

**LAPORAN**

**ALGORITMA PEMORGRAMAN**

**DISUSUN OLEH**

RIFAL FEBIYAN (2100018345)

SLOT SELASA 13.30 – KELAS G

**PROGRAM STUDI INFORMATIKA FAKULTAS TEKNOLOGI INDUSTRI**

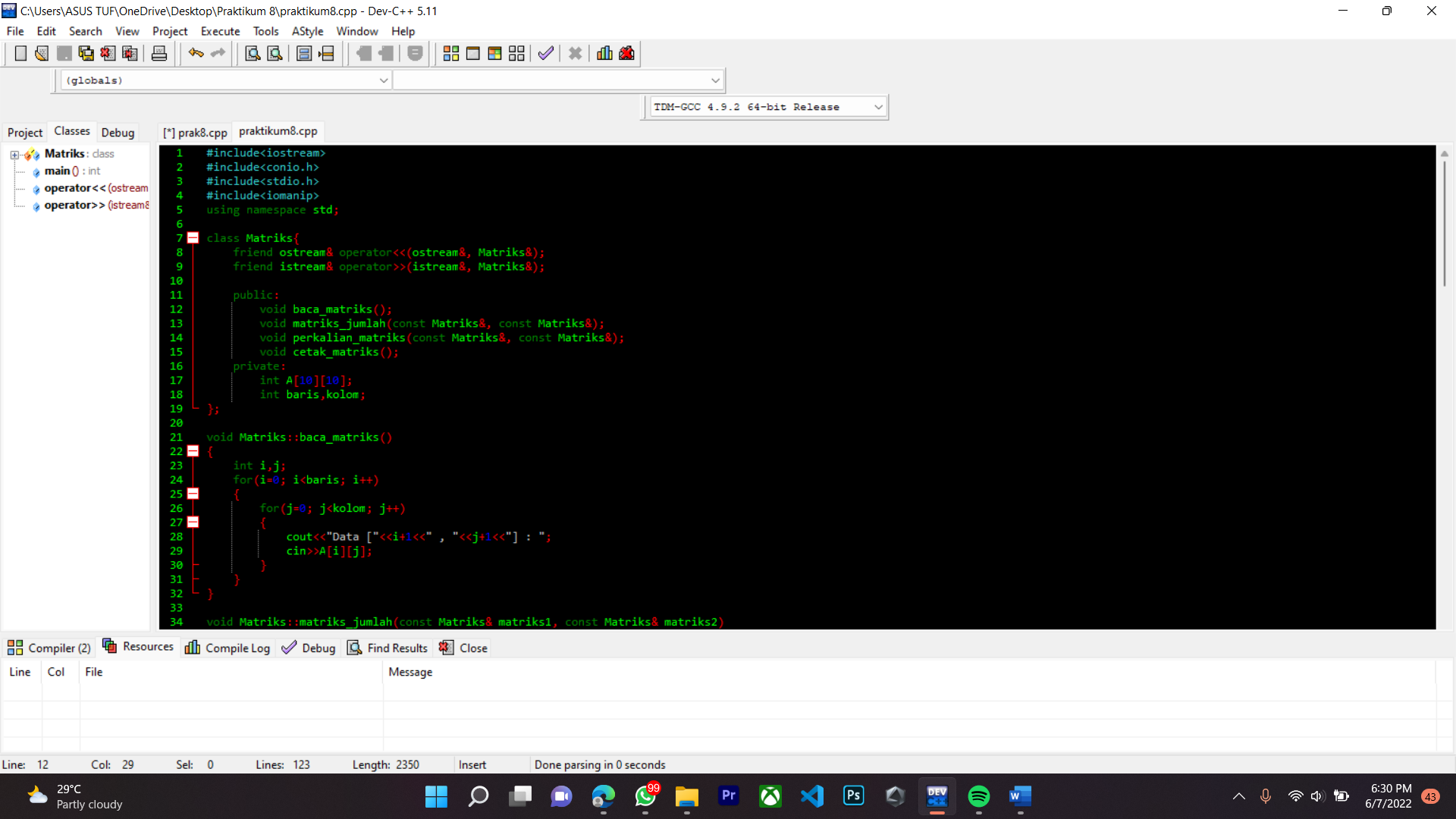
**UNIVERSITAS AHMAD DAHLAN**

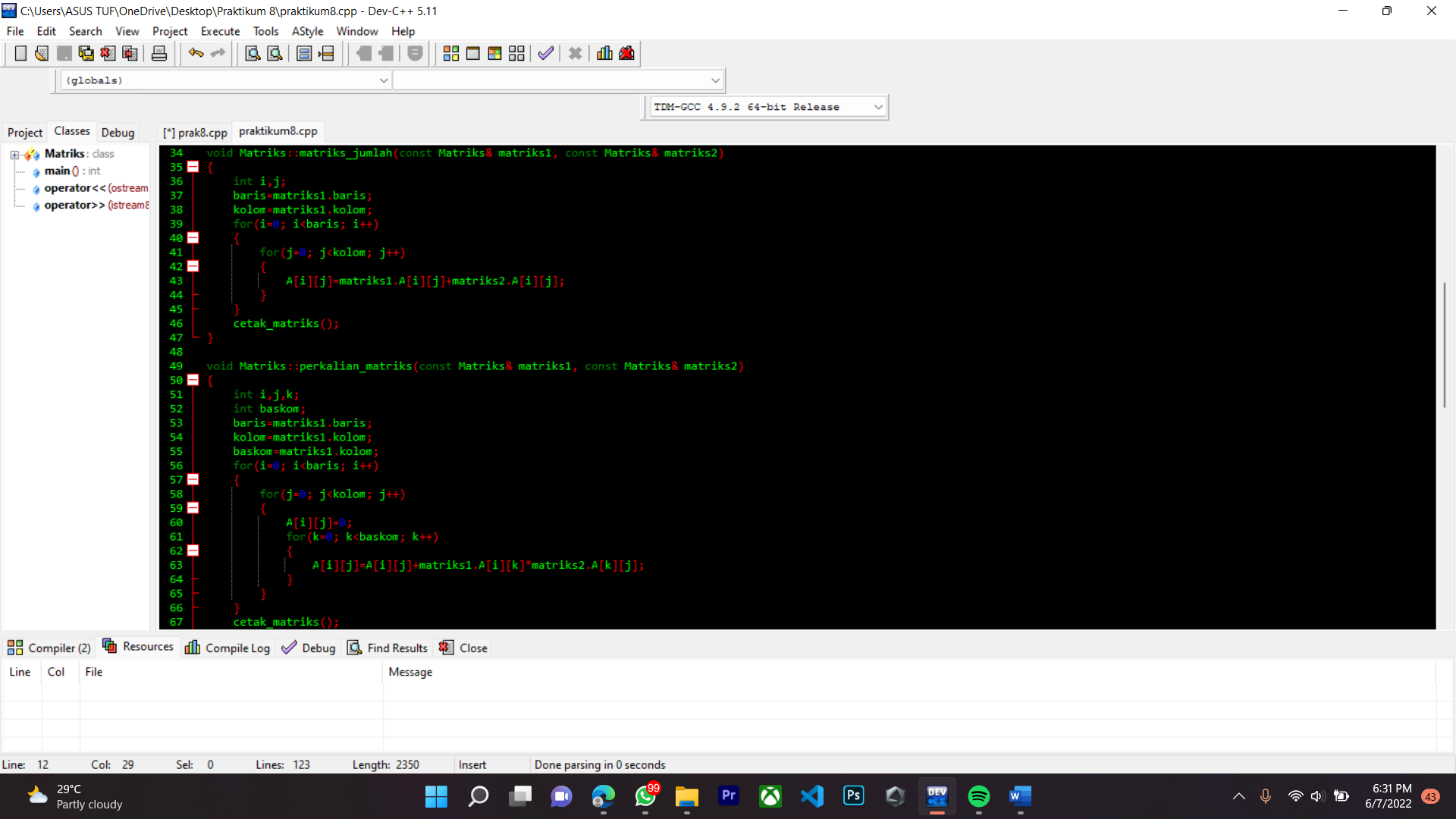
**TAHUN AJARAN 2021/2022**

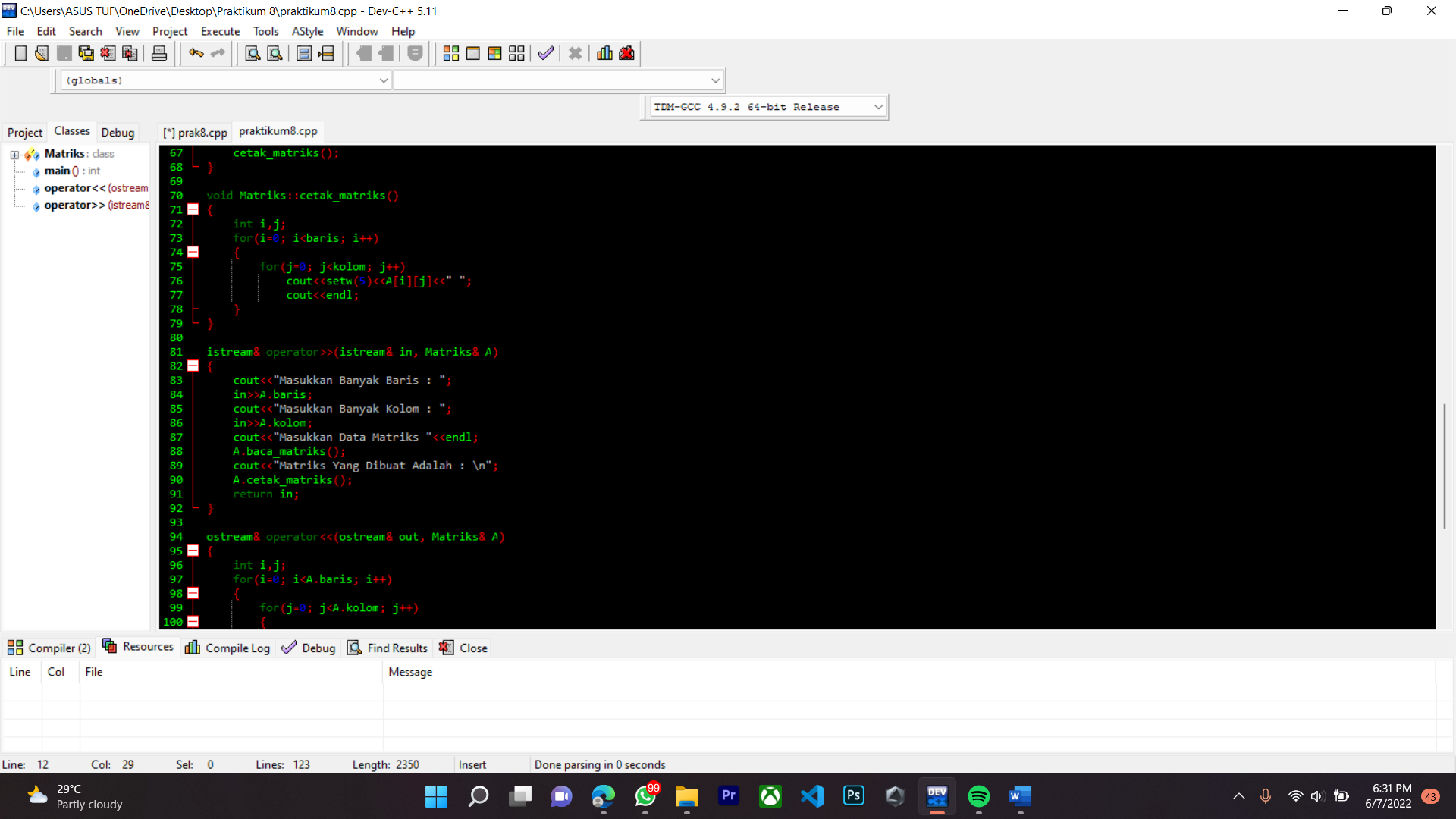
**KEGIATAN PRAKTIKUM 8 : ARRAY 2 DIMENSI**

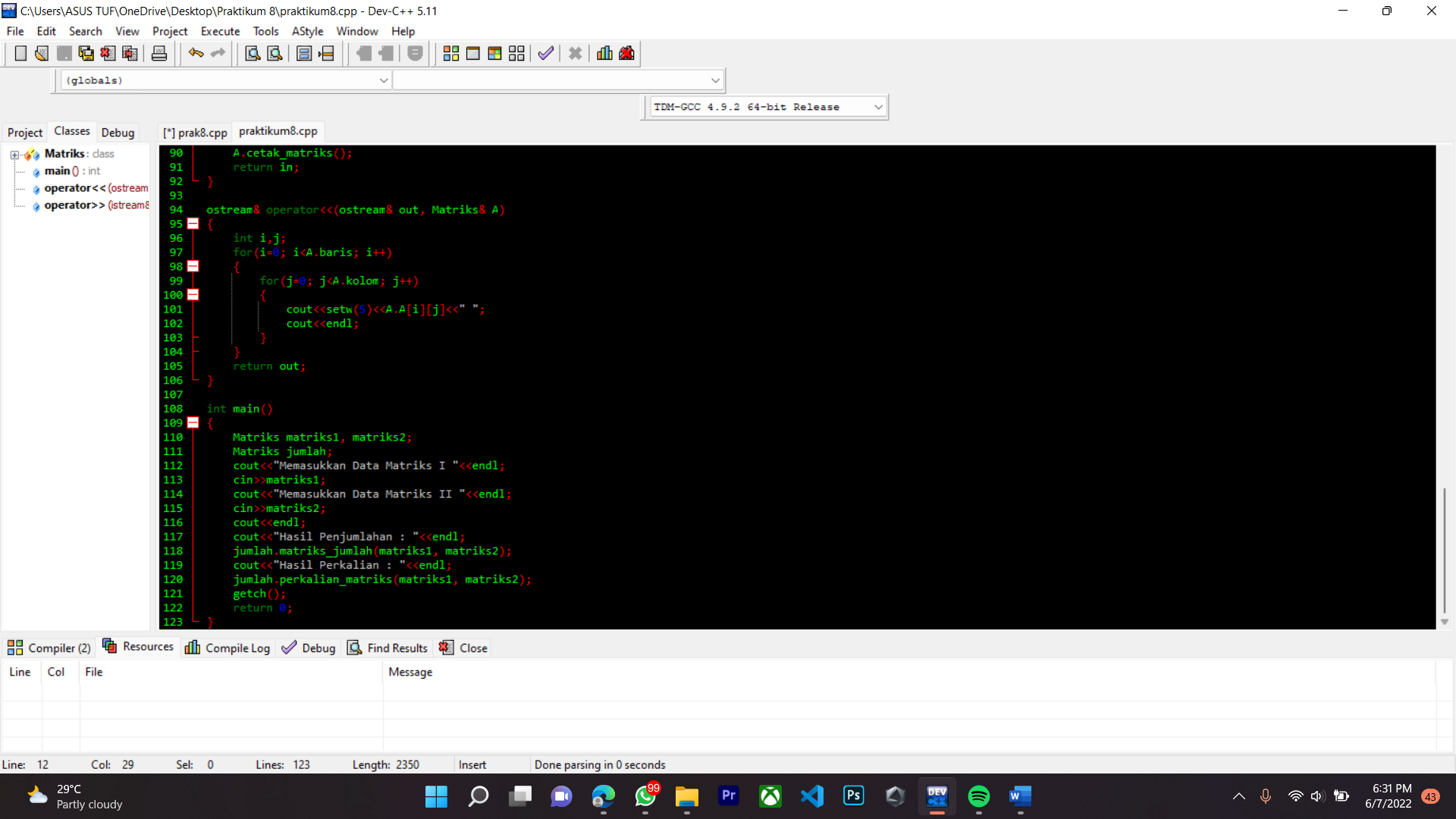
Ketiklah program operasi matriks berikut ini. Amati tata cara setiap operasi matriks berlaku pada setiap elemen matriks. Buat kesimpulan tentang pentingnya syarat setiap operasi matriks dapat dilakukan

* Tampilan pada Dev C++



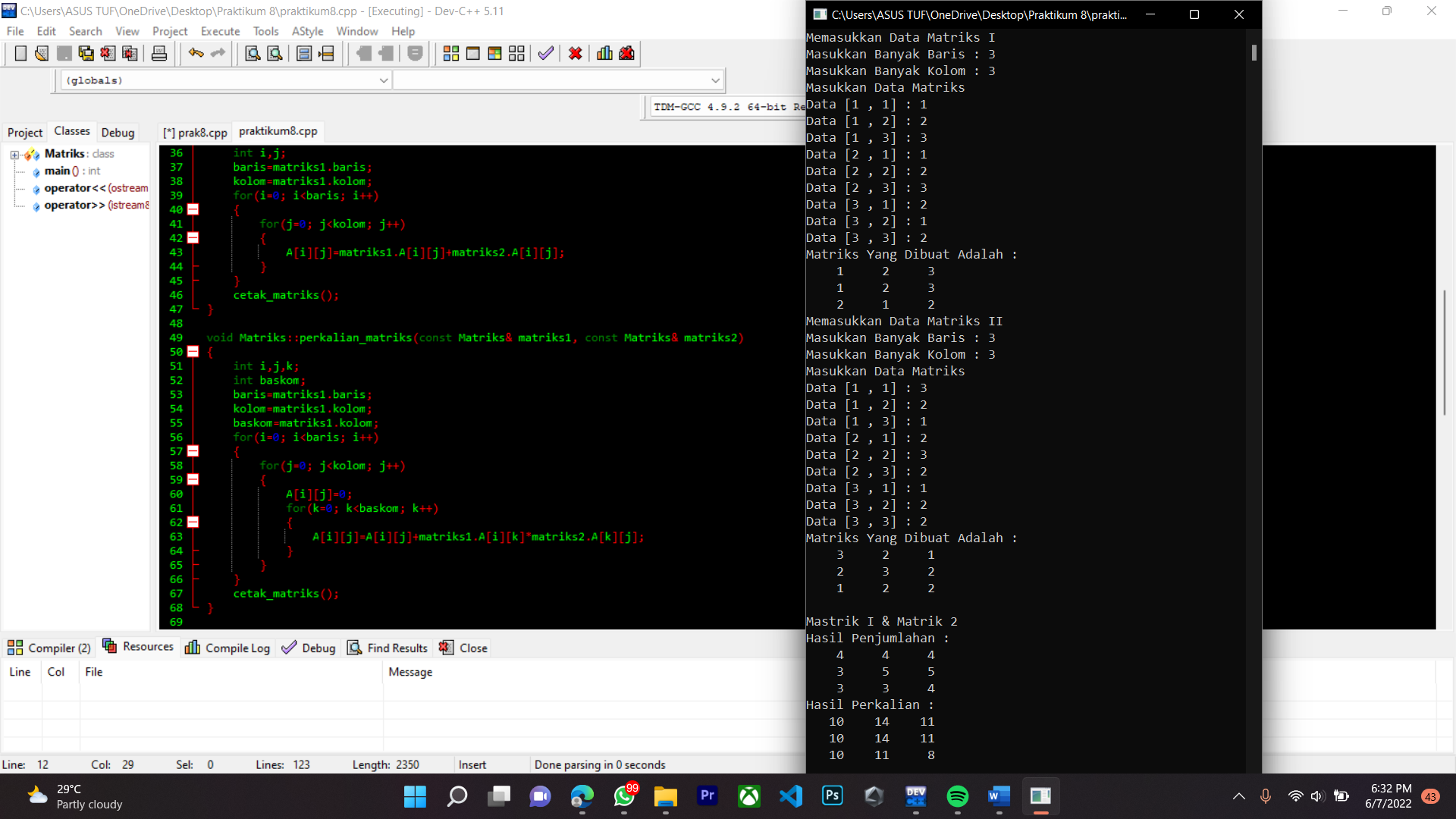




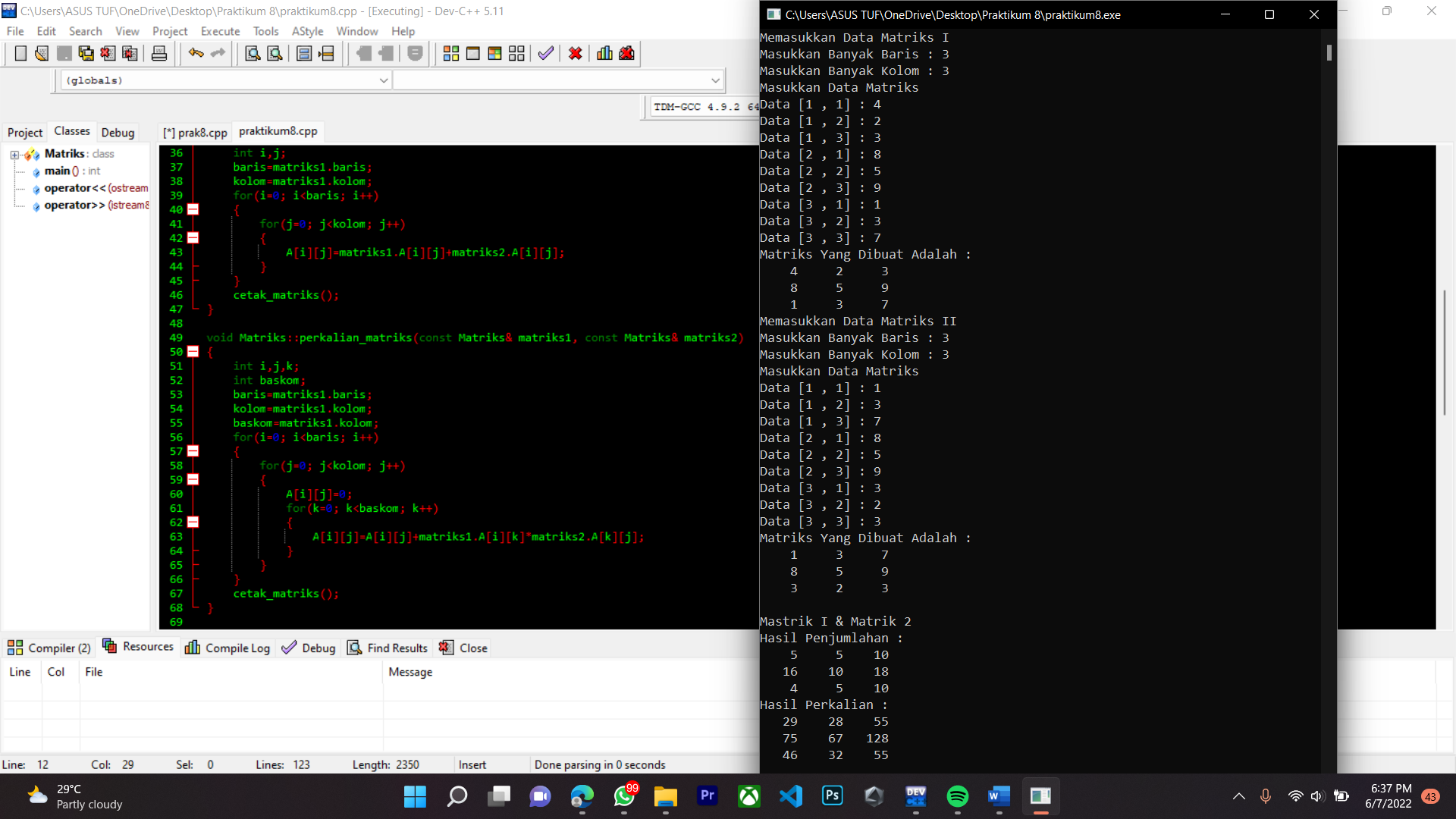


Output 🡺

Percobaan 1



Percobaan 2



Source Code

|  |
| --- |
| #include<iostream>  #include<conio.h>  #include<stdio.h>  #include<iomanip>  using namespace std;  class Matriks{  friend ostream& operator<<(ostream&, Matriks&);  friend istream& operator>>(istream&, Matriks&);    public:  void baca\_matriks();  void matriks\_jumlah(const Matriks&, const Matriks&);  void perkalian\_matriks(const Matriks&, const Matriks&);  void cetak\_matriks();  private:  int A[10][10];  int baris,kolom;  };  void Matriks::baca\_matriks()  {  int i,j;  for(i=0; i<baris; i++)  {  for(j=0; j<kolom; j++)  {  cout<<"Data ["<<i+1<<" , "<<j+1<<"] : ";  cin>>A[i][j];  }  }  }  void Matriks::matriks\_jumlah(const Matriks& matriks1, const Matriks& matriks2)  {  int i,j;  baris=matriks1.baris;  kolom=matriks1.kolom;  for(i=0; i<baris; i++)  {  for(j=0; j<kolom; j++)  {  A[i][j]=matriks1.A[i][j]+matriks2.A[i][j];  }  }  cetak\_matriks();  }  void Matriks::perkalian\_matriks(const Matriks& matriks1, const Matriks& matriks2)  {  int i,j,k;  int baskom;  baris=matriks1.baris;  kolom=matriks1.kolom;  baskom=matriks1.kolom;  for(i=0; i<baris; i++)  {  for(j=0; j<kolom; j++)  {  A[i][j]=0;  for(k=0; k<baskom; k++)  {  A[i][j]=A[i][j]+matriks1.A[i][k]\*matriks2.A[k][j];  }  }  }  cetak\_matriks();  }  void Matriks::cetak\_matriks()  {  int i,j;  for(i=0; i<baris; i++)  {  for(j=0; j<kolom; j++)  cout<<setw(5)<<A[i][j]<<" ";  cout<<endl;  }  }  istream& operator>>(istream& in, Matriks& A)  {  cout<<"Masukkan Banyak Baris : ";  in>>A.baris;  cout<<"Masukkan Banyak Kolom : ";  in>>A.kolom;  cout<<"Masukkan Data Matriks "<<endl;  A.baca\_matriks();  cout<<"Matriks Yang Dibuat Adalah : \n";  A.cetak\_matriks();  return in;  }  ostream& operator<<(ostream& out, Matriks& A)  {  int i,j;  for(i=0; i<A.baris; i++)  {  for(j=0; j<A.kolom; j++)  {  cout<<setw(5)<<A.A[i][j]<<" ";  cout<<endl;  }  }  return out;  }  int main()  {  Matriks matriks1, matriks2;  Matriks jumlah;  cout<<"Memasukkan Data Matriks I "<<endl;  cin>>matriks1;  cout<<"Memasukkan Data Matriks II "<<endl;  cin>>matriks2;  cout<<endl;  cout<<"Hasil Penjumlahan : "<<endl;  jumlah.matriks\_jumlah(matriks1, matriks2);  cout<<"Hasil Perkalian : "<<endl;  jumlah.perkalian\_matriks(matriks1, matriks2);  getch();  return 0;  } |

Penjelasan:

1). Pada operasi penjumlahan matriks, matriks dapat dijumlahkan apabila keduanya memiliki ordo yang sama. Hasil operasi penjumlahannya adalah matriks baru yang memiliki ordo sama dengan matriks semula, dengan elemen-elemennya terdiri dari hasil penjumlahan elemen-elemen pada matriks.

2). Pada operasi perkalian matriks, syarat dua buah matriks dapat dikalikan jika memiliki jumlah kolom matriks pertama yang sama dengan jumlah baris matriks ke dua. Ordo matriks hasil perkalian dua matriks adalah jumlah baris pertama dikali jumlah kolom ke dua.

Screenshot dan Link Repository