**LAPORAN RESMI**

**MODUL II**

**(PENYELEKSIAN KONDISI)**

**ALGORITMA PEMROGRAMAN**



**NAMA : Giraldo Stevanus Nainggolan**

**N.R.P : 220441100064**

**DOSEN : Sigit Susanto Putro, S.Kom., M.Kom.**

**ASISTEN : M . Badrut Tamam**

**TGL PRAKTIKUM : 4 Oktober 2022**

**Disetujui : 10 Oktober 2022**

**Asisten**

**M . Badrut Tamam**

**21.04.411.00032**

**LABORATORIUM BISNIS INTELIJEN SISTEM**

**PRODI SISTEM INFORMASI**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK**

**UNIVERSITAS TRUNOJOYO MADURA**

# **BAB I PENDAHULUAN**

## **Latar Belakang**

Penyeleksian kondisi adalah rangkaian proses dalam membuat sebuah kegiatan pengalokasian yang mampu merancang atau memilah kata menjadi terstruktur baik adanya (sempurna). Kata “Penyeleksian” sering di jumpai pada hamper semua kegiatan dalam aktifitas manusia termasuk: arsitektur, programan neraca sipil, bagian kesehatan, segala bentuk pelatihan atau webinar, perlombaan atau merebut sebuah gelar (kejuaraan), mencari kerja, instansi pendidikan, dll. Begitu juga dengan kata “Kondisi” sering juga kita jumpai dalam kegiatan sehari-hari dan banyak digunakan atau terlintas sebagai bahan pembicaraan seperti: menyatakan lokasi, kondisi fisik pada objek, tolak ukur nuansa instansi tersendiri, dan segala apa yang kita inginkan atau cita-citakan.

Pada umumnya kata “Penyeleksian Kondisi” menurut Menurut M. Manularang (2011) (seperti yang dikutip I Komang ardana, Ni Wayan Mujiati, dan I Wayan Mudiartha Utama) diartikan sebagai salah satu istilah :”***freepare of selections”***merupakan hal yang sudah mutlak menjadi penyandang kebebasan yang terstruktur namun terikat seperti: pandangan proses, cara pembuatan, penyeleksian, penyaringan, hasil distribusi dari segala sisi, dan pemilihan. Begitu banyak yang terjadi dalam hidup ini sehingga harus menjadi persiapan kita untuk berinovasi melakukan berubahan menuju hal yang lebih banyak karna diera modrn harus lebih bijak dan antusias

## **Tujuan**

* Mampu Menjelaskan Penyeleksian pada Algoritma Pemrograman
* Mampu Menjelaskan Fungsi Lambang Pada Alogaritma
* Mampu memahami macam – macam penyeleksian kondisi
* Menciptakan kemampuan pengoprasian dasar bahasa pemrograman

**Bab II**

**DASAR TEORI**

## **Pengertian Penyeleksian Kondisi**

Pada umumnya dalam membuat program, selalu ada seleksi dimana diperlukan pengecekan suatu kondisi untuk mengarahkan program agar berjalan sesuai keinginan. Pada Python untuk melakukan suatu pengecekan kondisi, terdapat tiga macam statemen. Antara lain :

1. Perintah if
2. Peintah if – elif – else
3. Perintah if bersarang
   1. **Perintah If**

Bentuk umum perintah if

1. If (kondisi) Statemen:

Statemen if digunakan untuk melakukan penyeleksian dimana jika kondisi bernilai benar maka progam akan mengeksekusi statemen dibawahnya. Dalam python, untuk penulisan pengkondisian dan statemen di pisahkan oleh tanda titik dua ( : ). Contohnya,



Untuk setiap penulisan perintah if setelah penentuan kondisi maka dilanjutkan dengan penulisan tanda titik dua ( : ). Tanda titik dua ini berarti jika kondisi bernilai benar maka statemen-statemen setelah tanda titik dua akan di jalankan.

**2.3 Perintah If – Else**

Statemen if – else digunakan untuk melakukan penyeleksian kondisi dimana jika kondisi bernilai benar maka program akan mengeksekusi statemen 1. Namun, jika nilai kondisi bernilai salah maka statemen 2 yang akan dieksekusi.

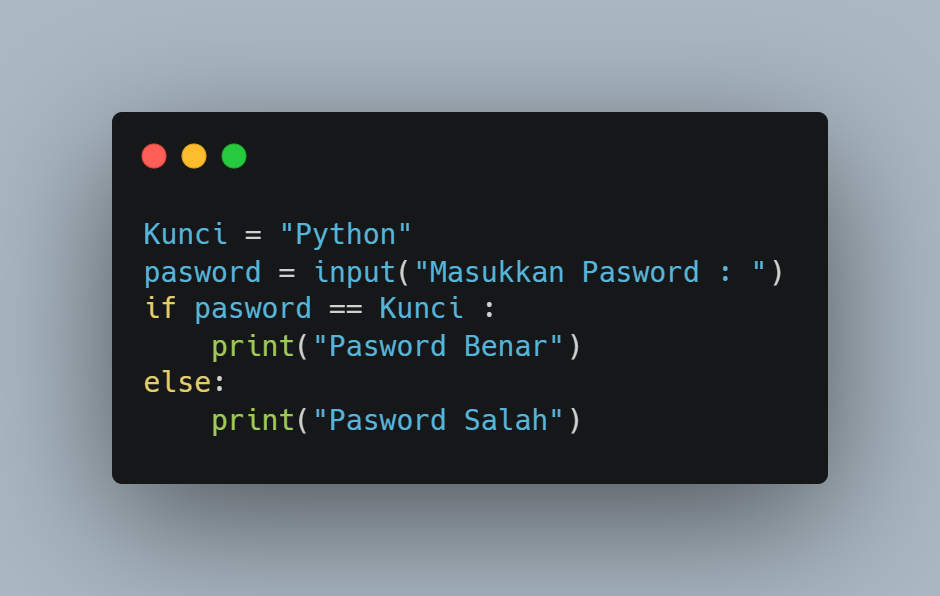
**Bentuk umum perintah if – else:**

**if ( kondisi ) :**

**statemen 1**

**else :**

**statemen 2**



**2.4 Perintah If – Elif - Else**

**If ( kondisi 1 ) :**

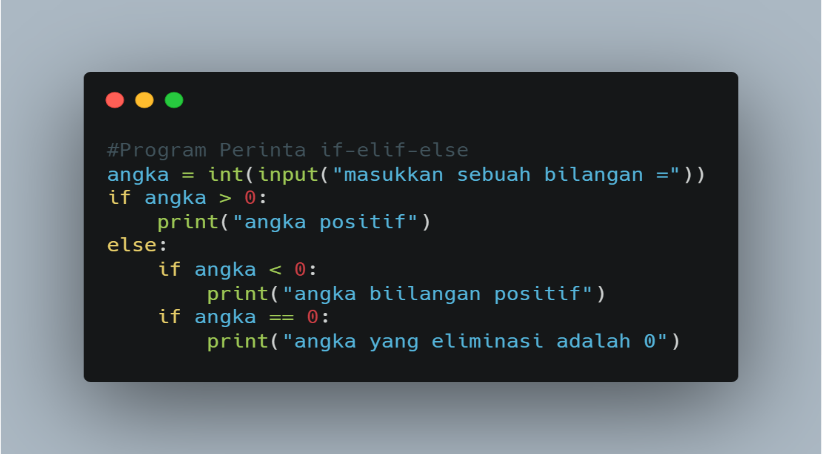
**statemen**

**Elif ( kondisi 2 ) :**

**statemen**

**Else:**

**Statemen**

****

**2.5 If Bersarang**

Kondisi bersarang adalah suatu kondisi di dalam kondisi tertentu, Jika terdapat 2 cabang kondisi maka di dalam salah satu cabang kondisi tersebut dapat pula diisi suatu komisi tertentu. Misalnya

**#Program If Bersarang**

**x = int (“input(“Masukkan nilai x=”))**

**y = int (“input(“Masukkan nilai y=”))**

**if x == y:**

**print (“nilai”, x , “dan “ ,y ,”mempunyai nilai yang sama”)**

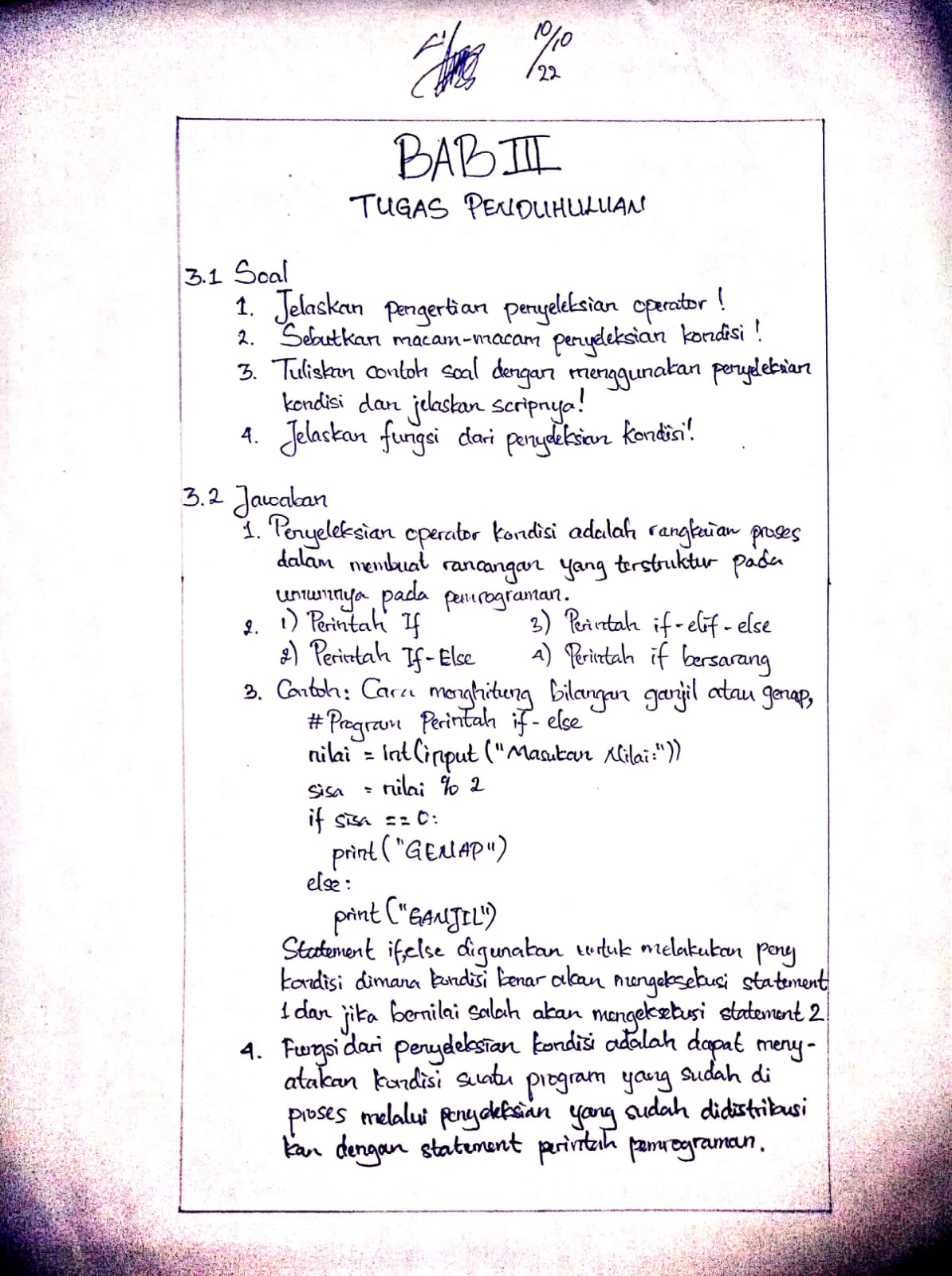
**else :**

**if x > y :**

**print (x, “lebih besar dari”, y)**

**if x < y :**

**print (x, “lebih besar dari”, y)**



# **BAB IV IMPLEMENTASI**

## **Source Code**

Menghitung program python untuk menentukan sebuah bilangan termasuk bilangan positif atau negatif. Yaitu langkah pertamanya adalah masukkan angka inputan berupa integer untuk memprogram bilangan. Kedua gunakan if pada angak > 0 untuk memerintahkan bilangan positif. Jika salah gunakan if pada angaka < 0 untuk memerintahkan bilagan negative.

Membuat sebuah program berupa permainan (gunting/batu/kertas) gunakan pemisah (“==========”) karna source codenya banyak. Masukkan perintah yaitu (0= Batu, 1 = Gunting, 2 = Kertas). Membuat dua perintah programan user dengan permainan 1 & 2 dengan mengeluarkan peintah sesuaikan usernya dalam integer. Untuk menentukan batu gunakan perintah if, elif, else untuk seri, menang/kalah. Untuk menentukan gunting gunakan perintah if, elif, else untuk seri, menang/kalah. Untuk menentukan kertas gunakan perintah if, elif, else untuk seri, menang/kalah. Lalu tambahkan else dengan perintah (“Angka Yang Anda Masukkan Salah”). Lalu gunakan penambahan pemisah (“===TERIMAKASIH===”)

## **Hasil**



Dalam pengaelolahan hasil di harapkan menyimpan dokumen (Save/save us). Lakukan refresh atau run program nanti akan di perintahkan memasukkan bilangan dan tinggal menyesuaikan dengan perintah codingan kitanya.

Dalam menghitung sebuah program berupa permainan (gunting/batu/kertas) dalam layar akan muncul perintah otomatis 0 = Batu, 1 = Gunting, 2 = Kertas. Kemudian masukkan pilihan mu dan masukkan pilihan lawanmu. Otomatis akan memunculkan inputan hasil: kedua pilihan hasilnya sama atau seimbang (SERI), dan akan muncul kalah/menang.

# **BAB V PENUTUP**

## **Analisa**

Dari hasil praktikum, praktikan menganalisa bahwa sebuah proses pengolahan data harus memiliki pola pikir yang teliti dan cepat merespon karna semua materi yang padat ini lebih mengacuh kepada penyederhanaan kalimat untuk menyelesaikan suatu masalah.

Sebuah praktikum harus melakukan pemahan dan memiliki perulangan yang banyak terutama perbanyak praktek atau menguji coba materi yang rutin supaya paham. Dalam pemahaman aritmatika yang berlandasan pada pemrograman dasar adalah aktivisa yang umum sehingga semua materi atau pelajar harus mampu memahaminya dengan cepat.

Dari hasil praktikum semua dituntut maksimal dan sempurna karna pada dasarkan kita harus mampu belajar lebih fleksibel dan berilmu menyesuaikan dengan keadaan, setiap seorang programmer mampu menganalisa sebuah kerusakan (eror) itu merupakan tuntutan yang menjadi keharusan mempelajarinya serta menahan sabar dalam proses mencari sebuah kesalahan baik kecil maupun besar.

## **Kesimpulan**

1. Materi perkuliahan jadi lebih paham
2. Memperkuat relasi kelas
3. Melatih cara berargumen dengan pemahaman sederhana
4. Melatih ketelitian & hati-hati dalam pengerjaan suatu tugas
5. Mengerti cara berlogika secara sederhana pada salah satu bahasa pemrograman begitu juga membuat perintah programan game (gunting, batu, dan kertas)