**LAPORAN PRAKTIKUM**

**STRUKTUR DATA**

**MODUL 2**

**PENGENALAN BAHASA C++**



**Disusun oleh:**

**RASYA SYAHRI RAMADHAN**

**103112430189**

**S1IF-12-02**

**PROGRAM STUDI S1 INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

**TUGAS**

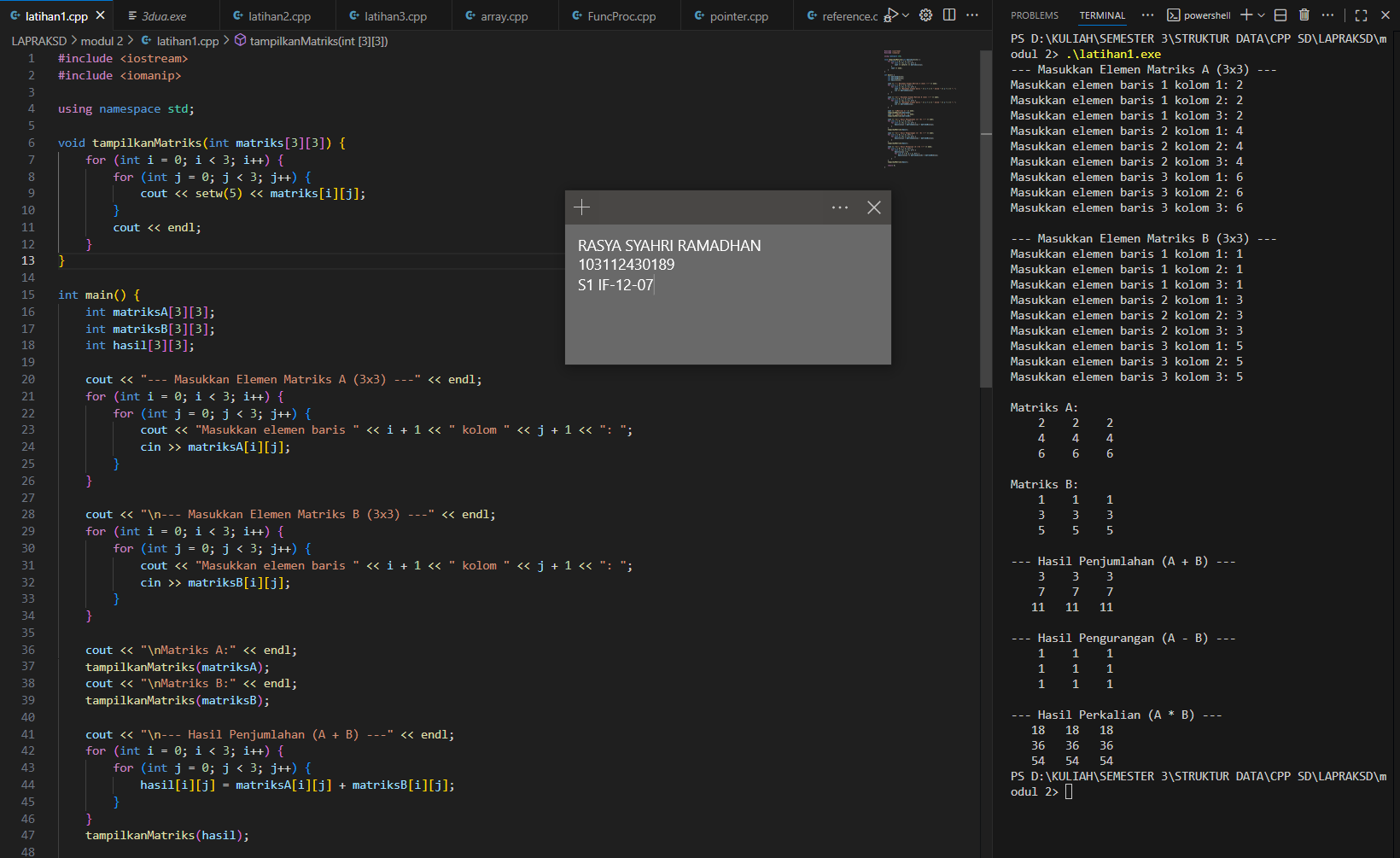
1. **Tugas 1**

Buatlah program yang dapat melakukan operasi penjumlahan, pengurangan, dan perkalian matriks 3x3

**Source code**

|  |
| --- |
| #include <iostream>  #include <iomanip>  using namespace std;  void tampilkanMatriks(int matriks[3][3]) {      for (int i = 0; i < 3; i++) {          for (int j = 0; j < 3; j++) {              cout << setw(5) << matriks[i][j];          }          cout << endl;      }  }  int main() {      int matriksA[3][3];      int matriksB[3][3];      int hasil[3][3];      cout << "--- Masukkan Elemen Matriks A (3x3) ---" << endl;      for (int i = 0; i < 3; i++) {          for (int j = 0; j < 3; j++) {              cout << "Masukkan elemen baris " << i + 1 << " kolom " << j + 1 << ": ";              cin >> matriksA[i][j];          }      }      cout << "\n--- Masukkan Elemen Matriks B (3x3) ---" << endl;      for (int i = 0; i < 3; i++) {          for (int j = 0; j < 3; j++) {              cout << "Masukkan elemen baris " << i + 1 << " kolom " << j + 1 << ": ";              cin >> matriksB[i][j];          }      }      cout << "\nMatriks A:" << endl;      tampilkanMatriks(matriksA);      cout << "\nMatriks B:" << endl;      tampilkanMatriks(matriksB);      cout << "\n--- Hasil Penjumlahan (A + B) ---" << endl;      for (int i = 0; i < 3; i++) {          for (int j = 0; j < 3; j++) {              hasil[i][j] = matriksA[i][j] + matriksB[i][j];          }      }      tampilkanMatriks(hasil);      cout << "\n--- Hasil Pengurangan (A - B) ---" << endl;      for (int i = 0; i < 3; i++) {          for (int j = 0; j < 3; j++) {              hasil[i][j] = matriksA[i][j] - matriksB[i][j];          }      }      tampilkanMatriks(hasil);      cout << "\n--- Hasil Perkalian (A \* B) ---" << endl;      for (int i = 0; i < 3; i++) {          for (int j = 0; j < 3; j++) {              hasil[i][j] = 0;              for (int k = 0; k < 3; k++) {                  hasil[i][j] += matriksA[i][k] \* matriksB[k][j];              }          }      }      tampilkanMatriks(hasil);      return 0;  } |

**Screenshoot program**

****

**Deskripsi program**

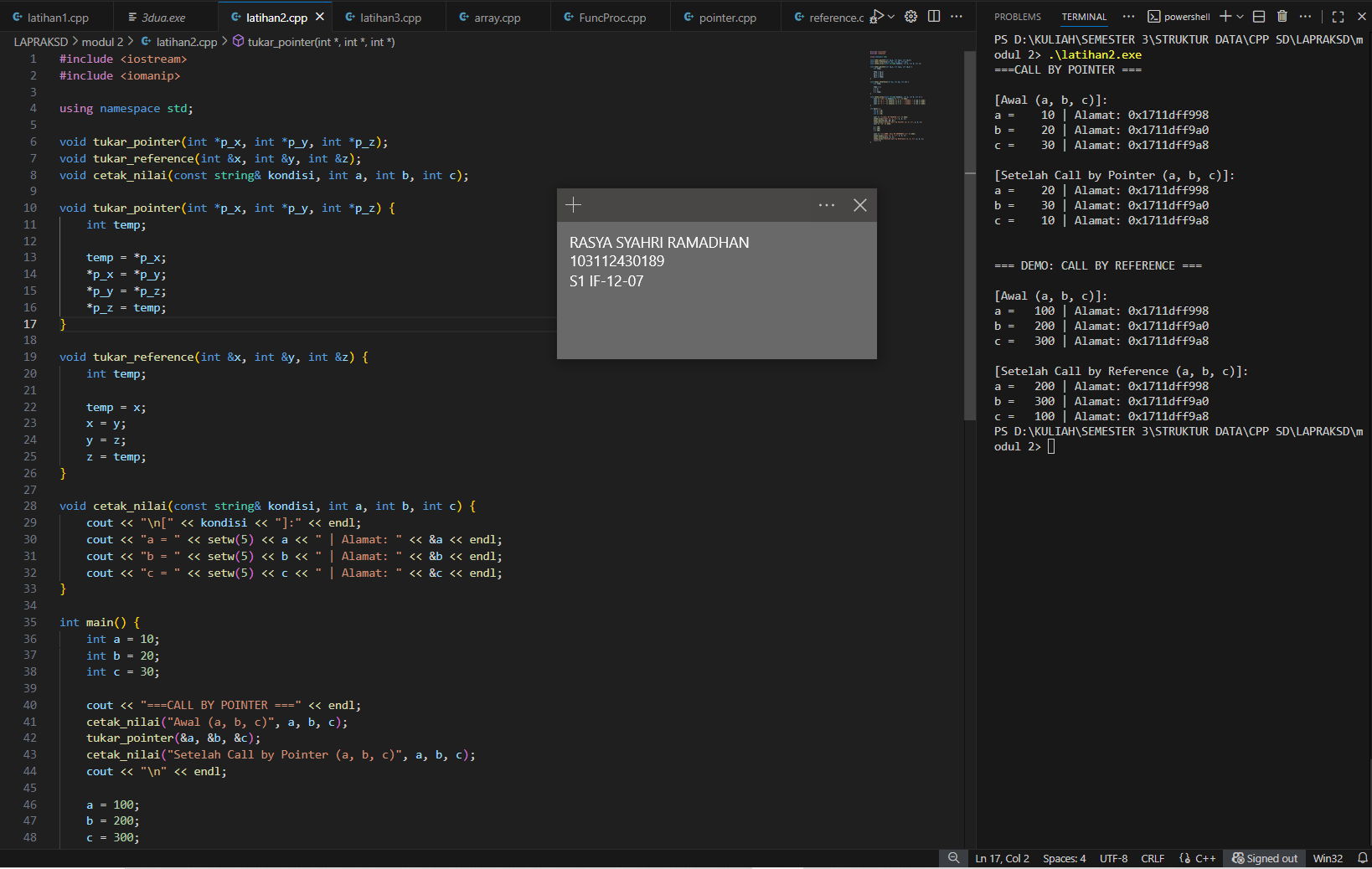
Program di atas adalah program C++ yang melakukan operasi dasar pada dua buah matriks berukuran 3x3, yaitu penjumlahan, pengurangan, dan perkalian. Program menggunakan library **<iostream>** untuk input-output dan **<iomanip>** untuk memformat tampilan agar hasil matriks terlihat rapi. Pengguna diminta untuk memasukkan elemen-elemen dari Matriks A dan Matriks B, kemudian program menampilkan kedua matriks tersebut. Setelah itu, program menghitung dan menampilkan hasil dari operasi A + B, A - B, dan A × B dengan menggunakan perulangan bersarang. Fungsi **tampilkanMatriks()** digunakan untuk menampilkan setiap matriks dengan jarak kolom yang seragam menggunakan manipulasi **setw()**.

1. **Latihan**

**Source code**

|  |
| --- |
| #include <iostream>  #include <iomanip>  using namespace std;  void tukar\_pointer(int \*p\_x, int \*p\_y, int \*p\_z);  void tukar\_reference(int &x, int &y, int &z);  void cetak\_nilai(const string& kondisi, int a, int b, int c);  void tukar\_pointer(int \*p\_x, int \*p\_y, int \*p\_z) {      int temp;      temp = \*p\_x;      \*p\_x = \*p\_y;      \*p\_y = \*p\_z;      \*p\_z = temp;  }  void tukar\_reference(int &x, int &y, int &z) {      int temp;      temp = x;      x = y;      y = z;      z = temp;  }  void cetak\_nilai(const string& kondisi, int a, int b, int c) {      cout << "\n[" << kondisi << "]:" << endl;      cout << "a = " << setw(5) << a << " | Alamat: " << &a << endl;      cout << "b = " << setw(5) << b << " | Alamat: " << &b << endl;      cout << "c = " << setw(5) << c << " | Alamat: " << &c << endl;  }  int main() {      int a = 10;      int b = 20;      int c = 30;      cout << "===CALL BY POINTER ===" << endl;      cetak\_nilai("Awal (a, b, c)", a, b, c);      tukar\_pointer(&a, &b, &c);      cetak\_nilai("Setelah Call by Pointer (a, b, c)", a, b, c);      cout << "\n" << endl;        a = 100;      b = 200;      c = 300;      cout << "=== DEMO: CALL BY REFERENCE ===" << endl;      cetak\_nilai("Awal (a, b, c)", a, b, c);      tukar\_reference(a, b, c);      cetak\_nilai("Setelah Call by Reference (a, b, c)", a, b, c);      return 0;  } |

**Screenshoot program**

****

**Deskripsi program**

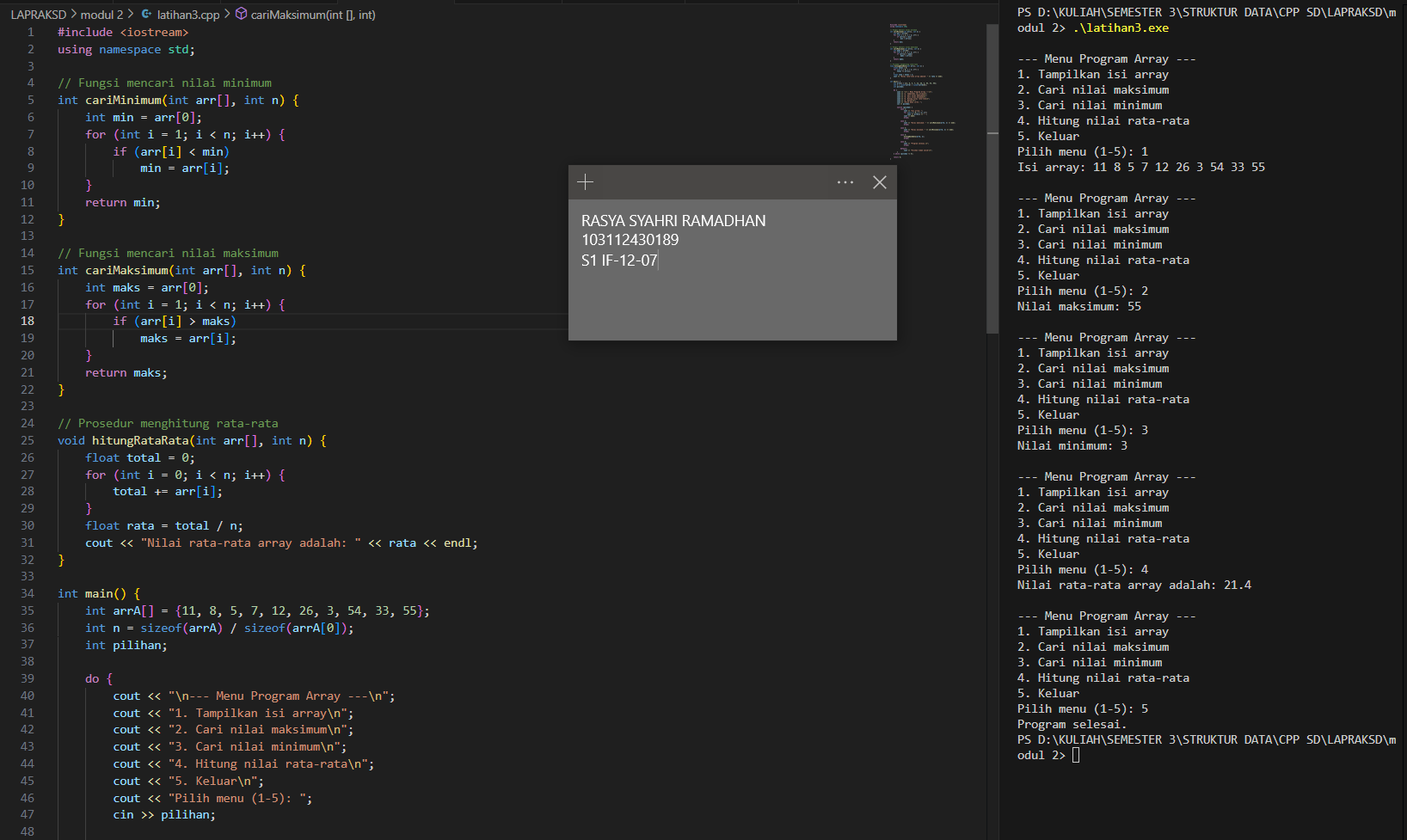
Program di atas adalah program C++ yang mendemonstrasikan perbedaan antara **call by pointer** dan **call by reference** dalam pertukaran nilai tiga variabel. Program menggunakan fungsi **tukar\_pointer()** yang menerima alamat variabel sebagai parameter **(int\*)** dan **tukar\_reference()** yang menggunakan referensi **(int&)**. Kedua fungsi tersebut menukar nilai variabel **a, b,** dan **c** secara bergantian dengan menggunakan variabel sementara **temp**. Fungsi **cetak\_nilai()** digunakan untuk menampilkan nilai dan alamat memori dari setiap variabel dengan format rapi menggunakan **setw()** dari library **<iomanip>.** Pada bagian main(), program pertama-tama menukar nilai **a, b,** dan **c** menggunakan metode pointer, lalu mengulangi proses dengan metode reference untuk menunjukkan bahwa keduanya dapat mengubah nilai variabel asli di luar fungsi.

1. **Latihan 3**

**Source code**

|  |
| --- |
| #include <iostream>  using namespace std;  int cariMinimum(int arr[], int n) {      int min = arr[0];      for (int i = 1; i < n; i++) {          if (arr[i] < min)              min = arr[i];      }      return min;  }  int cariMaksimum(int arr[], int n) {      int maks = arr[0];      for (int i = 1; i < n; i++) {          if (arr[i] > maks)              maks = arr[i];      }      return maks;  }  void hitungRataRata(int arr[], int n) {      float total = 0;      for (int i = 0; i < n; i++) {          total += arr[i];      }      float rata = total / n;      cout << "Nilai rata-rata array adalah: " << rata << endl;  }  int main() {      int arrA[] = {11, 8, 5, 7, 12, 26, 3, 54, 33, 55};      int n = sizeof(arrA) / sizeof(arrA[0]);      int pilihan;      do {          cout << "\n--- Menu Program Array ---\n";          cout << "1. Tampilkan isi array\n";          cout << "2. Cari nilai maksimum\n";          cout << "3. Cari nilai minimum\n";          cout << "4. Hitung nilai rata-rata\n";          cout << "5. Keluar\n";          cout << "Pilih menu (1-5): ";          cin >> pilihan;          switch (pilihan) {              case 1:                  cout << "Isi array: ";                  for (int i = 0; i < n; i++)                      cout << arrA[i] << " ";                  cout << endl;                  break;              case 2:                  cout << "Nilai maksimum: " << cariMaksimum(arrA, n) << endl;                  break;              case 3:                  cout << "Nilai minimum: " << cariMinimum(arrA, n) << endl;                  break;              case 4:                  hitungRataRata(arrA, n);                  break;              case 5:                  cout << "Program selesai.\n";                  break;              default:                  cout << "Pilihan tidak valid!\n";          }      } while (pilihan != 5);      return 0;  } |

**Screenshoot program**

****

**Deskripsi program**

Program di atas adalah program C++ yang menggunakan array untuk menampilkan, mencari, dan menghitung data numerik dengan beberapa pilihan menu. Program menyimpan 10 elemen bilangan dalam array **arrA**, kemudian menampilkan menu interaktif yang memungkinkan pengguna memilih operasi tertentu. Fungsi **cariMaksimum()** digunakan untuk mencari nilai terbesar dalam array, sedangkan **cariMinimum()** mencari nilai terkecil. Prosedur **hitungRataRata()** menghitung dan menampilkan nilai rata-rata elemen array. Program menggunakan struktur kontrol **do-while** dan s**witch** untuk menjalankan menu berulang kali hingga pengguna memilih keluar (opsi 5). Dengan demikian, program ini menunjukkan cara dasar penggunaan fungsi, array, dan kontrol alur dalam C++.