



## Penggunaan Statement IF-ELSE-END Pada Kode Program Kalkulator Fisika Untuk Materi Tumbukan

Puji Suharmanto\*, Sri Endang Wahyuni  
Universitas Indraprasta PGRI  
\* E-mail: harmansuharmanto@gmail.com

### **Abstract**

*In the physics calculator there is a Matlab program code which will be read as input and then executed to calculate physics problems computationally. Matlab has a program control structure as input, namely the statement IF-ELSE-END in the form of a conditional statement command with the ability to make different decisions depending on conditions and repeat operations without duplicating code so that commands can be skipped or executed. The research objective is that students can complete functions and create simple scripts using the statement IF-ELSE-END in physics problems. The method used in this research is developing Matlab program code in writing scripts using the statement IF-ELSE-END. Observations were made on the program code that had been created with each script on the radiobutton being converted into a simple script using the statement IF-ELSE-END. Based on the results of observations, students were able to run a physics calculator using the statement IF-ELSE-END on collision material with calculation results that were in accordance with manual calculations. So it can be concluded that using the statement IF-ELSE-END can be used to write simple scripts to problems solve physics.*

**Keywords:** Program code, Collision, Matlab.

**How to Cite:** Suharmanto, P., & Wahyuni, S.E. (2023). Penggunaan Statement IF-ELSE-END Pada Kode Program Kalkulator Fisika Untuk Materi Tumbukan. *Schrodinger Jurnal Ilmiah Mahasiswa Pendidikan Fisika*, 4(2), 109-117.

### **PENDAHULUAN**

Mata kuliah Fisika gerak terintegrasi dengan software Matlab, agar perhitungan rumus fisika secara manual dengan tingkat kerumitan yang cukup tinggi dapat diselesaikan secara komputerisasi. Khususnya bagi mahasiswa teknik informatika Matlab dapat digunakan sebagai alat pendukung pembelajaran dalam memahami rumus-rumus fisika. Dengan menggunakan matlab dapat dibuatkan sebuah pemrograman sederhana kalkulator fisika yang membantu peserta didik dalam menghitung permasalahan hitungan dalam fisika (Okryanida, Widiyatun, & Asih, 2021). Penggunaan kalkulator fisika berbasis matlab selain meminimalisir kesalahan hitung juga dapat menghemat waktu saat melakukan analisis ayunan matematis (Astuti, 2021). Dengan menggunakan KALFIS mampu membaca simulasi konsep fisika dan menjelaskan gambaran secara nyata, sehingga mampu memahami konsep mekanika dengan baik (Yosua, Fauzan, Kistiani, & Astuti, 2019).

MATLAB merupakan sistem interaktif dan sebuah aplikasi dengan bahasa pemrograman (Goeritno & Hendrawan, 2019). Matlab adalah *high-level programming* dan memiliki fasilitas yang interaktif untuk visualisasi, komputasi dan programming. Matlab juga berbasis *object-oriented-programming* sehingga cocok untuk keperluan pengajaran dan penelitian (Setiawan, 2022). Pemrograman dalam matlab akan tidak efektif jika dilakukan langsung di jendela kerja, karena fungsi pemrograman pada jendela kerja sangat terbatas. Oleh karena itu muncullah istilah file m (Noviansyah, 2019). Pada file.m dapat dilakukan menulis kode program berisi inputan data nilai-nilai asumsi, rumus perhitungan dan output

penyelesaian fisika. Kode program Matlab akan dibaca sebagai inputan. Seperti bahasa pemrograman lainnya, Matlab memiliki beberapa kontrol struktur. Kontrol struktur adalah perintah dalam bahasa pemrograman yang digunakan dalam pengambilan keputusan. Matlab memiliki empat kontrol struktur, yaitu IF-ELSE-END, SWITCH-CASE, FOR, dan WHILE (Fitriani & Hermadi, 2018)

Tumbukan merupakan salah satu sub bab materi fisika gerak dalam BAB Momentum. Kalkulator fisika berbasis matlab dibuat untuk menyelesaikan persoalan tumbukan. Pada kalkulator fisika terdapat kode program matlab yang akan dibaca sebagai inputan kemudian dieksekusi untuk mengitung persoalan fisika secara komputasi. Matlab memiliki struktur kontrol program sebagai inputan yaitu *Statement IF-ELSE-END* digunakan untuk memilih alternatif apabila suatu syarat atau kondisi dipenuhi, atau memilih satu alternatif dari 2 kemungkinan berdasarkan apakah syarat terpenuhi atau tidak (Faradiba, 2019). Untuk mengatur efektivitas dan efisiensi pada Pemrograman MATLAB di dalam script/m-file maka diperlukan adanya kontrol program (Busrah, 2019). Tujuan penelitian yaitu mahasiswa dapat menyelesaikan fungsi dan membuat *script* secara sederhana menggunakan *statement IF-ELSE-END* dalam persoalan fisika. Metode yang digunakan dalam penelitian ini adalah pengembangan kode program matlab dalam menulis *script* dengan menggunakan *statement IF-ELSE-END*. Diharapkan pembelajaran kode program menggunakan *Statement IF-ELSE-END* pada persoalan fisika dapat mempermudah mahasiswa teknik informatika dalam memahami rumus fisika materi tumbukan.

## METODE PENELITIAN

Metode yang digunakan dalam penelitian ini adalah pengembangan kode program matlab dalam menulis *script* dengan menggunakan *statement IF-ELSE-END*. Model pengembangan yang digunakan adalah 4-D (*Define, Design, Develop, and Disemination*). Untuk tahap penelitian pengembangan kali ini, media pembelajaran yang dibuat baru pengembangan kode program berupa fungsi dan script yang lebih sederhana menggunakan *statement IF-ELSE-END* belum sampai tahapan respon pengguna. Penelitian yang dilakukan melalui 3 tahap: Tahap pertama, merupakan tahap persiapan berupa studi literature dan mahasiswa dibekali pengetahuan akan materi tentang tumbukan dan dasar-dasar kode program pada Matlab 2013a yang akan diaplikasikan menjadi kalkulator fisika. Tahap kedua dilakukan praktek dengan pembuatan GUI MATLAB, selanjutnya membuat kode program untuk menyelesaikan fungsi dan pembuatan *script* pada masing-masing tombol radiobutton menggunakan fungsi off selanjutnya membuat *script* sederhana menggunakan *statement IF-ELSE-END* dalam persoalan fisika. Tahap ketiga dilakukan pengujian dari hasil kedua kode program (coding) yang telah dibuat. Jika masih terdapat kesalahan pada kode program maka kalkulator tidak dapat dijalankan dan jika kode program yang dibuat sudah benar maka kalkulator berhasil dijalankan. Dengan dilakukannya coding atau kode program pada materi tumbukkan diharapkan mahasiswa teknik informatika dapat lebih mudah dalam memahami materi didukung dengan komputasinya baik secara teori maupun praktek. Adapun materi tumbukan yang dijelaskan sebagai berikut :

- Materi Jenis – Jenis Tumbukan

Jenis tumbukan ini terbagi menjadi menjadi tiga yaitu: lenting sempurna, lenting sebagian, dan tak lenting. Tumbukan ini terkait dengan perpindahan energi kinetik. Tumbukan lenting sempurna adalah tumbukan yang dapat mempertahankan energi kinetiknya setelah mengalami tumbukan, dengan kata lain energi kinetiknya tetap. Untuk membedakan masing – masing tumbukan perlu didefinisikan sebuah koefisien restitusi. Selanjutnya bawah ini penurunan koefisien restitusi:

Berawal dari hukum kekekalan energi kinetik:

$$Ek_1 + Ek_2 = Ek_1' + Ek_2'$$

$$\frac{1}{2}m_1v_1^2 + \frac{1}{2}m_2v_2^2 = \frac{1}{2}m_1v_1'^2 + \frac{1}{2}m_2v_2'^2 \quad (1)$$

yang dapat ditulis dalam bentuk lain

$$m_1(v_1^2 - v_1'^2) = m_2(v_2'^2 - v_2^2) \quad (2)$$

Dan hukum kekekalan momentum

$$m_1v_1 + m_2v_2 = m_1v_1' + m_2v_2' \quad (3)$$

ditulis menjadi

$$m_1(v_1 - v_1') = m_2(v_2' - v_2) \quad (4)$$

Koefisien restitusi Persamaan (2) dibagi dengan (4) akan menjadi

$$(v_1 + v_1') = (v_2' + v_2) \quad (5)$$

Atau dapat juga ditulis

$$(v_1 - v_2) = (v_2' - v_1') \quad (6)$$

Persamaan (6) merupakan kecepatan relatif sebelum tumbukan dan sesudah tumbukan. Jika persamaan (6) dibuat perbandingan akan menjadi

$$e = \frac{-(v_1' - v_2')}{(v_1 - v_2)} \quad (7)$$

dengan  $e$  adalah koefisien restitusi.

Nilai koefisien restitusi ini pada interval 0 sampai dengan 1. Jika  $e = 0$  disebut tumbukan tak lenting,  $e = 1$  disebut tumbukan lenting sempurna, dan  $0 < e < 1$  adalah lenting sebagian.

Selanjutnya mencari persamaan kecepatan akhir partikel setelah tumbukan tanpa diketahui kecepatan akhir partikel lainnya, didapatkan dengan alternatif persamaan (5)

$$v_1 + v_1' = v_2' + v_2$$

$$v_2' = v_1 + v_1' - v_2 \quad (8)$$

$$v_1' = v_2' + v_2 - v_1 \quad (9)$$

Substitusikan persamaan (8) ke pers (3) maka didapatkan rumus mencari kecepatan pertama setelah tumbukan ( $v_1'$ ) sebagai berikut

$$v_1' = \left(\frac{m_1 - m_2}{m_1 + m_2}\right)v_1 + \left(\frac{2m_2}{m_1 + m_2}\right)v_2 \quad (10)$$

Substitusikan persamaan (9) ke pers (3) maka didapatkan rumus mencari kecepatan kedua setelah tumbukan ( $v_2'$ ) sebagai berikut

$$v_2' = \left(\frac{2m_1}{m_1 + m_2}\right)v_1 + \left(\frac{m_2 - m_1}{m_1 + m_2}\right)v_2 \quad (11)$$

- Penyeleksian Kondisi dengan Statement if – else – end

Selanjutnya dalam menyelesaikan permasalahan fisika secara komputasi diperlukan efektivitas dan efisiensi dari penulisan perintah-perintah yang akan di eksekusi. Untuk

mengatur efektivitas dan efisiensi pada Pemrograman MATLAB di dalam script/m-file maka diperlukan adanya kontrol program. Penyeleksian kondisi dan pengambilan keputusan serta perulangan yang melibatkan hukum-hukum logika menjadi satu bentuk kontrol program yang dapat dilakukan dalam pemrograman. Statement if-else-end merupakan salah satu struktur kontrol program pada Matlab, Kondisi-kondisi yang diseleksi berupa suatu ekspresi yang mempunyai nilai kebenaran (Benar-Salah). Benar atau salahnya suatu ekspresi didasarkan dengan menggunakan logika komputasi baik yang bersifat numerical maupun berkarakter string (Busrah, 2019).

Didalam MATLAB, biasanya ekspresi yang diuji berupa ekspresi yang bersifat matematis seperti yang disajikan pada contoh-contoh. Seringkali sederetan perintah harus dikerjakan dengan didasarkan pada hasil tes rasional . Dalam bahasa pemograman, logical ini dikerjakan dengan variasi kontruksi if – else - end. Bentuk paling sederhana kontruksi if-else-end pada MATLAB, biasanya tersusun atas :

```

if ekspresi (Benar/Salah)
perintah (Dijalankan jika ekspresi bernilai Benar)
end

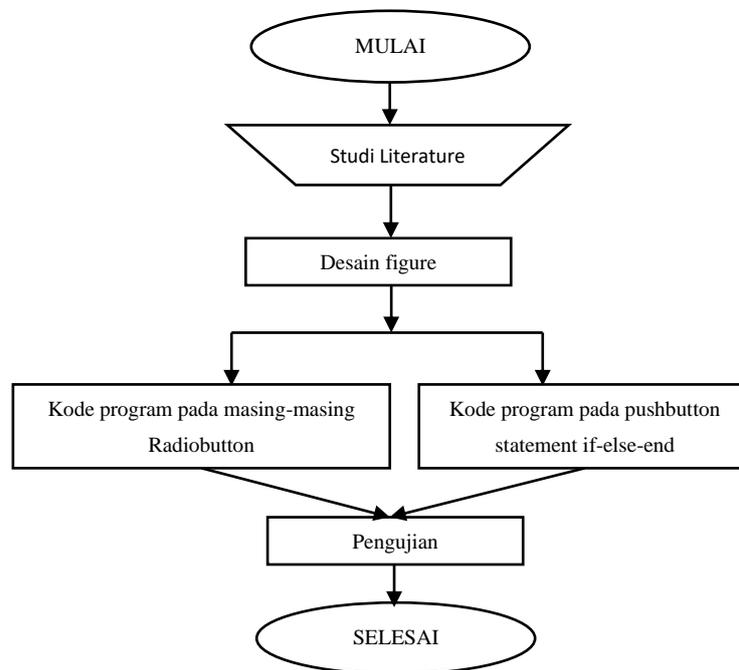
```

Selanjutnya jika kondisi yang diseleksi sebanyak dua kemungkinan mempunyai struktur seperti berikut :

```

if ekspresi (Benar/Salah)
perintah1 (Dijalankan jika ekspresi bernilai Benar)
else perintah2 (Dijalankan jika ekspresi tidak memenuhi ekspresi sebelumnya)
end

```



Gambar 1. Diagram alir penelitian

## HASIL DAN PEMBAHASAN

Aplikasi yang akan dibuat pada materi tumbukan perhitungan rumusnya menggunakan tools push button, radio button dan axes. Soal yang akan dibuat menjadi kalkulator fisika mengenai lenting sempurna dan penyelesaian secara manual sebagai berikut :

### Soal tumbukan lenting sempurna!

Benda A dan B bermassa sama yaitu 2 kg bergerak saling mendekati, di mana kelajuan benda A adalah 10 m/s menuju arah barat dan kelajuan benda B adalah 12 m/s menuju arah timur. Jika kedua benda bertumbukan lenting sempurna maka kelajuan benda A dan benda B setelah tumbukan ?

Penyelesaian :

Diketahui :

$$m_A = 2 \text{ kg}$$

$$v_A = -10 \text{ m/s}$$

$$m_B = 2 \text{ kg}$$

$$v_B = 12 \text{ m/s}$$

$$e = 1$$



**Ingat!** Jangan abaikan tanda (+) dan (-) pada arah kecepatan karena berpengaruh terhadap perhitungan.

Ditanya:  $v'_A$  &  $v'_B = ?$

Jawab

Dari koefisien restitusi (e) untuk lenting sempurna = 1 maka

$$1 = -\frac{(v'_A - v'_B)}{v_A - v_B}$$

$$1 = -\frac{(v'_A - v'_B)}{-10 - 12}$$

$$-22 = -v'_A + v'_B$$

$$v'_B = -22 + v'_A \dots\dots\dots(12)$$

$$m_A v_A + m_B v_B = m_A v'_A + m_B v'_B$$

$$2 \cdot (-10) + 2 \cdot 12 = 2 \cdot v'_A + 2 \cdot (-22 + v'_A)$$

$$-20 + 24 = 2v'_A - 44 + 2v'_A$$

$$4 + 44 = 2v'_A + 2v'_A$$

$$48 / 4 = v'_A$$

$$12 \text{ m/s} = v'_A$$

Atau  $v'_A$  diperoleh dari persamaan (10)

$$\begin{aligned} v'_A &= \left(\frac{m_A - m_B}{m_A + m_B}\right) v_A + \left(\frac{2m_B}{m_A + m_B}\right) v_B \\ &= \left(\frac{2-2}{2+2}\right) (-10) + \left(\frac{2(2)}{2+2}\right) (12) \\ &= 12 \text{ m/s} \end{aligned}$$

sehingga

$$v'_B = -22 + v'_A$$

$$v'_B = -22 + 12$$

$$v'_B = -10 \text{ m/s}$$

Maka setelah tumbukan, benda A berubah kecepatan menjadi 12 m/s ke arah kanan dan kecepatan benda B menjadi 10 m/s ke arah kiri. Sesuai dengan teori kecepatan untuk lenting sempurna  $v'_A = v_B$  dan  $v'_B = -v_A$

### a. Mendesain Figure

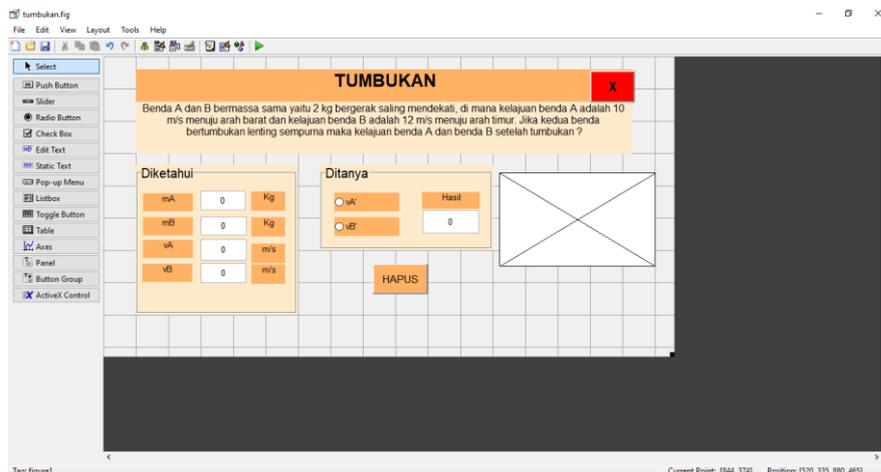
Dalam mendesain figure, kita harus dapat membayangkan komponen apa saja yang perlu kita tampilkan sebagai input (diketahui), output (ditanyakan) dan perhitungan rumus. Pada soal, variabel input (diketahui) dalam menyelesaikan soal lenting sempurna adalah  $m_A$ ,

$v_A$ ,  $m_B$  dan  $v_B$ . Kemudian yang menjadi variabel output (ditanyakan) adalah  $v'_A$  dan  $v'_B$ . Selanjutnya untuk rumus kelajuan benda pertama setelah bertumbukan dihitung menggunakan persamaan (10) dan rumus kelajuan benda kedua menggunakan persamaan (12) berikut :

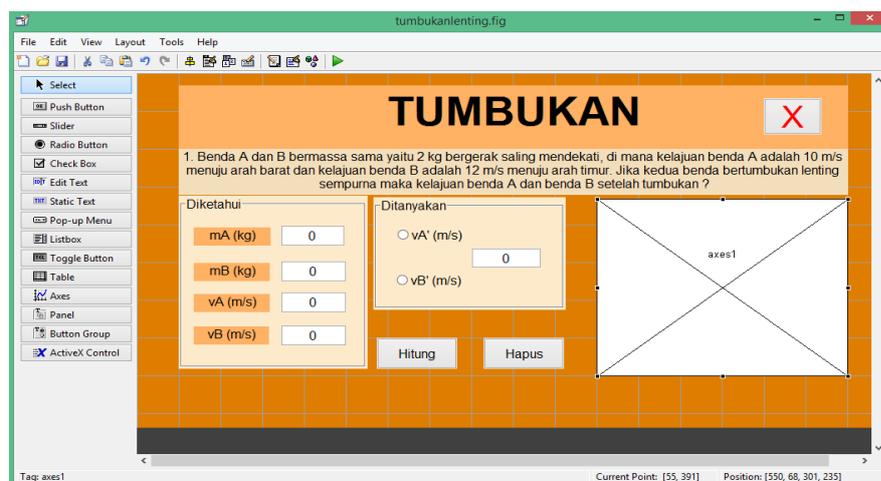
$$v'_A = \left( \frac{m_A - m_B}{m_A + m_B} \right) v_A + \left( \frac{2m_B}{m_A + m_B} \right) v_B$$

$$v'_B = -22 + v'_A$$

Dari variabel input (diketahui) dan variabel output (ditanyakan) lihat kembali penyelesaian lengkapnya pada soal diatas, maka kita memerlukan 5 edit text yang akan digunakan untuk menampilkan variabel-variabel tersebut. Kemudian untuk memunculkan hasil atau output penelitian ini melakukan dua program yang keduanya memiliki dua perbedaan yaitu pada tombol hitungnya, dimana pada gambar 1 Desain figure untuk coding fungsi off tidak ada tombol hitung tetapi proses hitung dilakukan langsung oleh masing-masing tools radiobutton sedangkan pada gambar 2 Desain figure untuk coding statement IF-ELSE-END, proses hitung dilakukan pada satu tombol hitung oleh tools pushbutton. Perhatikan desain figure gambar 2 dan gambar 3 sebagai berikut :



Gambar 2. Desain figure untuk coding fungsi off



Gambar 3. Desain figure untuk coding statement IF-ELSE-END

Tabel 1. Komparasi Figure Atribut

Atribut Matlab	Fungsi Off	Statement IF-ELSE-END
Static text	√	√
Edit Text	√	√
Radiobutton	√	√
Axes	√	√
Push button Hitung	X	√
Push button Hapus	√	√
Push button Keluar	√	√

## b. Kode Program

Setelah selesai menyimpan figure, langkah selanjutnya adalah melakukan kode program berupa tulisan script. Script file atau M-File adalah kumpulan perintah yang ditulis menggunakan bahasa pemrograman MATLAB yang dapat disimpan dan dijalankan berulang-ulang. Script file dibuat dengan nama tanpa spasi dan dengan ekstensi dot m (Yosua,2019). Pada penelitian ini penulisan kode program dibuat komparasi dua macam kode program bertujuan agar penulisan script lebih efektif dan efisien dalam menulis script kode program. Kode program pertama untuk menggunakan fungsi off pada masing-masing tombol radiobutton sebagai berikut :

### 1. Script File Pertama menggunakan kode program fungsi off

#### ❖ Untuk Tombol Radiobutton

Klik kanan tombol radiobutton1 atau radiobutton2 → view callback → callback

#### Kode Program pada radiobutton 1

```
% --- Executes on button press in radiobutton1.
function radiobutton1_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of radiobutton1
off = [handles.radiobutton2];
mA=str2double(get(handles.edit1,'string'));
mB=str2double(get(handles.edit2,'string'));
vA=str2double(get(handles.edit3,'string'));
vB=str2double(get(handles.edit4,'string'));
%Rumus
vA2=(((mA-mB)/(mA+mB))*(vA)) + (((2*mB)/(mA+mB))*(vB));
set(handles.edit5,'string',vA2);
pilih=1;
handles.pilih = pilih;
guidata(hObject, handles);
```

#### Kode Program pada radiobutton 2

```
% --- Executes on button press in radiobutton2.
function radiobutton2_Callback(hObject, eventdata, handles)
% hObject    handle to radiobutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% Hint: get(hObject,'Value') returns toggle state of radiobutton2
off = [handles.radiobutton1];
mA=str2double(get(handles.edit1,'string'));
mB=str2double(get(handles.edit2,'string'));
vA=str2double(get(handles.edit3,'string'));
```

```

vB=str2double(get(handles.edit4,'string'));
%Rumus
vB2= (-22+vA2)
set (handles.edit5,'string',vB2);
pilih=2;
handles.pilih = pilih;
guidata(hObject, handles);

```

## 2. Script File Pertama menggunakan kode program statement IF-ELSE-END

selanjutnya membuat *script* fungsi off menjadi lebih sederhana dengan menggunakan *statement* IF-ELSE-END dalam persoalan soal yang sama.

### ❖ Untuk Tombol Hitung

Klik kanan tombol hitung → view callback → callback

#### Kode Program pada pushbutton 1

```

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject handle to pushbutton1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
mA=str2double(get(handles.edit1,'string'));
mB=str2double(get(handles.edit2,'string'));
vA=str2double(get(handles.edit3,'string'));
vB=str2double(get(handles.edit4,'string'));
%Rumus
vA2=(((mA-mB)/(mA+mB))*(vA)) + (((2*mB)/(mA+mB))*(vB));
vB2= (-22+vA2);

a=get(handles.radiobutton1,'value');
b=get(handles.radiobutton2,'value');

if(a==1)
    set(handles.edit5,'string',vA2)
else(b==1)
    set(handles.edit5,'string',vB2)
end

```

Hasil penulisan script pertama menggunakan kode program fungsi off pada dibutuhkan dua kali penulisan script file secara berulang pada masing-masing tools radiobutton, sedangkan penulisan script kedua menggunakan kode program statement IF-ELSE-END hanya dilakukan satu kali penulisan script file pada tombol hitung tools pushbutton sehingga dapat disimpulkan kode program statement IF-ELSE-END sebagai kontrol program yang dapat digunakan dalam script file untuk mengatur efektivitas dan efisiensi pada Pemrograman MATLAB (Busrah, 2019). Selanjutnya proses akhir penelitian ini kedua kode program yang telah dibuat mahasiswa berhasil dijalankan pada kalkulator fisika materi tumbukan dengan hasil perhitungan yang sesuai dengan perhitungan manual.

## PENUTUP

Berdasarkan hasil penelitian kode program statement IF-ELSE-END sebagai kontrol program yang dapat digunakan dalam script file lebih efektivitas dan efisiensi pada Pemrograman MATLAB daripada penggunaan kode program fungsi off dan hasil pengamatan mahasiswa mampu menjalankan kalkutor fisika menggunakan *Statement IF-ELSE-END* pada materi tumbukan dengan hasil perhitungan yang sesuai dengan perhitungan manual. Sehingga dapat disimpulkan penggunaan *statement IF-ELSE-END* dapat digunakan untuk

menulis *script* secara sederhana dalam menyelesaikan persoalan fisika. Diharapkan pembelajaran kode program menggunakan *Statement IF-ELSE-END* pada persoalan fisika dapat mempermudah mahasiswa teknik informatika dalam memahami rumus fisika materi tumbukan.

## DAFTAR PUSTAKA

- Astuti, S. P. (2021). Pemanfaatan Aplikasi Kalkulator Fisika Untuk Meningkatkan Motivasi Belajar Pada Materi Fisika Gerak. *Susunan Artikel Pendidikan*, 105-111.
- Busrah, Z. (2019). *Matematika Komputasi Berbasis Pemrograman Matlab*. Parepare: Percetakan Kaaffah.
- Faradiba. (2019). *PENGGUNAAN APLIKASI MATLAB UNTUK ANALISA KOMPUTASI*. Jakarta: PROGRAM STUDI PENDIDIKAN FISIKA FAKULTAS KEGURUAN DAN ILMU PENDIDIKAN UNIVERSITAS KRISTEN INDONESIA.
- Fitriani, R. A., & Hermadi, I. (2018). Instrumentasi Kode Program Secara Otomatis untuk Path Testing. *Jurnal Ilmu Komputer Agri-Informatika*, 40-50.
- Goeritno, A., & Hendrawan, T. (2019). Perubahan Daya Beban di Setiap Bus Disimulasikan dengan Aplikasi MATLAB untuk Analogi Fenomena Steady State Stability. *Jurnal Ilmiah Setrum*, 35-54.
- Noviansyah, M. (2019). Pengenalan Dasar Matlab. Jakarta, Jakarta, Indonesia.
- Okyanida, I. Y., Widiyatun, F., & Asih, D. A. (2021). PERANCANGAN APLIKASI KALKULATOR FISIKA PADA MATERI GAYA. *Jurnal Inovasi Penelitian*, 1663-1668.
- Setiawan, I. M. (2022). *Graphical user Interface (GUI) Matlab dan Interface dengan Arduino*. Belitung: Politeknik Manufaktur Negeri Bangka Belitung.
- Yosua, R., Fauzan, A., Kistiani, & Astuti, I. A. (2019). Aplikasi KALFIS (Kalkulator Fisika) Berbasis Matlab Untuk Membantu Analisis Eksperimen Fisika. *Navigation Physics Journal Of Physics Education*, 59 - 62.