**Praktikum Algoritma dan Struktur Data**

“Insertion Selection”

**Oleh:**

Marlon Immanuel Sarapang / 5223600010

**Program Studi Sarjana Terapan Teknologi Game**

**Departemen Teknologi Multimedia Kreatif**

**Politeknik Elektronika Negeri Surabaya**

**2024**

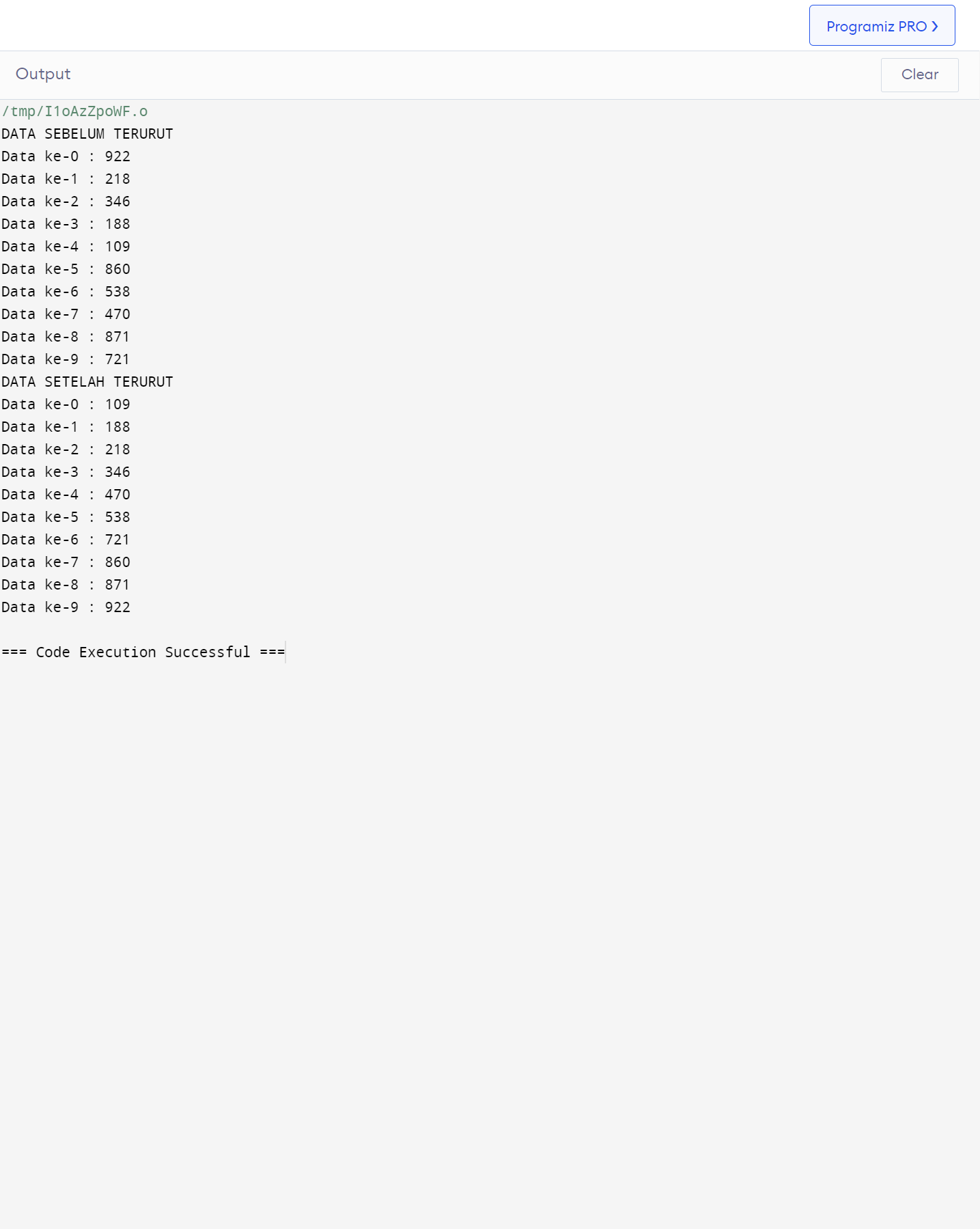
**Percobaan**

1. Percobaan Straight Insertion Sort

**Source code**

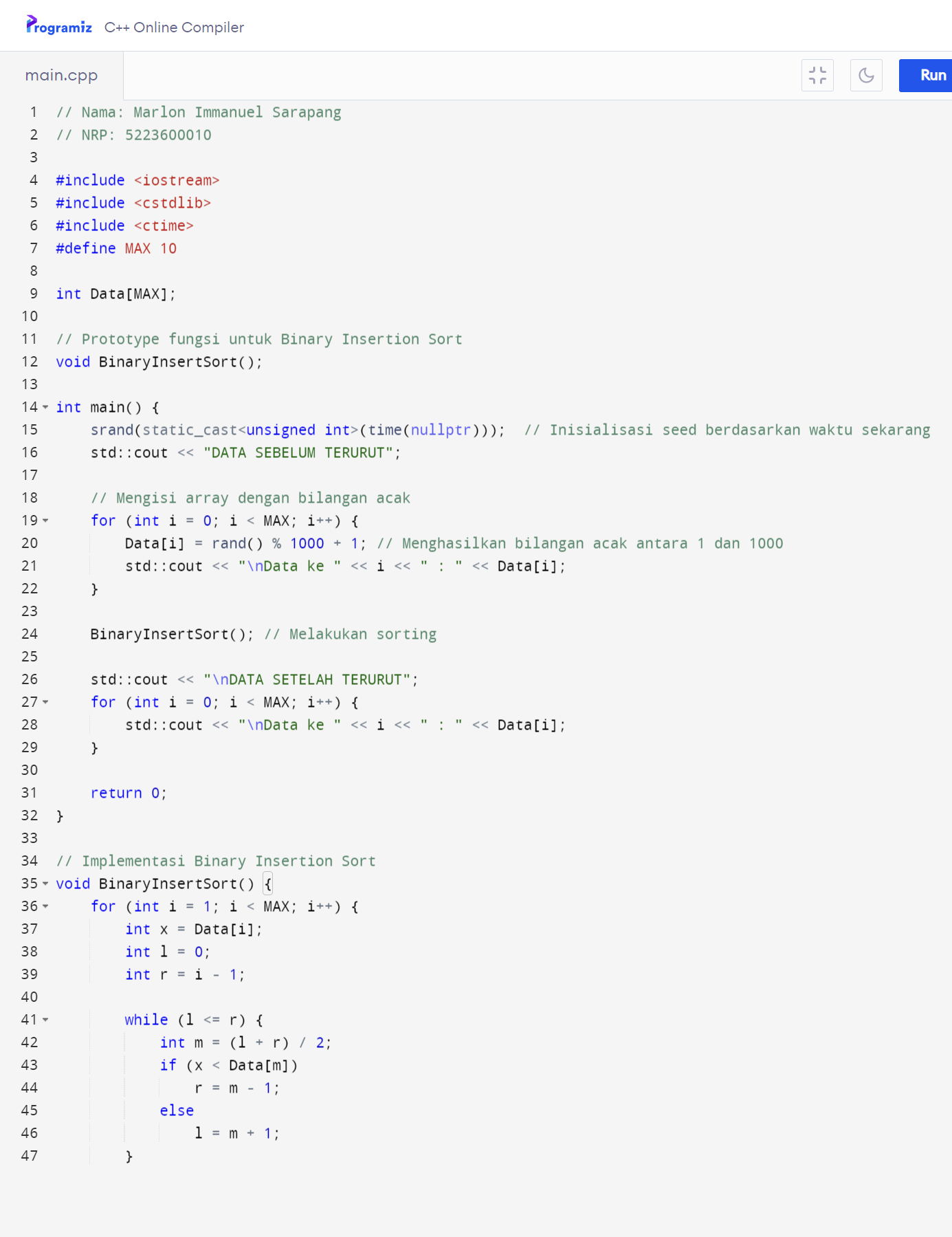


**Hasil**

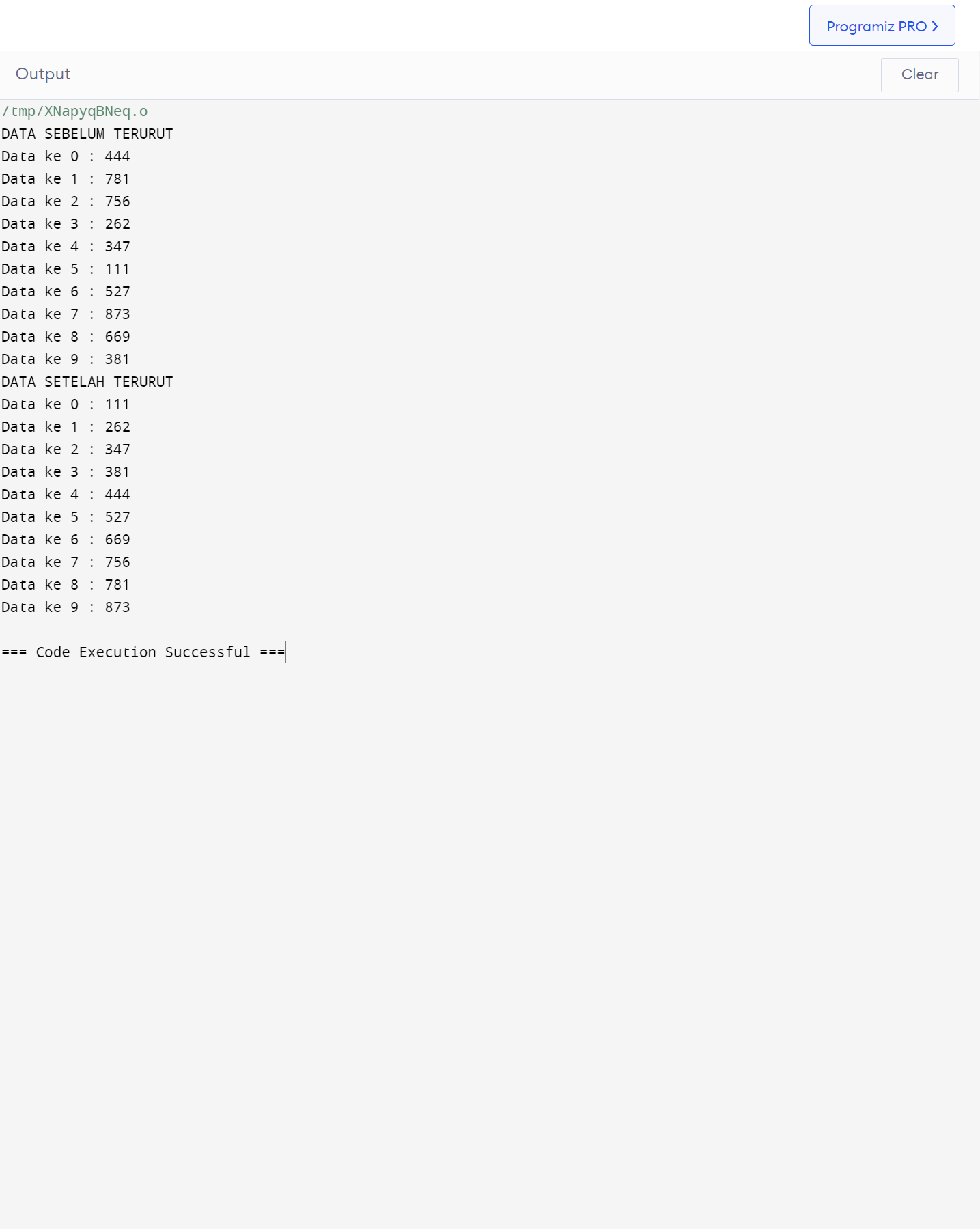


1. Percobaan Binary Insertion Sort

**Source code**

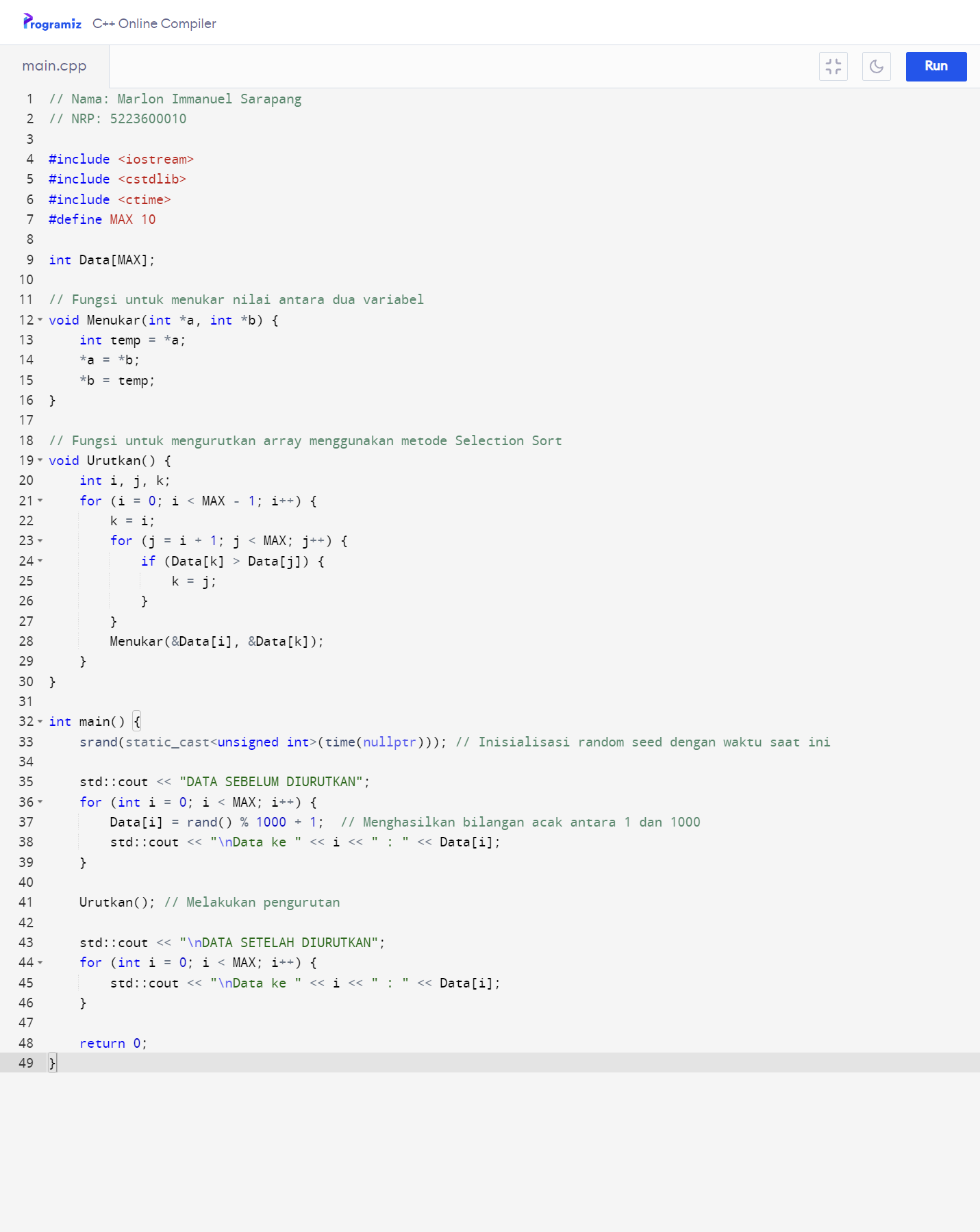


**Hasil**

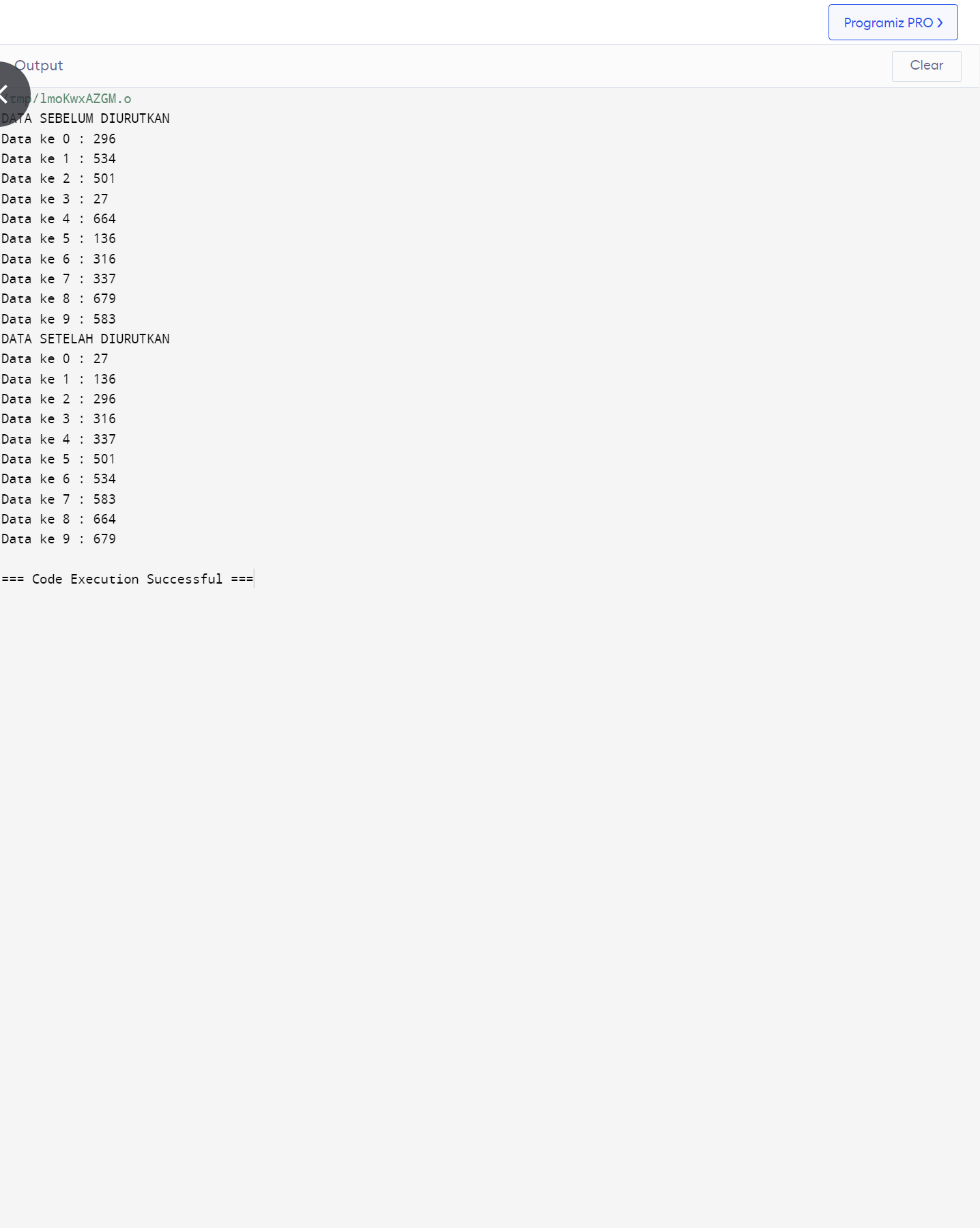


1. Percobaan Selection Sort

**Source code**



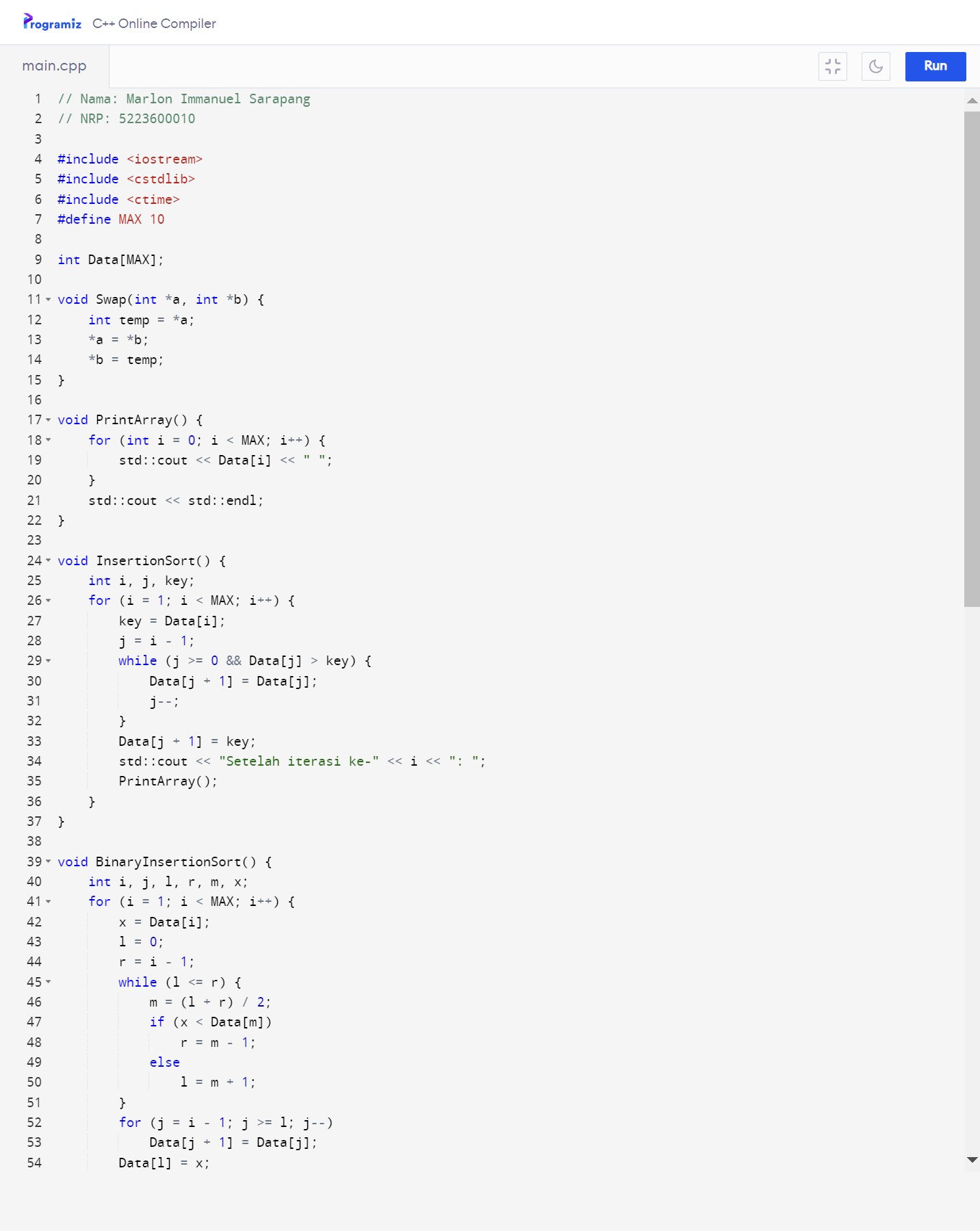
**Hasil**



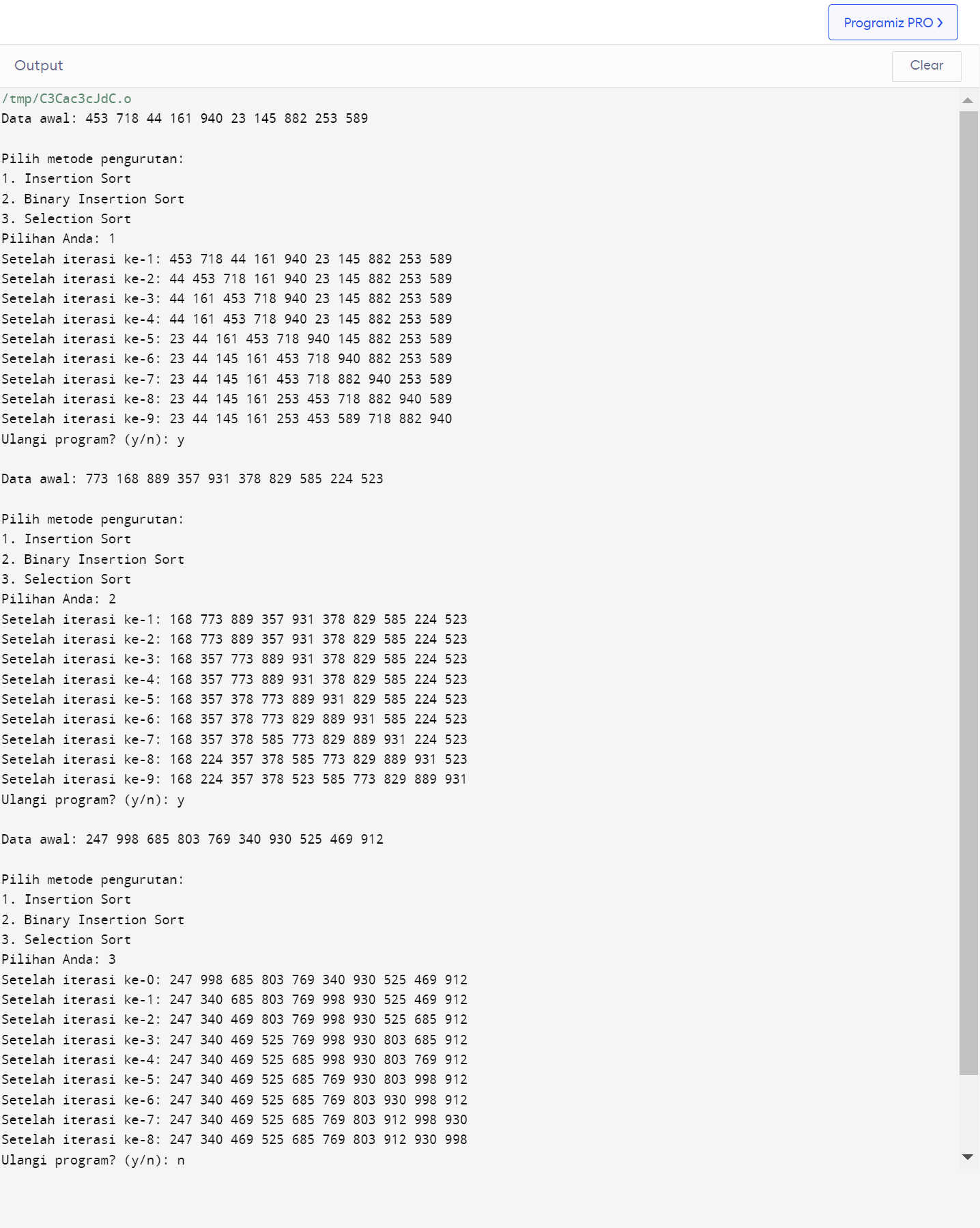
**Latihan**

1. Tambahkan kode program untuk menampilkan perubahan setiap iterasi dari proses pengurutan dengan penyisipan langsung, penyisipan biner dan seleksi.

**Source code**

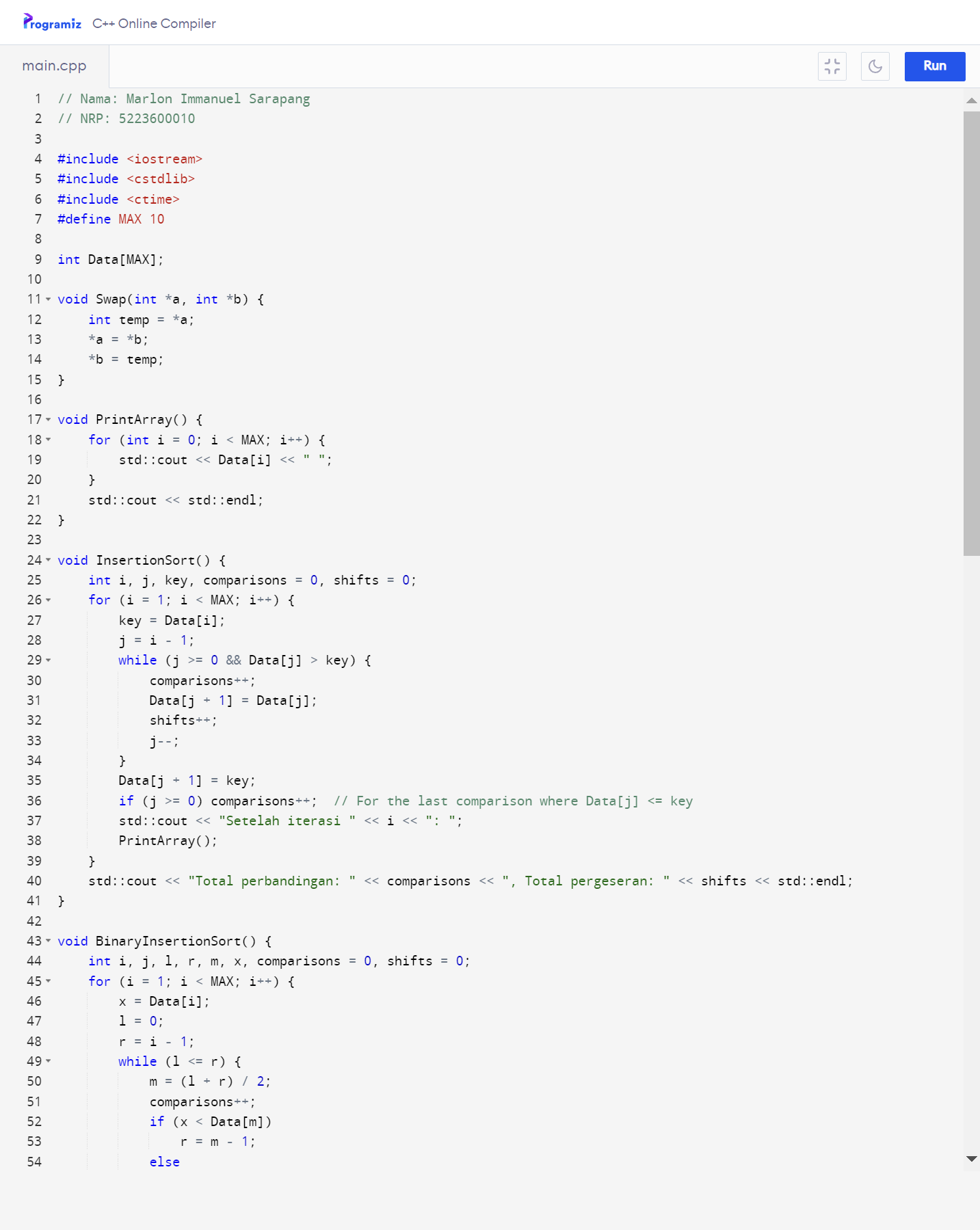


**Hasil**

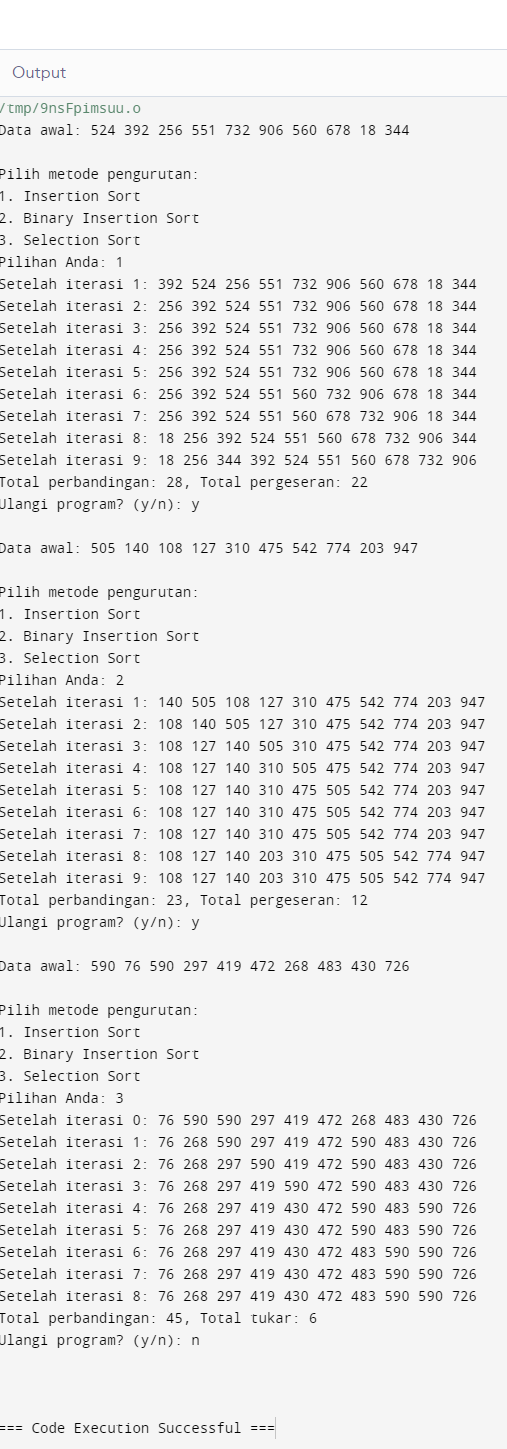


1. Tambahkan kode program untuk menghitung banyaknya perbandingan dan pergeseran pada algoritma pengurutan penyisipan langsung, penyisipan biner dan seleksi.

**Source code**

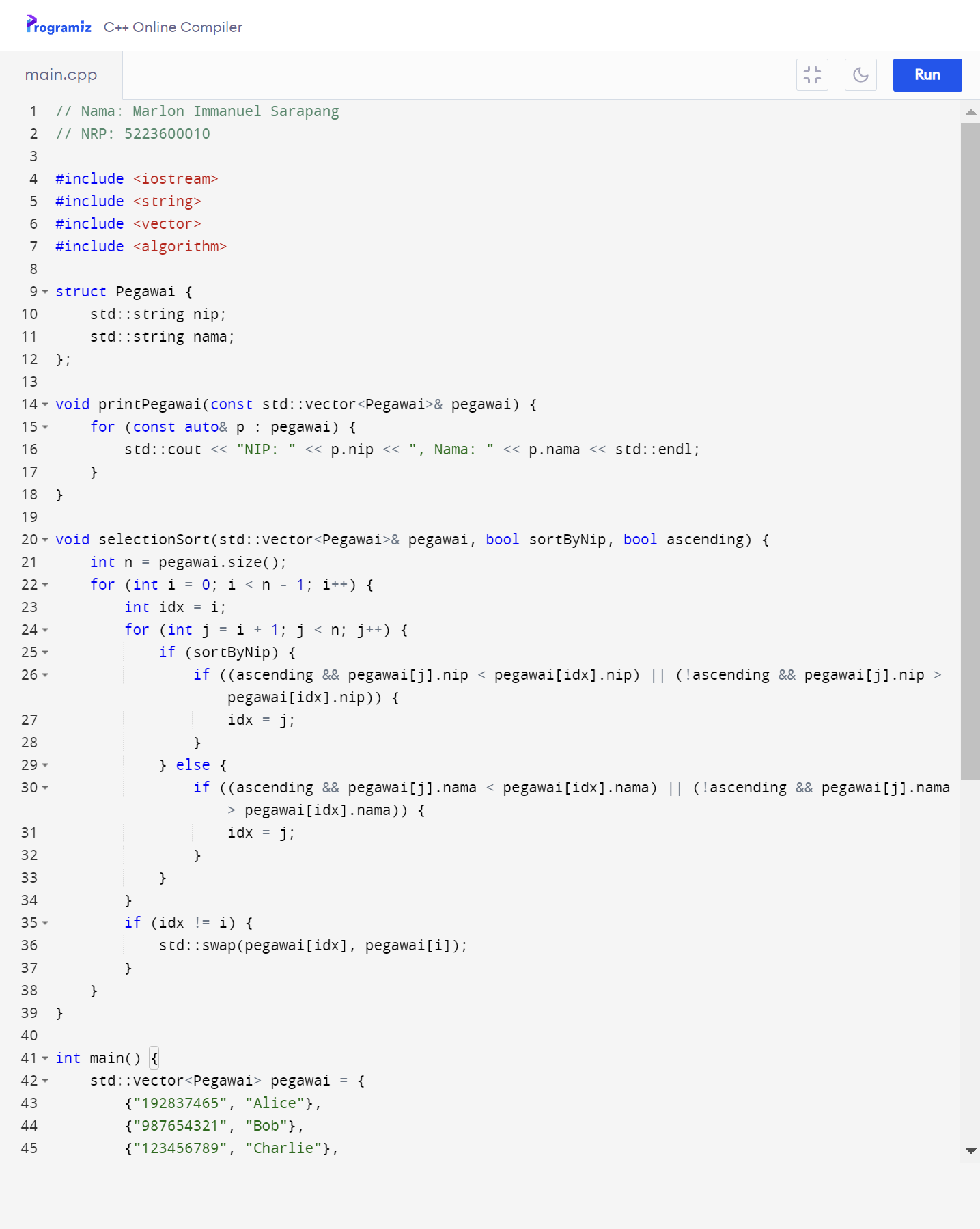


**Hasil**

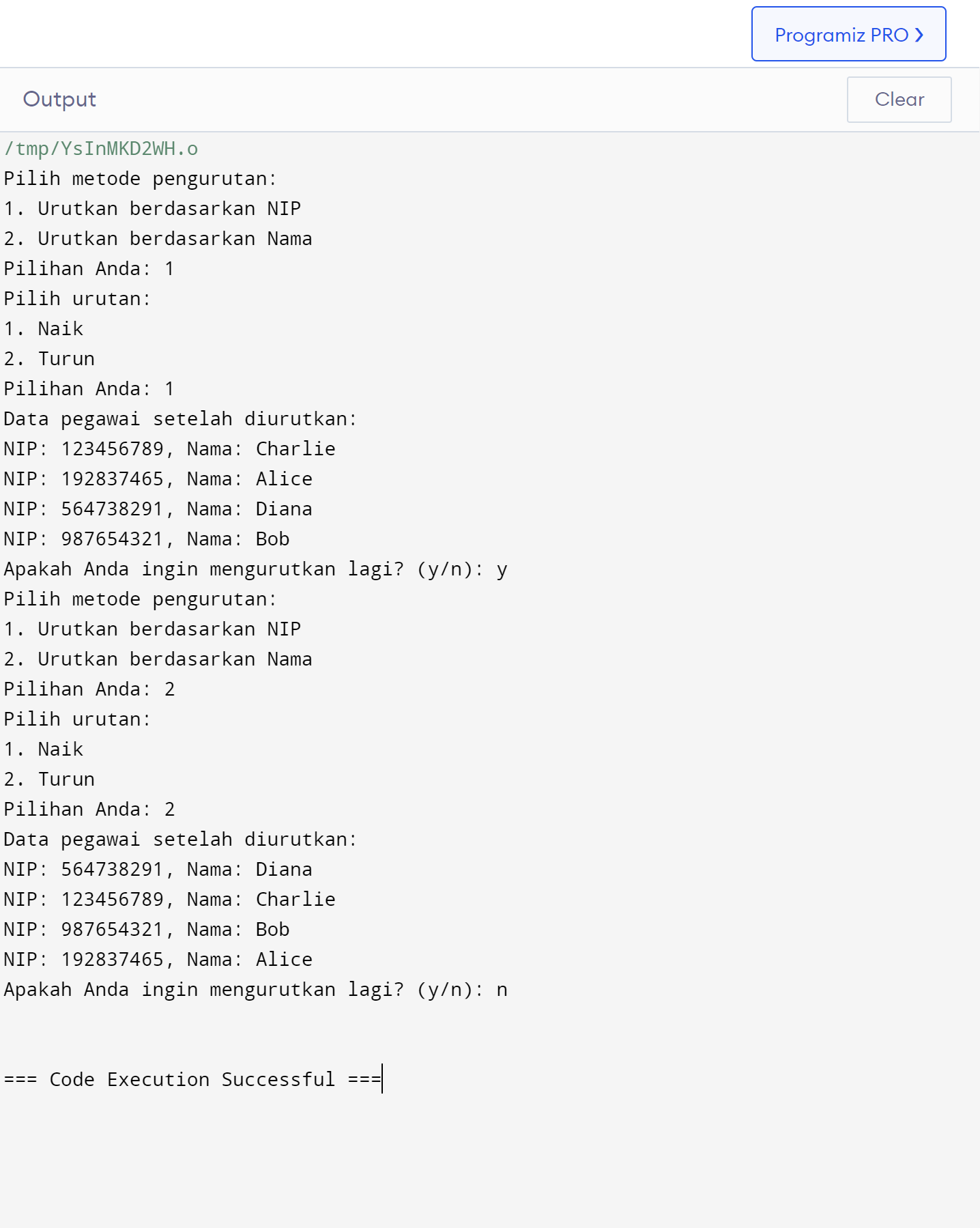
****

1. Buatlah project baru untuk Latihan dan implementasikan pengurutan data Pegawai pada tugas pendahuluan dengan ketentuan :
2. Metode pengurutan dapat dipilih.
3. Pengurutan dapat dipilih secara urut naik atau turun.
4. Pengurutan dapat dipilih berdasarkan NIP dan NAMA.
5. Gunakan struktur data array.

**Source code**



**Hasil**



1. Berikan kesimpulan dari percobaan dan latihan yang telah Anda lakukan.

Program-program yang telah kita eksplorasi memberikan wawasan mendalam tentang cara mengimplementasikan dan mengelola data menggunakan berbagai metode pengurutan dalam C++. Kita telah menggunakan struktur Pegawai dengan atribut nip dan nama, yang ditampung dalam std::vector untuk memanfaatkan fleksibilitasnya dalam menyimpan koleksi data yang dinamis. Implementasi algoritma seperti Selection Sort menunjukkan adaptabilitasnya dalam mengurutkan data berdasarkan kriteria yang berbeda dan urutan yang bisa diatur, meskipun algoritma ini mungkin kurang efisien untuk set data yang besar.

Interaksi pengguna dengan program ditingkatkan melalui penggunaan loop do-while, memungkinkan pengguna untuk mengulangi pemilihan metode pengurutan tanpa perlu menjalankan ulang program, meningkatkan pengalaman pengguna dan memudahkan pengujian fungsi pengurutan. Selain itu, program tersebut menunjukkan praktik pemrograman yang solid, seperti penanganan input pengguna yang efektif dan pemisahan logika dengan fungsi helper seperti printPegawai.

Kedua aspek ini tidak hanya meningkatkan kejelasan dan pemeliharaan kode tetapi juga memberikan umpan balik langsung kepada pengguna, memperkaya pemahaman mereka tentang operasi pengurutan. Keseluruhan inisiatif ini memperlihatkan bagaimana teori pengurutan dan manipulasi data dapat diaplikasikan dalam skenario nyata, memperkuat keterampilan yang sangat penting dalam pengembangan software dan pemecahan masalah menggunakan C++.