (TanpaInputan)

Nomor Soal : 54

Materi :

- Selection/Seleksi
- Repetition/Pengulangan
- Array
- Function
- String/File
- Selection Sort-Ascending

### 2.2 Pembahasan Tugas Besar

#### 2.2.1 Program/Source Code

```
/*TUGAS BESAR: Algoritma dan Pemrograman A(Group2)
Program - SI Pembayaran PAM (TanpaInputan)
          |           | (SelectionSort-Ascending)
Zahra - 4523210145
*/
#include <iostream> //library cin/cout
#include <fstream> //library file
#include <iomanip> //library stew
#include <windows.h> //library sistem windows
#include <string.h> //library string
#define JUMLAH_BULAN 12 //nyatakan variable penyimpan array bernilai 12
using namespace std;

void lihatRekapTahunLalu() {
    int R, t, s = 0, h, z;
    int totalTagihanPemakaiAir = 0,
        totalTerlambat = 0;
    int bulan[JUMLAH_BULAN] = {12,3,4,5,6,8,2,11,10,7,1,9},
        pakaiAir[JUMLAH_BULAN] = {4,5,6,7,8,42,4,10,49,34,11,20};
    int tagihanAir[JUMLAH_BULAN], tagihanPakaiAir[JUMLAH_BULAN];
    char bayar[JUMLAH_BULAN][10] = {"CASH","BCN","BRI","CASH", //menyimpan data pada array 2 dimensi
                                     "BRI","CASH","BCN","CASH",
                                     "BCN","BCN","BRI","CASH");
    string tanggal[JUMLAH_BULAN] = {"04-01-2023","29-02-2023","17-03-2023",
                                     "05-04-2023",
                                     "18-05-2023","20-06-2023","01-07-2023","03-08-2023",
                                     "30-09-2023","26-10-2023","27-11-2023","10-12-2023"};
    string status[JUMLAH_BULAN] = {"TEPAT WAKTU","TERLAMBAT(!)","TEPAT WAKTU","TEPAT WAKTU",
                                    "TEPAT WAKTU","TEPAT WAKTU","TEPAT WAKTU","TEPAT WAKTU",
                                    "TERLAMBAT(!)","TEPAT WAKTU","TERLAMBAT(!)","TEPAT WAKTU"};

    /*Sistem proses SelectionSort-Ascending untuk mengurutkan
    data berdasarkan variable array*/
    while (s < JUMLAH_BULAN - 1) {
        t = s;
        R = s + 1;
        while (R < JUMLAH_BULAN) {
            if (bulan[R] < bulan[t])
                t = R;
            R++;
        }

        h = bulan[t];
        bulan[t] = bulan[s];
        bulan[s] = h;
        s++;
    }
}
```

```

//Menampilkan rekapan pembayaran tahun lalu
cout<<"----- REKAPAN PEMBAYARAN PAM TAHUN 2023 -----"
cout<<""
cout<<""-----><<en
cout<<"Nama Pelanggan : ZAHRA TSABITAH"<<endl;
cout<<"Alamat Pelanggan : JL. PAM BARU RAYA NO.7A"<<endl;
cout<<"ID Pelanggan : 45-23-2-101-45"<<endl;
cout<<"Jenis Golongan : Rumah Tangga"<<endl;
//menampilkan header
cout << setfill('-') << setw(124) << "-" << endl; // garis pembatas otomatis
cout << setfill(' ') // mengambilkan karakter pengisi ke spasi
//header judul diratakan dengan setw untuk merapikan tabel
cout << " | " << setw(6) << left << "Bulan"
    << " | " << setw(15) << left << "Pemakaian Air"
    << " | " << setw(12) << left << "Tagihan Air"
    << " | " << setw(22) << left << "Tagihan Pemakaian Air"
    << " | " << setw(15) << left << "Tanggal Pembayaran"
    << " | " << setw(10) << left << "Status Pembayaran"
    << " | " << setw(17) << left << "Metode Pembayaran" << "|" << endl;
cout << setfill('-') << setw(124) << "-" << endl; // garis pembatas otomatis
cout << setfill(' ') ;

/*Sistem proses menampilkan data berdasarkan array looping
variable z sebagai patokan*/
for (z = 0; z < JUMLAH_BULAN; z++) {
    //sistem menghitung biaya
    if(pakaiAir[z] <= 35){
        tagihanAir[z] = 7450;
        tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] + 1195 + 3000;
    } else {
        tagihanAir[z] = 8250;
        tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] + 2400 + 3000;
    }

    /* - Menambahkan biaya denda jika terlambat
    - menghitung berapa kali terlambat*/
    if(status[z] == "TERLAMBAT()"){
        tagihanPakaiAir[z] += 5000;
        totalTerlambat++;
    }

    cout << setw(4) << " | " << setw(4) << bulan[z]
    //menggunakan fungsi string untuk mengubah tipedata menjadi string
    << setw(7) << " | " << setw(10) << to_string(pakaiAir[z]) + "Rp"
    << setw(8) << " | " << "Rp" << setw(8) << tagihanAir[z]
    << setw(8) << " | " << "Rp" << setw(12) << tagihanPakaiAir[z]
    << setw(8) << " | " << setw(15) << tanggal[z]
    << setw(8) << " | " << setw(10) << status[z]
    << setw(7) << " | " << setw(12) << bayar[z] << "|" << endl;

    //hitung total tagihan pemakaian air dengan operator penugasan yang bertugas
    //untuk Pengisian dan Penambahan variable dinyatakan dengan symbol += */
    totalTagihanPemakaianAir += tagihanPakaiAir[z];
}

cout << setfill('-') << setw(124) << "-" << endl; // garis pembatas otomatis
//menampilkan footer
cout << ""-----><<en
cout << " TOTAL TAGIHAN PEMAKAIAN AIR 1.THN : " << "Rp" << totalTagihanPemakaianAir << endl;
cout << " TOTAL KETERLAMBATAN PEMBAYARAN : " << totalTerlambat << " kali" << endl;
cout << setfill('-') << setw(124) << "-" << endl; // garis pembatas otomatis
cout << setfill(' ') ;

}

void PembayaranBulanIni(){
const char *bayar[] = {"BRI", "BCA", "CASH"}; //deklarasi array dari pointer ke kontanta string
char nama[50] = "Zahra Tsabitah", //Menyimpan data pada array 1 dimensi
alamat[50] = "Jl. Pam Baru Raya No.7a",
ID[50] = "45-23-2-101-45",
jenis[50] = "Rumah Tangga",
tanggal[50] = "2 Januari 2024",
code[35];

int pakaiAir = 34,
tagihanAir,
tagihanPakaiAir,
metode,
panjang_array;

strand(time());
panjang_array = sizeof(bayar) / sizeof(bayar[0]); //menghitung panjang character array
metode = rand() % panjang_array; //menghasilkan kata random pada data yang ada di dalam variable bayar

cout<<"----- PEMBAYARAN PAM BULANINI -----"
cout<<""
cout<<""-----><<en
cout<<"Nama Pelanggan : ...<<endl;
Sleep(2000):cout << "\x1b[1A\x1b[2K";
cout<<"Nama Pelanggan : "<<strupr(nama)<<endl; //string untuk huruf besar semua
cout<<"Alamat Pelanggan : ...<<endl;
Sleep(2000):cout << "\x1b[1A\x1b[2K";
cout<<"Alamat Pelanggan : "<<strupr(alamat)<<endl; // str-> string upr-> upper
cout<<"ID Pelanggan : ...<<endl;
Sleep(2000):cout << "\x1b[1A\x1b[2K";
cout<<"ID Pelanggan : "<<ID<<endl;
cout<<"Jenis Golongan : ...<<endl;
Sleep(2000):cout << "\x1b[1A\x1b[2K";
cout<<"Jenis Golongan : "<<jenis<<endl;
cout<<"Tanggal : ...<<endl;
Sleep(2000):cout << "\x1b[1A\x1b[2K";
cout<<"Tanggal : "<<tanggal<<endl;
}

```

```

cout<<"Pemakaian Air (m3) : ..."<<endl;
Sleep(2000);cout << "\x1b[1A\x1b[2K";
cout<<"Pemakaian Air (m3) : "<<pakaiAir<<"m3"<<endl;
cout<<"Sistem Pembayaran (CASH/BRI/BCA) : ..."<<endl;
Sleep(2000);cout << "\x1b[1A\x1b[2K";
cout<<"Sistem Pembayaran : "<< bayar[metode]<<endl;

//sistem penghitung tagihanAir dan tagihanPakaiAir
if(pakaiAir <= 35){
    tagihanAir = 7450;
    tagihanPakaiAir = pakaiAir * tagihanAir + 1195 + 3000;
} else {
    tagihanAir = 9550;
    tagihanPakaiAir = pakaiAir * tagihanAir + 2400 + 3000;
}

if(bayar[metode] == "BRI"){
    tagihanPakaiAir += 2500;
    strcpy(code, "00097 8823 2791 1111 0273"); //copy nilai string ke variable code
}else if (bayar[metode] == "BCA"){
    tagihanPakaiAir += 2500;
    strcpy(code, "4444 6793 8712 0853 4365");
} else{
    strcpy(code, "-");

//cetak tagihan pembayaran
cout<<endl;
cout<<"----- TAGIHAN PEMBAYARAN AIR -----"<<endl;
cout<<"-----<<endl;
cout<<"Pemakaian Air (m3) : "<<pakaiAir<<"m3"<<endl;
cout<<"Tagihan Air : "<<tagihanAir<<endl;
cout<<"Biaya Admin : "<<"Rp3000"<<endl;
cout<<"Biaya Pajak : "<<"Rp1195"<<endl;
cout<<"-----<<endl;
cout<<"TOTAL TAGIHAN PEMAKAIAN AIR : Rp"<<tagihanPakaiAir<<endl;
cout<<"SISTEM PEMBAYARAN : "<<bayar[metode]<<endl;
cout<<"CODE PEMBAYARAN : "<<code<<endl;
cout<<"-----<<endl;

ofstream file("TAGIHAN_PEMBAYARAN.txt"); // membuat/membuat file "TAGIHAN_PEMBAYARAN"
if (file.is_open()) { //nyatakan kondisi apakah file terbuka
    //Masukan seluruh data kedalam file
    file << "-----<< endl;
    file << "----- TAGIHAN PEMBAYARAN AIR-----<< endl;
    file << "-----<< endl;
    file << setw(25) << left << "Nama Pelanggan" << endl;
    file << setw(25) << left << "Alamat Pelanggan" << endl;
    file << setw(25) << left << "ID Pelanggan" << endl;
    file << setw(25) << left << "Jenis Golongan" << endl;
    file << setw(25) << left << "Tanggal" << endl;
    file << setw(25) << left << "Pemakaian Air (m3)" << endl;
    file << setw(25) << left << "Tagihan Air" << endl;
    file << setw(25) << left << "Biaya Admin" << endl;
    file << setw(25) << left << "Biaya Pajak" << endl;
    file << "-----<< endl;
    file << setw(25) << left << "TOTAL TAGIHAN PEMAKAIAN AIR" << endl;
    file << setw(25) << left << "SISTEM PEMBAYARAN" << endl;
    file << setw(25) << left << "CODE PEMBAYARAN" << endl;
    file << "-----<< endl;

    file.close(); //menutup file
    cout << "SEDANG MENCETAK TAGIHAN MOHON TUNGGU SEBENTAR.....\n";
    Sleep(5000);
    cout << "\x1b[1A\x1b[2K";
    cout << "\nBERHASIL MENCETAK TAGIHAN KE DALAM FILE | TAGIHAN_PEMBAYARAN.txt |";
} else {
    cout << "GAGAL MENCETAK TAGIHAN" << endl; //kondisi jika file gagal dibuka
}

int main() {
    srand(time()); //refresh random setiap kali program dijalankan
    int pilihan = rand() % 2 + 1; //menyatakan nilai pilihan secara random

    cout<<"----- SISTEM PEMBAYARAN PAM -----<< endl;
    cout<<"-----<< endl;
    cout<<"nSilahkan Login untuk melanjutkan "<<endl;
    cout<<"Username : ...<<endl;
    Sleep(2000); //Menggunakan sistem windows program menunggu selama beberapa detik
    cout << "\x1b[1A\x1b[2K"; //Hapus baris:\x1b[1A - Naik 1 Baris || \x1b[2K - Hapus Seluruh Baris
    cout<<"Username : Zahra Tsabitah<<endl; //menampilkan baris code setelah baris sebelumnya dihapus menggunakan sistem ANSI
    cout<<"Password : ...<<endl;
    Sleep(2000);cout << "\x1b[1A\x1b[2K";
    cout<<"Password : 4523210145<<endl;
    cout<<"nTERIMA KASIH BERHASIL LOGIN!!<<endl;
//cout<<"n

    cout<<"nSilahkan Pilih Aktivitas yang dinginkan sesuai dengan menu dibawah ini<<endl;
    cout<<"1. Lihat Rekap Tahun Lalu<<endl;
    cout<<"2. Melakukan Pembayaran Bulan Ini <<endl;
    cout<<"nMasukan Pilihan : .....<<endl;
    Sleep(2000);cout << "\x1b[1A\x1b[2K";
    cout<<"nAnda memilih pilihan ke-<<pilihan<<endl;
    cout<<endl;
    cout<<"MOHON TUNGGU SEBENTAR SEDANG BERALIH.....<<endl;
    Sleep(2000);cout << "\x1b[1A\x1b[2K";
    cout<<endl;
    cout<<endl;

//sistem pemilihan aktivitas dengan menampilkan fungsi yang sesuai
switch (pilihan) { //sesuai dengan inisialisasi nilai random diatas
    case 1:
        lihatRekapTahunLalu(); //memanggil function
        break;
    case 2:
        PembayaranBulanIni();
        break;
    default:
        cout << "Pilihan tidak valid" << endl;
}
cin.get();
}

```

## 2.2.2 Hasil Compile/Run Code

### Kondisi 1 : Memilih Lihat Rekap Tahun Lalu

```
C:\ProjectMinGW\TUGASBESAR_Praktek>PembayaranPAM
-----
----- SISTEM PEMBAYARAN PAM -----
-----



Silahkan Login untuk melanjutkan
Username : Zahra Tsabitah
Password : 4523210145

TELAH BERHASIL LOGIN!!!

Silahkan Pilih Aktivitas yang dinginkan sesuai dengan menu dibawah ini
1. Lihat Rekap Tahun Lalu
2. Melakukan Pembayaran Bulan Ini

Anda memilih pilihan ke-1
```

Gambar 1.1 : Tampilan menu awal untuk memilih aktivitas

----- REKAPAN PEMBAYARAN PAM TAHUN 2023 -----											
Nama Pelanggan	:	ZAHRA TSABITAH	Alamat Pelanggan	:	JL. PAM BARU RAYA NO.7A	ID Pelanggan	:	45-23-2-101-45	Jenis Golongan	:	Rumah Tangga
<hr/>											
Bulan	Pemakaian Air	Tagihan Air	Tagihan Pemakaian Air	Tanggal Pembayaran	Status Pembayaran	Metode Pembayaran					
1	4m3	Rp7450	Rp33995	04-01-2023	TEPAT WAKTU	CASH					
2	5m3	Rp7450	Rp46445	29-02-2023	TERLAMBAT(!)	BCA					
3	6m3	Rp7450	Rp48895	17-03-2023	TEPAT WAKTU	BRI					
4	7m3	Rp7450	Rp5345	05-04-2023	TEPAT WAKTU	CASH					
5	8m3	Rp7450	Rp63795	18-05-2023	TEPAT WAKTU	BRI					
6	42m3	Rp8250	Rp351900	20-06-2023	TEPAT WAKTU	CASH					
7	4m3	Rp7450	Rp33995	01-07-2023	TEPAT WAKTU	BCA					
8	10m3	Rp7450	Rp78695	03-08-2023	TEPAT WAKTU	CASH					
9	49m3	Rp8250	Rp414658	30-09-2023	TERLAMBAT(!)	BCA					
10	34m3	Rp7450	Rp257495	26-10-2023	TEPAT WAKTU	BCA					
11	11m3	Rp7450	Rp91145	27-11-2023	TERLAMBAT(!)	BRI					
12	20m3	Rp7450	Rp153195	10-12-2023	TEPAT WAKTU	CASH					
<hr/>											
TOTAL TAGIHAN PEMAKAIAN AIR 1.THN : Rp1630550											
TOTAL KETERLAMBATAN PEMBAYARAN : 3 kali											

Gambar 1.2 : Ouput function lihatRekapTahunLalu() dengan pilihan = 1

### Kondisi 2 : Memilih Melakukan Pembayaran Bulan Ini

```
C:\ProjectMinGW\TUGASBESAR_Praktek>PembayaranPAM
-----
----- SISTEM PEMBAYARAN PAM -----
-----



Silahkan Login untuk melanjutkan
Username : Zahra Tsabitah
Password : 4523210145

TELAH BERHASIL LOGIN!!!

Silahkan Pilih Aktivitas yang dinginkan sesuai dengan menu dibawah ini
1. Lihat Rekap Tahun Lalu
2. Melakukan Pembayaran Bulan Ini

Anda memilih pilihan ke-2
```

Gambar 2.1 : Tampilan menu awal untuk memilih aktivitas ke-2

```

----- PEMBAYARAN PAM BULANINI -----
Nama Pelanggan : ZAHRA TSABITAH
Alamat Pelanggan : JL. PAM BARU RAYA NO.7A
ID Pelanggan : 45-23-2-101-45
Jenis Golongan : Rumah Tangga
Tanggal : 2 Januari 2024
Pemakaian Air (m3) : 34m3
Sistem Pembayaran : BRI
-----TAGIHAN PEMBAYARAN AIR-----
Pemakaian Air (m3) : 34m3
Tagihan Air : 7450
Biaya Admin : Rp3000
Biaya Pajak : Rp1195
TOTAL TAGIHAN PEMAKAIAN AIR : Rp259995
SISTEM PEMBAYARAN : BRI
CODE PEMBAYARAN : 0097 8823 2791 1111 0273

----- PEMBAYARAN PAM BULANINI -----
Nama Pelanggan : ZAHRA TSABITAH
Alamat Pelanggan : JL. PAM BARU RAYA NO.7A
ID Pelanggan : 45-23-2-101-45
Jenis Golongan : Rumah Tangga
Tanggal : 2 Januari 2024
Pemakaian Air (m3) : 34m3
Tagihan Air : 7450
Biaya Admin : Rp3000
Biaya Pajak : Rp1195
TOTAL TAGIHAN PEMAKAIAN AIR: Rp259995
SISTEM PEMBAYARAN : BRI
CODE PEMBAYARAN : 0097 8823 2791 1111 0273

----- TAGIHAN PEMBAYARAN AIR -----
Isi file TAGIHAN PEMBAYARAN.txt

```

BERHASIL MENCETAK TAGIHAN KE DALAM FILE | TAGIHAN\_PEMBAYARAN.txt ||

Gambar 2.2 : Tampilan function PembayaranBulanIni() – Sistem Pembayaran (BRI)

```

----- PEMBAYARAN PAM BULANINI -----
Nama Pelanggan : ZAHRA TSABITAH
Alamat Pelanggan : JL. PAM BARU RAYA NO.7A
ID Pelanggan : 45-23-2-101-45
Jenis Golongan : Rumah Tangga
Tanggal : 2 Januari 2024
Pemakaian Air (m3) : 34m3
Sistem Pembayaran : BCA
-----TAGIHAN PEMBAYARAN AIR-----
Pemakaian Air (m3) : 34m3
Tagihan Air : 7450
Biaya Admin : Rp3000
Biaya Pajak : Rp1195
TOTAL TAGIHAN PEMAKAIAN AIR : Rp259995
SISTEM PEMBAYARAN : BCA
CODE PEMBAYARAN : 4444 6793 8712 0853 4365

----- PEMBAYARAN PAM BULANINI -----
Nama Pelanggan : ZAHRA TSABITAH
Alamat Pelanggan : JL. PAM BARU RAYA NO.7A
ID Pelanggan : 45-23-2-101-45
Jenis Golongan : Rumah Tangga
Tanggal : 2 Januari 2024
Pemakaian Air (m3) : 34m3
Tagihan Air : 7450
Biaya Admin : Rp3000
Biaya Pajak : Rp1195
TOTAL TAGIHAN PEMAKAIAN AIR: Rp259995
SISTEM PEMBAYARAN : BCA
CODE PEMBAYARAN : 4444 6793 8712 0853 4365

----- TAGIHAN PEMBAYARAN AIR -----
Isi file TAGIHAN PEMBAYARAN.txt

```

BERHASIL MENCETAK TAGIHAN KE DALAM FILE | TAGIHAN\_PEMBAYARAN.txt |

Gambar 2.2 : Tampilan function PembayaranBulanIni() – Sistem Pembayaran (BCA)

```

----- PEMBAYARAN PAM BULANINI -----
Nama Pelanggan : ZAHRA TSABITAH
Alamat Pelanggan : JL. PAM BARU RAYA NO.7A
ID Pelanggan : 45-23-2-101-45
Jenis Golongan : Rumah Tangga
Tanggal : 2 Januari 2024
Pemakaian Air (m3) : 34m3
Sistem Pembayaran : CASH
-----TAGIHAN PEMBAYARAN AIR-----
Pemakaian Air (m3) : 34m3
Tagihan Air : 7450
Biaya Admin : Rp3000
Biaya Pajak : Rp1195
TOTAL TAGIHAN PEMAKAIAN AIR : Rp257495
SISTEM PEMBAYARAN : CASH
CODE PEMBAYARAN : -

----- PEMBAYARAN PAM BULANINI -----
Nama Pelanggan : ZAHRA TSABITAH
Alamat Pelanggan : JL. PAM BARU RAYA NO.7A
ID Pelanggan : 45-23-2-101-45
Jenis Golongan : Rumah Tangga
Tanggal : 2 Januari 2024
Pemakaian Air (m3) : 34m3
Tagihan Air : 7450
Biaya Admin : Rp3000
Biaya Pajak : Rp1195
TOTAL TAGIHAN PEMAKAIAN AIR: Rp257495
SISTEM PEMBAYARAN : CASH
CODE PEMBAYARAN : -

----- TAGIHAN PEMBAYARAN AIR -----
Isi file TAGIHAN PEMBAYARAN.txt

```

BERHASIL MENCETAK TAGIHAN KE DALAM FILE | TAGIHAN\_PEMBAYARAN.txt |

Gambar 2.3 : Tampilan function PembayaranBulanIni() – Sistem Pembayaran (CASH)

### 2.2.3 Pseudocode

#### Pseudocode

##### Kamus/Deklarasi Variable Function lihatRekapTahunLalu

```
R, t, s, h, z : int
totalTagihanPemakaianAir, totalTerlambat : int
bulan[JUMLAH_BULAN], pakaiAir[JUMLAH_BULAN] : int
tagihanAir[JUMLAH_BULAN], tagihanPakaiAir[JUMLAH_BULAN] : int
bayar[JUMLAH_BULAN][10] : char
tanggal[JUMLAH_BULAN], string status[JUMLAH_BULAN] : string
```

##### Algoritma/Deksripsi Function lihatRekapTahunLalu

```
s = 0
bulan[JUMLAH_BULAN] = {12,3,4,5,6,8,2,11,10,7,1,9}
pakaiAir[JUMLAH_BULAN] = {4,5,6,7,8,42,4,10,49,34,11,20}
tagihanAir[JUMLAH_BULAN], tagihanPakaiAir[JUMLAH_BULAN]:
bayar[JUMLAH_BULAN][10] = {"CASH", "BCA", "BRI", "CASH", "BRI", "CASH", "BCA",
                            "CASH", "BCA", "BCA", "BRI", "CASH"}
tanggal[JUMLAH_BULAN] = {"04-01-2023", "29-02-2023", "17-03-2023", "05-04-2023",
                        "18-05-2023", "20-06-2023", "01-07-2023", "03-08-2023",
                        "30-09-2023", "26-10-2023", "27-11-2023", "10-12-2023"}
status[JUMLAH_BULAN] = {"TEPAT WAKTU", "TERLAMBAT()", "TEPAT WAKTU",
                        "TEPAT WAKTU", "TEPAT WAKTU", "TEPAT WAKTU",
                        "TEPAT WAKTU", "TEPAT WAKTU", "TERLAMBAT()", "TEPAT WAKTU",
                        "TEPAT WAKTU", "TERLAMBAT()", "TEPAT WAKTU"}

while (s < JUMLAH_BULAN - 1)
    t = s
    R = s + 1
    while (R < JUMLAH_BULAN)
        if (bulan[R] < bulan[t])
            t = R
        endif
        R++
    endwhile
    h = bulan[t]
    bulan[t] = bulan[s]
    bulan[s] = h
    s++
endwhile

for (z = 0; z < JUMLAH_BULAN; z++)
    if(pakaiAir[z] <= 35)
        tagihanAir[z] = 7450
        tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] + 1195 + 3000
    else
        tagihanAir[z] = 8250
        tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] + 2400 + 3000
    endif
```

Pseudocode : Part1

```

if(status[z] == "TERLAMBAT(!)")
    tagihanPakaiAir[z] += 5000
    totalTerlambat++
endif
print(bulan[z])
print(to_string(pakaiAir[z]))
print(tagihanAir[z])
print(tagihanPakaiAir[z])
print(tanggal[z])
print(status[z])
print(bayar[z])
print(tanggal[z])
totalTagihanPemakaianAir += tagihanPakaiAir[z]
print(totalTagihanPemakaianAir)
print(totalTerlambat)

```

#### **Kamus/Deklarasi Variable Function PembayaranBulanIni**

```

const *bayar[] = {"BRI", "BCA", "CASH"} : char
nama[50], alamat[50], ID[50], jenis[50], : char
tanggal[50], code[25] : char
pakaiAir, tagihanAir, tagihanPakaiAir, : int
metode, panjang_array : int

```

#### **Algoritma/Deksripsi Function PembayaranBulanIni**

```

const *bayar[] = {"BRI", "BCA", "CASH"}
nama[50] = "Zahra Tsabitah"
alamat[50] = "Jl. Pam Baru Raya No.7a"
ID[50] = "45-23-2-101-45"
jenis[50] = "Rumah Tangga"
tanggal[50] = "2 Januari 2024"
pakaiAir = 34
panjang_array = sizeof(bayar) / sizeof(bayar[0])
metode = rand() % panjang_array
print(strupr(nama))
print(strupr(alamat))
print(ID)
print(jenis)
print(tanggal)
print(pakaiAir)
print(bayar[metode])
if(pakaiAir <= 35)
    tagihanAir = 7450
    tagihanPakaiAir = pakaiAir * tagihanAir + 1195 + 3000
else
    tagihanAir = 9550
    tagihanPakaiAir = pakaiAir * tagihanAir + 2400 + 3000
endif

```

Pseudocode : Part2

```

if(bayar[metode] == "BRI")
    tagihanPakaiAir += 2500
    strcpy(code, "0097 8823 2791 1111 0273")
else
    if (bayar[metode] == "BCA")
        tagihanPakaiAir += 2500
        strcpy(code, "4444 6793 8712 0853 4365")
    else
        strcpy(code, "-")
    endif
endif
print(pakaiAir, "m3")
print(tagihanAir)
print("Rp3000")
print("Rp1195")
print(tagihanPakaiAir)
print(bayar[metode])
print(code)
ofstream file("TAGIHAN PEMBAYARAN.txt")
if (file.is_open()) {
    file("-----")
    file("-----TAGIHAN PEMBAYARAN AIR-----")
    file("-----")
    file("Nama Pelanggan", ": ", strupr(nama))
    file("Alamat Pelanggan", ": ", strupr(alamat))
    file("ID Pelanggan", ": ", ID)
    file("Jenis Golongan", ": ", jenis)
    file("Tanggal", ": ", tanggal)
    file("Pemakaian Air (m3)", ": ", pakaiAir, "m3")
    file("Tagihan Air", ": ", tagihanAir)
    file("Biaya Admin", ": ", "Rp3000")
    file("Biaya Pajak", ": ", "Rp1195")
    file("-----")
    file("TOTAL TAGIHAN PEMAKAIAN AIR", ": Rp", tagihanPakaiAir)
    file("SISTEM PEMBAYARAN", ": ", bayar[metode])
    file("CODE PEMBAYARAN", ": ", code)
    file("-----")
    file.close()
    print("BERHASIL MENCETAK TAGIHAN KE DALAM FILE |")
    TAGIHAN_PEMBAYARAN.txt |")
else
    print("GAGAL MENCETAK TAGIHAN")
endif

```

Pseudocode : Part3

**Kamus/Deklarasi Variable**

pilihan : int

**Algoritma/Deksripsi**

```
srand(time(0))
pilihan = rand() % 2 + 1
print(pilihan)
switch (pilihan)
    case 1:
        lihatRekapTahunLalu()
        break;
    case 2:
        PembayaranBulanIni()
        break;
    default:
        print("Pilihan tidak valid")
```

Pseudocode : Part4

## 2.2.4 Algoritma

### Algoritma

```
1. Membuat function lihatRekapTahunLalu()
2. s = 0
3. bulan[JUMLAH_BULAN] = {12,3,4,5,6,8,2,11,10,7,1,9}
4. pakaiAir[JUMLAH_BULAN] = {4,5,6,7,8,42,4,10,49,34,11,20}
5. tagihanAir[JUMLAH_BULAN], tagihanPakaiAir[JUMLAH_BULAN];
6. bayar[JUMLAH_BULAN][10] = {"CASH","BCA","BRI","CASH","BRI","CASH","BCA",
    "CASH","BCA","BCA","BRI","CASH"}
7. tanggal[JUMLAH_BULAN] = {"04-01-2023","29-02-2023","17-03-2023","05-04-2023",
    "18-05-2023","20-06-2023","01-07-2023","03-08-2023", "30-09-2023","26-10-
    2023","27-11-2023","10-12-2023"}
8. status[JUMLAH_BULAN] = {"TEPAT WAKTU","TERLAMBAT(!)","TEPAT
    WAKTU","TEPAT WAKTU","TEPAT WAKTU","TEPAT WAKTU","TEPAT WAKTU","TEPAT
    WAKTU","TERLAMBAT(!)","TEPAT WAKTU","TERLAMBAT(!)","TEPAT WAKTU"}
9. Selama (s < JUMLAH_BULAN - 1), maka kerjakan baris 10 s.d 19 kalau tidak baris 20
10. t = s
11. R = s + 1
12. Selama (R < JUMLAH_BULAN), maka kerjakan baris 13 s.d 15 kalau tidak baris 16
13. jika (bulan[R] < bulan[t]), maka kerjakan baris 14 s.d 15 kalau tidak baris 16
14. t = R
15. R++
16. h = bulan[t]
17. bulan[t] = bulan[s]
18. bulan[s] = h
19. s++
20. z = 0
21. jika (z < JUMLAH_BULAN), maka kerjakan baris 22 s.d 38 kalau tidak baris 39
22. jika (pakaiAir[z] <= 35), maka kerjakan baris 23 s.d 24 kalau tidak baris 25
23. tagihanAir[z] = 7450
24. tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] + 1195 + 3000
25. tagihanAir[z] = 8250
26. tagihanPakaiAir[z] = pakaiAir[z] * tagihanAir[z] + 2400 + 3000
27. jika (status[z] == "TERLAMBAT(!)"), maka kerjakan baris 27 s.d 29 kalau tidak 30
28. tagihanPakaiAir[z] += 5000
29. totalTerlambat++
30. Mencetak nilai variable bulan[z]
31. Mencetak nilai variable menjadi string to_string(pakaiAir[z]) + "m3"
32. Mencetak nilai variable tagihanAir[z]
33. Mencetak nilai variable tagihanPakaiAir[z]
34. Mencetak nilai variable tanggal[z]
35. Mencetak nilai variable status[z]
36. Mencetak nilai variable bayar[z]
37. totalTagihanPemakaianAir += tagihanPakaiAir[z]
38. z++
39. Mencetak nilai variable totalTagihanPemakaianAir
40. Mencetak nilai variable totalTerlambat
```

Algoritma : Part1

```

41. Membuat function PembayaranBulanIni()
42. const array pointer *bayar[] = {"BRI", "BCA", "CASH"}
43. nama[50] = "Zahra Tsabitah"
44. alamat[50] = "Jl. Pam Baru Raya No.7a"
45. ID[50] = "45-23-2-101-45"
46. jenis[50] = "Rumah Tangga"
47. tanggal[50] = "2 Januari 2024"
48. pakaiAir = 34
49. panjang_array = sizeof(bayar) / sizeof(bayar[0])
50. metode = rand() % panjang_array
51. Mencetak nilai variable menjadi Huruf Besar strupr(nama)
52. Mencetak nilai variable menjadi Huruf Besar strupr(alamat)
53. Mencetak nilai variable ID
54. Mencetak nilai variable jenis
55. Mencetak nilai variable tanggal
56. Mencetak nilai variable pakaiAir
57. Mencetak nilai variable bayar[metode]
58. Jika (pakaiAir <= 35), maka kerjakan baris 59 s.d 60 kalau tidak baris 61
59. tagihanAir = 7450
60. tagihanPakaiAir = pakaiAir * tagihanAir + 1195 + 3000
61. tagihanAir = 9550
62. tagihanPakaiAir = pakaiAir * tagihanAir + 2400 + 3000
63. Jika (bayar[metode] == "BRI"), maka kerjakan baris 64 s.d 65 kalau tidak baris 66
64. tagihanPakaiAir += 2500
65. strcpy(code, "0097 8823 2791 1111 0273")
66. Jika (bayar[metode] == "BCA"), maka kerjakan baris 67 s.d 68 kalau tidak baris 69
67. tagihanPakaiAir += 2500
68. strcpy(code, "4444 6793 8712 0853 4365")
69. strcpy(code, "-")
70. Mencetak nilai variable pakaiAir dan "m3"
71. Mencetak nilai variable tagihanAir
72. Mencetak "Rp3000"
73. Mencetak "Rp1195"
74. Mencetak nilai variable tagihanPakaiAir
75. Mencetak nilai variable bayar[metode]
76. Mencetak nilai variable code
77. Membuat file dan menyatakan file sebagai file("TAGIHAN PEMBAYARAN.txt")
78. Jika (file.is_open()), maka kerjakan baris 78 s.d 97 kalau tidak baris 98
79. Merekam data ke file ("-----")
80. Merekam data ke file ("-----TAGIHAN PEMBAYARAN AIR-----")
81. Merekam data ke file ("-----")
82. Merekam data ke file ("Nama Pelanggan", ": ", strupr(nama))
83. Merekam data ke file ("Alamat Pelanggan", ": ", strupr(alamat))
84. Merekam data ke file ("ID Pelanggan", ": ", ID)
85. Merekam data ke file ("Jenis Golongan", ": ", jenis)
86. Merekam data ke file ("Tanggal", ": ", tanggal)
87. Merekam data ke file ("Pemakaian Air (m3)", ": ", pakaiAir, "m3")
88. Merekam data ke file ("Tagihan Air", ": ", tagihanAir)

```

Algoritma : Part2

```
89.Merekam data ke file ("Biaya Admin", ": ","Rp3000")
90.Merekam data ke file ("Biaya Pajak", ": ","Rp1195")
91.Merekam data ke file ("-----")
92.Merekam data ke file ("TOTAL TAGIHAN PEMAKAIAN AIR", ": Rp", tagihanPakaiAir)
93.Merekam data ke file ("SISTEM PEMBAYARAN", ": ",bayar[metode])
94.Merekam data ke file ("CODE PEMBAYARAN", ": ",code)
95.Merekam data ke file ("-----")
96.Menutup file dengan file.close()
97.Mencetak "BERHASIL MENCETAK TAGIHAN KE DALAM FILE |"
    TAGIHAN_PEMBAYARAN.txt |"
98.Mencetak "GAGAL MENCETAK TAGIHAN"
99.srand(time(0))
100.pilihan = rand() % 2 + 1
101.Mencetak nilai variable pilihan
102.Melakukan seleksi dengan switch berdasarkan nilai variable pilihan
103.Jika (case 1), maka kerjakan baris 104 jika tidak baris 105
104.Memanggil function lihatRekapTahunLalu()
105.Jika (case 2), maka kerjakan baris 106 jika tidak baris 107
106.Memanggil function PembayaranBulanIni()
107.Mencetak default "Pilihan tidak valid"
108.Selesai
```

### Algoritma : Part3

## **BAB III**

## **PENUTUP**

### **3.1 Kesimpulan**

Kemajuan dan perkembangan teknologi zaman ini memang sudah tidak bisa terlepas lagi dari umat manusia, mau dalam kegiatan apapun itu teknologi pasti akan selalu ada mengambil peran dalam keseharian kita, kita pun merasakan bagaimana mudahnya melakukan sesuatu dengan adanya teknologi, seperti program sederhana yang telah penulis buat tersebut, jika program tersebut di pantenkan kemudian dikembangkan kembali maka kemudahan dalam bertransaksi, mendapatkan informasi, dan memberikan data akan jauh lebih mudah, bahkan kita pun tidak perlu kesusahan untuk mengurus hal-hal tersebut.

Oleh karena itulah Algoritma dalam suatu pemograman diperlukan untuk mengasah jalan pikir seorang programmer agar membuat program dengan sistem sesuai serta mencari cara untuk menyelesaikan beberapa permasalahan dengan membuatkan sistemnya. Pendukung dalam hal ini selain mempelajari teori-teori tentang bagaimana suatu program dibuat dan komponen pendukungnya adalah dengan melakukan praktik secara langsung, salah satunya adalah tugas besar berupa program Sistem Pembayaran PAM yang memuat seluruh materi yang ada pada tugas praktik sebelumnya.

Dengan adanya laporan praktikum ini saya sebagai penulis mendapatkan pengetahuan baru dan perkembangan skill yang meningkat pada bahasa pemograman C++ dengan materi berupa materi Sequence, Selection, Repetition, Rekursif, Array, Function, Sort, Pointer, File Biner dan Teks. Selain itu dikarenakan adanya tugas-tugas praktik sebelum tugas besar sangat membantu karena dapat merasa terlatih ataupun terbiasa dalam hal Problem Solving dan berpikir kritis dengan menggunakan logika.

Semoga perkembangan teknologi selanjutnya dan sistem-sistem pendukung untuk pembuatan sistem pemograman menjadi lebih baik dan lebih memudahkan untuk membantu para mahasiswa dalam menempuh jalannya sebagai seorang programmer.

### **3.2 Kritik dan Saran**

#### **3.2.1 Kritik**

Kritik sangat diperlukan sebagai bentuk tinjauan kembali mengenai laporan praktikum maupun proses pembuatan laporan praktikum.

1. Dibutuhkannya waktu yang lebih dalam pembuatan tugas serta laporan agar mendapatkan hasil yang sesuai.
2. Penggunaan beberapa fitur yang kurang relevan dengan kondisi saat ini yang memungkinkan terjadinya bug pada program.
3. Tidak memberikan contoh mengenai sistem yang lebih rumit sesuai dengan materi sehingga jika mendapatkan tugas ataupun mencoba membuat sistem program dengan lebih kompleks mendapatkan sedikit kesulitan.
4. Perbedaan format materi yang diajarkan dengan yang ditugaskan terkadang berbeda sehingga mengalami sedikit kesulitan dalam membaca ataupun mengartikannya.

### **3.2.2 Saran**

Untuk pengembangan laporan praktikum yang lebih baik kedepannya saya sebagai penulis memiliki beberapa saran.

1. Memberikan kesempatan fleksibilitas terhadap pelajar dalam waktu pengeroaan tugas dan kreativitas.
2. Memberikan arahan atau sosialisasi yang jelas dan terperinci dalam penyerahan tugas maupun dalam pemberian materi.
3. Menyimpan seluruh tugas praktikum yang telah dibuat dalam 1 folder untuk memudahkan saat pembuatan laporan praktikum.
4. Mengerjakan tugas besar praktik sesuai sub topik materi yang diberikan untuk pembahasan teori.
5. Gunakan 1 format yang sama dalam pengeroaan tugas praktikum agar dari awal praktik sampai akhir memiliki tampilan yang sama dan bagus karena tersusun rapih.

## **DAFTAR PUSTAKA**

- GreeksforGreeks. (2023). "Rand() and Srand() in C++". Diakses tanggal 15 Desember 2023 dari <https://www.geeksforgeeks.org/rand-and-srand-in-ccpp/>
- Nursari, Sri Rezeki Candra. (2023). "20231-Algoritma dan Pemograman-A". LMS E-learning Universitas Pancasila. Diakses tanggal 26 Desember 2023 dari <https://lms.univpancasila.ac.id/course/view.php?id=16422>
- Nursari, Sri Rezeki Candra. (2023). "20231-Prak.Algoritma dan Pemograman-A". LMS E-learning Universitas Pancasila. Diakses tanggal 26 Desember 2023 dari <https://lms.univpancasila.ac.id/course/view.php?id=16428>
- Petersen, C. (2021). "ANSI escape mode". Github Gist. Diakses tanggal 19 Desember 2023 dari <https://gist.github.com/fnky/458719343aab01cfb17a3a4f7296797>
- Saini, A. (2023). "C++ sleep() function". Scaler Topics. Diakses tanggal 15 Desember 2023 dari <https://www.scaler.com/topics/cpp-sleep/>
- Sianturi, Tasya Ilmelia Sabarwati. (2020). "LAPORAN PRAKTIKUM ALGORITMA & PEMROGRAMAN A "PENJUALAN KAOS SABLON"".