

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 : Openbook

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

Nama : Muhammad Hanif Hibatullah
NIM : 13220051

1. a. Representasi integer dengan **unsigned** akan dimulai dari angka 0 sampai 2^n-1 dengan n merupakan banyak bit yang direpresentasikan, hal ini mirip dengan range dari unsigned int pada bahasa C. Sehingga jika kita memiliki 16 bit digit yang akan direpresentasikan dalam **unsigned** maka nilai tertinggi yang didapatkan adalah $2^{16}-1 = 65535$ dan nilai terendah adalah 0. Untuk representasi integer dengan **2's-complement** akan dimulai dari angka -2^{n-1} sampai dengan $2^{n-1}-1$ hal ini mirip dengan range integer pada bahasa C. Sehingga jika kita memiliki 16 bit digit yang akan direpresentasikan dalam **2's-complement** maka nilai tertingginya adalah $2^{15}-1 = 32767$ dan nilai terendahnya adalah -32768.

b. Bilangan yang saya dapatkan: 32051

Dalam desimal dapat dinyatakan sebagai:

$$32051 = 3 \times 10^4 + 2 \times 10^3 + 0 \times 10^2 + 5 \times 10^1 + 1 \times 10^0$$

Sehingga dalam desimal 32051 dapat direpresentasikan sebagai 32051_{10} .

Dalam biner dapat dinyatakan sebagai:

$$32051 = 1 \times 2^{14} + 1 \times 2^{13} + 1 \times 2^{12} + 1 \times 2^{11} + 1 \times 2^{10} + 0 \times 2^9 + 1 \times 2^8 + 0 \times 2^7 + 0 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$

Sehingga dalam biner 32051 dapat direpresentasikan sebagai 111110100110011_2 .

c. least significant byte dari jawaban b adalah A = 0011 0011 (karena byte = 8 bits).

d. A = 0011 0011

x	y	Operasi	Binary (least significant byte)	Hexa	Unsigned	Signed 2C
1010 0101	1110 1111	$(A \& y) \wedge x$	1000 0110	86	134	-122
1101 1011	1111 0011	$(\sim(y \gg 4) \& x) + A$	1001 1101	9D	157	-99
0010 1111	1010 0101	$((x \gg 2) y) * A$	0101 1101	5D	93	93

Diasumsikan x, y, A merupakan **signed binary** (bukan **signed 2's complement**) dan diasumsikan pula hasilnya hanya diambil least significant bytenya.

Operasi operasi = & (and), ^ (xor), >> (shift right), ~ (negation), + (add), | (or), * (multiplication)

Soal pertama:

$$(A \& y) = (0011\ 0011 \& 1110\ 1111) = (0010\ 0011)$$

$$(A \& y) \wedge x = (0010\ 0011) \wedge (1010\ 0101) = 1000\ 0110$$

Soal kedua:

$$(\sim(y \gg 4) \& x) \text{ diasumsikan hanya satu tanda } \& = (\sim(1111\ 0011 \gg 4) \& 1101\ 1011)$$

$$(\sim(y \gg 4) \& x) = (\sim(0000\ 1111) \& 1101\ 1011) = (1111\ 0000 \& 1101\ 1011) = 1101\ 0000$$

$$(\sim(y \gg 4) \& x) + A = 1101\ 0000 + 0011\ 0011 = -80 + 51 = -29 = 1001\ 1101$$

Soal ketiga:

$((x \gg 2) | y) = ((0010\ 1111 \gg 2) | 1010\ 0101) = (0000\ 1011 | 1010\ 0101) = 1010\ 1111$

$((x \gg 2) | y) * A = 1010\ 1111 * 0011\ 0011 = (-47) * 51 = -2397 = 1\ 1001\ 0101\ 1101$

Dimana ketika diambil least significant bytenya didapatkan hasil dari soal ketiga yaitu 0101 1101

e. Untuk program yang dibuat dengan bahasa C dapat diakses pada link berikut:

<https://github.com/HanifHBK/SolusiUTSPMC> dengan nama file 1e.c

f. Hasil yang didapatkan dari program:

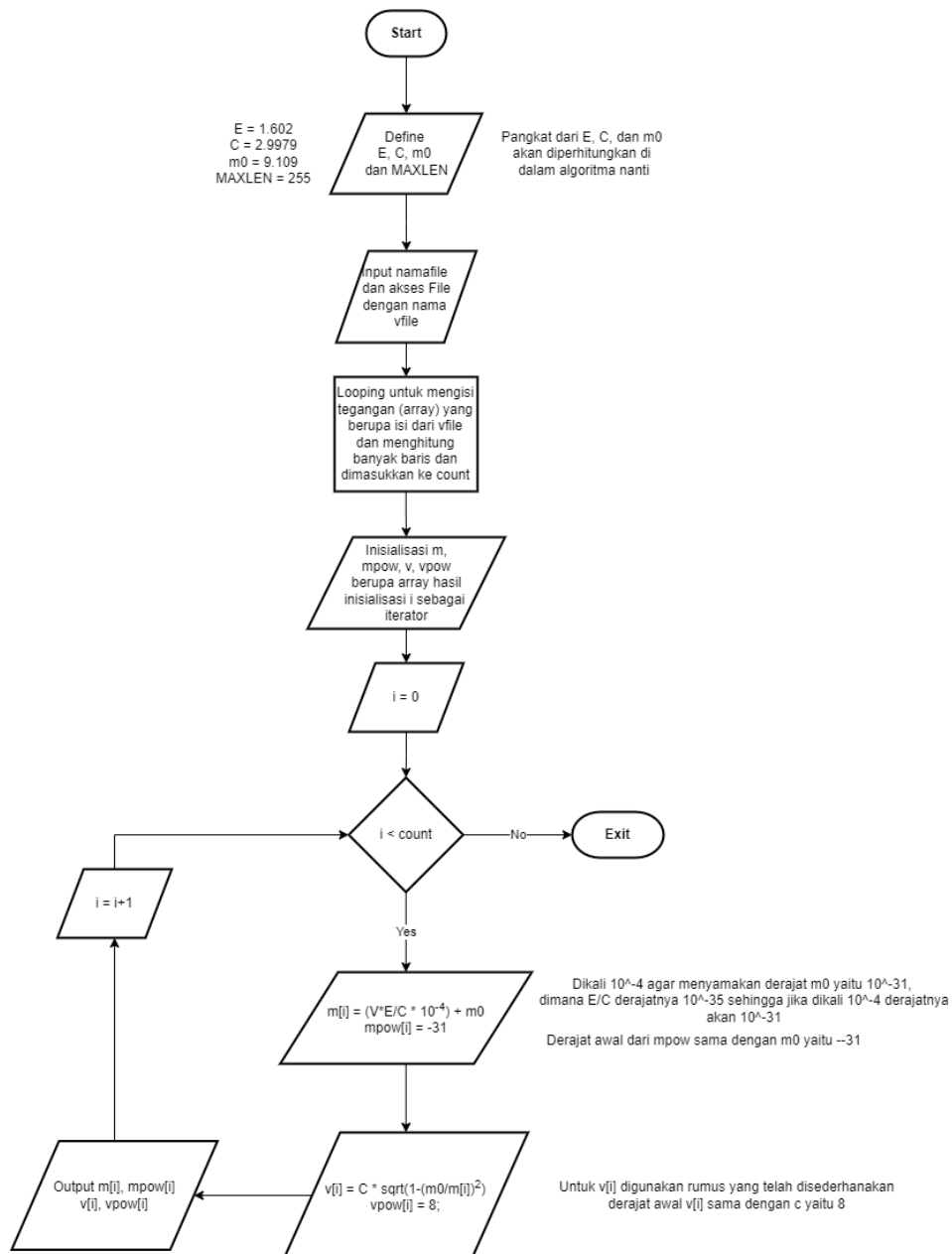
```
; if ($?) { gcc 1e.c -o 1e } ; if ($?) { .\1e }ro\Pemecahan Masalah dengan C\UTS>
Program untuk menghitung operasi bitwise dengan masukan NIM (diambil digit kedua ketiga, dan 3 digit terakhir).
Masukkan NIM anda : 13220051
NIM anda dalam bentuk binary (32051) : 111110100110011
A dalam binary (least significant byte) : 00110011
-----
x = 10100101 dan y = 11101111
Soal pertama : (A & y) ^ x
Binary : 10000110
Hexa : 86
Unsigned : 134
Signed 2C : -122
-----
x = 11011011 dan y = 11110011
Soal Kedua : ~(y >> 4) & x) + A
Binary : 10011101
Hexa : 9d
Unsigned : 157
Signed 2C : -99
-----
x = 00101111 dan y = 10100101
Soal Kedua : ((x >> 2) | y) * A
Binary : 01011101
Hexa : 5d
Unsigned : 93
Signed 2C : 93
-----
PS C:\Users\muham\Downloads\ITB Semester 4 Elektro\Pemecahan Masalah dengan C\UTS> 
```

Dimana dapat dilihat bahwa tidak terdapat perbedaan hasil untuk NIM saya (tidak dilakukan pengecekan dengan NIM lain).

2. Terlebih dahulu rumus untuk mencari m dan v disederhanakan

$$m = \frac{V \cdot e}{c^2} + m_o$$
$$v = c \sqrt{1 - \left(\frac{m_o}{m}\right)^2}$$

a. Flowchart yang digunakan:



Program dalam bahasa C (kode yang lebih jelas dapat diakses pada link):

<https://github.com/HanifHBK/SolusiUTSPMC> berupa file 2a.c

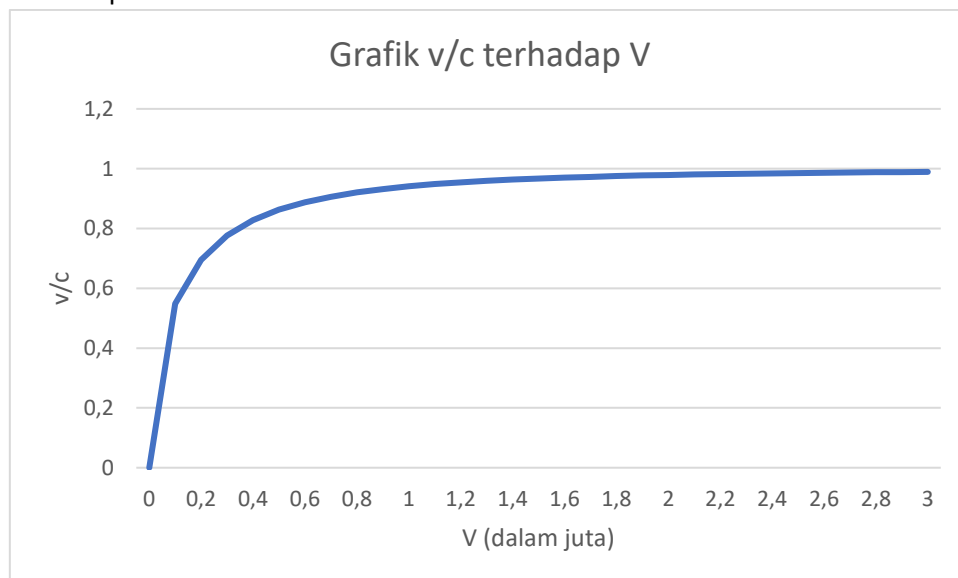
```

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

#define C 2.9979
#define E 1.602
#define m0 9.109
#define MAXLEN 255
int main(){
    char namafile[MAXLEN];
    printf("Masukkan nama file berisi Tegangan: ");
    scanf("%s", &namafile);
    FILE* vfile = fopen(namafile, "r");
    char line[MAXLEN];
    float Tegangan[MAXLEN];
    int count = 0;
    while(fgets(line, MAXLEN, vfile)){
        Tegangan[count] = atof(line);
        count++;
    }
    int i;
    float m[count], mpow[count], v[count], vpow[count];
    for(i = 0; i < count; i++){
        m[i] = (Tegangan[i]*E*pow(10,-4)/(C*C)) + m0;
        mpow[i] = -31;
        while(m[i] >= 10){
            mpow[i] += 1;
            m[i] /= 10;
        }
        v[i] = C * pow((1-pow((m0*pow(10,-31-mpow[i])/m[i]),2)), 0.5);
        vpow[i] = 8;
        while(v[i] <= 1){
            vpow[i] -= 1;
            v[i] *= 10;
        }
        printf("Untuk tegangan %.0f, didapatkan m = %.3f x 10^(%.0f) dan v = %.3f x 10^(%.0f)\n", Tegangan[i], m[i], mpow[i], v[i], vpow[i]);
    }
}

```

b. Plot v/c terhadap V:



c. Isi file tegangan.txt:

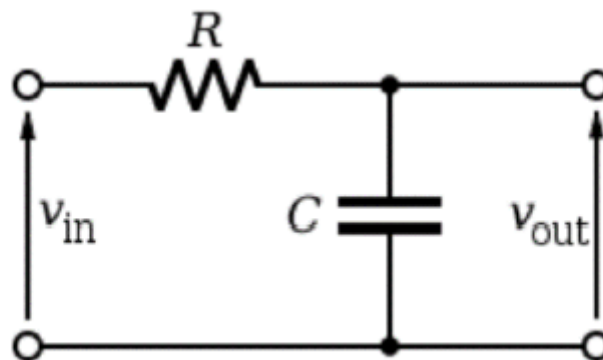
tegangan.txt :

1000000
1100000
1200000
1300000
1400000
1500000
1600000
1700000
1800000
1900000

Output dari input berupa tegangan.txt:

```
PS C:\Users\muham\Downloads\ITB Semester 4 Elektro\Pemecahan Masalah dengan C\UTS>
; if ($?) { gcc 2a.c -o 2a } ; if ($?) { .\2a }
Masukkan nama file berisi Tegangan: tegangan.txt
Untuk tegangan 1000000, didapatkan m = 2.693 x 10(-30) dan v = 2.821 x 10(8)
Untuk tegangan 1100000, didapatkan m = 2.872 x 10(-30) dan v = 2.843 x 10(8)
Untuk tegangan 1200000, didapatkan m = 3.050 x 10(-30) dan v = 2.861 x 10(8)
Untuk tegangan 1300000, didapatkan m = 3.228 x 10(-30) dan v = 2.876 x 10(8)
Untuk tegangan 1400000, didapatkan m = 3.406 x 10(-30) dan v = 2.889 x 10(8)
Untuk tegangan 1500000, didapatkan m = 3.585 x 10(-30) dan v = 2.899 x 10(8)
Untuk tegangan 1600000, didapatkan m = 3.763 x 10(-30) dan v = 2.909 x 10(8)
Untuk tegangan 1700000, didapatkan m = 3.941 x 10(-30) dan v = 2.917 x 10(8)
Untuk tegangan 1800000, didapatkan m = 4.119 x 10(-30) dan v = 2.924 x 10(8)
Untuk tegangan 1900000, didapatkan m = 4.298 x 10(-30) dan v = 2.930 x 10(8)
```

3.



$R = 20 \text{ k}\Omega$

$C = 132051 \text{ pF}$

$V_{in} = 5 \text{ Volt}$

a. Dengan KVL didapatkan

$$-v_{in} + RI + v_{out} = 0$$

Ingat bahwa $I = I_c = C \frac{dv_{out}}{dt}$ sehingga

$$-v_{in} + RC \frac{dv_{out}}{dt} + v_{out} = 0$$

$$RC \frac{dv_{out}}{dt} + v_{out} = v_{in}$$

Dengan menganggap solusi umum dari adalah $v_{out} = v_p + v_h$

Solusi particular:

$$RC \frac{dv_{out}}{dt} + v_{out} = 5$$

Dapat dipilih $v_{out} = 5$ sehingga persamaan memenuhi.

Solusi homogen:

$$RC \frac{dv_{out}}{dt} + v_{out} = 0$$

Dengan menganggap solusi umum dari v_{out} adalah Ae^{-St} didapatkan

$$(-SRC + 1)Ae^{-St} = 0$$

$$S = \frac{1}{RC} \approx 378,642$$

maka didapatkan

$$v_{out} = 5 + Ae^{-378,642t} \text{ V}$$

Dengan asumsi kondisi awal ($t = 0$) kapasitor sudah lama kosong sehingga $v(0) = 0$ didapatkan $A = -5$

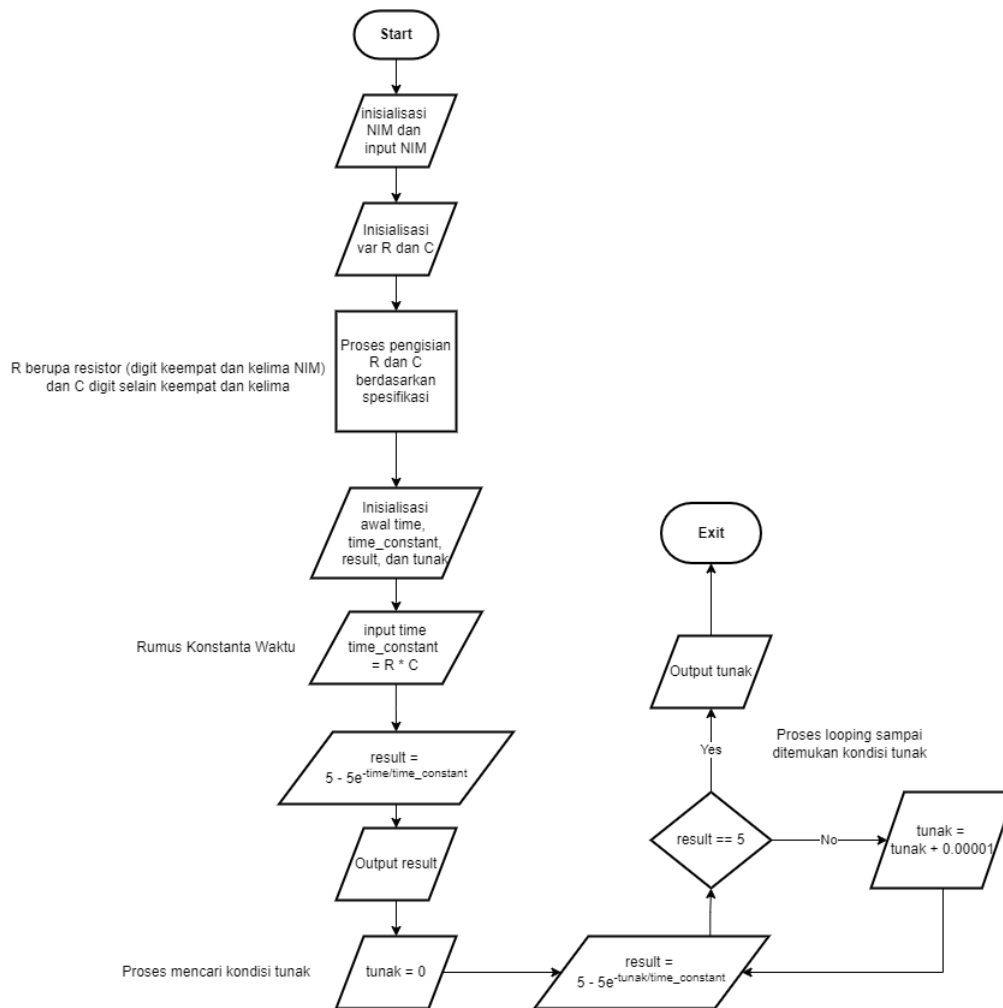
$$v_{out} = 5 - 5e^{-378,642t} \text{ V}$$

Saat kondisi tunak atau $t \rightarrow \infty$ maka didapatkan $v_{out} = 5 \text{ V}$

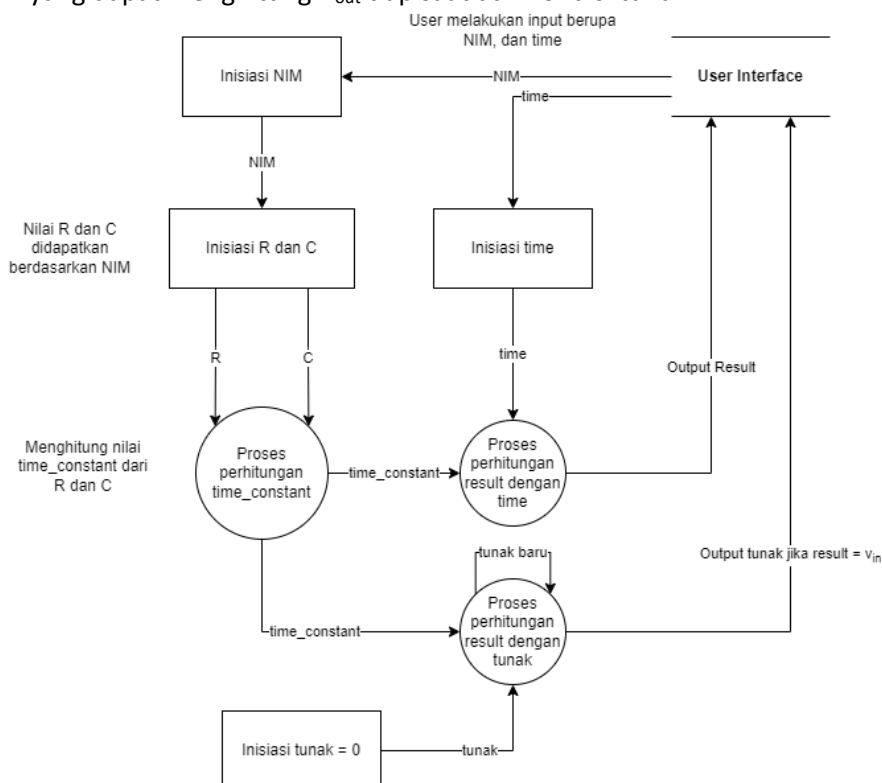
Dalam domain s dengan menggunakan transformasi laplace didapatkan fungsinya adalah

$$V_{out}(s) = V_{in}(s) \times \frac{1}{RCs + 1} = \frac{5}{s} \times \frac{1}{0,002641s + 1} \text{ V}$$

b. Flowchart program yang dapat menghitung V_{out} tiap saat dan kondisi tunak.



c. DFD program yang dapat menghitung V_{out} tiap saat dan kondisi tunak.



d. Program untuk menghitung V_{out} pada setiap saat dan saat kondisi tunak dengan masukan berupa NIM dan t yang akan dicari. Output yang dihasilkan berupa tegangan saat t input dan nilai t saat kondisi tunak.

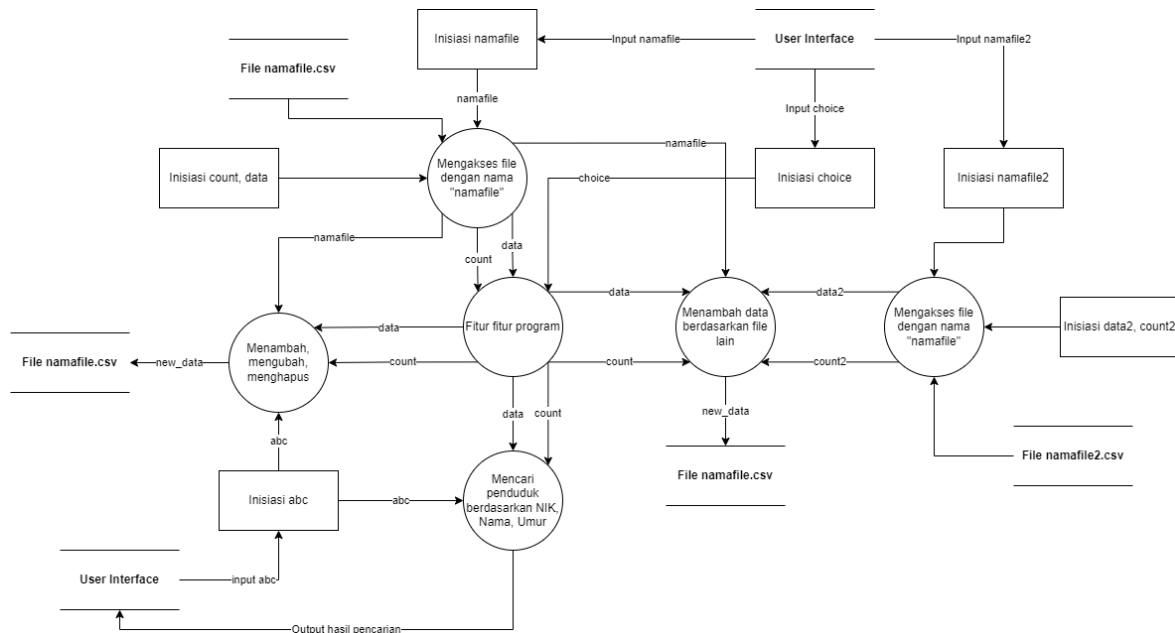
```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>

int main(){
    char NIM[8];
    printf("Program untuk menghitung nilai Vout rangkaian RC dengan spesifikasi sebagai berikut.\n");
    printf("Nilai R adalah digit keempat dan kelima NIM yang akan anda input (dalam kOhm)\n");
    printf("Nilai C adalah digit pertama, kedua, ketiga, keenam, ketujuh dan kedelapan NIM anda (dalam pF)\n");
    printf("Nilai Vin adalah tetap yaitu 5V.\n");
    printf("Masukkan NIM anda : ");
    scanf("%s", &NIM);
    char Rs[2];
    char Cs1[2];
    char Cs2[4];
    strncpy(Rs, NIM + 3, 2);
    strncpy(Cs1, NIM, 3);
    strncpy(Cs2, NIM + 5, 3);
    char Cs[6];
    strncpy(Cs, Cs1, 3);
    strcat(Cs, Cs2);
    double R, C;
    R = atof(Rs);
    C = atof(Cs);
    printf("Dari data NIM anda, maka diperoleh nilai R dan C berturut turut adalah %.0f kOhm dan %.0f pF\n", R, C);
    double time;
    printf("Masukkan nilai waktu (dalam s) yang ingin anda cari tegangan Voutnya (gunakan . sebagai pemisah desimal) : ");
    double result;
    scanf("%lf", &time);
    double time_constant = R * C * pow(10, -9);
    result = 5 - (5 * exp(-time/time_constant));
    printf("Nilai time constant berdasarkan NIM anda : %lf\n", time_constant);
    printf("Pada saat t = %lf s nilai Tegangan Vout adalah = %lf V\n", time, result);
    double tunak = 0;
    result = 5 - (5 * exp(-tunak/time_constant));
    while(result != 5){
        tunak += 0.000001;
        result = 5 - (5 * exp(-tunak/time_constant));
    }
    printf("Rangkaian tunak pada saat t = %lf s", tunak);
}
```

Kode yang lebih jelas dapat dilihat pada link berikut:

<https://github.com/HanifHBK/SolusiUTSPMC> dengan nama file 3d.c

e. Grafik V_{out} terhadap waktu (t):



c. Aplikasi dengan bahasa C tersebut dapat diakses pada link berikut:

<https://github.com/HanifHBK/SolusiUTSPMC> dengan nama file 4c.c

Note:

1. Semua grafik dibuat dengan excel dan filenya ada di github.
2. Semua flowchart dan DFD dibuat dengan draw.io yang dimana file pngnya ada di github.
3. Semua source code ada di github dan untuk source code yang lumayan panjang tidak dilampirkan di file jawaban.

Link github: <https://github.com/HanifHBK/SolusiUTSPMC>