

## PERANGKAT LUNAK APLIKASI PEMBELAJARAN FLOWCHART

*Oleh:*

**Agustinus Zalukhu<sup>1</sup>, Swingly Purba<sup>2</sup>, Dedi Darma<sup>3</sup>**

Mahasiswa Teknik Informatika, Fakultas Teknologi Industri<sup>1</sup>

Dosen Teknik Informatika, Fakultas Teknologi Industri<sup>2,3</sup>

Institut Sains Dan Teknologi TD Pardede

E-mail:

<sup>1</sup>[agustinuszalukhu93@gmail.com](mailto:agustinuszalukhu93@gmail.com), <sup>2</sup>[swinglypurba@istp.ac.id](mailto:swinglypurba@istp.ac.id),

<sup>3</sup>[dedidarma@istp.ac.id](mailto:dedidarma@istp.ac.id)

### ABSTRAK

*Flowchart* adalah penggambaran secara grafik dari langkah-langkah dan urutan-urutan prosedur dari suatu program. *Flowchart* menolong analis dan programmer untuk memecahkan masalah kedalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternative-alternatif lain dalam pengoperasian. *Flowchart* biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan di evaluasi lebih lanjut. Perangkat lunak pembelajaran implementasi *flowchart* ini dapat membantu dalam memahami algoritma-algoritma yang ada dalam menampilkan proses dan tahapan-tahapan dalam pembuatan *flowchart*. Perangkat lunak pembelajaran implementasi *flowchart* ini mampu merancang suatu *flowchart* berbasis objek tanpa perlu mengisi kode program isi instruksi dari masing-masing objek *flowchart*. Perangkat lunak pembelajaran implementasi *flowchart* ini juga menyediakan objek-objek yang dibutuhkan dalam pembuatan suatu algoritma *flowchart* sehingga mempermudah pengguna dalam pembuatan suatu diagram alir (*flowchart*).

**Kata kunci :** Aplikasi, Pembelajaran, Algoritma, Flowchart dan Visual Basic 6

### ABSTRACT

*flowchart* is a graphic depiction of the steps and procedural sequences of a program. *Flowcharts* help analysts and programmers to break down problems into smaller segments and help in analyzing other alternatives in operations. *Flowcharts* usually make it easier to solve a problem, especially problems that need further study and evaluation. This *flowchart* implementation learning software can help in understanding existing algorithms in displaying the process and stages in making flowcharts. This *flowchart* implementation learning software is capable of designing an object-based flowchart without the need to fill in the program code containing the instructions for each flowchart object. This *flowchart* implementation learning software also provides the objects needed to create a flowchart algorithm, making it easier for users to create a flowchart.

**Keywords:** Applications, Learning, Algorithms, Flowcharts and Visual Basic 6

## 1. PENDAHULUAN

Teknologi di zaman sekarang telah semakin berkembang pesat. dengan perkembangan teknologi, akan berpengaruh dan dirasakan oleh manusia pada memenuhi kebutuhan sehari-hari (Ova Vaturonman Galih, 2022), salah satunya menggunakan hadirnya berbagai aplikasi yang mampu digunakan oleh user di menyelesaikan tugasnya. perangkat lunak ialah suatu perangkat yg dirancang sang programmer buat bisa menuntaskan tugas-tugas eksklusif buat bidang tersendiri.

Algoritma memegang peranan penting dalam bidang pemrograman. Sebegitu pentingnya suatu algoritma, sehingga perlu dipahami konsep dasar algoritma. Apalagi untuk seorang programmer, tentu diperlukan suatu algoritma sehingga dapat membuat program yang lebih efektif dan efisien. Bagi kebanyakan orang, algoritma sangat membantu dalam memahami konsep logika pemrograman.

Bagi seorang programmer, algoritma digunakan untuk membuat modul-modul program. Salah satu metode untuk menyajikan algoritma secara tepat adalah dengan menggunakan flowchart (diagram alir). Flowchart adalah gambaran dalam bentuk diagram alir dari algoritma-algoritma dalam suatu program, yang menyatakan arah alur program tersebut.

Terkadang seorang mengalami kesulitan dalam membuat flowchart khususnya mahasiswa yang baru melekat pemrograman. Untuk itu penulis merancang perangkat lunak yang dapat mengimplentasikan flowchart menjadi output program. Adapun penulis mengambil judul **“Perangkat Lunak Aplikasi Pembelajaran Flowchart”**.

## 2. TINJAUAN PUSTAKA

### 2.1. Aplikasi

Aplikasi merupakan perangkat lunak yang dimasukkan atau terdapat dalam komputer dan

memiliki fungsi-fungsi khusus. perangkat lunak (software) adalah program komputer yang terasosiasi dengan dokumentasi perangkat lunak seperti dokumentasi kebutuhan, model desain, dan cara penggunaan (user manual). Sebuah program komputer tanpa terasosiasi dengan dokumentasinya maka belum dapat dikatakan perangkat lunak (software). Proses perangkat lunak (software process) adalah sekumpulan aktivitas yang memiliki tujuan mengembangkan atau mengubah perangkat lunak.

Secara umum proses perangkat lunak terdiri dari :

1. Pengumpulan Spesifikasi (Specification), yaitu mengetahui apa saja yang harus dapat dikerjakan sistem perangkat lunak dan batasan pengembangan perangkat lunak.
2. Pengembangan (Development), yaitu pengembangan perangkat lunak untuk menghasilkan perangkat lunak.
3. Validasi (Validation), yaitu memeriksa apakah perangkat lunak sudah memenuhi kebutuhan pelanggan (Customer).
4. Evolusi (Evolution) yaitu mengubah perangkat lunak untuk memenuhi perubahan kebutuhan pelanggan (Customer).

### 2.2. Pembelajaran

Yang dimaksud dengan pembelajaran adalah suatu proses atau kegiatan belajar. Proses pembelajaran mengandung lima komponen komunikasi, guru (komunikator), bahan

pembelajaran, media pembelajaran, siswa (Komunikasikan), dan tujuan pembelajaran. Jadi, Media pembelajaran adalah segala sesuatu yang dapat digunakan untuk menyalurkan pesan (bahan pembelajaran), sehingga dapat merangsang perhatian, minat, dan perasaan siswa dalam kegiatan belajar untuk mencapai tujuan belajar.

### 2.3. Flowchart










Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart sistem merupakan suatu urutan proses dalam sistem dengan menunjukkan alat dari media input, output serta jenis media yang digunakan untuk penyimpanan dalam proses pengolahan data sedangkan flowchart program merupakan suatu bagan dengan simbol-simbol tertentu yang menggambarkan suatu urutan dari proses secara detail dan berhubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program. Jika seseorang analis dan programmer yang akan membuat flowchart, terdapat beberapa petunjuk yang harus diperhatikan, seperti berikut:

1. Flowchart dibuat mengikuti prosesnya dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktivitas yang tergambar harus didefinisikan secara hati-hati dan didefinisikan harus dapat dipahami oleh pembacanya.
3. Setiap aktivitas yang dimulai dan diakhiri harus ditentukan secara jelas.
4. Setiap langkah-langkah dari aktivitas harus dijabarkan dengan menggunakan

deskripsi kata kerja.

5. Langkah-langkah dari setiap aktivitas harus berada pada urutan yang benar.
6. Lingkup dan range dari aktivitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktivitas yang sedang digambarkan tidak perlu digambarkan pada flowchart yang sama. Simbol konektor harus digunakan dan percabangannya diletakkan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.
7. Menggunakan simbol-simbol flowchart yang standar.

Tabel 1 Simbol Flowchart  
Simbol Nama Fungsi  
Terminator Untuk mengawali.

SIMBOL	NAMA	FUNGSI
	TERMINATOR	Permulaan/akhir program
	GARIS ALIR (FLOW LINE)	Arah aliran program
	PREPARATION	Proses inisialisasi/pemberian harga awal
	PROSES	Proses perhitungan/proses pengolahan data
	INPUT/OUTPUT DATA	Proses input/output data, parameter, informasi
	PREDEFINED PROCESS (SUB PROGRAM)	Permulaan sub program/proses menjalankan sub program
	DECISION	Perbandingan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	ON PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada satu halaman
	OFF PAGE CONNECTOR	Penghubung bagian-bagian flowchart yang berada pada halaman berbeda

### 3. METODE PENELITIAN

- a. Program Adalah suatu rancangan struktur, desain, kode skema, maupun bentuk yang lainnya dengan yang disusun sesuai alur Algoritma dengan tujuan mempermudah suatu permasalahan. sebuah program biasanya disebut juga dengan istilah Aplikasi, tujuannya adalah mempermudah suatu hal agar pekerjaan bisa lebih produktif dan lebih efisien. Adapun istilah programmer yaitu seseorang yang membuat atau merancang suatu Program tersebut. Sebuah program yang dapat dibaca oleh manusia biasa disebut dengan Kode Sumber atau source code.
- b. Secara istilah, pengertian Aplikasi adalah suatu program yang siap untuk digunakan yang di buat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta pengguna aplikasi lain yang di gunakan oleh suatu sasaran yang akan di tuju, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu teknik pemrosesan data aplikasi yang di ininginkan atau di harapkan maupun pemresesan data yang di harapkan.
- c. Algoritma adalah kunci dari bidang ilmu komputer, dan pada dasarnya setiap hari kita melakukan aktivitas algoritma. Kata algoritma berasal dari sebutan Algorizm (Abu Abdullah Muhammad Ibn Musa Al Khwarizmi, ahli matematika Uzbekistan). Algoritma adalah kumpulan instruksi yang dibuat secara jelas untuk menunjukan langkah-langkah penyelesaian suatu masalah. Langkah-langkah dalam algoritma harus logis dan harus dapat ditentukan bernilai salah atau benar.

Pembuatan algoritma pada sistem komputer harus selalu dikaitkan dengan kebenaran algoritma kompleksitas (lama dan jumlah waktu proses dan penggunaan memori). Untuk itu agar algoritma berjalan dengan baik diperlukan kriteria Algoritma yang baik, yaitu:

1. Input : algoritma dapat memiliki nol atau lebih inputan dari luar.
2. Output : algoritma harus memiliki minimal satu buah output keluaran.
3. Definiteness (pasti) : algoritma memiliki instruksi-instruksi yang jelas dan tidak ambigu.
4. Finiteness (ada batas) : algoritma harus memiliki titik berhenti (stopping role).
5. Effectiveness (tepat dan efisien) : algoritma sebisa mungkin harus dapat dilaksanakan dan efektif. Contoh instruksi yang tidak efektif adalah:  $A = A + 0$  atau  $A = A * 1$

Berikut dibawah ini contoh algoritma sederhana untuk menulis surat, maka kita perlu melakukan beberapa langkah yaitu:

1. Mempersiapkan kertas dan amplop
2. Mempersiapkan alat tulis (pena, pensil)
3. Mulai menulis
4. Memasukkan kertas ke dalam amplop
5. Pergi ke kantor pos

Adapun beberapa contoh singkat algoritma pada kehidupan sehari-hari, seperti Tabel 1. dibawah ini

**Tabel 2. Beberapa contoh algoritma pada kehidupan sehari-hari**

No	Proses	Algoritma	Contoh langkah dalam algoritma
1.	Membuat Kue	Resep kue	Masukan telur ke dalam wajan, kocok sampai mengembang
2.	Membuat pakaian	Pola pakaian	Gunting kain dari pinggir kiri bawah ke arah kanan sejauh 5 cm
3.	Kegiatan sehari-hari	Jadwal harian	Pukul 15.00: tidur siang Pukul 16.00: membuat PR
4.	Memainkan musik	Papanot balok	Not balok
5.	Mengisi voicer HP	Panduan pengisian	Tekan 888 Masukan kode voucher

Pada tabel contoh diatas dijelaskan secara besar cara menyelesaikan suatu masalah dengan algoritma. Masih ada detail-detail yang harus dilakukan untuk dapat menyelesaikan masalah. Namun pada sistem komputer, tidak mengenal algoritma dengan langkah besar seperti pada tabel diatas. Komputer hanya dapat menyelesaikan satu per satu langkah dengan instruksi yang jelas dan terperinci sesuai dengan logika.

Pada Gambar 1. mengilustrasikan suatu masalah akan dapat diatasi dengan algoritma hingga menghasilkan solusi yang berguna.

**Gambar 1. Proses Penyelesaian Masalah**

#### d. Aturan Penulisan Teks Algoritma

Terdapat aturan penulisan teks algoritma agar komputer dapat menjalankan instruksi yang diberikan oleh pengguna komputer. Aturannya adalah setiap algoritma akan selalu terdiri dari :

1. **JUDUL ALGORITMA** → Algoritma  
NAMA ALGORITMA  
Yakni menjelaskan tentang algoritma, berisi uraian singkat cara kerja

program, kondisi awal dan akhir dari program → spesifikasi algoritma.

#### 2. DEKLARASI

Maksudnya semua nama yang dipakai, meliputi nama file, nama variable, nama konstanta, nama prosedur serta nama fungsi.

#### 3. DESKRIPSI

Yaitu semua langkah/aksi yang digunakan pada algoritma

Contoh:

a). Kepala algoritma: Algoritma Luas\_Lingkaran { Menghitung luas lingkaran dengan ukuran jari-jari tertentu .Algoritma menerima masukan jejari lingkaran, menghitung luasnya, dan menyajikan hasilnya ke piranti keluaran }

b) Deklarasi algoritma:

DEKLARASI { nama konstanta }

const PHI = 3.14; { Nilai phi = 22/7 }

{ nama peubah } var R : real; { input jejari lingkaran bilangan riil }  
l\_Lingkaran : real; { luas lingkaran bilangan riil }

{ nama sub program } procedure TUKAR (input/output A:integer, input/output B:integer)

{Mempertukarkan nilai A dan B.Parameter A dan B sudah terdefinisi nilainya.Setelah pertukaran, A berisi nilai B dan B berisi nilai A }

c) Deskripsi algoritma:

{ Baca data jejari lingkaran R.Jika R <= 0 tulis pesan data salah, selain itu hitung luas lingkaran. Tampilkan luas lingkaran. }  
baca(R);  
jika R <= 0 then tulis ("Data salah !")

selain itu  $l_{\text{Lingkaran}} = \text{PHI} \times R \times R$ ;  
 tulis( $l_{\text{Lingkaran}}$ );

#### e. Metode Penyajian Algoritma

French, C.S. (1984)

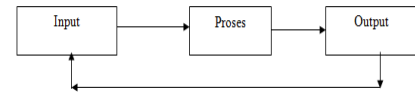
menyatakan sejumlah konsep yang mempunyai relevansi dengan masalah rancangan program yaitu kemampuan komputer, kesulitan dan ketepatan. Penerapan dari konsep tersebut biasanya digunakan dalam rancangan algoritma. Dalam merancang sebuah algoritma, Fletcher (1991) memberikan beberapa cara atau metode yaitu kumpulan perintah, ekspresi, tabel instruksi, program komputer, kode semu dan flow chart, sedangkan Knuth (1973) menyarankan algoritma fundamental.

- Kode Semu Dalam merancang sebuah algoritma menggunakan kode semu atau pseudo code, komponen-komponen input, output dan proses harus terdefinisi secara jelas.
- Flowchart merupakan gambar atau bagan yang memperlihatkan urutan dan hubungan antar proses beserta instruksinya.

Dalam pembuatan flowchart tidak ada rumus atau patokan yang bersifat mutlak. Karena flowchart merupakan gambaran hasil pemikiran dalam menganalisa suatu masalah dengan komputer.

Namun secara garis besar setiap pengolahan selalu terdiri dari 3 bagian utama, yaitu:

- Input,
- Proses pengolahan dan
- Output



**Gambar 2. Proses pengolahan pada flowchart**

Untuk pengolahan data dengan komputer, urutan dasar pemecahan suatu masalah:

- START, berisi instruksi untuk persiapan peralatan yang diperlukan sebelum menangani pemecahan persoalan.
- READ, berisi instruksi kegiatan untuk membaca data dari suatu peralatan input.
- PROSES, berisi kegiatan yang berkaitan dengan pemecahan persoalan sesuai dengan data yang dibaca.
- WRITE, berisi instruksi untuk merekam hasil kegiatan ke peralatan output
- END, mengakhiri kegiatan pengolahan.

#### f. Struktur Dasar Algoritma

Dalam proses pembuatan struktur pemecahan masalah dengan algoritma, akan lebih sering menggunakan logika sehingga menjadi suatu alur yang menyesuaikan dengan proses empiris yang ada.

Proses empiris pada algoritma dan pemrograman dapat dibagi ke dalam 3 jenis proses, yaitu:

- Sekuensial (Runtutan) Proses penyederhanaan suatu persamaan menjadi suatu variable tertentu sehingga dalam proses selanjutnya kita tidak perlu mengulangi persamaan tersebut.
- Percabangan (Pengkondisian) Proses memilah / memisahkan proses

satu dengan proses lainnya yang berbeda kondisi dan perlakuan.

4. Loop(Pengulangan) Proses pengulangan dengan syarat sama yang diberlakukan untuk variable tertentu.

#### 4. HASIL DAN PEMBAHASAN

##### A. Algoritma

Algoritma perangkat lunak merupakan bagian dari langkah-langkah kerja dari perangkat lunak yang dirancang.

###### 1. Algoritma Menu Utama

Algoritma Menu Utama adalah algoritma menjalankan form splash, module, class module dan form lainnya yang terkait di form main.

MULAI

Tampilkan Menu Utama  
Memanggil procedure module-module

SELESAI

###### 2. Algoritma Input

Algoritma input ini adalah langkah-langkah yang dijalankan untuk memberi input berupa pesan teks, nilai dan variabel yang digunakan.

MULAI

Masukkan pesan teks  
Masukkan nilai dan variabel

SELESAI

###### 3. Algoritma

Proses Aritmatika dan Operasi String

MULAI

If TabStrip1 Then  
Frame

Definisi Load

Else if TabStrip2  
Then

Frame

OperasiAritmatika Load

Else if TabStrip3  
then

Frame

OperasiString Load

END

###### 4. Algoritma Proses Pengkondisian MULAI

Masukkan Operand1

Masukkan Operand2

Operator

Pembanding Antara  
Operand1 Dan Operand2

END

###### 5. Algoritma Output

MULAI

Masukkan pesan  
teks

Masukkan nilai dan  
variabel

END

###### 6. Algoritma Stack

```
Public Sub clearStack() Dim i As Integer ' set semua sel dengan isian "?": For i = 0 To grid.Rows - 1 grid.TextMatrix(i, 2) = "?" Next i ' clear lama dengan tanda "SP >" If (SP >= 0) And (SP < grid.Rows) Then grid.TextMatrix(SP, 1) = "" ElseIf SP < 0 Then ' di case "stack overflow" terjadi sebelum: grid.TextMatrix(0, 1) = "" End If ' set SP menuju top dari stack: SP = grid.Rows - 1 ' set label SP: grid.TextMatrix(SP, 1) = "SP >"End SubPublic Sub PUSH(sValue As String) ' clear label "SP >" sebelumnya: If SP > 0 Then grid.TextMatrix(SP, 1) = "" SP = SP - 1 ' periksa apakah terjadi overflow: If SP < 0 Then ' tampilkan kotak pesan hanya jika aliran tetap berjalan, If Not (ArahPanahPenunjukAliran Is Nothing) Then MsgBox "STACK OVERFLOW!" End If frmMain.EksekusiDiterminasikan Exit Sub End If ' set label "SP >" baru: grid.TextMatrix(SP, 1) = "SP >" ' simpan nilai kedalam stack: grid.TextMatrix(SP, 2) = sValue End Sub Public Function POP() As String ' periksa jika stack kosong (tidak memiliki data): If SP = (grid.Rows -
```



```

1) Then mBox "STACK
KOSONG!"
frmMain.EksekusiDiterminasikan
Exit Function End If ' pop nilai
point dengan SP: POP =
grid.TextMatrix(SP, 2) ' clearkan
label "SP >" sebelumnya:
grid.TextMatrix(SP, 1) = "" SP = SP
+ 1 ' update SP (tidak dengan +2
seperti yang gue buat, tapi +1). ' set
label "SP >" baru:
grid.TextMatrix(SP, 1) = "SP >"End
Function
    
```

## B. Implementasi

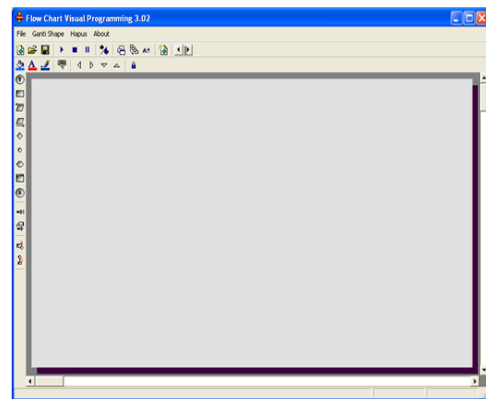
Implementasi merupakan hasil akhir dari perangkat lunak yang dirancang. Program ini dirancang untuk menganalisis implementasi flowchart. Terdiri dari tampilan About, Menu Utama, Layar, Input, Output, Proses, Pengkondisian dan Stack. Cara penggunaan program ini sama seperti mengerjakan flowchart pada kertas, hanya saja pada program ini disimulasikan dengan lebih prosesnya mulai start hingga end.

1. Tampilan About Berikut ini adalah tampilan about:



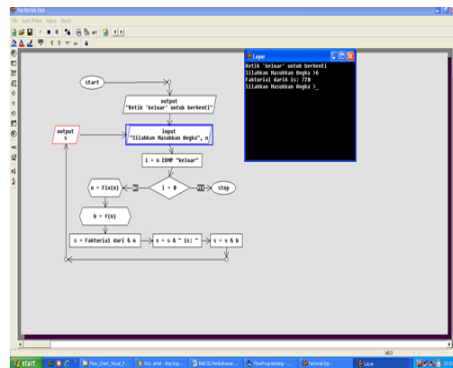
Gambar 3. Implementasi Tampilan Form About

## 3. Tampilan Menu Utama



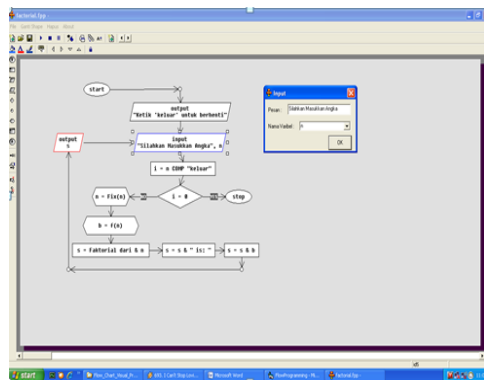
Gambar 4. Implementasi Tampilan Form Menu Utama

## 4. Implementasi Tampilan Layar



Gambar 5. Implementasi Tampilan Layar

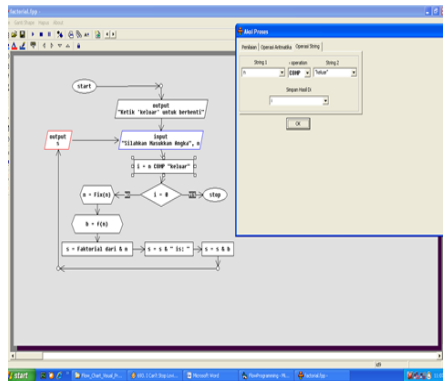
## 5. Implementasi Tampilan Input



Gambar 6. Implementasi Tampilan Input

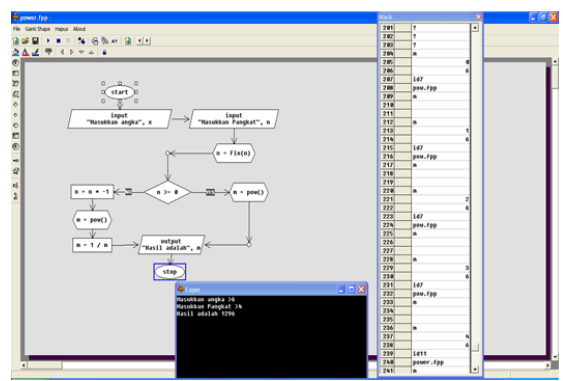


## 6. Implementasi Tampilan Proses



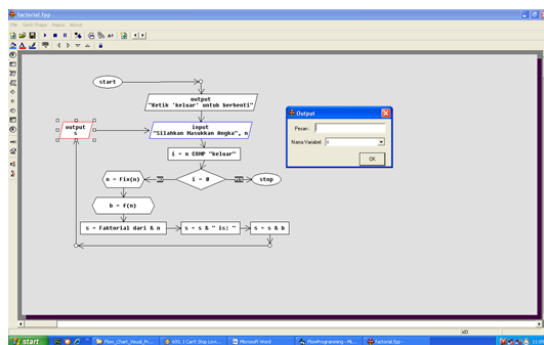
Gambar 7. Implementasi Tampilan Proses

## 9. Implementasi Tampilan Stack



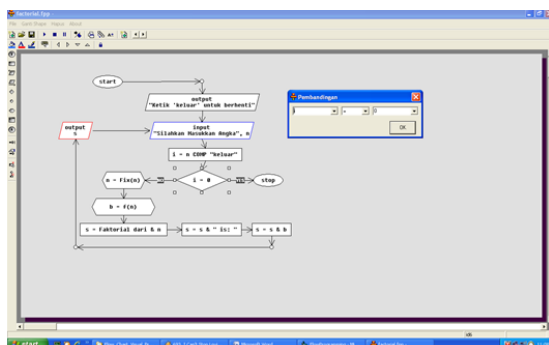
Gambar 10. Implementasi Tampilan Stack

## 7. Implementasi Tampilan Output



Gambar 8 Implementasi Tampilan Ouput

## 8. Implementasi Tampilan Pengkondisian



Gambar 9 Implementasi Tampilan Pengkondisian

## 5. KESIMPULAN DAN SARAN

### 1. Kesimpulan

Dalam tugas akhir ini telah dibangun Perangkat Lunak Aplikasi Pembelajaran Flowchart. Setelah mengaplikasikan Program yang telah dibangun, maka dapat diambil kesimpulan sebagai berikut:

1. Hasil yang diperoleh dari perancangan ini yaitu program yang dapat mensimulasikan Flowchart pada berbagai kasus dimana terjadi beberapa kejadian (*event*) ketika dijalankan.
2. Perangkat lunak yang dirancang dapat mensimulasikan struktur data stack.

### 2. Saran

Untuk pengembangan program lebih lanjut, dapat diberikan beberapa saran yaitu:

1. Adanya pengembangan dari program yang dirancang dengan menambahkan menu latihan. Misalnya user ingin membuat flowchart rumus luas persegi jika user salah merancang maka program dapat memberi solusi dan penjelasan sesuai kamus flowchart yang dirancang programmer.

2. Diharapkan program dapat menambahkan operator dalam operasi matematika dengan lebih lengkap.

#### DAFTAR PUSTAKA

1. Koswara Eko, **Visual Basic Beginner Guide**, PT. MediaKom, 2011.
2. Ario Suryokusumo, **Microsoft Visual Basic**, PT. Elex Media Komputindo, 2001.
3. Djoko Pramono, **Mudah menguasai Visual Basic**, PT. Elex Media Komputindo, 2002.
4. Rony Setiawan, **Teknik Pemecahan Masalah dengan Algoritma dan Flowchart**, PT. Lentera Ilmu Cendekia, 2009.
5. Warsita, Bambang. 2008. Teknologi Pembelajaran dan Aplikasinya. Jakarta: Rineka Cipta.
6. Priyanto Hidayatulla, visual basic.net; membuat aplikasi dan program kreatif, and ed. Bandung: informatika, 2015. Aurino,
7. D., Sridhani. 2012. Algoritma dan Flowchart. Diambil dari: <http://abc.aurino.com/?p=272> .(05 juli 2023).
8. Donal, R. 2013. Pengenalan flowchart dan symbol flowchart. Diambil dari: <https://sites.google.com/site/pagentukilmu2/tutorial/pemrograman/pengenalanflowchartdefinisidansymbolflowchart>. (10juli 2023).
10. Sulaeman,A. 2011.Algoritma flowchart dan pseudocode. Diambil dari: <http://rivsoe666.blog.unsoed.ac.id/2011/10/pengertian-algoritma-flow-chart-dan-pseudocode/> . (18 Juli 2023)