



PEMBELAJARAN PEMROGRAMAN PYTHON DENGAN PENDEKATAN LOGIKA ALGORITMA

Muhammad Romzi¹, Budi Kurniawan²

^{1,2}. Program Studi Manajemen Informatika, AKMI Baturaja, Baturaja

Korespondensi Email: ujromzi@gmail.com

Abstrak

Intisari-Bahasa pemrograman yang cukup populer saat ini salah satunya adalah python. Python digunakan oleh perusahaan besar dan juga para pengembang perangkat lunak untuk membuat aplikasi berbasis desktop, aplikasi berbasis web dan aplikasi mobile. Python juga banyak dipelajari di kampus, terutama kampus yang berbasis teknologi informasi. Metode yang digunakan adalah studi pustaka, dengan mempelajari konsep-konsep algoritma pemrograman, selanjutnya konsep tersebut diimplementasikan kedalam pembuatan program menggunakan python. Hasil penelitian ini menunjukkan bahwa pendekatan logika dan algoritma dapat memudahkan dalam pembelajaran bahasa pemrograman, kemudahan yang dimaksud adalah kemudahan dalam membuat program per struktur, yaitu struktur program runtunan, struktur program percabangan, dan struktur program perulangan. Pemahaman atas ketiga struktur ini diharapkan dapat menjadi bekal dalam membuat program yang lebih kompleks dengan menggunakan bahasa pemrograman, salah satunya yaitu pemrograman python.

Kata Kunci: Python, Logika, Algoritma

PYTHON PROGRAMMING LEARNING WITH A LOGICAL APPROACH ALGORITHM

Abstract

One of the most popular programming languages is python. Python is used by large companies as well as software developers to create desktop-based applications, web-based applications and mobile applications. Python is also widely studied on campus, especially on information technology-based campuses. The method used is literature study, by studying the concepts of programming algorithms, then these concepts are implemented into programming using python. The results of this study indicate that the logic and algorithm approach can facilitate learning programming languages, the ease in question is the ease in making programs per structure, namely the structure of the series program, the structure of the branching program, and the structure of the repetition program. An understanding of these three structures is expected to be a provision in making more complex programs using programming languages, one of which is python programming.

Keywords: Python, Logic, Algorithm

PENDAHULUAN

Python merupakan salah satu bahasa pemrograman yang banyak digunakan oleh perusahaan besar maupun para developer untuk mengembangkan berbagai macam aplikasi berbasis desktop, web dan mobile. Python diciptakan oleh Guido van Rossum di Belanda pada tahun 1990 dan namanya diambil dari acara televisi kesukaan *Guido Monty Python's Flying Circus*. Van Rossum mengembangkan Python sebagai hobi, kemudian Python menjadi bahasa pemrograman yang dipakai secara luas dalam industri dan pendidikan karena sederhana, ringkas, sintak intuitif dan memiliki pustaka yang luas (Schuerer & Maufrais 2010).

Kepopuleran penggunaan Python menempatkannya menjadi bahasa pemrograman yang mulai banyak dipelajari oleh mahasiswa terutama mahasiswa di kampus yang berbasis IT, guna menyelesaikan tugas kuliah, tugas akhir maupun tugas penelitian, untuk dapat menyelesaikan berbagai tugas pemrograman, seseorang perlu memahami algoritma, karena pada dasarnya program computer adalah implementasi dari algoritma.

Menurut Munir, algoritma berasal dari nama seorang ilmuwan Arab Abu Ja'far Muhammad ibnu Musa al Khuwarizmi seorang penulis kitab *Al Jabar Wal Muqabala*. Al Khuwarizmi dibaca orang barat menjadi algorism. Perubahan kata algorism menjadi algorithm disebabkan dikelirkannya kata algorism dengan arithmetic, sehingga akhiran *-sm* berubah

menjadi-*thm*, dalam bahasa Indonesia, kata *algorithm* menjadi *algoritma*.

Menurut Kadir (2018), algoritma adalah sekumpulan langkah rinci yang ditujukan untuk computer dalam menyelesaikan suatu masalah. Algoritma dibuat pada tahapan perancangan program. Algoritma memiliki peranan penting untuk menghubungkan antara keluaran yang dikehendaki dan masukan-masukan yang tersedia.

Karakteristik algoritma menurut Donal E. Knuth (dalam Munir, 2001), yaitu:

1. Finiteness (Keterbatasan): Algoritma harus berakhir setelah melakukan sejumlah langkah proses.
2. Definiteness (Kepastian): Setiap langkah algoritma harus didefinisikan dengan tepat dan tidak menimbulkan makna ganda.
3. Input (Masukan): Sebuah algoritma memiliki nol atau lebih masukan (input) yang diberikan kepada algoritma sebelum dijalankan.
4. Output (Keluaran): Setiap algoritma memberikan satu atau beberapa hasil keluaran
5. Effectiveness (Efektivitas)

Langkah-langkah algoritma dikerjakan dalam waktu yang “wajar”

Menurut Munir,dkk (2016) sebuah algoritma dapat dibangun dari tiga buah struktur dasar, yaitu struktur runtunan (*sequence*), struktur pemilihan (*selection*) 1. dan struktur pengulangan (*repetition*). Runtunan terdiri satu atau lebih instruksi yang dikerjakan secara berurutan dengan urutannya dalam algoritma. Pemilihannya itu instruksi yang dikerjakan berdasarkan kondisi tertentu atau syarat tertentu dimana suatu kondisi atau syarat tersebut dapat bernilai benar atau salah. Intruksi akan dilaksanakan manakala kondisi atau syarat bernilai benar, dan suatu instruksi tidak akan dikerjakan apabila kondisi atau syarat tidak terpenuhi. Pengulangan merupakan pengulangan sejumlah aksi yang sama sebanyak jumlah yang ditentukan atau kondisi yang diinginkan.

Salah satu teknik atau cara penulisan algoritma adalah dengan menggunakan kodesemu (*pseudo-code*) yang mendekati bahasa pemrograman tertentu, dan akan ditranslasi ke dalam kode program untuk dipraktekkan pada computer untuk melihat keluaran dari algoritma yang disusun.

Menurut Rachmat *pseudocode* berasal dari kata *pseudo* dan *code*, yang berarti kode semu. *Pseudocode* menyerupai kode program, bukannya kode program sebenarnya. *Pseudocode* menggunakan symbol-simbol yang mirip kode program yang ditulis dengan menggunakan suatu bahasa tertentu. Perbedaan *pseudocode* dengan

flowchart adalah terletak pada simbol yang digunakan, *flowchart* menggunakan symbol yang berbentuk gambar, sedangkan *pseudocode* menggunakan bahasa tingkat tinggi yang menyerupai kode program.

Penelitian ini dimaksudkan untuk memudahkan dalam pembelajaran pemrograman Python dengan menggunakan pendekatan logika dan algoritma, yang terdiri dari tiga struktur dasar, yaitu runtunan, pemilihan, dan perulangan.

METODE

Metode yang digunakan adalah studi pustaka, dengan mempelajari konsep-konsep algoritma pemrograman, selanjutnya konsep tersebut diimplementasikan kedalam pembuatan program menggunakan python. Pembuatan program dimulai dari : 1). Menjalankan Python Shell, 2). Pengenalan operator dasar pada python, 3). Penggunaan variabel, 4). Pengenalan tipe data, 5). Algoritma runtunan dan implementasi dalam pemrograman python, 6). Algoritma percabangan dan implementasi dalam pemrograman python, 7). Algoritma perulangan dan implementasi dalam pemrograman python

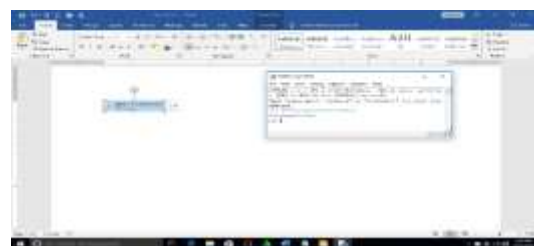
HASIL DAN PEMBAHASAN

Menjalankan Python dengan *Python Shell*
Pilih *Python Shell* pada start menu



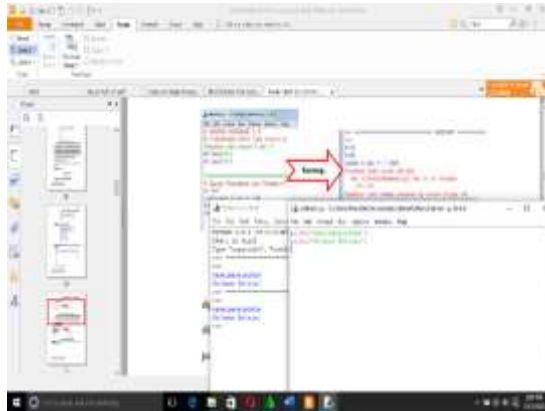
Gambar 1. Tools Python Shell

IDLE(PythonGUI) *Integrated Development and Learning Environment Python GUI* merupakan lingkungan belajar dengan tampilan GUI



Gambar 2. Python Shell

Statemen *print* digunakan untuk menampilkan tulisan, contoh diatas menampilkan tulisan 'Assalamualaikum' dilanjutkan dengan menekan tombol enter, maka akan diperoleh hasil tulisan tepat dibawah kode program. Program ini dijalankan oleh interpreter yang langsung menampilkan hasil setelah menekan tombol enter.



Gambar 3. Tampilan Editor Compiler

Pada gambar 3, menampilkan kode penulisan program menggunakan file, kode program dituliskan sekaligus dan akan dieksekusi setelah penulisan kode program selesai seluruhnya.

2. Operator

Tabel 1. Operator dasar pada python

Simbol	Arti
+	jumlahan
-	kurangan
*	kalikan
/	bagian

Contoh penggunaan operator pada python

```

rasiPenjumlahan
>> 4 + 5
10

rasiPerkalian
>> 9 * 9
81

rasiPembagian
>> 9 / 3
3

rasiCampuran
>> 3 * 3 / 3
3
    
```

```

rasiCampuran
>> 5 + 4 * 25
125
    
```

3. Penggunaan Variabel

Contoh penggunaan variable pada python

```

>>> harga = 65000
>>> jumlah = 10
>>> harga * jumlah
650000
    
```

4. Tipe Data

Tabel 2. Tipe Data

Tipe Data	Keterangan
String	Tipe data dasar yang dapat berupa karakter/kata/kalimat, penggunaan tipe data ini disertai dengan tanda petik tunggal / ganda
Integer	Tipe data yang menyatakan bilangan bulat
Float	Tipe data yang menyatakan bilangan pecahan (mengandung koma)
Boolean	Tipe data yang menyatakan dua buah nilai True (benar) bernilai satu atau False (salah) bernilai nol
Complex	Tipe data dengan angka real dan imajiner
Dictionary	Tipe data yang menyimpan data penunjuk dan nilai
Hexadecimal	Tipe data dalam format hexa yang memiliki base 16
List	Tipe data yang menyimpan berbagai tipe data yang isinya dapat berubah
Tuple	Tipe data yang menyimpan berbagai tipe data yang isinya tak dapat berubah

5. Algoritma Runtunan

Runtunan (*sequence*) terdiri satu atau lebih instruksi yang dikerjakan secara berurutan sesuai dengan urutannya dalam algoritma. Contoh program:

1). Menghitung Keliling Persegi Panjang

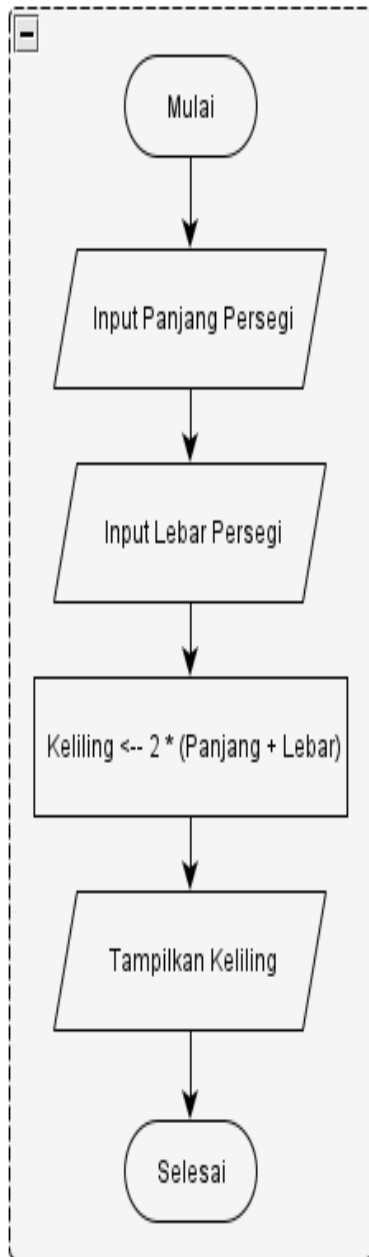
Algoritma Hitung Keliling Persegi Panjang

PJ □ Panjang persegi panjang

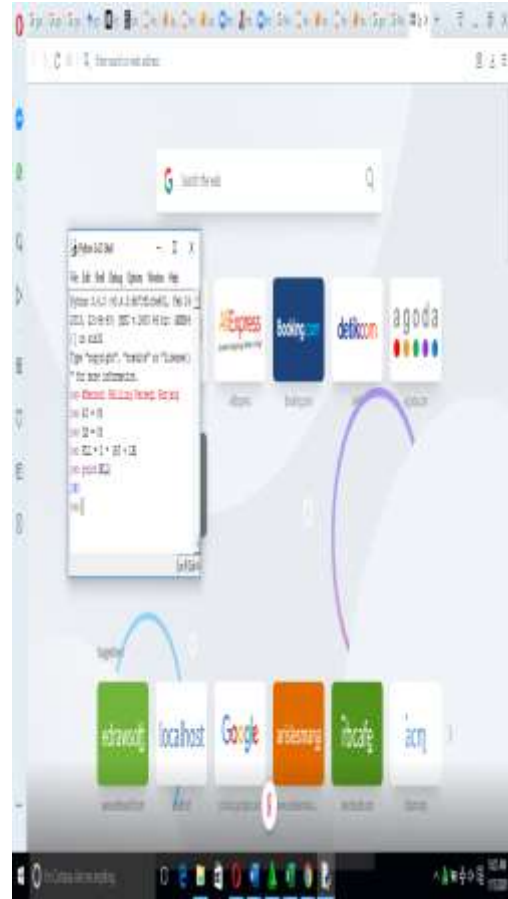
LB □ Lebar persegi Panjang

KLL □ 2 * (PJ + LB)

Tampilkan KLL

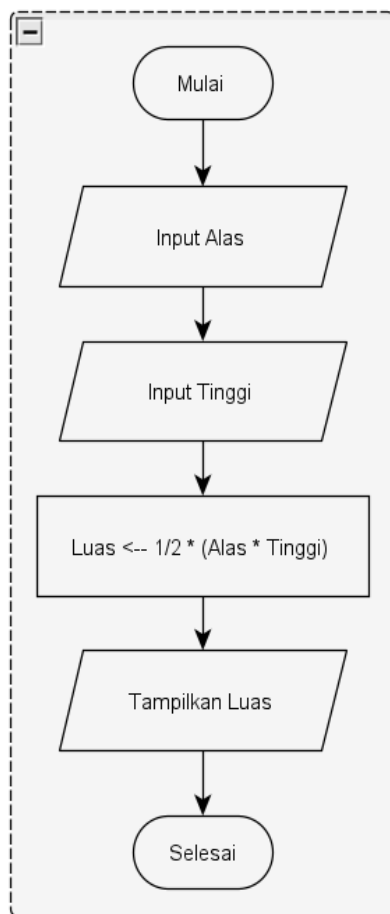


Gambar 4. Flowchart Hitung Keliling Persegi

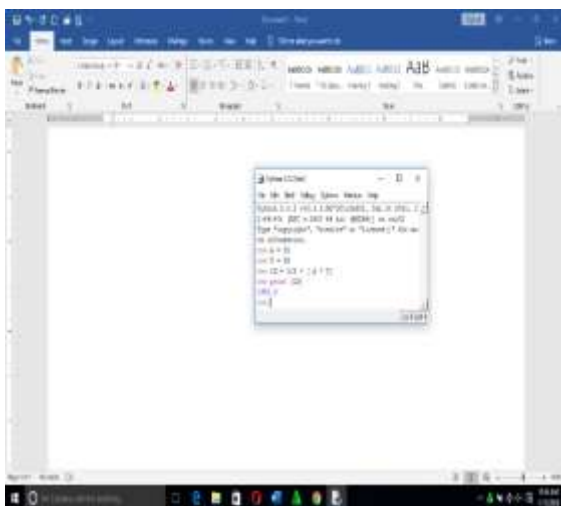


Gambar 5. Program Hitung Keliling Persegi

2). Menghitung Luas Segitiga
 Algoritma Hitung Luas Segitiga
 $A \square$ Alas Segitiga
 $T \square$ Tinggi Segitiga
 $LS \square \frac{1}{2} * (A * T)$
 Tampilkan LS



Gambar 6. Flowchart Hitung Luas Segitiga



Gambar 7. Program Hitung Luas Segitiga

Pengulangan merupakan pengulangan sejumlah aksi yang sama sebanyak jumlah yang ditentukan atau kondisi yang diinginkan.

6. Algoritma percabangan / pilihan

Pemilihan (*selection*), yaitu instruksi yang dikerjakan berdasarkan kondisi tertentu atau syarat tertentu dimana suatu kondisi atau syarat tersebut dapat bernilai benar atau salah. Instruksi akan dilaksanakan manakala kondisi atau syarat bernilai benar, dan suatu instruksi tidak akan dikerjakan apabila kondisi atau syarat tidak terpenuhi.

Pemilihan atau percabangan menggunakan statemen If, If/Else, If/Elif/Else. Statemen If digunakan saat terdapat satu pilihan, statemen If/Else digunakan saat terdapat dua pilihan, statemen If/Elif/Else digunakan apabila terdapat lebih dari dua pilihan. Lebih jelasnya perhatikan contoh program berikut:

Contoh program

1). Penggunaan statemen IF

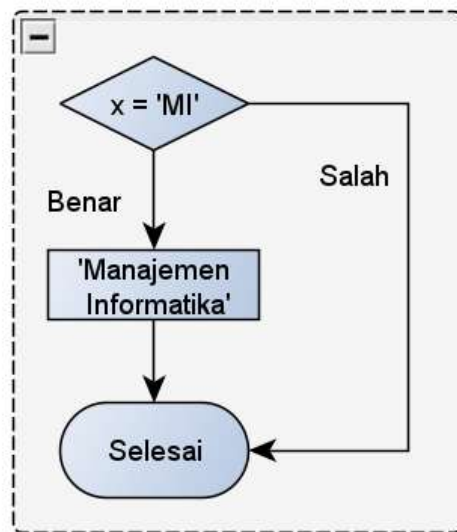
Menentukan jurusan berdasarkan kode jurusan

Algoritma Menentukan NamaJurusan

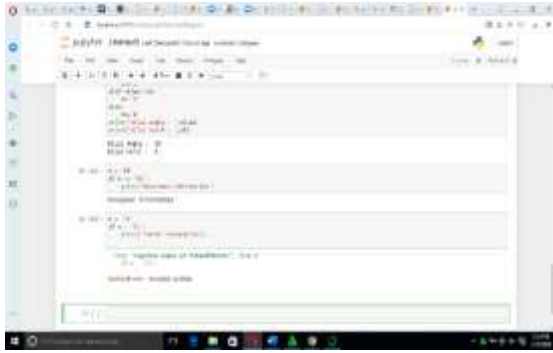
$x \leftarrow \text{KodeJurusan}$

If $x == \text{'MI'}$ then

Tampilkan 'Manajemen Informatika'



Gambar 8. Flowchart Menampilkan Nama Jurusan



Gambar 9. Program Menampilkan Jurusan

2). Penggunaan statemen IF/Else

Menentukan kelulusan berdasarkan nilai

Algoritma Menentukan Kelulusan Berdasarkan Nilai

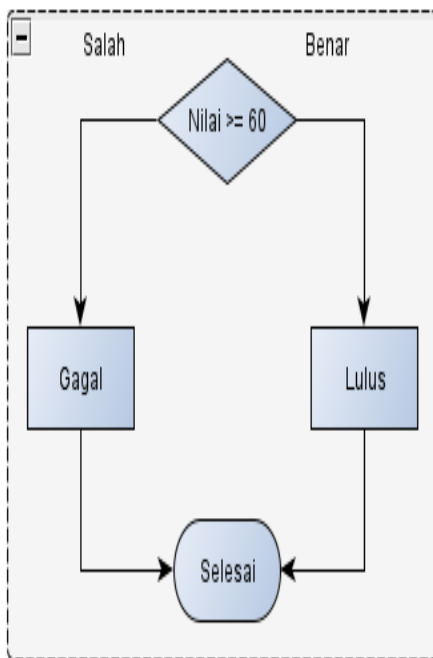
Nilai \square Nilai yang diperoleh

If (Nilai ≥ 60) then

Tampilkan 'Anda Lulus'

Else

Tampilkan 'Anda Gagal'



Gambar 10. Flowchart Menentukan Kelulusan



Gambar 11. Program Menampilkan Kelulusan

3). Penggunaan statemen IF/Elif/Else

Menentukan Nilai Huruf (A,B,C,D, dan E)

Algoritma Menentukan Nilai Huruf Berdasarkan

Nilai Angka

Nilai \square Nilai yang diperoleh

If (NA ≥ 80) then

NH 'A'

Else

If (NA ≥ 66) then

NH 'B'

Else

If (NA ≥ 56) then

NH 'C'

Else

If (NA ≥ 40) then

NH 'D'

Else

NH='E'

Tampilkan NH



Gambar 12. Program Menampilkan Nilai Huruf

7. Algoritma perulangan

Pengulangan merupakan pengulangan sejumlah aksi yang sama sebanyak jumlah yang ditentukan atau kondisi yang diinginkan. Terdapat dua statemen perulangan pada python, yaitu statemen For dan statemen While

1). Statemen For

Contoh program menampilkan angka 0 s.d 9

Algoritma Menampilkan Deret Angka

X □ Range Nilai

For I □ in range (X)

Tampilkan(I)



Gambar 13. Program Menampilkan Deret Angka

2). Statemen While

Contoh program menampilkan angka 1 s.d 5

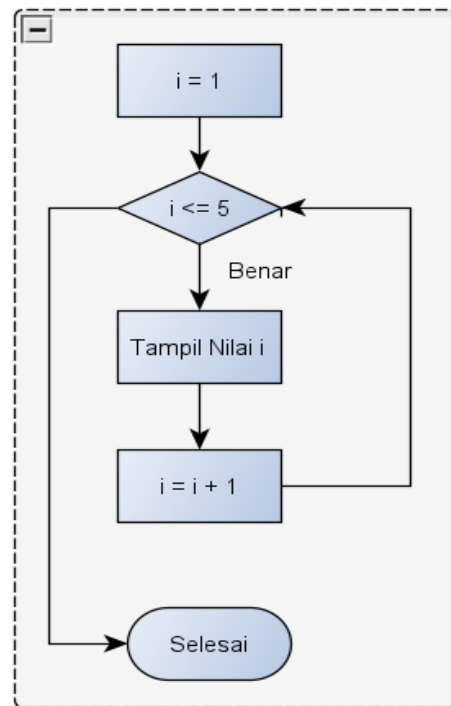
Algoritma Menampilkan Deret Angka

i □ Nilai Awal

while i <= n

Tampilkan(i)

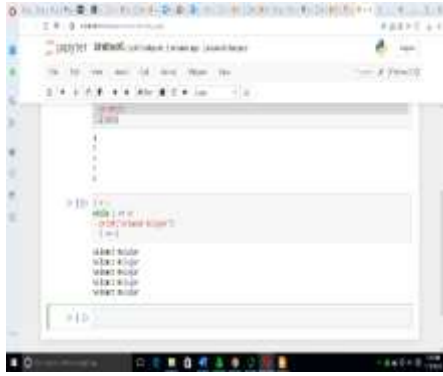
i □ i+1



Gambar 14. Flowchart Menampilkan Deret Angka



Gambar 15. Program Menampilkan Deret Angka



Gambar 16. Program Menampilkan Selamat Belajar Berulang

KESIMPULAN

Dari penjelasan di atas dapat disimpulkan bahwa Penguasaan algoritma pemrograman dan penguasaan bahasa pemrograman itu sendiri, dan memperbanyak latihan dalam menyelesaikan masalah pemrograman. Kode program python dapat dijalankan perbaris kode (*interpreter*), dan juga kode program dapat diterjemahkan sekaligus (*compiler*).

Pendekatan logika dan algoritma dapat memudahkan dalam pembelajaran bahasa pemrograman, di karenakan algoritma memberikan langkah-langkah logis dalam penyelesaian masalah. Algoritma terdiri dari tiga struktur, yaitu runtunan, struktur percabangan dan struktur perulangan.

Statemen yang digunakan untuk menampilkan hasil pada python adalah statemen ***print()***, statemen percabangan pada python dapat menggunakan statemen If, If/Else, dan If/Elif/Else, sedangkan statemen yang digunakan untuk perulangan adalah ***For***, dan statemen ***While***.

DAFTAR PUSTAKA

- Kadir, A. 2018. Dasar Logika Pemrograman Komputer. Cetakan Kedua. Elexmedia Komputindo.
- Munir,R. 2001. Algoritma dan Pemrograman Dalam Bahasa Pascal dan C, Buku 1. Informatika Bandung
- Munir, Rinaldi dan Lidya 2016. Algoritma dan Pemrograman Dalam Bahasa Pascal dan C, dan C++. Informatika Bandung
- Schuerer, Katja dan Corinne Maufrais (2010). Introduction to Programming using Python. Boston: Pearson, hal. 1–242. ISBN: 0132747189.