UJIAN PRAKTIKUM FISIKA KOMPUTASI
SEMESTER 1 2007/2008

Soal dan Solusi

oleh:

Ahmad Ridwan (10204001)

Program Studi Fisika
Institut Teknologi Bandung
Desember 2007

Ujian Praktikum Fisika Komputasi

Kerjakan seluruh perintah dalam soal. Apabila dalam soal diminta untuk membuat program atau grafik, simpanlah hasilnya dalam folder khusus sesuai kesepakatan dengan asisten pengawas. Untuk soal-soal yang mengharapkan jawaban analitik, tuliskan penjelasannya pada lembar jawaban.

Waktu: 2 jam, **Bahasa Pemrograman**: C/C++, **Sifat Ujian**: *Open Book*

■ KODE SOAL: FI3102-K

(Jika diperlukan, gunakan $c = 3 \times 10^8$ m/s, $h = 6,63 \times 10^{-34}$ J s, $k = 1,38 \times 10^{-23}$ J/K, 1 eV $= 1,6 \times 10^{-19}$ J, dan $\pi = 3,14$.)

Kerapatan energi spektral $u(\lambda)$ dari suatu benda hitam dirumuskan oleh hukum radiasi Planck:

$$u(\lambda)d\lambda = \frac{8\pi hc}{\lambda^5}\frac{d\lambda}{\left[\exp\left(\frac{hc}{\lambda kT}\right) - 1\right]}.$$

(A) Turunkan hukum pergeseran Wien dari rumusan Planck tersebut, yaitu temukan:

$$\lambda_m.T = C_w,$$

dengan λ_m adalah panjang gelombang yang menghasilkan rapat energi maksimum pada temperatur T tertentu dan C_w adalah konstanta. Langkah-langkah yang perlu dilakukan adalah:

(A1) Supaya rapat energi jadi maksimum, maka bagian 'penyebut' dari ruas kanan hukum radiasi Planck harus minimum. Tuliskan bentuk matematik yang tepat untuk kondisi tersebut (ingat materi kalkulus)! Selesaikan (secara analitik) hingga diperoleh bentuk yang harus menggunakan metode numerik tertentu! **[nilai maksimum: 25]**

(A2) Hitung nilai C_w (konstanta Wien) dengan metode numerik yang tepat dalam bentuk sebuah program berdasarkan langkah terakhir soal A1! **[nilai maksimum: 35]**

(B) Tunjukkan bahwa bentuk rapat energi yang bergantung pada frekuensi adalah

$$u(\nu)d\nu = \frac{8\pi h}{c^3}\frac{\nu^3 d\nu}{\left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]}.$$

Manfaatkan hubungan frekuensi, kecepatan cahaya, dan panjang gelombang: $\nu = c/\lambda$.
[nilai maksimum: 10]

(C) Buatlah kurva $u(\nu)$ terhadap ν untuk frekuensi $0,1 \leq \nu \leq 4,5(\times 10^{14})$ Hz. Gunakan interval penambahan frekuensi sebesar 0,1 (dalam satuan 10^{14} Hz). Satuan $u(\nu)$ yang diinginkan pada kurva adalah dalam eV s/m^3, sedangkan kurva yang diinginkan ada dua buah: $u(\nu)$ vs ν untuk $T = 1200$ K dan $u(\nu)$ vs ν untuk $T = 1800$ K. Sebisa mungkin kedua kurva dibuat dalam satu grafik agar mudah dibandingkan. Grafik boleh dibuat dengan menggunakan $spreadsheets$ (semacam MS Excel atau OpenOffice Calc). **[nilai maksimum: 25]**

(D) Jelaskan terjadinya 'pergeseran' puncak kerapatan energi pada temperatur yang berbeda, sesuai hukum Wien, untuk soal C! **[nilai maksimum: 5]**

Ujian Praktikum Fisika Komputasi

Kerjakan seluruh perintah dalam soal. Apabila dalam soal diminta untuk membuat program atau grafik, simpanlah hasilnya dalam folder khusus sesuai kesepakatan dengan asisten pengawas. Untuk soal-soal yang mengharapkan jawaban analitik, tuliskan penjelasannya pada lembar jawaban.

Waktu: 2 jam, **Bahasa Pemrograman**: C/C++, **Sifat Ujian**: *Open Book*

■ KODE SOAL: FI3102-L

(A) Pada sebuah devais nonlinear yang iseng dibuat di laboratorium, arus dan tegangan dihubungkan oleh persamaan

$$i(v) = k\left(1 + \frac{v}{V_{DC}}\right)^{1,5},$$

dengan k adalah konstanta dan V_{DC} komponen searah dari tegangan sesaat v.

(A1) Uraikan fungsi ini ke dalam bentuk deret pangkat sampai orde pangkat tiga, yang dapat digunakan untuk menghitung arus i secara lebih eksak! Asumsikan v cukup kecil dan bervariasi di sekitar titik nol.

(A2) Misalkan $k = 1$ dan $V_{DC} = 1$. Buatlah program untuk menghitung $i(v)$ pada selang $v = [-1, 1]$ sebanyak 101 titik dengan partisi yang sama!

(A3) Buatlah kurva i terhadap v sesuai keluaran dari program A2. Kurva boleh dibuat dengan alat bantu *spreadsheets* (semacam MS Excel atau OpenOffice Calc).

(B) Sebuah gelombang pada suatu saat dapat dimodelkan dengan memperluas fungsi

$$f(x) = -x^2 + 4x,$$

pada $0 \leq x \leq 4$ menjadi fungsi *genap* yang periodik.

(B1) Tunjukkan bahwa fungsi genap yang dimaksud adalah

$$f(x) = \frac{8}{3} - 32\sum_{n=1}^{\infty} \frac{\cos n\pi + 1}{(n\pi)^2} \cos\left(\frac{n\pi x}{4}\right),$$

dengan cara uraian deret Fourier!

(B2) Buatlah program untuk menghitung $f(x)$ pada selang $-4 \leq x \leq 4$ sebanyak 101 titik partisi yang sama! Program harus disertai dengan toleransi *error* 10^{-4}, yaitu selisih antara nilai fungsi sebenarnya yang merupakan perluasan *genap* dari $f(x) = -x^2 + 4x$ terhadap nilai fungsi dari deret Fourier. Tentukanlah jumlah n iterasi yang perlu dilakukan untuk keperluan tersebut! (Program yang benar tentu tidak perlu menghitung sampai $n = \infty$).

(B3) Buatlah kurva $f(x)$ terhadap x sesuai keluaran dari program B2.

Ujian Praktikum Fisika Komputasi

Kerjakan seluruh perintah dalam soal. Apabila dalam soal diminta untuk membuat program atau grafik, simpanlah hasilnya dalam folder khusus sesuai kesepakatan dengan asisten pengawas. Untuk soal-soal yang mengharapkan jawaban analitik, tuliskan penjelasannya pada lembar jawaban.

Waktu: 2 jam, **Bahasa Pemrograman**: C/C++, **Sifat Ujian**: *Open Book*

■ KODE SOAL: FI3102-M

(A) Diketahui data posisi setiap saat dari sebuah partikel pada sumbu-x positif:

t (detik)	0	2	3	10	14	15
x (meter)	3,01	6,99	12,04	103,1	199,2	228,13

(A1) Lengkapi tabel tersebut dengan metode interpolasi polinom Lagrange untuk $0 \leq t \leq 15$ dan selang penambahan waktu sebesar 1 detik! Hitunglah kecepatan dan percepatan partikel setiap detik untuk $1 \leq t \leq 14$ dengan metode beda sentral orde terendah! Buatlah programnya dalam satu *file*.

(A2) Gambarkan kurva posisi, kecepatan, dan percepatan partikel terhadap waktu dalam selang $1 \leq t \leq 14$. Usahakan ketiganya berada dalam satu grafik dan kurva-kurva tersebut harus berdasar pada keluaran program dari soal A1. Penggunaan *spreadsheets* (semacam MS Excel atau OpenOffice Calc) diperbolehkan sebagai alat bantu.

(A3) Tafsirkan jenis gerak partikel yang mungkin terjadi, dengan melihat hasil grafik yang telah dibuat! (Ingat materi fisika dasar.)

(B) Seorang penerjun payung jatuh dari sebuah pesawat. Kecepatannya sebagai fungsi dari waktu adalah

$$v(t) = \frac{gm}{b}\left[1 - e^{(-b/m)t}\right],$$

dengan $g = 9,8$ m/s dan $b = 12,5$ kg/s.

(B1) Tuliskan rumusan yang tepat untuk jarak penerjun sebagai fungsi dari waktu! (Nyatakan dalam bentuk integral saja)

(B2) Hitung seberapa jauh penerjun jatuh setelah waktu $t = 10$ detik dengan metode integrasi trapesium! Buat programnya.

Ujian Praktikum Fisika Komputasi

Kerjakan seluruh perintah dalam soal. Apabila dalam soal diminta untuk membuat program atau grafik, simpanlah hasilnya dalam folder khusus sesuai kesepakatan dengan asisten pengawas. Untuk soal-soal yang mengharapkan jawaban analitik, tuliskan penjelasannya pada lembar jawaban.

Waktu: 2 jam, **Bahasa Pemrograman**: C/C++, **Sifat Ujian**: *Open Book*

■ KODE SOAL: FI3102-N

Bola besi bermassa 0,5 kg digantungkan pada seutas tali dengan panjang 1 meter dan memiliki massa yang sangat kecil. Sistem (bandul) tersebut kemudian diberi simpangan 60^o terhadap garis vertikal dan dilepaskan dari keadaan diam. Bandul diasumsikan dapat kembali ke posisinya semula.

(A) Tuliskan persamaan gerak bandul dinyatakan dalam besaran sudut!

(B) Hitung waktu yang dibutuhkan bandul untuk mencapai kecepatan maksimumnya pertama kali, dengan cara membuat sebuah program yang menggunakan metode numerik tertentu untuk pemecahan persamaan diferensial!

(C) Gambarkan kurva simpangan sudut dan kecepatan bandul dalam satu grafik, selama perjalanan bandul dari posisi awal hingga tercapainya kecepatan maksimum berdasarkan soal B!

(D) Hitunglah waktu yang dibutuhkan bandul untuk kembali ke posisi semula dengan modifikasi program pada soal B dan dibuat dalam *file* terpisah! Apakah waktu tersebut sama dengan 4 kali waktu yang dibutuhkan bandul untuk mencapai kecepatan maksimum pertama kalinya? Jelaskan!

(E) Gambarkan kurva simpangan sudut dan kecepatan bandul dalam satu grafik, untuk perjalanan bandul dari posisi awal hingga kembali lagi ke posisi tersebut! Data untuk keperluan ini harus diperoleh dari keluaran program soal D.

Ujian Praktikum Fisika Komputasi

Kerjakan seluruh perintah dalam soal. Apabila dalam soal diminta untuk membuat program atau grafik, simpanlah hasilnya dalam folder khusus sesuai kesepakatan dengan asisten pengawas. Untuk soal-soal yang mengharapkan jawaban analitik, tuliskan penjelasannya pada lembar jawaban.

Waktu: 2 jam, **Bahasa Pemrograman**: C/C++, **Sifat Ujian**: *Open Book*

■ KODE SOAL: FI3102-O

(A) Berikan uraian singkat tentang metode dekomposisi LU dan gambarkan algoritmanya! Buatlah sebuah program dengan metode tersebut untuk mencari nilai arus I_1, I_2, dan I_3:

$$\begin{bmatrix} 2 & 3 & -2 \\ 1 & -6 & 4 \\ 4 & -1 & 6 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}.$$ **(nilai maks = 40)**

(B) Kerapatan energi spektral suatu benda hitam terhadap frekuensinya dirumuskan oleh hukum radiasi Planck:

$$u(\nu)d\nu = \frac{8\pi h}{c^3} \frac{\nu^3 d\nu}{\left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]}.$$

(B1) Ada frekuensi ν_m tertentu agar rapat energi suatu benda hitam bernilai maksimum. Syarat yang harus dipenuhi sesuai hukum radiasi Planck tersebut adalah:

$$\frac{\nu^3}{\exp\left(\frac{h\nu}{kT}\right) - 1}$$

harus bernilai *maksimum*.

Tuliskan syarat tersebut secara matematis (ingat materi kalkulus) dan pecahkan hingga ditemui bentuk akhir persamaan nonlinear yang harus diselesaikan dengan metode numerik tertentu! Buatlah program untuk menyelesaikannya dan buktikan bahwa:

$$\frac{\nu_m}{T} = \text{konstanta},$$

kemudian temukan nilai konstanta yang dimaksud! **(nilai maks = 40)**

(B2) Buatlah kurva $u(\nu)$ terhadap ν untuk frekuensi $0,1 \leq \nu \leq 4,5(\times 10^{14})$ Hz. Gunakan interval penambahan frekuensi sebesar 0,1 (dalam satuan 10^{14} Hz). Satuan $u(\nu)$ yang diinginkan pada kurva adalah dalam eV s/m^3, sedangkan kurva yang diinginkan ada dua buah: $u(\nu)$ vs ν untuk $T = 1200$ K dan $u(\nu)$ vs ν untuk $T = 2000$ K. Sebisa mungkin kedua kurva dibuat dalam satu grafik agar mudah dibandingkan. Grafik boleh dibuat dengan menggunakan $spreadsheets$ (semacam MS Excel atau OpenOffice Calc). Jika diperlukan, gunakan data $c = 3 \times 10^8$ m/s, $h = 6,63 \times 10^{-34}$ J s, $k = 1,38 \times 10^{-23}$ J/K, 1 eV $= 1,6 \times 10^{-19}$ J, dan $\pi = 3,14$. **(nilai maks = 15)**

(B3) Tafsirkan kurva yang telah dibuat pada soal B2, berdasarkan rumusan hasil B1! **(nilai maks = 5)**

SOLUSI Ujian Praktikum Fisika Komputasi

■ KODE SOAL: FI3102-K

Hukum radiasi Planck:

$$u(\lambda)d\lambda = \frac{8\pi hc}{\lambda^5}\frac{d\lambda}{\left[\exp\left(\frac{hc}{\lambda kT}\right) - 1\right]}.$$

(A) Mencari hukum pergeseran Wien: $\lambda_m.T = C_w$

(A1) Supaya rapat energi jadi maksimum, maka bagian 'penyebut' dari ruas kanan hukum radiasi Planck harus minimum. Bentuk matematik yang tepat adalah:

$$\frac{d}{d\lambda}\left(\lambda^5\left[\exp\left(\frac{hc}{\lambda kT}\right) - 1\right]\right) = 0.$$

Dari rumusan tersebut, dapat diperoleh λ_m:

$$\begin{aligned}
\lambda_m^4\left[\exp\left(\frac{hc}{\lambda_m kT}\right) - 1\right] + \lambda_m^5\left(-\frac{hc}{\lambda_m^2 kT}\right)\exp\left(\frac{hc}{\lambda_m kT}\right) &= 0\\
\Leftrightarrow 5\lambda_m^4\left[\exp\left(\frac{hc}{\lambda_m kT}\right) - 1\right] &= \lambda_m^5\left(\frac{hc}{\lambda_m^2 kT}\right)\exp\left(\frac{hc}{\lambda_m kT}\right)\\
\Leftrightarrow 5\left[\exp\left(\frac{hc}{\lambda_m kT}\right) - 1\right] &= \frac{hc}{\lambda_m kT}\exp\left(\frac{hc}{\lambda_m kT}\right).
\end{aligned}$$

Misalkan

$$x = \frac{hc}{\lambda_m kT},$$

sehingga

$$5[\exp(x) - 1] = x\exp(x) \Leftrightarrow \exp(x) - 1 = \frac{x}{5}\exp(x),$$

$$\therefore 1 - \exp(-x) = \frac{x}{5}.$$

(A2) Untuk menghitung konstanta Wien, metode numerik yang paling tepat digunakan adalah metode iterasi. Nilai x dari bentuk terakhir soal A1 harus dicari, kemudian kembalikan lagi ke permisalan $x = \frac{hc}{\lambda_m kT}$ sehingga ditemukan hukum pergeseran Wien. Untuk keperluan tersebut, tulislah:

$$x_{n+1} = 5[1 - \exp(-x_n)],$$

dan pecahkan x dengan program metode iterasi:

```
/* iterasi.c */
#include <stdio.h>
#include <math.h>
main()
{
    float x0, x, y, dx;
    x0 = 5.; /* tebakan awal */
    do {
        x = 5 * (1 - exp(-x0));
        dx = fabs(x - x0); /* selisih untuk konvergensi */
        x0 = x; /* metode iterasi */
```

```
    } while (dx >= 10e-6); /* konvergensi sampai 0,000001 */
    printf("x = %.4f", x);
    getchar();
}
```

sehingga diperoleh $x = 4,9651$. Substitusikan $x = \frac{hc}{\lambda_m kT}$, lalu masukkan nilai h, c, serta k:

$$
\begin{aligned}
4,9651 &= \frac{6,63 \times 10^{-34} \times 3 \times 10^8}{1,38 \times 10^{-23} \times \lambda_m T} \\
\lambda_m T &= C_w = 2,898 \times 10^{-3} \text{ m K.}
\end{aligned}
$$

(B) Rapat energi bergantung frekuensi dapat diperoleh dari hukum radiasi Planck dalam panjang gelombang dengan menggunakan hubungan

$$\nu = \frac{c}{\lambda}.$$

Dari hubungan tersebut, dapat diturunkan

$$\lambda = \frac{c}{\nu} \Leftrightarrow d\lambda = \frac{c}{\nu^2} d\nu.$$

Substitusikan pada hukum radiasi Planck:

$$
\begin{aligned}
u(\lambda) d\lambda &= \frac{8\pi hc}{\lambda^5} \frac{d\lambda}{\left[\exp\left(\frac{hc}{\lambda kT}\right) - 1\right]}, \\
u(\nu) d\nu &= \frac{8\pi hc}{(c/\nu)^5} \frac{(c/\nu^2) d\nu}{\left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]} \\
&= \frac{8\pi h\nu^5}{c^4} \frac{(cd\nu}{\nu^2 \left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]}, \\
\therefore u(\nu) d\nu &= \frac{8\pi h}{c^3} \frac{\nu^3 d\nu}{\left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]}.
\end{aligned}
$$

(C) Grafik yang ingin diplot adalah $u(\nu)$ terhadap ν:

$$u(\nu) = \frac{8\pi h}{c^3} \frac{\nu^3}{\left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]}.$$

Perhatikan bahwa pada rumus tersebut ada beberapa konstanta yang dapat disederhanakan terlebih dahulu.

$$\frac{8\pi h}{c^3} = \frac{8(3,14)(6,63 \times 10^{-34})}{(3 \times 10^8)^3} \approx 6,17 \times 10^{-58}.$$

Oleh karena satuan energi muncul dalam eV, maka konstanta tersebut perlu dibagi dengan $1,6 \times 10^{-19}$:

$$\frac{6,17 \times 10^{-58}}{1,6 \times 10^{-19}} \approx 3,86 \times 10^{-39}.$$

Perhatikan lagi, soal meminta untuk menggunakan frekuensi dalam satuan 10^{14} Hz. Artinya satuan di atas dapat disederhanakan lagi agar orde pangkat tidak terlalu kecil sehingga dapat dihitung di perangkat *spreadsheets* yang akan digunakan. Pada rumusan $u(\nu)$ terkandung suku ν^3, sehingga konstanta di atas dapat dikalikan dengan $(10^{14})^3$:

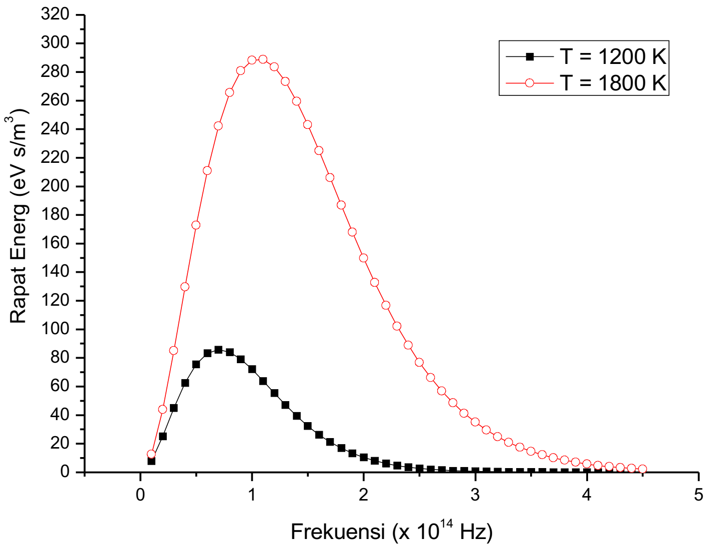
$$3,86 \times 10^{-39} \times (10^{14})^3 = 3,86 \times 10^3,$$

sedangkan suku di dalam eksponensial akan memiliki konstanta:

$$\frac{h}{k} \times 10^{14} = \frac{6,63 \times 10^{-34}}{1,38 \times 10^{-23}} \times 10^{14} \approx 4,8 \times 10^{3}.$$

Dengan demikian, rumus $u(\nu)$ jadi lebih sederhana dan dapat dihitung menggunakan program buatan sendiri atau menggunakan *spreadsheets*:

$$u(\nu) = \frac{3,86 \times 10^{3} \times \nu^{3}}{\exp\left(4,78 \times 10^{3} \times \nu/T\right) - 1}.$$



(D) Sesuai hukum Wien, panjang gelombang yang menyebabkan rapat energi maksimum akan bergeser menuju panjang gelombang yang lebih rendah jika temperaturnya naik. Atau, jika dinyatakan dalam frekuensi, maka frekuensi yang menyebabkan rapat energi maksimum akan bergeser menuju frekuensi yang lebih tinggi ketika temperaturnya naik. Hubungan ini ditunjukkan jelas pada gambar soal C.

SOLUSI Ujian Praktikum Fisika Komputasi

■ KODE SOAL: FI3102-L

(A) Fungsi yang ingin diuraikan:

$$i(v) = k\left(1 + \frac{v}{V_{DC}}\right)^{1,5}.$$

(A1) Deret pangkat yang dimaksud dalam soal adalah deret Maclaurin. Tuliskan dulu bentuk deret Taylor:

$$i(v) = i(v_0) + i'(v_0)(v - v_0) + \frac{i''(v_0)}{2!}(v - v_0)^2 + \frac{i'''(v_0)}{3!}(v - v_0)^3 + \ldots.$$

Tegangan v diasumsikan cukup kecil dan bervariasi di sekitar $v_0 = 0$, sehingga deret Taylor tereduksi menjadi deret Maclaurin. Potong suku deret sampai orde pangkat tiga saja:

$$i(v) = i(0) + i'(0)v + \frac{i''(0)}{2!}v^2 + \frac{i'''(v_0)}{3!}v^3.$$

Sesuai dengan

$$i(v) = k\left(1 + \frac{v}{V_{DC}}\right)^{3/2},$$

maka $i(0) = k$ dan

$$\begin{aligned}
i'(v) &= \frac{3k}{2V_{DC}}\left(1 + \frac{v}{V_{DC}}\right)^{1/2} \Rightarrow i'(0) = \frac{3k}{2V_{DC}},\\
i''(v) &= \frac{3k}{4V_{DC}^2}\left(1 + \frac{v}{V_{DC}}\right)^{-1/2} \Rightarrow i''(0) = \frac{3k}{4V_{DC}^2},\\
i'''(v) &= -\frac{3}{8V_{DC}^3}\left(1 + \frac{v}{V_{DC}}\right)^{-3/2} \Rightarrow i'''(0) = -\frac{3k}{8V_{DC}^3}.
\end{aligned}$$

Subsitusikan pada deret Maclaurin:

$$\begin{aligned}
i(v) &= k + \frac{3k}{2V_{DC}}v + \frac{3k}{8V_{DC}^2}v^2 - \frac{k}{16V_{DC}^3}v^3\\
&= k\left(1 + \frac{3}{2V_{DC}}v + \frac{3}{8V_{DC}^2}v^2 - \frac{1}{16V_{DC}^3}v^3\right).
\end{aligned}$$

(A2) Misalkan $k = 1$ dan $V_{DC} = 1$, maka

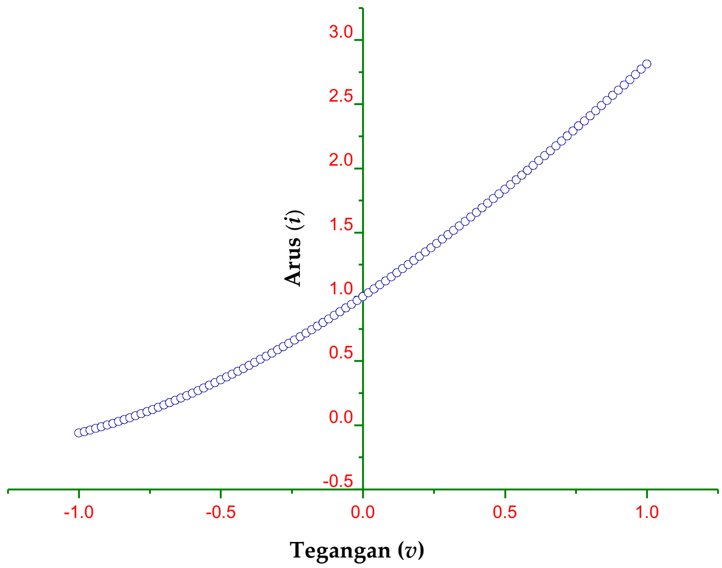
$$i(v) = 1 + \frac{3}{2}v + \frac{3}{8}v^2 - \frac{1}{16}v^3.$$

Dari soal diminta untuk dihitung nilai $i(v)$ pada selang $v = [-1, 1]$ dengan 101 titik partisi yang sama. Artinya besar setiap partisi adalah $\Delta v = 0,02$.

```
#include <stdio.h>
main()
{
    float i, v;
    FILE *f;
    if((f = fopen("kurva_iv.csv", "w")) == NULL) printf("error\n");
    v = -1; /* batas kiri v */
```

```
        do {
            i = 1 + (3./2)*v + (3./8)*v*v - (1./16)*v*v*v;
            printf("%.4f\t%.4f\n", v, i);
            fprintf(f, "%.4f,%.4f\n", v, i);
            v = v + 0.02;
        } while(v <= 1); /* batas kanan v */
        getchar();
        fclose(f);
    }
```

(A3) Kurva $i - v$:



(B) Sebuah gelombang pada suatu saat dapat dimodelkan dengan memperluas fungsi

$$f(x) = -x^2 + 4x,$$

pada $0 \leq x \leq 4$ menjadi fungsi *genap* yang periodik.

(B1) Sembarang fungsi periodik dengan periode $2L$ dapat diuraikan dalam bentuk deret Fourier, yaitu:

$$f(x) = \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos\frac{n\pi x}{L} + \sum_{n=1}^{\infty} b_n \sin\frac{n\pi x}{L},$$

dengan

$$a_0 = \frac{1}{L}\int_{-L}^{L} f(x)dx, \quad a_n = \frac{1}{L}\int_{-L}^{L} f(x)\cos\frac{n\pi x}{L}dx, \quad b_n = \frac{1}{L}\int_{-L}^{L} f(x)\sin\frac{n\pi x}{L}dx.$$

Fungsi yang ditinjau pada soal ini adalah fungsi genap sehingga $b_n = 0$, dan kalau fungsi tersebut digambar akan dapat diketahui bahwa selang $0 \leq x \leq 4$ hanya mengandung setengah periode ($L = 4$). Akibatnya,

$$a_0 = \frac{2}{L}\int_0^L f(x)dx, \quad a_n = \frac{2}{L}\int_0^L f(x)\cos\frac{n\pi x}{L}dx$$

Hitung a_0:

$$\begin{aligned} a_0 &= \frac{2}{4}\int_0^4 f(x)dx = \frac{1}{2}\left[-\frac{1}{3}x^3 + 2x^2\right]_0^4 \\ &= \frac{1}{2}\left(-\frac{64}{3} + 32\right) = \frac{16}{3}. \end{aligned}$$

Hitung a_n:

$$\begin{aligned} a_n &= \frac{2}{4}\int_0^4 (-x^2 + 4x)\cos\frac{n\pi x}{4}dx \\ &= -\frac{1}{2}\int_0^4 x^2\cos\frac{n\pi x}{4}dx + 2\int_0^4 x\cos\frac{n\pi x}{4}dx \\ &= -\frac{1}{2}\left(\left[\frac{4x^2}{n\pi}\sin\frac{n\pi x}{4}\right]_0^4 - \frac{8}{n\pi}\int_0^4 x\sin\frac{n\pi x}{4}dx\right) + 2\left(\left[\frac{4x}{n\pi}\sin\frac{n\pi x}{4}\right]_0^4 - \frac{4}{n\pi}\int_0^4 \sin\frac{n\pi x}{4}dx\right) \\ &= -\frac{1}{2}\left(0 - \frac{8}{n\pi}\int_0^4 x\sin\frac{n\pi x}{4}dx\right) + 2\left(0 + \frac{16}{(n\pi)^2}\left[\cos\frac{n\pi x}{4}\right]_0^4\right) \\ &= \frac{4}{n\pi}\left(\left[-\frac{4x}{n\pi}\cos\frac{n\pi x}{4}\right]_0^4 + \frac{4}{n\pi}\int_0^4 \cos\frac{n\pi x}{4}dx\right) + \frac{32}{(n\pi)^2}(\cos n\pi - 1) \\ &= -\frac{16}{(n\pi)^2}(4\cos n\pi - 0) + 0 + \frac{32}{(n\pi)^2}\cos n\pi - \frac{32}{(n\pi)^2} \\ &= \frac{(32 - 64)\cos n\pi}{(n\pi)^2} - \frac{32}{(n\pi)^2}. \\ &= -\frac{32}{(n\pi)^2}(\cos n\pi + 1). \end{aligned}$$

Substitusikan a_0 dan a_n pada $f(x)$ sesuai deret Fourier:

$$\begin{aligned} f(x) &= \frac{a_0}{2} + \sum_{n=1}^{\infty} a_n \cos\frac{n\pi x}{L} \\ &= \frac{8}{3} - 32\sum_{n=1}^{\infty}\frac{\cos n\pi + 1}{(n\pi)^2}\cos\frac{n\pi x}{4}. \quad \textbf{(terbukti)} \end{aligned}$$

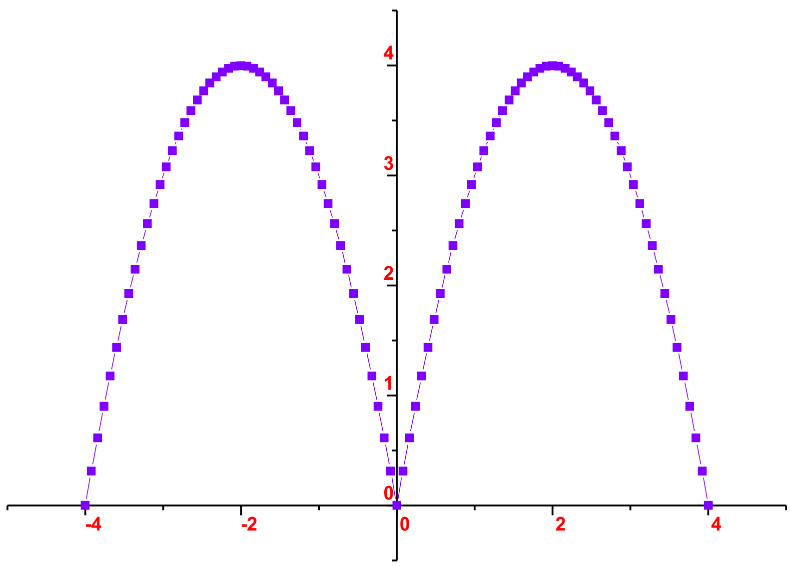
(B2) Lebar partisi yang digunakan untuk 101 titik pada selang $-4 \leq x \leq 4$ adalah $0,08$. Toleransi kesalahan yang diperkenankan adalah 10^{-4}. Programnya sebagai berikut:

```
/* fourier.c */
#include <stdio.h>
#include <math.h>
main()
{
    float x, fx, fx0;
    int n;
    FILE *f;
    if((f = fopen("fourier.csv", "w")) == NULL) printf("error\n");
```

```
    x = -4;
    do {
        n = 1;
        fx = 8/3.;
        fx0 = -pow(fabs(x),2) + 4*fabs(x); /* fungsi genap acuan*/
        do {
            fx = fx - 32.*(cos(n*M_PI) + 1)*cos(n*M_PI*x/4)/pow(n*M_PI,2);
            n = n + 1;
        } while(fabs(fx - fx0) > 1e-4);
        printf("%.2f \t %.4f\n", x, fx);
        fprintf(f,"%.2f,%.4f\n", x, fx);
        x = x + 0.08;
    } while(x <= 4);
    printf("Jumlah iterasi yang dibutuhkan: n = %d", n - 1);
    getchar();
    fclose(f);
}
```

Berdasarkan program tersebut, jumlah iterasi yang dibutuhkan adalah 29166.

(B3) Kurva $f(x)$ terhadap x:



SOLUSI Ujian Praktikum Fisika Komputasi

■ KODE SOAL: FI3102-M

(A) Data posisi setiap saat dari sebuah partikel pada sumbu-x positif adalah:

t (detik)	0	2	3	10	14	15
x (meter)	3,01	6,99	12,04	103,1	199,2	228,13

(A1) Untuk melengkapi data posisi pada selang waktu $0 \leq t \leq 15$, dapat digunakan interpolasi polinom Lagrange. Rumusannya adalah:

$$p_n(x) = \sum_{i=0}^{n} a_i L_i(x) = a_0 L_0(x) + a_1 L_1(x) + \ldots + a_n L_n(x),$$

dengan $a_i = y_i$, $i = 0, 1, 2, \ldots, n$, dan

$$L_i(x) = \prod_{j=0, j\neq i}^{n} \frac{x - x_j}{x_i - x_j}.$$

Untuk melengkapi data kecepatan dan percepatan pada selang waktu $1 \leq t \leq 14$, dapat digunakan metode turunan numerik beda sentral. Kecepatan merupakan turunan pertama dari fungsi posisi terhadap waktu, dan percepatan merupakan turunan kedua dari fungsi posisi terhadap waktu. Dengan metode beda sentral yang ordenya paling rendah, rumus turunan pertama adalah:

$$f_0' = \frac{f_1 - f_{-1}}{2h},$$

sedangkan rumus turunan kedua adalah:

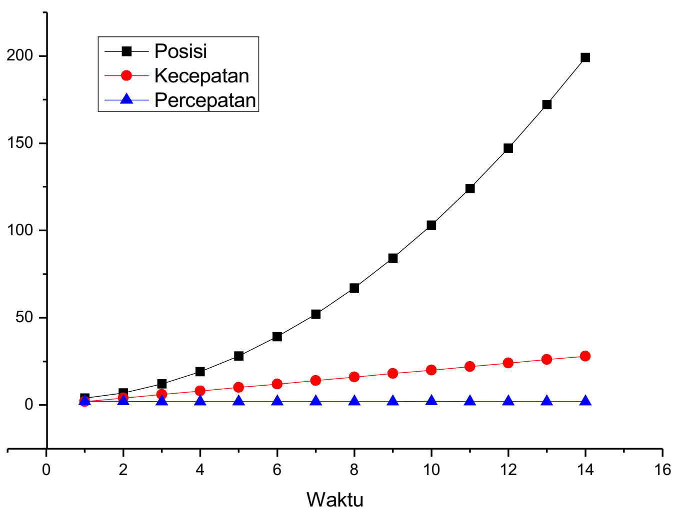
$$f_0'' = \frac{f_1 - 2f_{-1} + f_{-1}}{h^2}.$$

Simbol-simbol f yang digunakan pada rumusan tersebut masing-masing merupakan kependekan dari $f(x)$, yaitu f_0 menyatakan $f(x_i)$, kemudian $f_1 = f(x_{i+1})$, dan $f_{-1} = f(x_{i-1})$. Program untuk melengkapi data posisi, kecepatan, dan percepatan adalah:

```
/* lagrange.c */
#include <stdio.h>
#include <conio.h>
main()
{
     float  posisi, p_x[20], t[20], x[20], p_i, v, a, h;
     int i, j, n, waktu;
     FILE *f;
     if((f = fopen("x-v-a.csv", "w")) == NULL) printf("error\n");
     printf("Banyak titik data yang akan dimasukkan: "); scanf("%d", &n);
     for(i = 0; i <= (n-1); i++) { /* masukkan data */
          printf("t[%d] = ", i); scanf("%f", &t[i]);
          printf("x[%d] = ", i); scanf("%f", &x[i]);
     }
```

```
    printf("\nWaktu dan Posisi: \n");
    for(waktu = 0; waktu <= 15; waktu++) {
        posisi = 0.; /* nilai awal polinom lagrange */
        for(i = 0; i <= (n-1); i++) {
            p_i = 1.;
            for(j = 0; j <= (n-1); j++) {
                if(i != j) p_i = p_i*(waktu - t[j])/(t[i] - t[j]);
            }
            posisi = posisi + x[i]*p_i;
        }
        printf("%d\t%.2f\n", waktu, posisi);
        p_x[waktu] = posisi;
    }
    h = 1.; /* partisi */
    printf("\nWaktu, Posisi, Kecepatan, Percepatan: \n");
    for(waktu = 1; waktu <= 14; waktu++) {
        v = (p_x[waktu+1] - p_x[waktu-1]) / (2*h);
        a = (p_x[waktu+1] - 2*p_x[waktu] + p_x[waktu-1]) / (h*h);
        printf("%d\t%.2f\t%.2f\t%.2f\n", waktu, p_x[waktu], v, a);
        fprintf(f, "%d,%.2f,%.2f,%.2f\n", waktu, p_x[waktu], v, a);
    }
    getch();
    fclose(f);
}
```

(A2) Kurva posisi, kecepatan, dan percepatan partikel:



(A3) Gerak partikel merupakan gerak lurus dipercepat beraturan karena percepatannya cenderung tetap, yaitu 2 m/s^2.

(B) Seorang penerjun payung jatuh dari sebuah pesawat. Kecepatannya sebagai fungsi dari waktu adalah

$$v(t) = \frac{gm}{b}\left[1 - e^{(-b/m)t}\right],$$

dengan $g = 9,8$ m/s dan $b = 12,5$ kg/s.

(B1) Kecepatan merupakan turunan pertama dari fungsi jarak, maka jarak penerjun dari titik terjun ($t = 0$) adalah:

$$y = \int_0^t v(t)dt = \int_0^t \frac{gm}{b}\left[1 - e^{(-b/m)t}\right].$$

(B2) Misalkan daerah integrasi dari c sampai d dibagi menjadi n partisi yang sama berbentuk trapesium. Tiap partisi memiliki lebar $h = (d - c)/n$, sehingga bentuk integral dapat didekati oleh:

$$\int_c^d f(x)dx \approx \frac{h}{2}\left(f(x_0) + 2\sum_{i=1}^{n-1} f(x_i) + f(x_n)\right),$$

dengan $x_0 = c$, $x_n = d$, dan $x_i = x_0 + i \times h$.

Program untuk menghitung jarak tempuh penerjun setelah $t = 10$ detik:

```
#include <stdio.h>
#include <math.h>
main()
{
     float h, x, sigma, I, c, d, b, g, m;
     int i, n;
     g = 9.8; b = 12.5;
     c = 0.; d = 10.; /* batas integrasi t = 0 sampai t = 10 */
     printf("Masukkan massa penerjun: "); scanf("%f", &m);
     printf("Masukkan jumlah partisi: "); scanf("%d", &n);
     h = (d - c) / n;
     x = c;
     I = (g*m/b)*(1-exp(-(b/m)*c))+(g*m/b)*(1-exp(-(b/m)*d)); /* f(c) + f(d) */
     sigma = 0.;
     for(i = 1; i <= (n-1); i++) {
          x = x + h; /* atau x = c + i*h */
          sigma = sigma + 2*(g*m/b)*(1-exp(-(b/m)*x));
     }
     I = (I + sigma) * h/2;
     printf("\nJarak penerjun (hasil integrasi): %.4f", I);
     getchar(); getchar();
}
```

Perhatikan bahwa massa m dan jumlah partisi n menjadi parameter masukan yang bebas diambil sesuka hati dalam program.

Sebagai contoh, ambil $m = 68.1$ dan $n = 128$, maka program yang benar harus menghasilkan perhitungan jarak $d = 289,4310$ meter.

SOLUSI Ujian Praktikum Fisika Komputasi

■ KODE SOAL: FI3102-N

Perhatikan bahwa massa beban tidak akan berpengaruh pada gerak sistem bandul karena kasus ini bukan bandul fisis, melainkan bandul matematis biasa. Sifat ini akan dibuktikan pada jawaban A.

(A) Untuk menuliskan persamaan gerak bandul, misalkan beban akan menempuh jarak busur s yang berkaitan dengan perpindahan sudut θ. Uraikan gaya pemulih pada bandul dan gunakan hukum Newton, sehingga:

$$m\frac{d^2s}{dt^2} = -mg\sin\theta,$$

lalu sederhanakan melalui hubungan $s = L\theta$, dengan L merupakan panjang bandul yang konstan:

$$\begin{aligned} \frac{d^2s}{dt^2} &= -g\sin\theta \\ \Leftrightarrow \frac{d^2(L\theta)}{dt^2} &= -g\sin\theta \\ \therefore \frac{d^2\theta}{dt^2} &= -\frac{g}{L}\sin\theta. \end{aligned}$$

Terbukti bahwa massa beban yang 0,5 kg (atau berapapun) tidak akan berpengaruh pada gerak sistem bandul.

(B) Waktu yang dibutuhkan bandul untuk mencapai kecepatan maksimumnya setara dengan waktu yang dibutuhkan bandul untuk mencapai posisi terendah $\theta = 0$ karena pada posisi tersebut kecepatan bandul akan maksimum. Jadi di sini diperlukan pemecahan persamaan diferensial hasil dari soal A melalui dua langkah, yang dapat dilakukan simultan dengan metode Euler. Langkah pertama adalah pemecahan persamaan kecepatan sudut ω:

$$\frac{d^2\theta}{dt^2} = \frac{d\omega}{dt} = -\frac{g}{L}\sin\theta.$$

Langkah kedua adalah pemecahan persamaan posisi sudut θ:

$$\frac{d\theta}{dt} = \omega.$$

Dari kedua persamaan tersebut, dapat dibuat hubungan Euler, yaitu:

$$\Delta\omega = -\frac{g}{L}\sin\theta\Delta t \Rightarrow \omega_{i+1} = \omega_i - \left(\frac{g}{L}\sin\theta_i\right)\Delta t,$$

dan

$$\Delta\theta = \omega\Delta t \Rightarrow \theta_{i+1} = \theta_i + \omega_i\Delta t.$$

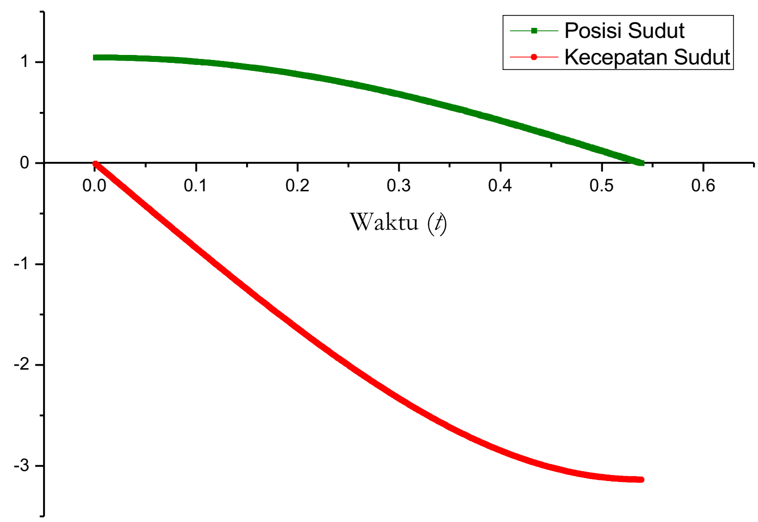
Perhatikan bahwa dengan rumusan tersebut, kecepatan sudut ω akan bernilai negatif. Tapi itu tidak masalah karena dapat diambil besarnya saja. Kecepatan sudut maksimum di sini adalah 'besar' kecepatan yang maksimum, yaitu ketika sudut $\theta = 0$. Programnya adalah:

```
/* euler.c */
#include <stdio.h>
#include <math.h>
```

```
main()
{
    float g, L, omega_i, omega_t, t_i, t_t, sudut;
    FILE *f;
    if ((f = fopen("hasileuler.csv", "w")) == NULL) printf("error!");
    g = 9.8; L = 1.;
    omega_i = 0.; t_i = 0.;
    sudut = M_PI / 3.; /* sudut awal 60 derajat dalam radian */
    do {
       t_t = t_i + 0.001; /* jadi delta_t digunakan 0.001 */
       omega_t = omega_i - (g / L) * sin(sudut) * (t_t - t_i);
       sudut = sudut + omega_i * (t_t - t_i);
       t_i = t_t;
       omega_i = omega_t;
       printf("%.3f\t%.4f\t%.4f\n", t_t, sudut, omega_t);
       fprintf(f, "%.3f,%.4f,%.4f\n", t_t, sudut, omega_t);
    } while(sudut >= 1e-6);
    getchar();
    fclose(f);
}
```

Dari keluaran program, waktu yang dibutuhkan adalah $0,54$ detik.

(C) Kurva simpangan (posisi) sudut dan kecepatan sudut terhadap waktu sesuai soal B:

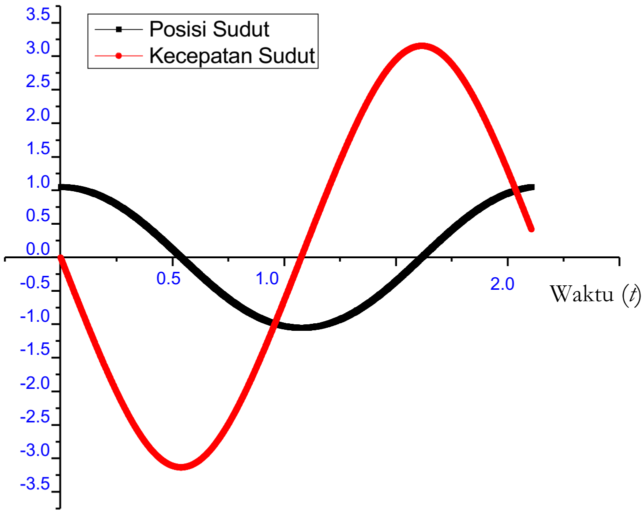


(D) Waktu dapat dihitung dengan asumsi awal bandul dapat kembali ke posisi semula. Dengan demikian, perhitungan dipecah menjadi dua bagian, yaitu untuk $\theta = 60^o$ hingga $\theta = -60^o$ dan sebaliknya dari $\theta = -60^o$ hingga $\theta = 60^o$. Programnya adalah:

```c
/* eulermod.c */
#include <stdio.h>
#include <math.h>
main()
{
    float g, L, omega_i, omega_t, t_i, t_t, sudut;
    FILE *f;
    if ((f = fopen("hasileulermod.csv", "w")) == NULL) printf("error!");
    g = 9.8; L = 1.;
    omega_i = 0.; t_i = 0.;
    sudut = M_PI / 3.; /* sudut awal 60 derajat dalam radian */
    do {
       t_t = t_i + 0.001; /* jadi delta_t digunakan 0.001 */
       omega_t = omega_i - (g / L) * sin(sudut) * (t_t - t_i);
       sudut = sudut + omega_i * (t_t - t_i);
       t_i = t_t;
       omega_i = omega_t;
       printf("%.3f\t%.4f\t%.4f\n", t_t, sudut, omega_t);
       fprintf(f, "%.3f,%.4f,%.4f\n", t_t, sudut, omega_t);
    } while(sudut >= -M_PI / 3.);
    do {
       t_t = t_i + 0.001; /* jadi delta_t digunakan 0.001 */
       omega_t = omega_i - (g / L) * sin(sudut) * (t_t - t_i);
       sudut = sudut + omega_i * (t_t - t_i);
       t_i = t_t;
       omega_i = omega_t;
       printf("%.3f\t%.4f\t%.4f\n", t_t, sudut, omega_t);
       fprintf(f, "%.3f,%.4f,%.4f\n", t_t, sudut, omega_t);
    } while(sudut <= M_PI / 3.);
    getchar();
    fclose(f);
}
```

Dari hasil perhitungan program, waktu yang diperlukan adalah $2,11$ detik. Ternyata waktunya memang empat kali kasus soal sebelumnya. Alasannya sederhana saja, yaitu bandul memang diasumsikan untuk bisa kembali ke posisinya semula atau satu periode, sementara soal sebelumnya adalah kasus gerak seperempat periode.

(E) Kurva yang diinginkan sesuai soal D:



SOLUSI Ujian Praktikum Fisika Komputasi

Waktu: 2 jam, **Bahasa Pemrograman**: C/C++, **Sifat Ujian**: *Open Book*

■ KODE SOAL: FI3102-O

(A) Metode dekomposisi LU bertujuan untuk menguraikan matriks persegi A berukuran $n \times n$ menjadi matriks segitiga bawah L yang seluruh elemen diagonal utamanya bernilai 1 dan matriks segitiga atas U. Misalkan elemen dari matriks A adalah a, elemen matriks L adalah l, dan elemen matriks U adalah u. Untuk menguraikan matriks A menjadi matriks L dan matriks U yang memenuhi A = LU, rumusan yang digunakan adalah (algoritma Crout):

$$
\begin{aligned}
j &= 1, 2, \dots, N \\
u_{1j} &= a_{1j} \\
u_{ij} &= a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj}; \quad i = 2, \dots, j \\
l_{ij} &= \frac{1}{u_{jj}} \left(a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj} \right); \quad i = j+1, \dots, N.
\end{aligned}
$$

Jika ada suatu sistem persamaan linear AX = B, dengan X matriks yang elemen-elemennya ingin ditentukan, maka matriks LU dapat dimanfaatkan untuk memecahkan sistem persamaan linear tersebut.

Tuliskan: AX = LU X = B, sehingga dapat dilakukan substitusi maju: LY = B dan substitusi mundur UX = Y. Dengan demikian, X dapat dicari.

Program untuk mencari arus I_1, I_2, I_3 berdasarkan persamaan

$$
\begin{bmatrix} 2 & 3 & -2 \\ 1 & -6 & 4 \\ 4 & -1 & 6 \end{bmatrix} \begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}
$$

adalah:

```
#include <stdio.h>
main()
{
    int i,j,k;
    float A[3][3], L[3][3], U[3][3], X[3], Y[3], B[3];
    /* nol-kan dulu seluruh matriks */
    for(i = 1; i <= 3; i++) {
        X[i] = 0.;
        Y[i] = 0.;
        B[i] = 0.;
        for(j = 1; j <= 3; j++) {
            A[i][j] = 0.;
            U[i][j] = 0.;
            L[i][j] = 0.;
```

```
    }
}
A[1][1] = 2; A[1][2] = 3  ; A[1][3] = -2; B[1] = 2;
A[2][1] = 1; A[2][2] = -6 ; A[2][3] = 4 ; B[2] = 4;
A[3][1] = 4; A[3][2] = -1 ; A[3][3] = 6 ; B[3] = 6;
/* Mulai Dekomposisi LU: */
for(j = 1; j <= 3; j++) {
    U[1][j] = A[1][j];
    for(i = 2; i <= j; i++) {
         U[i][j] = A[i][j];
         for(k = 1; k <= i-1; k++) {
              U[i][j] = U[i][j] - (L[i][k] * U[k][j]);
         }
    }
    for(i = j+1; i <= 3; i++) {
        L[i][j] = A[i][j] / U[j][j];
        for(k = 1; k <= j-1; k++) {
            L[i][j] = L[i][j] - L[i][k] * U[k][j] / U[j][j];
        }
    }
    L[j][j]=1;
}
/* Tampilkan hasil */
printf("Matriks L:\n");
for(i = 1; i <= 3; i++) {
    for(j = 1; j <= 3; j++) printf("% 6.2f ", L[i][j]);
    printf("\n");
}
printf("Matriks U:\n");
for(i = 1; i <= 3; i++) {
    for(j = 1; j <= 3; j++) printf("% 6.2f ", U[i][j]);
    printf("\n");
}
printf("Matriks Y:\n");
for(i = 1; i <= 3; i++) { /* forward substitution */
    Y[i] = B[i] / L[i][i];
    for(j = 1; j <= i-1; j++) Y[i] = Y[i] - L[i][j] * Y[j] / L[i][i];
    printf("%6.2f\n", Y[i]);
}
printf("Matriks X (Nilai Arus):\n");
for(i = 3; i >= 1; i--) { /* backward substitution */
    X[i] = Y[i] / U[i][i];
    for(j = i+1; j <= 3; j++) X[i] = X[i] - U[i][j] * X[j] / U[i][i];
```

```
        printf("I%i = %6.2f\n", i, X[i]);
    }
    getchar();
}
```

Hasilnya diperoleh $I_1 = 1,60$; $I_2 = -0,50$; dan $I_3 = -0,15$.

(B1) Syarat matematis agar rapat energi maksimum adalah:

$$\frac{d}{d\nu_m}\left[\frac{\nu_m^3}{\exp\left(\frac{h\nu_m}{kT}\right)-1}\right] = 0.$$

Selesaikan:

$$\begin{aligned}
\frac{d}{d\nu}\left[\frac{\nu_m^3}{\exp\left(\frac{h\nu_m}{kT}\right)-1}\right] &= \frac{d}{d\nu}\left(\nu_m^3\left[\exp\left(\frac{h\nu_m}{kT}\right)-1\right]^{-1}\right) = 0\\
\Leftrightarrow 3\nu_m^2\left[\exp\left(\frac{h\nu_m}{kT}\right)-1\right]^{-1} &- \frac{h\nu_m^3}{kT}\exp\left(\frac{h\nu_m}{kT}\right)\left[\exp\left(\frac{h\nu_m}{kT}\right)-1\right]^{-2} = 0\\
\Leftrightarrow \frac{3\nu_m^2}{\exp\left(\frac{h\nu_m}{kT}\right)-1} &= \frac{h\nu_m^3}{kT}\frac{\exp\left(\frac{h\nu_m}{kT}\right)}{\left[\exp\left(\frac{h\nu_m}{kT}\right)-1\right]^2}\\
\Leftrightarrow 3\left[\exp\left(\frac{h\nu_m}{kT}\right)-1\right] &= \frac{h\nu_m}{kT}\exp\left(\frac{h\nu_m}{kT}\right).
\end{aligned}$$

Misalkan $x = (h\nu_m)/(kT)$, sehingga

$$\begin{aligned}
3[\exp(x)-1] &= x\exp(x)\\
\therefore 3[1-\exp(-x)] &= x.
\end{aligned}$$

Metode yang tepat untuk menyelesaikan persamaan nonlinear tersebut adalah dengan iterasi. Untuk keperluan itu, tulislah:

$$x_{n+1} = 3[1-\exp(-x_n)].$$

Cari nilai x dengan program berikut:

```
#include <stdio.h>
#include <math.h>
main()
{
     float x0, x, y, dx;
     x0 = 5.; /* tebakan awal */
     do {
          x = 3 * (1 - exp(-x0));
          dx = fabs(x - x0); /* selisih untuk konvergensi */
          x0 = x; /* metode iterasi */
     } while (dx >= 10e-6); /* konvergensi sampai 0,000001 */
     printf("x = %.4f", x);
     getchar();
}
```

Dari hasil program tersebut, diperoleh $x = 2,8214$. Substitusikan nilai ini pada $x = \frac{h\nu_m}{kT}$:

$$\frac{6,63 \times 10^{-34} \times \nu_m}{1,38 \times 10^{-23} \times T} = 2,8214$$
$$\therefore \frac{\nu_m}{T} = 2,8214 \times \frac{1,38 \times 10^{-23}}{6,63 \times 10^{-34}} \approx 5,873 \times 10^{10}\ \text{Hz/K.}$$

(B2) Grafik yang ingin diplot adalah $u(\nu)$ terhadap ν:

$$u(\nu) = \frac{8\pi h}{c^3} \frac{\nu^3}{\left[\exp\left(\frac{h\nu}{kT}\right) - 1\right]}.$$

Perhatikan bahwa pada rumus tersebut ada beberapa konstanta yang dapat disederhanakan terlebih dahulu.

$$\frac{8\pi h}{c^3} = \frac{8(3,14)(6,63 \times 10^{-34})}{(3 \times 10^8)^3} \approx 6,17 \times 10^{-58}.$$

Oleh karena satuan energi muncul dalam eV, maka konstanta tersebut perlu dibagi dengan $1,6 \times 10^{-19}$:

$$\frac{6,17 \times 10^{-58}}{1,6 \times 10^{-19}} \approx 3,86 \times 10^{-39}.$$

Perhatikan lagi, soal meminta untuk menggunakan frekuensi dalam satuan 10^{14} Hz. Artinya satuan di atas dapat disederhanakan lagi agar orde pangkat tidak terlalu kecil sehingga dapat dihitung di perangkat *spreadsheets* yang akan digunakan. Pada rumusan $u(\nu)$ terkandung suku ν^3, sehingga konstanta di atas dapat dikalikan dengan $(10^{14})^3$:

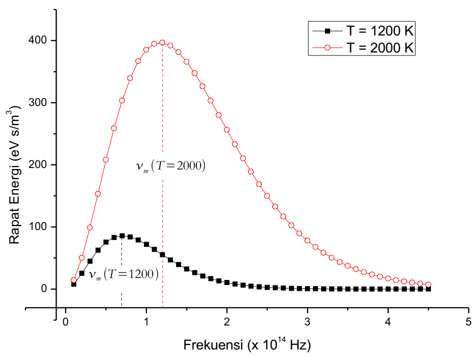
$$3,86 \times 10^{-39} \times (10^{14})^3 = 3,86 \times 10^3,$$

sedangkan suku di dalam eksponensial akan memiliki konstanta:

$$\frac{h}{k} \times 10^{14} = \frac{6,63 \times 10^{-34}}{1,38 \times 10^{-23}} \times 10^{14} \approx 4,8 \times 10^3.$$

Dengan demikian, rumus $u(\nu)$ jadi lebih sederhana dan dapat dihitung menggunakan *spreadsheets*:

$$u(\nu) = \frac{3,86 \times 10^3 \times \nu^3}{\exp\left(4,78 \times 10^3 \times \nu/T\right) - 1}.$$



(B3) Dari hubungan $\nu_m/T =$ konstan, atau $\nu_m =$ konstanta $\times T$, dapat disimpulkan bahwa frekuensi yang menyebabkan kerapatan energi maksimum akan bergeser ke kanan (membesar) seiring dengan meningkatnya suhu T.