

Kelompok 12:

Daffa Hardhan (2306161763)

Muhammad Bryan Farras (2306230975)

Muhammad Riyan Satrio Wibowo (2306229323)

A. METODE PENYELESAIAN

Untuk menyelesaikan tugas ini, kami menggunakan metode pencocokan kurva (curve fitting) dengan pendekatan persamaan polinomial. Langkah-langkah penyelesaian yang dilakukan meliputi:

1. Analisis Dataset
 - Dataset berisi data tahun, persentase pengguna Internet, dan jumlah populasi Indonesia dari tahun 1960 hingga 2023.
 - Ditemukan data yang hilang pada tahun 2005, 2006, 2015, dan 2016 untuk kolom populasi dan persentase pengguna Internet.
2. Pemilihan Metode Curve Fitting
 - Digunakan metode polinomial regresi karena data populasi dan pengguna Internet menunjukkan pola pertumbuhan yang dapat dimodelkan dengan polinomial.
 - Derajat polinomial dipilih berdasarkan kecocokan terhadap data (uji visualisasi atau nilai error terkecil).
3. Implementasi Program dengan Bahasa C
 - Data yang tersedia dimasukkan ke dalam program.
 - Program menghitung koefisien polinomial menggunakan metode Least Squares (Kuadrat Terkecil).
 - Setelah didapatkan persamaan polinomial, program digunakan untuk: menghitung estimasi data yang hilang pada tahun 2005, 2006, 2015, dan 2016; Menyusun persamaan polinomial final; Melakukan prediksi untuk tahun 2030 (populasi) dan tahun 2035 (pengguna Internet).
4. Validasi Hasil
 - Hasil estimasi dibandingkan dengan tren data yang tersedia untuk memastikan bahwa polinomial yang digunakan memberikan hasil yang logis.
 - Program diuji untuk memproyeksikan tahun-tahun lain guna memastikan kestabilan prediksi.

A. HASIL ESTIMASI DATA YANG HILANG

1) Populasi

2005: 230971878

2006: 234017107

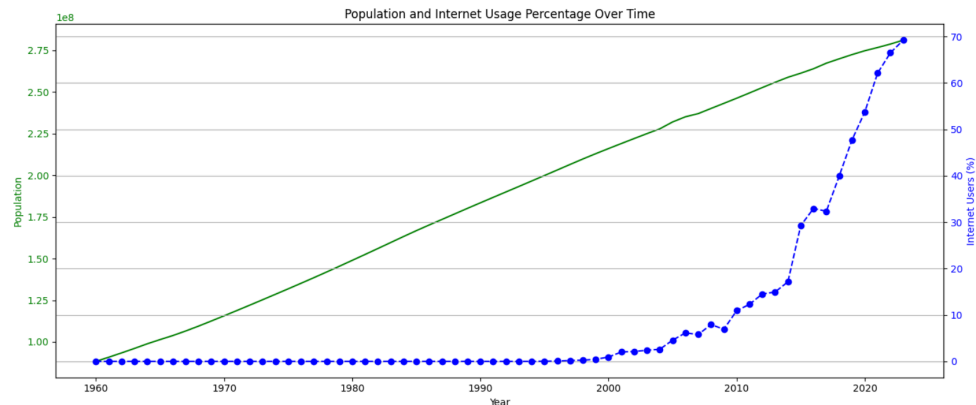
2015: 261700485

2016: 264523571

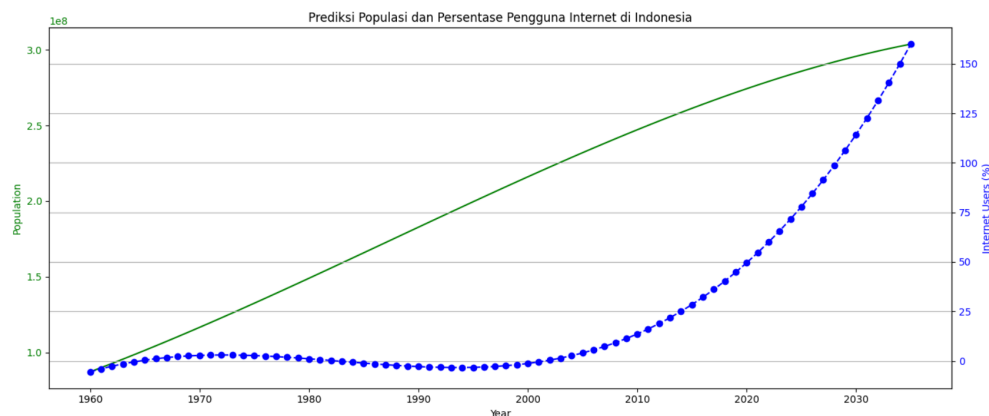
2) Persentase

2005: 3.662283
 2006: 4.724277
 2015: 22.207400
 2016: 27.271600

B. PERSAMAAN POLINOMIAL



Gambar 1. Grafik Populasi dan Persentase Pengguna Internet



Gambar 2. Grafik Populasi dan Persentase Pengguna Internet Berdasarkan Persamaan yang didapatkan

1) Persentase Pengguna Internet di Indonesia

$$y = 0.001294x^3 + -0.088823x^2 + 1.588394x + -5.485204$$

2) Pertumbuhan Populasi Indonesia

$$y = -291.948831x^3 + 24015.143096x^2 + 2729202.035580x + 87129932.412743$$

C. PENJELASAN TEKNIS

```

DataPoint* read_csv(const char* filename, int* count) {
    FILE* file = fopen(filename, "r");
    if (!file) {
        perror("Error opening file");
        exit(1);
    }
}
  
```

```

char line[1024];
fgets(line, sizeof(line), file);

DataPoint* data = malloc(sizeof(DataPoint) * 100);
int capacity = 100;
int n = 0;

while (fgets(line, sizeof(line), file)) {
    int year;
    double percentage;
    long long population;

    if (sscanf(line, "%d,%lf,%lld", &year, &percentage, &population) == 3) {
        if (n >= capacity) {
            capacity *= 2;
            data = realloc(data, sizeof(DataPoint) * capacity);
        }
        data[n].year = year;
        data[n].percentage = percentage;
        data[n].population = population;
        data[n].is_interpolated = 0;
        n++;
    }
}

fclose(file);
*count = n;
return data;
}

```

Function tersebut berfungsi untuk membaca file Data Tugas Pemrograman A.csv dan memasukkannya ke dalam array *struct* DataPoint. Function tersebut menggunakan *file handling* untuk membuka file .csv, kemudian memasukkan masing-masing datanya ke dalam array menggunakan while loop.

```

int compare_years(const void* a, const void* b) {
    DataPoint* da = (DataPoint*)a;
    DataPoint* db = (DataPoint*)b;
    return da->year - db->year;
}

```

Function tersebut berfungsi untuk mengurutkan data pada array berdasarkan *field* tahun pada data tersebut.

```

CubicCoefficients fit_cubic(int* years, double* y_values, int count) {
    double sum_x = 0, sum_x2 = 0, sum_x3 = 0, sum_x4 = 0, sum_x5 = 0, sum_x6 = 0;
    double sum_y = 0, sum_xy = 0, sum_x2y = 0, sum_x3y = 0;
    int i;
    for (i = 0; i < count; i++) {
        int x = years[i] - 1960;
        double y = y_values[i];

        double x2 = x * x;

```

```

double x3 = x2 * x;
double x4 = x3 * x;
double x5 = x4 * x;
double x6 = x5 * x;

sum_x += x;
sum_x2 += x2;
sum_x3 += x3;
sum_x4 += x4;
sum_x5 += x5;
sum_x6 += x6;

sum_y += y;
sum_xy += x * y;
sum_x2y += x2 * y;
sum_x3y += x3 * y;
}

double matrix[4][5] = {
    {sum_x3, sum_x2, sum_x, count, sum_y},
    {sum_x4, sum_x3, sum_x2, sum_x, sum_xy},
    {sum_x5, sum_x4, sum_x3, sum_x2, sum_x2y},
    {sum_x6, sum_x5, sum_x4, sum_x3, sum_x3y}
};

for (i = 0; i < 4; i++) {
    int k;
    int max_row = i;
    for (k = i + 1; k < 4; k++) {
        if (fabs(matrix[k][i]) > fabs(matrix[max_row][i])) {
            max_row = k;
        }
    }

    for (k = i; k < 5; k++) {
        double temp = matrix[i][k];
        matrix[i][k] = matrix[max_row][k];
        matrix[max_row][k] = temp;
    }
    int j;
    for (k = i + 1; k < 4; k++) {
        double factor = matrix[k][i] / matrix[i][i];
        for (j = i; j < 5; j++) {
            matrix[k][j] -= factor * matrix[i][j];
        }
    }
}

CubicCoefficients coeffs;
coeffs.d = matrix[3][4] / matrix[3][3];

```

```

    coeffs.c = (matrix[2][4] - matrix[2][3] * coeffs.d) / matrix[2][2];
    coeffs.b = (matrix[1][4] - matrix[1][3] * coeffs.d - matrix[1][2] * coeffs.c) /
matrix[1][1];
    coeffs.a = (matrix[0][4] - matrix[0][3] * coeffs.d - matrix[0][2] * coeffs.c -
matrix[0][1] * coeffs.b) / matrix[0][0];

    return coeffs;
}

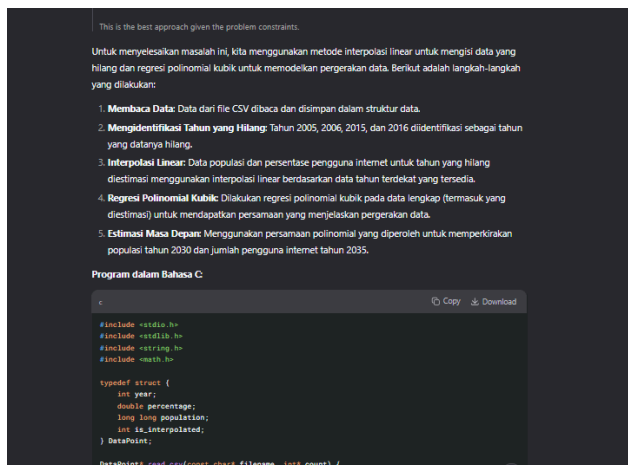
```

Function ini berfungsi untuk menghasilkan persamaan polinomial pangkat 3 untuk dataset.

D. DOKUMENTASI PENGGUNAAN AI

Kami menggunakan Deepseek dan ChatGPT untuk melakukan beberapa iterasi perhitungan untuk membandingkan akurasi.

Salah satu hasil *prompting* yang telah kami implementasikan:



Deepseek

| | | |
|------|----------|-----------|
| 2005 | 3.662283 | 230971878 |
| 2006 | 4.724277 | 234017107 |
| 2007 | 5.78627 | 237062337 |
| 2008 | 7.91748 | 240157903 |
| 2009 | 6.92 | 243220028 |
| 2010 | 10.92 | 246305322 |
| 2011 | 12.28 | 249470032 |
| 2012 | 14.52 | 252698525 |
| 2013 | 14.94 | 255852467 |
| 2014 | 17.1432 | 258877399 |
| 2015 | 22.2074 | 261700485 |
| 2016 | 27.2716 | 264523571 |

ChatGPT

| | | |
|------|-------|-------------|
| 2005 | 4.56 | 232050443.8 |
| 2006 | 6.13 | 235149865.4 |
| 2007 | 5.79 | 237062337 |
| 2008 | 7.92 | 240157903 |
| 2009 | 6.92 | 243220028 |
| 2010 | 10.92 | 246305322 |
| 2011 | 12.28 | 249470032 |
| 2012 | 14.52 | 252698525 |
| 2013 | 14.94 | 255852467 |
| 2014 | 17.14 | 258877399 |
| 2015 | 29.24 | 261331223 |
| 2016 | 32.94 | 264017040.4 |

E. REPOSITORI GITHUB

Link GitHub: https://github.com/DHard4114/Kelompok12_TugasPemrogramanA/