

UTS EL2008 - Pemecahan Masalah dengan C

Hari/Tanggal : Sabtu/19 Maret 2022
Dosen Pengampu : Arif Sasongko
Reza Darmakusuma
M. Ogin Hasanuddin
Waktu : 19 Maret 2022 (12.00) – 20 Maret 2022 (12.00)
Sifat : Open book

“Saya menyatakan bahwa saya melaksanakan ujian ini dengan jujur. Apabila nanti ditemukan kecurangan dalam pelaksanaan ujian ini, saya bersedia menerima sanksi yang diberikan.”

Nama : Bayu Aji Nugroho
NIM : 13221601



Link repository program dan file:

https://github.com/bayunugroho1990/UTS_EL2008.git

NOMOR 1 : SHORT INTEGER 16 BIT

a. Nilai terendah dan tertinggi

1. Unsigned short integer 16 bit = 2 byte

Nilai terendah = $\underbrace{0000\ 0000}_{1\text{ byte}}\ \underbrace{0000\ 0000}_{1\text{ byte}} = 0$

Nilai tertinggi = $\underbrace{1111\ 1111}_{1\text{ byte}}\ \underbrace{1111\ 1111}_{1\text{ byte}} = 65535$

2. Signed short integer 2's complement 16 bit = 2 byte

Nilai terendah = $\underbrace{1000\ 0000}_{1\text{ byte}}\ \underbrace{0000\ 0001}_{1\text{ byte}} = -32,768$

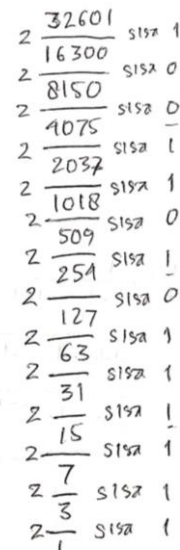
Nilai tertinggi = $\underbrace{0111\ 1111}_{1\text{ byte}}\ \underbrace{1111\ 1111}_{1\text{ byte}} = 32,767$

b. Convert decimal to biner NIM

NIM = 13221601 → diambil digit 2 & 3 digabung 3 digit terakhir

Nilai desimal = 32601

Nilai biner = 1111 1110 1011 001



Handwritten long division of 32601 by 2 to convert decimal to binary. The result is 1111 1110 1011 001.

c. $A = 1111 \ 1110 \ 1011 \ 001 = \mathbf{0101 \ 1001}$
LSByte

d. Perhitungan x, y, A sesuai operator

x	y	A	Operasi	Biner	Hexa	Unsigned	Signed 2C
1010 0101	1110 1111	0101 1001	$(A \& y)^x$	1110 1100	EC	0001 0011	1110 1101
1101 1011	1111 0011	0101 1001	$(\sim(y \gg 4) \& x) + A$	0001 1010	1A	0001 1010	1110 0110
0010 1111	1010 0101	0101 1001	$((x \gg 2) y) * A$	1111 1010 0001 0111	FA17	0000 0101 1110 1000	1111 1010 0001 1000

1. $A \& y$ $(A \& y)^x$

$$\begin{array}{r} A \ 0101 \ 1001 \\ y \ 1110 \ 1111 \\ \hline 0100 \ 1001 \end{array} \quad \times \quad \begin{array}{r} 0100 \ 1001 \\ 1010 \ 0101 \\ \hline 1110 \ 1100 \end{array}$$

2. $(y \gg 4)$ $\sim(y \gg 4)$ $(\sim(y \gg 4) \& x)$ $(\sim(y \gg 4) \& x) + A$

$$\begin{array}{r} 1111 \ 0011 \\ 0011 \ 1111 \\ \hline 0011 \ 1111 \end{array} \gg 4 \quad \begin{array}{r} 0011 \ 1111 \\ 1100 \ 0000 \\ \hline 1100 \ 0000 \end{array} \quad \begin{array}{r} 1100 \ 0000 \\ 1101 \ 1011 \\ \hline 1100 \ 0000 \end{array} \quad \begin{array}{r} 1100 \ 0000 \\ 0101 \ 1001 \\ \hline 1100 \ 0000 \\ \text{Carry} \quad 1 \\ \hline 0001 \ 1010 \end{array}$$

3. $(x \gg 2)$ $(x \gg 2) | y$ $((x \gg 2) | y) * A$

$$\begin{array}{r} 0010 \ 1111 \\ 1100 \ 1011 \\ \hline 1100 \ 1011 \end{array} \gg 2 \quad \begin{array}{r} 1100 \ 1011 \\ 1010 \ 0101 \\ \hline 1110 \ 1111 \end{array} \quad \begin{array}{r} 1110 \ 1111 \\ 0101 \ 1001 \\ \hline 111110 \ 1111 \\ 100000 \ 0001 \\ \hline 101110 \ 1111 \end{array}$$

e. Program C

```
#include<stdio.h>
#include<stdlib.h>
#define MAX_LEN 8

int A[MAX_LEN], biner_1[MAX_LEN], biner_2[MAX_LEN], biner_3[16];

int main() {

printf("Masukkan nilai biner A (8bits)=\n");
for(int i=0; i<MAX_LEN; i++) {
printf("Bit ke %d : ", i+1);
scanf("%d", &A[i]);
}

printf("\n\nOperasi 1:\n");
printf("-----\n");
int x1[8]={1,0,1,0,0,1,0,1};
int y1[8]={1,1,1,0,1,1,1,1};
printf("Hasil binary =\n");
for(int i=0; i<MAX_LEN; i++) {
biner_1[i]=(A[i]&y1[i])^x1[i];
printf("%d", biner_1[i]);
}

printf("\n\nOperasi 2:\n");
```

```

printf("-----\n");
int x2[8]={1,1,0,1,1,0,1,1};
int y2[8]={1,1,1,1,0,0,1,1};
printf("Hasil binary = ");
for(int i=0;i<MAX_LEN;i++){
biner_2[i]=(~(y2[i]>>4)&& x2[i])+A[i];
printf("%d", biner_2[i]);
}

printf("\n\nOperasi 3: ");
printf("-----\n");
int x3[8]={0,0,1,0,1,1,1,1};
int y3[8]={1,0,1,0,0,1,0,1};
printf("Hasil binary = ");
for(int i=0;i<16;i++){
biner_3[i]=((x3[i]>>2)|y3[i])*A[i];
printf("%d", biner_3[i]);
}
printf("\n\n");
return 0;
}

```

Hasil Program:

```

Masukkan nilai biner A (8bits)=
Bit ke 1 : 0
Bit ke 2 : 1
Bit ke 3 : 0
Bit ke 4 : 1
Bit ke 5 : 1
Bit ke 6 : 0
Bit ke 7 : 0
Bit ke 8 : 1

Operasi 1:
-----
Hasil binary =
11101100

Operasi 2:
-----
Hasil binary = 12022012

Operasi 3: -----
Hasil binary = 0000000100002012

Process returned 0 (0x0)   execution time : 10.860 s
Press any key to continue.

```

f. Bandingkan nilai

Program tidak berjalan dengan baik dan hasil tidak valid

NOMOR 2 : Electron Gun

Persamaan I:

$$V \cdot e = m \cdot c^2 - m_0 \cdot c^2$$

$$m = \frac{V \cdot e}{c^2} + m_0$$

$$m = \frac{V \cdot 1.601 \times 10^{-19}}{(2.99 \times 10^{-8})^2} + 9.109 \times 10^{-31}$$

$$m = (V \cdot 17.919 + 9.109) \times 10^{-31} \text{ Kg}$$

Persamaan II:

$$\frac{m}{m_0} = \frac{1}{\sqrt{1 - \left(\frac{v}{c}\right)^2}}$$

$$\frac{v}{c} = 1 - \left(\frac{m_0}{m}\right)^2$$

$$v = \left(1 - \left(\frac{m_0}{m}\right)^2\right) \cdot c$$

Dimana:

$$e = 1.601 \times 10^{-19} \text{ C}$$

$$m_0 = 9.109 \times 10^{-31} \text{ Kg}$$

$$c = 2.99 \times 10^{-8} \text{ m/s}$$

V = tegangan (Volt)

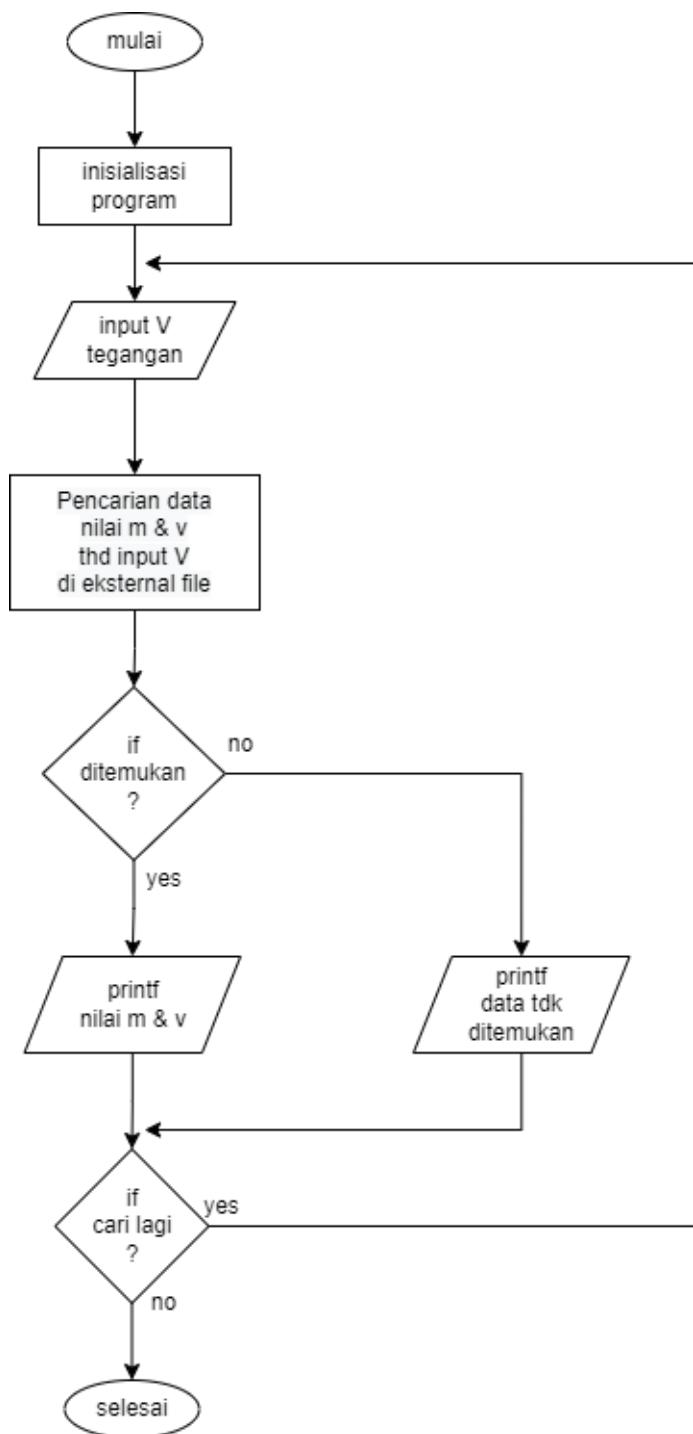
m = massa relative (Kg)

v = Kecepatan bermassa relative

Didapatkan perhitungan sbb:

V $\times 10^6$ volt	m $\times 10^{-31}$ Kg	v=(m/m0)	(m0/m)^2	v/c	v $\times 10^8$ m/s
0	9.109	1.0000	1.00000	0.00000	0.00000
0.5	18.069	0.5041	0.25415	0.74585	2.23008
1	27.028	0.3370	0.11358	0.88642	2.65039
1.5	35.988	0.2531	0.06407	0.93593	2.79844
2	44.947	0.2027	0.04107	0.95893	2.86720
2.5	53.907	0.1690	0.02855	0.97145	2.90463
3	62.866	0.1449	0.02099	0.97901	2.92723
3.5	71.826	0.1268	0.01608	0.98392	2.94191
4	80.785	0.1128	0.01271	0.98729	2.95199
4.5	89.745	0.1015	0.01030	0.98970	2.95920
5	98.704	0.0923	0.00852	0.99148	2.96454

a. Flowchart nomor 2



Kode Program C:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_LEN 100

int i = 0;
float massa[MAX_LEN];
float kecepatan[MAX_LEN];
char tegangan[MAX_LEN];
char filename[]="tegangan.txt";
char names[MAX_LEN][MAX_LEN];
char line[MAX_LEN];
char* token;

int main()
{
    //----->>membuka file eksternal
    FILE* stream = fopen(filename, "r");

    while(fgets(line, MAX_LEN, stream))
    {
        token = strtok(line, ",");
        strcpy(names[i], token);

        token = strtok(NULL, ",");
        massa[i]= atof(token);

        token = strtok(NULL, "\n");
        kecepatan[i]= atof(token);

        ++i;
    }
    //----->>pencarian
    int found = 0;
    int j = 0;
    printf("\nNilai tegangan antara 0 s/d 5 (Volt) dengan kelipatan 0.5: ");
    printf("\nMasukkan nilai tegangan (Volt): ");
    scanf("%s", &tegangan);
    while (j < i && !found)
    {
        if (!strcmp(tegangan, names[j]))
        {
            found = 1;
        }
        else
            ++j;
    }

    if (found){
        printf("-----\n");
        printf("ketika diberi tegangan\t = %sx10^6 Volt\n", tegangan);
        printf("elektron memiliki massa\t = %fx10^-31 Kg\n", massa[j]);
        printf("dengan kecepatan relatif = %fx10^8 m/s\n", kecepatan[j]);
        printf("-----\n");
    }
    else
        printf("%s tidak ada. masukkan kembali nilai!!\n", tegangan);

    fclose(stream);
    return 0;
}
```

Hasil Program:

```

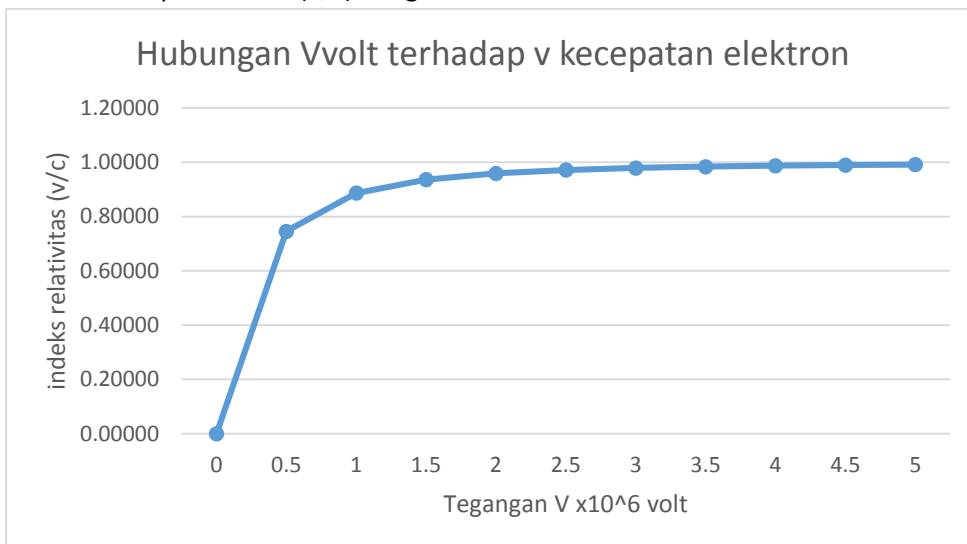
"E:\Filess\FILE_BAYU\5_KULIAH_ITB\2_SEMESTER_4\PMC\UTS\No...
Nilai tegangan antara 0 s/d 5 (Volt) dengan kelipatan 0.5:
Masukkan nilai tegangan (Volt): 2

-----
ketika diberi tegangan      = 2x10^6 Volt
elektron memiliki massa    = 44.946999x10^-31 Kg
dengan kecepatan relatif   = 2.867200x10^8 m/s
-----

Process returned 0 (0x0)   execution time : 3.141 s
Press any key to continue.

```

b. Grafik plot antara (v/c) dengan V volt



c. File eksternal : *tegangan.txt*

```

tegangan.txt - Notepad
File Edit Format View Help
0,9.109,0.00000
0.5,18.069,2.23008
1,27.028,2.65039
1.5,35.988,2.79844
2,44.947,2.86720
2.5,53.907,2.90463
3,62.866,2.92723
3.5,71.826,2.94191
4,80.785,2.95199
4.5,89.745,2.95920
5,98.704,2.96454

```

```

0,9.109,0.00000
0.5,18.069,2.23008
1,27.028,2.65039
1.5,35.988,2.79844
2,44.947,2.86720
2.5,53.907,2.90463
3,62.866,2.92723
3.5,71.826,2.94191
4,80.785,2.95199
4.5,89.745,2.95920
5,98.704,2.96454

```

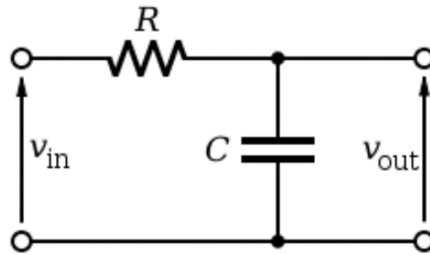
NOMOR 3 : FILTER RC

NIM = 13221601 = XXXYYZZZ

R = YY K Ω = 21K Ω

C = XXXZZZ pF = 132601 pF

V_{in} = 5 volt



a. Persamaan respon transien dan analisis

asumsi kondisi mula kapasitor tidak ada tegangan $V(0^-) = 0$

KVL :

$$-V_{in} + V_R + V_{out} = 0$$

$$i_R + V_{out} = V_{in} \quad \text{dimana } i = C \frac{dV_{out}}{dt}$$

$$RC \frac{dV_{out}}{dt} + V_{out} = V_{in}$$

dalam domain waktu (t) $\rightarrow RC \frac{dV_o(t)}{dt} + V_o(t) = V_i(t)$

dalam domain s $\rightarrow RCs V_o(s) + V_o(s) = V_i(s)$

$$V_o(s) (RCs + 1) = V_i(s)$$

$$\frac{V_o(s)}{V_i(s)} = \frac{1}{RCs + 1}$$

$$H(s) = \frac{1}{RC(s + \frac{1}{RC})}$$

$$= \frac{1}{RC} \cdot \frac{1}{(s + \frac{1}{RC})}$$

$$h(t) = \frac{1}{RC} e^{-\frac{1}{RC}t} u(t)$$

respon impuls $h(t) = 3591 e^{-3591t} u(t)$

$$\begin{aligned} * \frac{1}{RC} &= \frac{1}{21K \cdot 132601 pF} \\ &= \frac{1}{21 \cdot 10^3 \cdot 1,32601 \cdot 10^{-7}} \\ &= \frac{1}{27,84621 \cdot 10^{-4}} \\ &= 0,0359115 \cdot 10^4 \\ &= 3591,15 \\ &\approx 3591 \end{aligned}$$

* ketika mendapat input $V_{in} x(t) = 5 u(t)$

$$y(t) = h(t) * x(t)$$

$$Y(s) = (3591 e^{-3591t} * 5) u(t)$$

$$Y(s) = 3591 \frac{1}{s + 3591} \cdot \frac{5}{s}$$

$$= \frac{17955}{(s + 3591)s}$$

$$= \frac{A}{(s + 3591)} + \frac{B}{s}$$

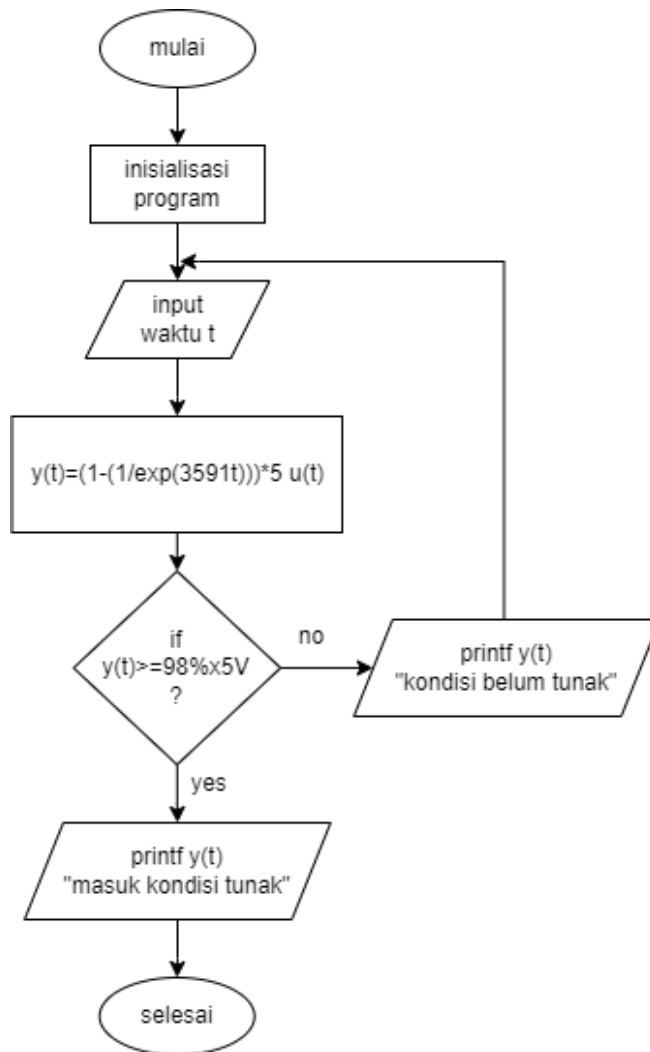
$$= \frac{-5}{(s + 3591)} + \frac{5}{s}$$

$$y(t) = (-5 e^{-3591t} + 5) u(t)$$

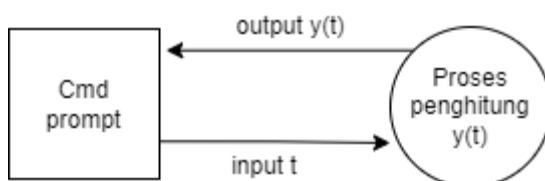
$$y(t) = (1 - e^{-3591t}) 5 u(t)$$

analisis : saat $t=0$ kondisi mula V_{out} akan 0 kemudian bergerak naik sesuai grafik fungsi diatas dan mencapai kondisi tunak saat t dimana $y(t)$ mendekati 5 volt

b. Flowchart Program



c. Data Flow Diagram (DFD)



d. Kode Program C

```

#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include<math.h>

float t;
float y;
float tunak;
float x;
float e=2.718;
  
```

```

int main(){
printf("Diketahui persamaan respons transient rangkaian RC sbb:\n");
printf("\n\t y(t)=(1-(1/exp(3591t)))*5u(t)\n\n");
printf("nilai 0.0001< t >0.0015\n");
printf("Untuk mengetahui kondisi tunak dan output y(t),\n");
start:
printf("masukkan input waktu (t)= ");
scanf("%f", &t);

x=3591*t;
y=5-(5/pow(e,x));
tunak=4.9;

if(y>=tunak && y<=5)
{
    printf("\nt=%f sekon telah masuk dalam kondisi tunak\n", t);
    printf("y(t)=%f volt\n", y);
    printf("program selesai\n");
}
else
{
    printf("\nt=%f sekon belum masuk dalam kondisi tunak\n", t);
    printf("kondisi tunak=%f volt\n", tunak);
    printf("y(t)=%f volt\n\n", y);
    goto start;
}
return 0;
}

```

Hasil Program:

```

E:\Filess\FILE_BAYU\5_KULIAH_ITB\2_SEMESTER_4\PMC\UTS\Nomor 3\progra...
Diketahui persamaan respons transient rangkaian RC sbb:
      y(t)=(1-(1/exp(3591t)))*5u(t)
nilai 0.0001< t >0.0015
Untuk mengetahui kondisi tunak dan output y(t),
masukkan input waktu (t)= 0.0005

t=0.000500 sekon belum masuk dalam kondisi tunak
kondisi tunak=4.900000 volt
y(t)=4.169623 volt

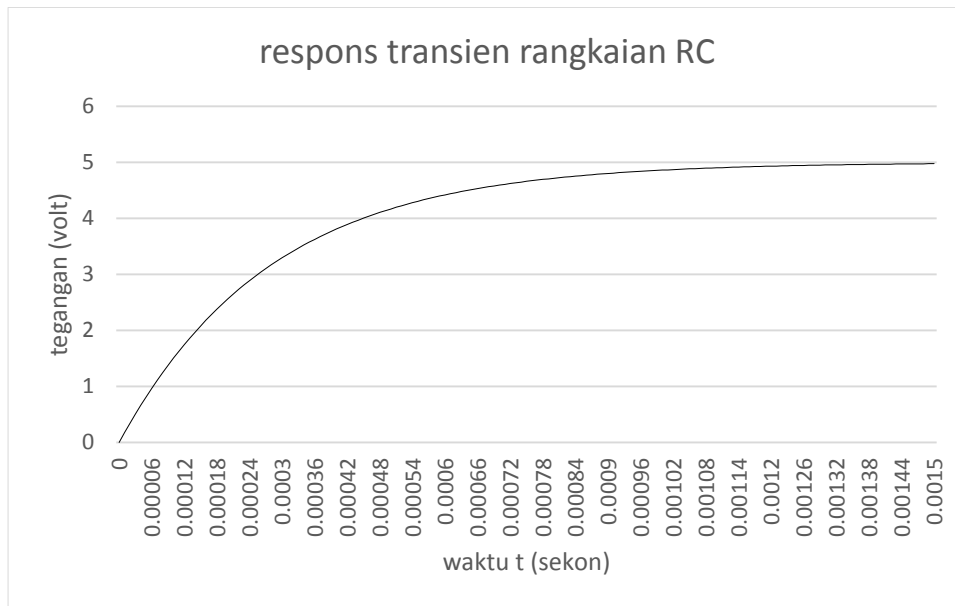
masukkan input waktu (t)= 0.0011

t=0.001100 sekon telah masuk dalam kondisi tunak
y(t)=4.903697 volt
program selesai

Process returned 0 (0x0)   execution time : 19.018 s
Press any key to continue.

```

e. Grafik Vout terhadap waktu (t)



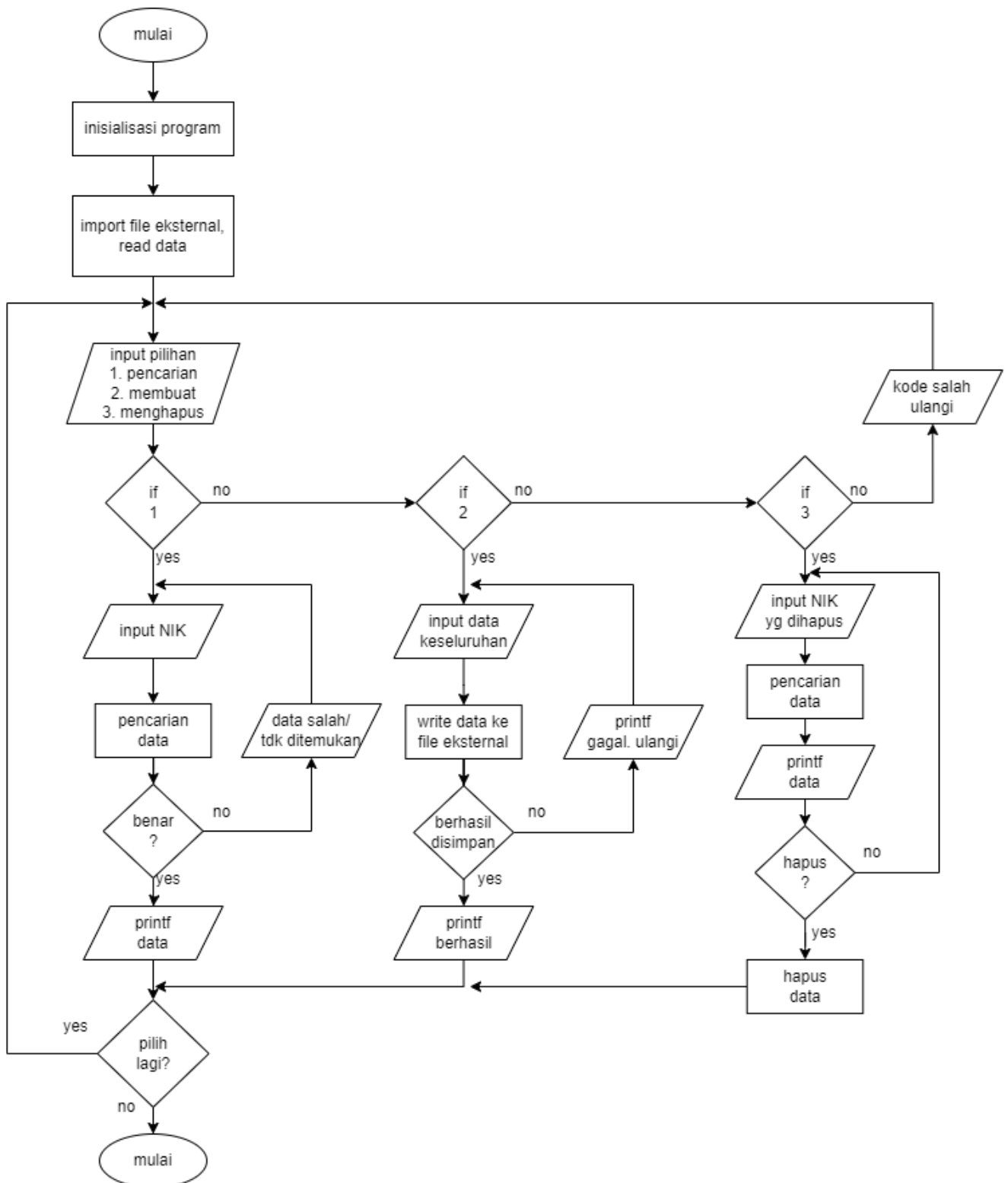
- f. konstanta waktu persamaan a adalah $1/RC=3591$. Konstanta waktu dalam program adalah $1/RC=3591$. Tanda pangkat (-) dalam e merupakan $1/e$. Oleh karena itu, tidak ada perbedaan dengan konstanta waktu dalam program.

$$y(t) = (-5e^{-3591t} + 5)u(t)$$

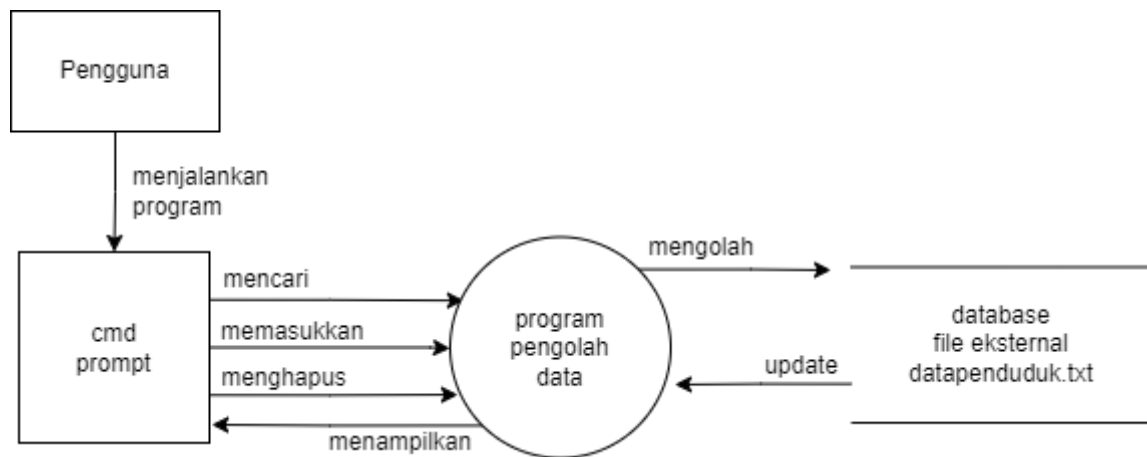
$$y(t) = (1 - e^{-3591t})5u(t)$$

NOMOR 4 : DATA KEPENDUDUKAN

a. Flowchart Pengolah Data Kependudukan



b. Data Flow Diagram (DFD)



c. Kode Program dalam Bahasa C (Program belum selesai)

```

#include<stdio.h>
#include <stdlib.h>
#include <string.h>
#define MAX_LEN 255

char filename[]="datapenduduk.txt";
int kode;

//----->>baca & cari
void baca_cari(){
printf("void baca\n");
//----->>baca
}

//----->>tulis
void tulis(){
printf("void tulis\n");
}

//----->>hapus
void hapus(){
printf("void hapus\n");
printf("hapus data?(ya=y/tidak=t): ");
char hps;
scanf("%s", &hps);
if(hps=='y')
{
printf("data terhapus\n\n");
}
else
{
printf("data tidak terhapus. silakan ulang!!\n\n");
}
}

int main(){

//----->>membuka file
FILE* stream = fopen(filename, "r");
  
```

```

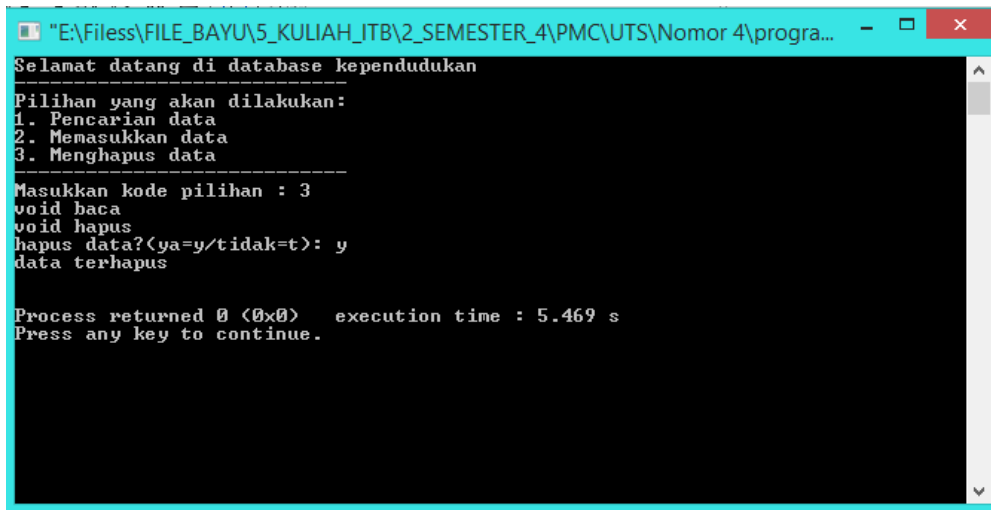
start:
printf("Selamat datang di database kependudukan\n");
printf("-----\n");
printf("Pilihan yang akan dilakukan:\n");
printf("1. Pencarian data\n");
printf("2. Memasukkan data\n");
printf("3. Menghapus data\n");
printf("-----\n");
printf("Masukkan kode pilihan : ");
scanf("%d", &kode);

if(kode==1){
    baca_cari();
}
else if(kode==2){
    tulis();
}
else if(kode==3)
{
    baca_cari();
    hapus();
}
else
{
    printf("kode salah. silahkan diulang!!\n\n");
    goto start;
}

return 0;
}

```

Hasil belum selesai:



```

E:\Fileless\FILE_BAYU\5_KULIAH_ITB\2_SEMESTER_4\PMC\UTS\Nomor 4\progra...
Selamat datang di database kependudukan
Pilihan yang akan dilakukan:
1. Pencarian data
2. Memasukkan data
3. Menghapus data
-----
Masukkan kode pilihan : 3
void baca
void hapus
hapus data?<ya=y/tidak=t>: y
data terhapus

Process returned 0 (0x0)   execution time : 5.469 s
Press any key to continue.

```