

Remidi

Pemograman Dasar



Nama : Faustina Chelloana T.
Kelas : X RPL
No : 15

A. Alur Pemograman Komputer

1. Algoritma Pemograman

Dalam matematika dan ilmu komputer, algoritma adalah urutan atau langkah-langkah untuk penghitungan atau untuk menyelesaikan suatu masalah yang ditulis secara berurutan. Sehingga, algoritma pemrograman adalah urutan atau langkah-langkah untuk menyelesaikan masalah pemrograman komputer.

Contoh : Algoritma untuk menghitung nilai y dari persamaan $y = 3x + 8$

Algoritmanya adalah:

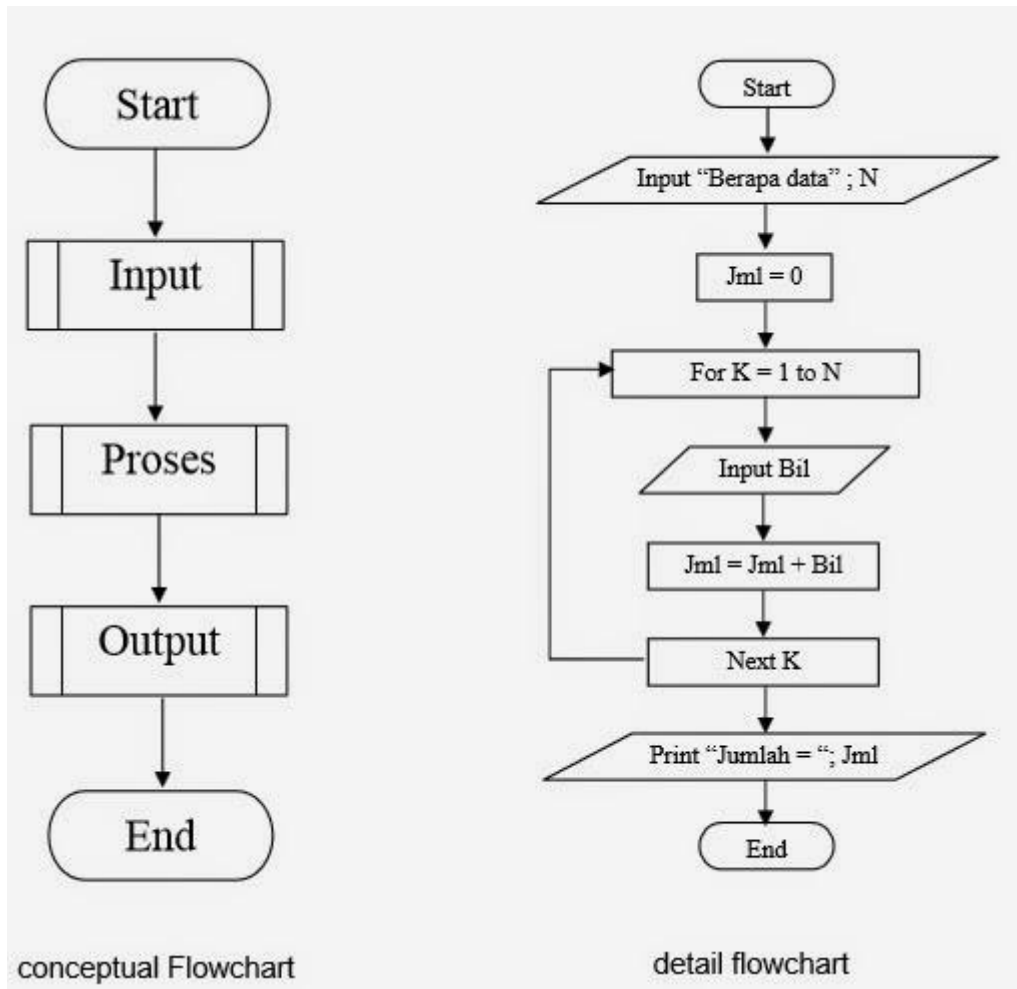
- Mulai
- Tentukan nilai x
- Hitung nilai $y = 3x + 8$
- Cetak nilai x dan y
- Selesai

2. Flowchart

Flowchart adalah Bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Flowchart merupakan cara penyajian dari suatu algoritma.

Tujuan Membuat Flowchat :

- Menggambarkan suatu tahapan penyelesaian masalah
- Secara sederhana, terurai, rapi dan jelas
- Menggunakan simbol-simbol standar



B. Perangkat Lunak Bahasa Pemrograman

Bahasa pemrograman, atau sering diistilahkan juga dengan bahasa komputer atau bahasa pemrograman komputer, adalah instruksi standar untuk memerintah komputer. Bahasa pemrograman ini merupakan suatu himpunan dari aturan sintaks dan semantik yang dipakai untuk mendefinisikan program komputer. Bahasa ini memungkinkan seorang programmer dapat menentukan secara persis data mana yang akan diolah oleh komputer, bagaimana data ini akan disimpan/diteruskan, dan jenis langkah apa yang akan diambil dalam berbagai situasi secara persis.

Contoh : HTML / CSS, PHP, Java, Visual Basic

C. Konsep Penggunaan Tipe Data, Variabel, Konstanta, Operasi, dan Ekspresi

1. Tipe Data

Tipe data biasanya digunakan untuk menentukan jenis nilai yang dapat ditampung oleh suatu variabel, umumnya telah menyediakan tipe-tipe data yang sederhana (simple) maupun yang terstruktur. Apabila kita membutuhkan tipe data yang belum tersedia, kita dapat mendefinisikan sendiri tipe data baru yang disebut enumerated type.

Setiap data yang ada pada program pascal memiliki tipe data, baik berupa angka bulat (integer), angka biasa (real), atau berupa karakter (char),(String) dan sebagainya.

Jenis - Jenis data yang di kenal dalam Bahasa Pascal yaitu :

- Tipe Data Sederhana
- Tipe Data Terstruktur

Tipe Data Sederhana

1. Bilangan bulat (integer)

Tipe data ini menerima angka dan bisa dilakukan proses aritmatika. Tipe data ini adalah tipe data bilangan bulat, yaitu Bilangan atau angka yang tidak memiliki titik desimal atau pecahan.

Contoh :

```
Program                                display_maxint;uses                                wincrt;
begin                                  writeln                                (maxint)
end.
```

Hasilnya: 32.767

2. Bilangan biasa (real)

Bilangan atau angka yang bisa memiliki titik desimal atau pecahan, Tipe data ini menerima angka dan koma serta bisa dilakukan proses aritmatika.

Contoh :

a' 'B' '+', dsb.

3. Boolean

Boolean adalah tipe data yang hanya mempunyai dua kemungkinan, yaitu true dan false. Biasanya digunakan untuk operasi logika dengan ungkapan AND, OR, atau NOT

Contoh :

```

Program display_bool;uses wincrt;
begin writeln(ord(true));
writeln(ord(false));
end.

```

Hasilnya: 1 0 3.2.

4. Karakter (char)

Karakter adalah data tunggal yang mewakili semua huruf, simbol baca, dan juga simbol angka yang tidak dapat dioperasikan secara matematis, misalnya: 'A', 'B',, 'Z', '?', '!', dst. Tipe data ini adalah tipe data yang menerima kumpulan karakter (huruf, angka, symbol) tipe data ini tidak bisa dilakukan proses aritmatika.

5. String

String adalah tipe data bentukan yang merupakan deretan karakter yang membentuk satu kata atau satu kalimat, yang biasanya diapit oleh dua tanda kutip. Tipe data yang menerima kumpulan karakter (huruf, angka, symbol) tipe data ini tidak bisa dilakukan proses aritmatika

Tipe Data Terstruktur

1. Array

Array atau lark adalah tipe terstruktur yang mempunyai komponen dalam jumlah yang tetap dan setiap komponen mempunyai tipe data yang sama. Posisi masing-masing komponen dalam array dinyatakan sebagai nomor index.

Contoh:

Type

Tipe_saya=array[1..100] of integer

Var

Var_baru : tipe_saya;

2. Record

Definisi Record hampir sama dengan Array, tetapi yang membedakan antara keduanya adalah kalau Array semua komponennya bertipe data yang sama sedangkan Record setiap komponennya bisa bertipe data yang beda-beda.

Contoh:

TYPE my_rec = RECORD

Nama : STRING[90]; Npm : STRING[8] ;

Nilai : REAL;

END;

VAR

Data nya : my re c;

3. Set

Himpunan (set) adalah kumpulan obyek yang mempunyai tipe data yang sama dan urutan penulisannya tidak diperhatikan. Setiap obyek dalam suatu himpunan disebut dengan anggota atau elemen himpunan.

Contoh:

TYPE irama = (JAZZ, POP, ROCK ,DANGDUT, SKA);

Musik = SET OF irama;

VAR

Jenis musik : Musik;

4. File

File (Berkas) adalah kumpulan sejumlah komponen yang bertipe data sama, yang jumlahnya tidak tertentu, dan biasanya tersimpan dalam suatu media penyimpan luar

Tipe Data Penunjuk

Pointer

Tipe data pointer merupakan tipe data yang berbeda dengan yang lainnya. Semua tipe data yang lainnya bersifat STATIS sedangkan tipe Pointer bersifat DINAMIS. Tipe ini merupakan tipe data untuk suatu variabel, dimana isi variabel ini berupa alamat variabel lain, dalam artian variabel pointer adalah variabel yang menunjuk nilai atau alamat variabel lainnya.

2. VARIABEL

Variabel adalah identifier yang berisi data yang dapat berubah-ubah nilainya di dalam program. Variabel dapat disebut juga sebagai perubah nilai, yaitu sesuatu yang dapat digunakan untuk menyimpan nilai, dimana nilai yang disimpan ini dapat selalu berubah-ubah.

JENIS-JENIS VARIABEL :

1. Boolean

Data boolean disebut juga data bilangan logik. Tipe data boolean hanya mengenal dua buah nilai yaitu benar (true) dan salah (false).

2. Integer

Data integrer merupakan nilai bilangan bulat dalam bentuk desimal misalnya 1, 2, 34, 765, 0, -89, -70. tipe data integrer mempunyai ranah nilai yang tidak terbatas. Dalam

Turbo Pascal terdapat lima macam tipe data integrer, yaitu: byte, shortint, word, integer dan longint.

3. Real

Data real adalah data yang mengandung data desimal misalkan 3.65, 9.0876, 2.00, dan lain-lain. Bilangan real juga dapat dituliskan dengan notasi E, yang artinya perpangkatan sepuluh. Misalnya 230.40E32 artinya 230.40×10^{32} .

4. Karakter (char)

Data karakter ditulis dengan menggunakan tanda petik (" ") misalkan "A", "@", "z", "#", "1", "-234", "+", "<" dan lain-lain. Tapi perlu diperhatikan "9" adalah karakter tetapi jika di tulis 9 maka tipenya menjadi integer. Maka sarat mutlak menuliskan karakter harus menggunakan tanda baca petik (" ").

5. String

Data string adalah untaian karakter-karakter dengan panjang tertentu yang terletak diantara tanda petik tunggal. nilai data string akan menepati memori sebesar banyaknya karakter stringnya ditambah dengan 1 byte. Contoh dari data string adalah seperti 'teknik', 'informatika', '9876151' dan sebagainya.

6. Pointer

Pointer adalah suatu variabel yang berisi alamat yang akan disimpan pada memori.

7. Ordinal

ordinal adalah tipe data yang merupakan subset dari tipe data sederhana (tipe data standart dan tipe data yang di definisikan sendiri oleh pemakai). Yang dimaksud dengan tipe data sederhana adalah semua tipe data kecuali tipe data riil. Jadi yang termasuk tipe data ordinal adalah tipe data standart (integer, shortint, longint, byte, word, boolean, dan char), dan tipe data yang didefinisikan sendiri oleh pemakai

Pemberian nama variabel harus mengikuti aturan yang ditetapkan oleh bahasa pemrograman yang kita gunakan. Namun secara umum ada aturan yang berlaku untuk hampir semua bahasa pemrograman. Aturan-aturan tersebut yaitu:

- Nama variabel harus diawali dengan huruf.
- Tidak boleh menggunakan spasi pada satu nama variabel. Spasi bisa diganti dengan karakter underscore (_).

- Nama variabel tidak boleh mengandung karakter-karakter khusus, seperti : ., +, -, *, /, <, >, &, (,) dan lain-lain.
- Nama variabel tidak boleh menggunakan kata-kata kunci d bahasa pemrograman

3. KONSTANTA

Konstanta adalah variabel yang nilai datanya bersifat tetap dan tidak bisa diubah. Jadi konstanta adalah juga variabel bedanya adalah pada nilai yang disimpannya. Jika nilai datanya sepanjang program berjalan tidak berubahubah, maka sebuah varibel lebih baik diperlakukan sebagai konstanta. Pada sebuah kode program, biasanya nilai data dari konstanta diberikan langsung di bagian deklarasi konstanta. Sedangkan untuk variabel biasanya hanya ditentukan nama variabel dan tipe datanya tanpa isian nilai data. Aturan penamaan variabel juga berlaku untuk penamaan konstanta. Demikian juga aturan penetapan tipe data. konstanta dibagi menjadi 4 :

- Konstanta integer, berupa integer biasa, integer panjang (long int) dan integer tak bertanda (unsigned integer). Nilainya dapat berupa desimal, oktal atau hexadesimal
- Konstanta floating point, dapat mengandung nilai pecahan, yang biasa ditulis dalam bentuk pecahan biasa maupun bentuk eksponensial dan selalu dinyatakan dalam double, kecuali jika diakhiri dengan F atau f (menyatakan konstanta float).
- Konstanta string, merupakan deretan karakter yang diawali dan diakhiri dengan tanda petik ganda (“...”). Juga dapat mengandung karakter yang menggunakan tanda \ yang disebut karakter escape (escape sequence).
- Konstanta karakter, selalu diawali dan diakhiri dengan tanda petik tunggal (‘...’). Beberapa konstanta karakter dapat diawali dengan tanda \ (penempatannya setelah tanda petik tunggal).

Sebagai contoh, jika kita membuat program perhitungan matematik yang menggunakan nilai pi (3.14159) yang mungkin akan muncul dibanyak tempat pada kode program, kita dapat membuat pi sebagai konstanta. Penggunaan konstanta **pi** akan lebih memudahkan penulisan kode program dibanding harus mengetikkan nilai 3.14159 berulang-ulang.

Konstanta digunakan apabila anda sering menggunakan suatu nilai konstanta secara berulang-ulang dalam kode anda, dan anda tidak ingin kerepotan untuk mengetikkan nilai konstanta itu terus menerus. Untuk membuat sebuah konstanta pada dasarnya sama seperti Untuk membuat sebuah konstanta pada dasarnya sama seperti Variabel.

```
const NamaKonstanta: JenisKonstanta := NilaiKonstanta
```


4. EKSPRESI

Ekspresi adalah transformasi nilai menjadi keluaran yang dilakukan melalui suatu perhitungan (komputasi).

Ekspresi terdiri atas operand dan operator, contoh ekspresi: “a + b”. Hasil Evaluasi dari sebuah Ekpresi adalah nilai yang sesuai dengan type operand yang dipakai

MACAM – MACAM EKSPRESI

Ø Ekspresi aritmatika

Ekspresi yang baik operand-nya bertipe numerik dan hasilnya juga bertipe numerik.

- Ekspresi aritmatika memakai operator aritmatika.
Pada ekspresi aritmetik terdapat 2 buah operator yaitu :
- Operator biner, yaitu ekspresi yang operatornya membutuhkan 2 buah operand.
Contoh $a + b$
- Operator Uner, yaitu “-” atau operator yang punya 1 operand contoh “-2”
Contoh 2. (Penulisan ekspresi dengan notasi algoritma)
 $T = 5/9 * (c + 32)$
 $Z = (2*x + y) / (5 * w)$
 $Y = 5((a+b) / (c*d) + m (p + q))$

Ø Ekspresi Perbandingan/ Relasional

Ekspresi relasional adalah ekspresi dengan operator $<$, \leq , $>$, \geq , $=$, dan \neq , not, and, or dan xor dengan menghasilkan nilai bertipe boolean (true atau false). Biasanya ekspresi Relasional disebut ekspresi boolean.

Ø Ekspresi Logika (Menggunakan Operator Boolean)

Adalah sebuah perintah yang akan mengeksekusi suatu perintah apabila kondisi tertentu terpenuhi.

Operator boolean digunakan untuk menghubungkan dua buah ekspresi logika. Pada kesempatan ini operator boolean yang akan dibahas adalah AND dan OR. Operator AND digunakan untuk menghubungkan dua buah ekspresi logika dan hasilnya TRUE jika kedua ekspresi tersebut bernilai TRUE.

Logika I	Operator Boolean	Lgika II	Hasil Akhir
FALSE		FALSE	FALSE

FALSE	AND	TRUE	FALSE
TRUE		FALSE	FALSE
TRUE		TRUE	TRUE

Operator OR dipergunakan untuk menghubungkan dua buah ekspresi logika dan hasilnya bernilai TRUE jika minimal salah satu ekspresi logika bernilai TRUE.

Logika I	Operator Boolean	Logika II	Hasil Akhir
FALSE		FALSE	FALSE
FALSE	OR	TRUE	TRUE
TRUE		FALSE	TRUE
TRUE		TRUE	TRUE

Contoh01

1. Program PenerimaanPegawai;
2. Uses Crt;
3. Var Umur : Integer;
4. Ket : String;
5. Begin
6. ClrScr;
7. Write('Umur calon pegawai : '); Readln(Umur);
8. If (Umur >= 20) AND (Umur <= 30) THEN Ket:='DITERIMA'
9. ELSE Ket:='DITOLAK';
10. Writeln('Status pegawai : ',Ket);
11. End.

D. Operator Aritmatika dan Logika

Operasi aritmatika adalah operasi yang melibatkan beberapa operator aritmatika seperti misalnya penjumlahan, pengurangan, perkalian dan pembagian untuk memproses 2 buah nilai atau lebih.

- + (jumlah) adalah operator untuk menjumlahkan dua buah angka atau lebih

- -(kurang) adalah jenis operator untuk memproses operasi pengurangan antara dua angka atau lebih
- * (kali) adalah operator yang digunakan untuk proses perkalian
- / (bagi) adalah operator yang digunakan untuk mendapatkan hasil bagi dari 2 buah angka
- MOD atau % (modular) adalah operator yang sering digunakan di dunia algoritma yang digunakan untuk mendapatkan sisa hasil bagi.

Operator baru yang tidak umum adalah MOD atau bisa disimbolkan dengan %, operator ini akan menghasilkan sisa hasil pembagian, misal $4\%2=0$. $5\%3=2$. Operator aritmatika biasanya digunakan untuk memproses perhitungan matematika yang dilibatkan di dalam program.

Operasi logika proses yang melibatkan beberapa operator logika seperti AND, OR, NOR, XOR dan operator logika lainnya yang menghasilkan nilai true dan false atau menghasilkan nilai benar dan salah. operasi logika ini sangat sering sekali digunakan di algoritma dan bahasa pemrograman dasar.

- and : menghasilkan nilai TRUE jika kedua operand bernilai TRUE
- or : menghasilkan nilai TRUE jika salah satu operand bernilai TRUE
- xor : menghasilkan nilai TRUE jika salah satu operand bernilai TRUE tetapi bukan keduanya bernilai TRUE
- ! (simbol tidak): menghasilkan nilai tidak TRUE
- && (simbol AND): menghasilkan nilai TRUE jika kedua operand bernilai TRUE
- || (simbol OR): menghasilkan nilai TRUE jika salah satu operand bernilai TRUE

E. Struktur Kontrol Pencabangan/Conditional

1) Struktur If

Digunakan untuk membandingkan suatu kondisi tertentu, jika kondisi tersebut benar, maka akan menjalankan ekspresi tertentu. Tetapi jika kondisi salah ekspresi didalam if tidak akan dijalankan.

Syntaksnya sbb :

If (kondisi) Then

..... Pernyataan.....

End If

2) Struktur If.... else

Pada

If ... Else ... kita bisa menentukan ekspresi lain yang akan dieksekusi jika kondisi yang dibandingkan oleh if tidak memenuhi syarat.

Syntaksnya sbb :

```
If (kondisi ) Then
Pernyataan
Else
Pernyataan
End If
```

3) Select Case

Digunakan untuk melakukan pemilihan kondisi yang jumlahnya banyak atau bertingkat tingkat.

Syntax nya sebagai berikut :

```
Select Case
ekspresi
Case
nilai
.....Pernyataan1.....
Case
nilai
.....Pernyataan2.....
Case
nilai
.....Pernyataan3.....
Case
Else
.....Pernyataan4.....
End Selec
```

F. Struktur Kontrol Perulangan/Looping

Pengulangan atau Loop sering juga disebut dengan iterasi adalah instruksi program yang memerintahkan suatu tugas diulang-ulang berdasarkan kondisi tertentu.

Pengulangan ini dapat dikendalikan oleh aplikasi tergantung pada kondisi yang diinginkan

1. While.... Wend

```
While condition
[statements]
Wend
```

Melakukan pengulangan terus menerus selama (while) suatu kondisi memenuhi syarat atau bernilai True
Dan akan berhenti jika kondisi bernilai False

2. For... next

```
For counter=
Start To
```

```

end
[Step step]
[statements]
[Exit For]
[statements]
Next

```

Digunakan untuk mengulangi suatu perintah dalam jumlah yang ditentukan

3. Keluar dari Perulangan

Untuk keluar dari loop baik

pada do while, do until maupun for next dapat dilakukan dengan perintah "Exit"

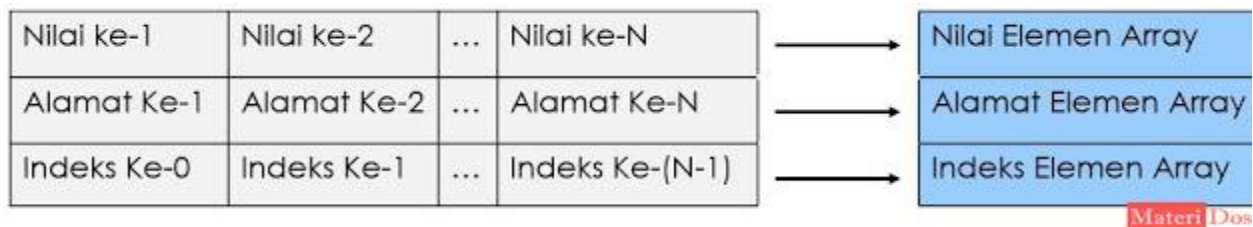
Contoh :

Exit Do

Exit Fo

G. Konsep Array

Array adalah sebuah variabel yang menyimpan sekumpulan data yang mempunyai tipe sama atau Array juga dapat disebut sebagai kumpulan dari nilai-nilai data yang bertipe sama (misal int) dalam urutan tertentu yang memakai sebuah nama yang sama. Setiap data tersebut menempati alamat memori atau lokasi yang berbeda-beda dan selanjutnya disebut dengan elemen array. Elemen array sendiri dapat diakses melalui indeks yang terdapat di dalamnya. Namun, penting sekali untuk di ingat bahwa dalam C++ indeks array selalu dimulai dari **0**, bukan **1**. Berikut ini gambar yang mengilustrasikan sebuah array.



Ilustrasi Array atau Ilustrasi Array Satu Dimensi

Macam-Macam Array

Array Satu Dimensi

Array satu dimensi merupakan array yang terdiri dari n buah kolom atau Array satu dimensi adalah array yang terdiri dari 1 subskrip array saja, setiap elemen array satu dimensi dapat diakses melalui indeks yang terdapat di dalamnya

Contoh Deklarasi Array Satu Dimensi:

```
tipe_data nama_array [jumlah_element];
```

Array Dua Dimensi

Array dua dimensi merupakan array yang terdiri dari n buah baris dan m buah kolom, bentuknya dapat sobat bayangkan seperti matriks atau tabel. dimana indeks pertama menunjukkan baris dan indeks kedua menunjukkan kolom. Array dua dimensi biasa digunakan untuk pendataan nilai, pendataan penjualan dan lain sebagainya.

Contoh Deklarasi Array Dua Dimensi:

```
tipe_data nama_array [jumlah_element_baris] [jumlah_element_kolom];
```

H.Konsep Fungsi Dalam Pemograman

Fungsi / Function adalah satu blok kode yang melakukan tugas tertentu atau satu blok instruksi yang di eksekusi ketika dipanggil dari bagian lain dalam suatu program. Tujuan pembuatan fungsi adalah : Baik data maupun fungsi harus dideklarasikan.

SUMBER

A. Alur Pemograman Komputer

Algoritma Pemograman : <https://teknojurnal.com/pengertian-algoritma-pemrograman/>

Flowchart : <https://www.it-jurnal.com/pengertian-flowchart/>

B. Perangkat Lunak Bahasa Pemograman

https://id.wikipedia.org/wiki/Bahasa_pemrograman

C. Konsep Penggunaan Tipe Data, Variabel, Konstanta, Operasi, dan Ekspresi

<http://cynthia281999.blogspot.com/2016/01/tipe-data-konstanta-ekspresi-operator.html>

D. Operator Aritmatika dan Logika

<http://referensisiswa.blogspot.com/2016/08/operasi-logika-dan-operasi-aritmatika.html>

E. Struktur Kontrol Pencabangan/Conditional

<https://docplayer.info/35901874-Bab-iii-pengenalan-struktur-kontrol-penyeleksian-dan-perulangan.html>

F. Struktur Kontrol Perulangan/Looping

<https://docplayer.info/35901874-Bab-iii-pengenalan-struktur-kontrol-penyeleksian-dan-perulangan.html>

G. Konsep Array

<http://www.materidosen.com/2017/05/konsep-array-dalam-c-lengkap-contoh.html>

H. Konsep Fungsi Dalam Pemograman

<http://setyo2511.blogspot.com/2017/01/bab-6-fungsi-by-setiyo-budi.html>