

Tugas Pendahuluan Modul 1

STRUKTUR DATA - Ganjil 2024/2025

"Pengenalan C++: Subprogram & Array"

Ketentuan Tugas Pendahuluan

1. Tugas Pendahuluan dikerjakan secara **Individu**.
2. TP ini bersifat **WAJIB**, tidak mengerjakan = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
3. Hanya **MENGUMPULKAN** tetapi **TIDAK MENGERJAKAN** = **PENGURANGAN POIN JURNAL / TES ASESMEN**.
4. Deadline pengumpulan TP Modul 2 adalah Senin, 30 September 2024 pukul 07.30 WIB.
5. **TIDAK ADA TOLERANSI KETERLAMBATAN, TERLAMBAT ATAU TIDAK MENGUMPULKAN TP MAKA DIANGGAP TIDAK MENGERJAKAN**.
6. **DILARANG PLAGIAT (PLAGIAT = E)**.
7. Kerjakan TP dengan jelas agar dapat dimengerti.
8. Codingan diupload di Github dan upload Laporan di Lab menggunakan format **PDF** dengan ketentuan:  
**TP\_MOD\_[XX]\_NIM\_NAMA.pdf**

**CP (WA):**

- Andini (082243700965)
- Imelda (082135374187)

**SELAMAT MENGERJAKAN^^**

**LAPORAN PRAKTIKUM**  
**PERTEMUAN 3**  
**ABSTRACT DATA TYPE**



**Nama :**

Zulfa Mustafa Akhyar Iswahyudi (2311104010)

**Dosen :**

Yudha Islami Sulistya

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2024**

## A. Tujuan

Untuk melatih kompetensi Mahasiswa untuk memperdalam skill pemrograman C++

## B. Tools

Codeblocks, VSCode, Github

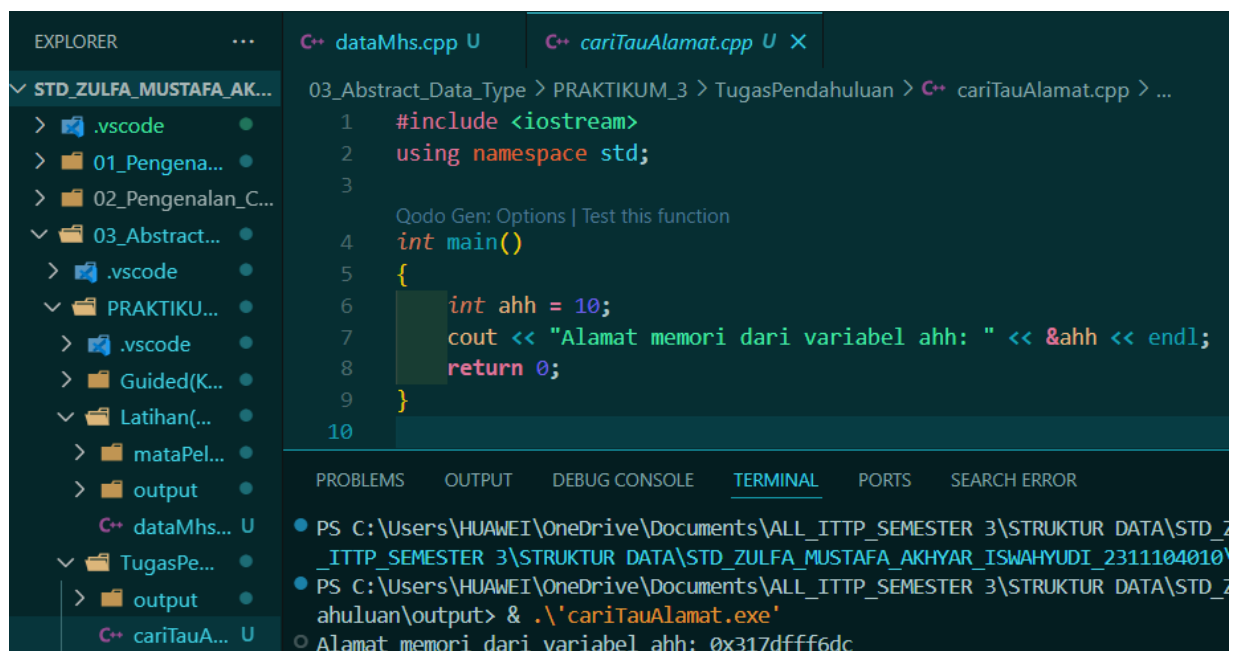
# TUGAS PENDAHULUAN

1.) Pointer itu gampangnya adalah fungsi untuk menyimpan alamat memori suatu variabel.

**Alamat ini sifatnya berubah-ubah dan tergenerasi secara RNG karena memang lokasi memori suatu variabel itu sifatnya tidak tetap.**

## 2.) Alamat Memori

Ini adalah Sintaks untuk mencari alamat memori suatu variabel. Kita buat variabel dari 'ahh' ini adalah integer dengan nilai 10 lalu buat output dengan mendeklarasikan '&ahh' yang merupakan alamat memori dari variabel 'ahh'.



```
EXPLORER  ...  C++ dataMhs.cpp U  C++ cariTauAlamat.cpp U X
03_Abstract_Data_Type > PRAKTIKUM_3 > TugasPendahuluan > C++ cariTauAlamat.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  Qodo Gen: Options | Test this function
5  int main()
6  {
7      int ahh = 10;
8      cout << "Alamat memori dari variabel ahh: " << &ahh << endl;
9      return 0;
10
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR
• PS C:\Users\HUAWEI\OneDrive\Documents\ALL_ITTP_SEMESTER 3\STRUKTUR DATA\STD_ZULFA_MUSTAFA_AKHYAR_ISWAHYUDI_2311104010\
• PS C:\Users\HUAWEI\OneDrive\Documents\ALL_ITTP_SEMESTER 3\STRUKTUR DATA\STD_ZULFA_MUSTAFA_AKHYAR_ISWAHYUDI_2311104010\
ahuluan\output> & .\'cariTauAlamat.exe'
o Alamat memori dari variabel ahh: 0x317dffff6dc
```

## 3.) Pemakaian Pointer

Ini adalah program yang menampilkan nilai dari suatu variabel dengan menggunakan pointer. Kita buat variabel integer 'oh' dengan nilai 75, lalu kita deklarasikan pointer yang diisi nilai dari alamat variabel 'oh'. Setelahnya buat output dengan memanggil pointer yang akan menampilkan isi nilai dari variabel 'oh'.

```
03_Abstract_Data_Type > PRAKTIKUM_3 > TugasPendahuluan > C++ pakePointer.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  Qodo Gen: Options | Test this function
5  int main()
6  {
7      int oh = 75;
8      int *ptr = &oh;
9      cout << "Nilai 'oh' melalui pointer: " << *ptr << endl;
10     return 0;
11 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR

● PS C:\Users\HUAWEI\OneDrive\Documents\ALL\_ITTP\_SEMESTER 3\STRUKTUR DATA\\_ITTP\_SEMESTER 3\STRUKTUR DATA\STD\_ZULFA\_MUSTAFA\_AKHYAR\_ISWAHYUDI\_23111  
PS C:\Users\HUAWEI\OneDrive\Documents\ALL\_ITTP\_SEMESTER 3\STRUKTUR DATA  
● ahuluan\output> & .\'pakePointer.exe'  
○ Nilai 'oh' melalui pointer: 75  
PS C:\Users\HUAWEI\OneDrive\Documents\ALL\_ITTP\_SEMESTER 3\STRUKTUR DATA  
ahuluan\output> |

4.) **ADT (Abstract Data Type)** itu simpelnya adalah sebuah model sistem yang memerhatikan operasi tipe data yang berbeda-beda dan memisahkan interface dari implementasinya.

5.) Contoh gampang implementasi ADT adalah mesin Kasir dalam pertokoan. **Dimana kita hanya bisa melihat proses uang masuk dan keluar beserta dengan detail dari total pembayaran dari belanja kita, tanpa memperlihatkan implementasi atau penerapan dari sistem yang berjalan dibelakang sebuah antarmuka(interface).**

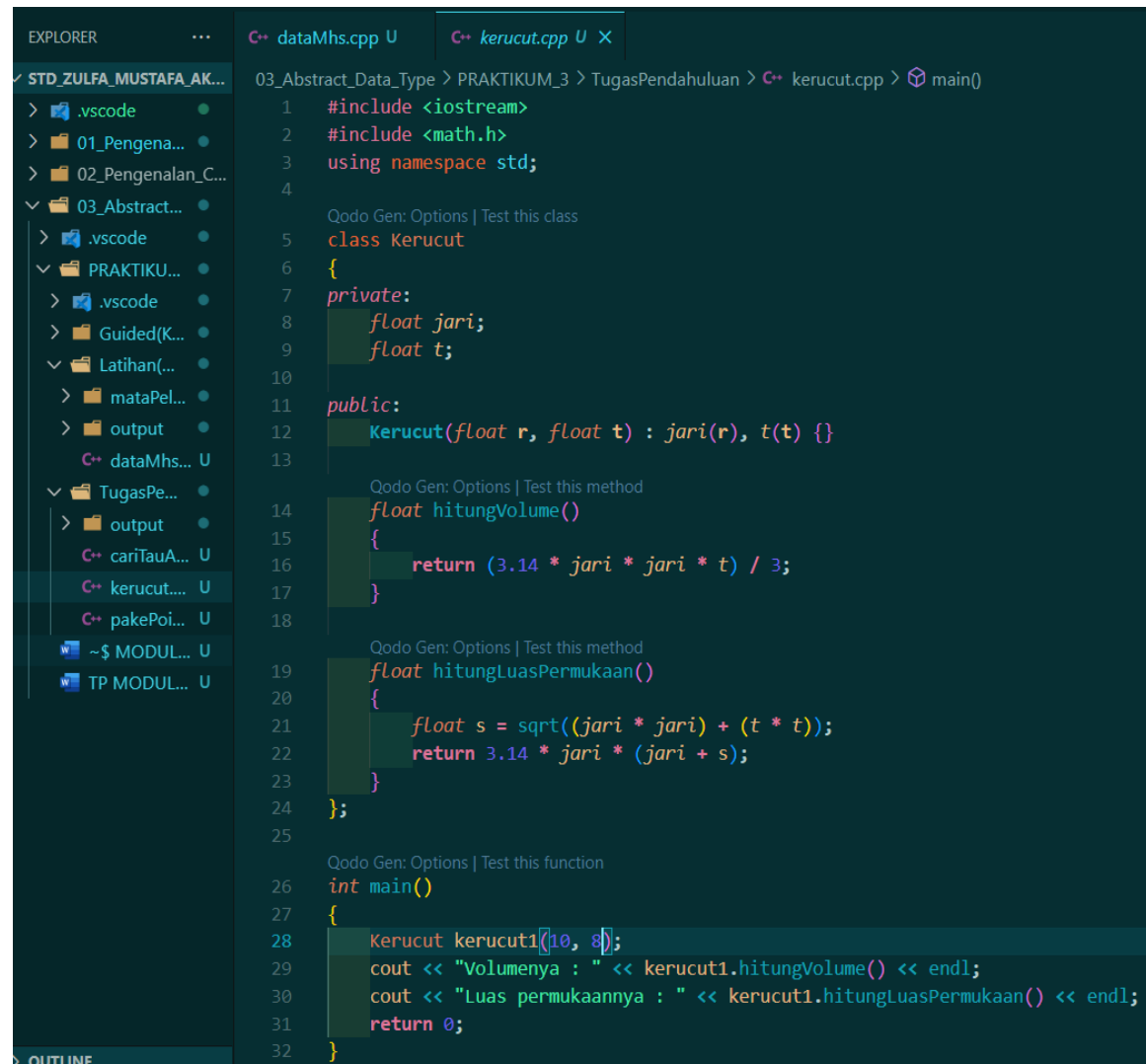
6.) Kerucut

Ini adalah program yang mengoperasikan rumus hitung dari volume dan luas permukaan kerucut. **Buat variabel float dari jari dan t sebagai private agar variabel ini tak**

bisa diakses sembarangan dikelas lain, lalu deklarasikan variabel-variabel tadi dalam satu parameter.

Selanjutnya buat rumus fungsi hitung untuk volume dan luas permukaan.

Terakhir buat output dengan pemanggilan rumus dan inputan yang akan diproses.



```
1  #include <iostream>
2  #include <math.h>
3  using namespace std;
4
5  Qodo Gen: Options | Test this class
6  class Kerucut
7  {
8  private:
9      float jari;
10     float t;
11 public:
12     Kerucut(float r, float t) : jari(r), t(t) {}
13
14     Qodo Gen: Options | Test this method
15     float hitungVolume()
16     {
17         return (3.14 * jari * jari * t) / 3;
18     }
19
20     Qodo Gen: Options | Test this method
21     float hitungLuasPermukaan()
22     {
23         float s = sqrt((jari * jari) + (t * t));
24         return 3.14 * jari * (jari + s);
25     }
26 };
27
28 Qodo Gen: Options | Test this function
29 int main()
30 {
31     Kerucut kerucut1(10, 8);
32     cout << "Volumenya : " << kerucut1.hitungVolume() << endl;
33     cout << "Luas permukaannya : " << kerucut1.hitungLuasPermukaan() << endl;
34     return 0;
35 }
```

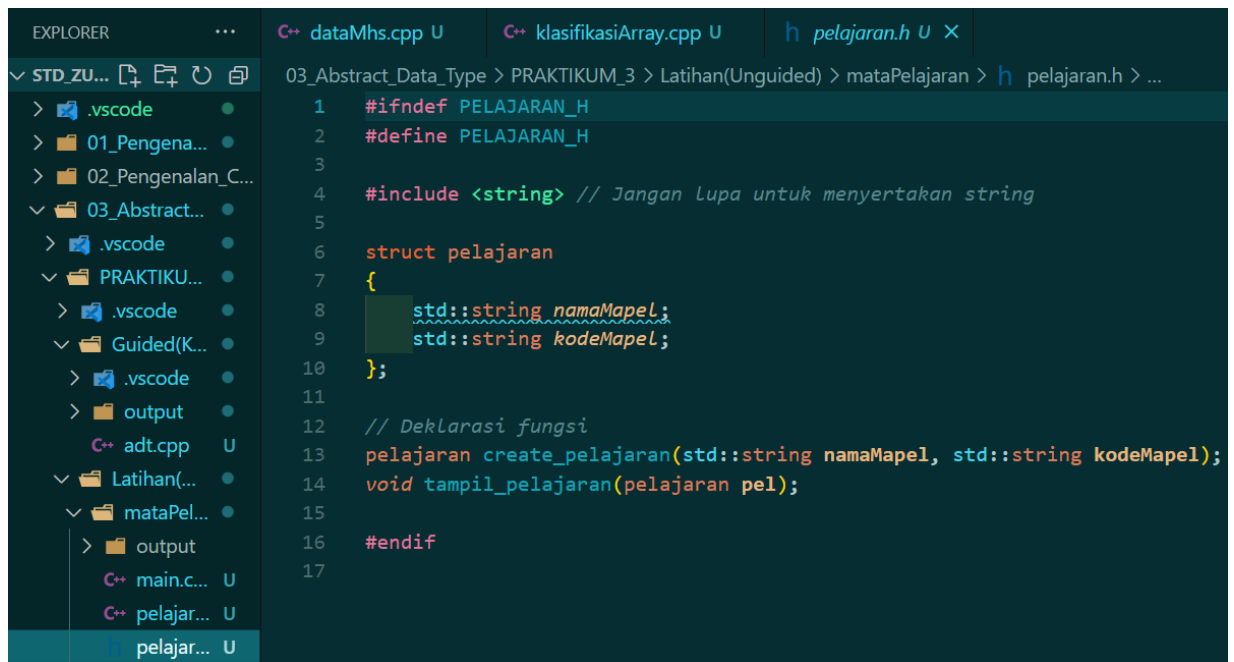
Output :

```
Volumenya : 837.333
Luas permukaannya : 716.116
```

# LATIHAN – UNGUIDED

## 1.) Mata Pelajaran

Ini adalah rangkaian program rumit yang melibatkan dua pemanggilan sub-class pada program utama 'main.cpp'. Untuk kelas pertama adalah 'pelajaran.h' yang menyimpan deklarasi variabel namaMapel dan kodeMapel. Kemudian deklarasikan fungsi 'create\_pelajaran' yang isinya pemanggilan variabel string dari dua variabel diatas dan method 'tampil\_pelajaran' yang isinya pemanggilan constructor dan gunakan definisi baru yaitu 'pel'.




The screenshot shows the Visual Studio Code interface. On the left, the Explorer sidebar displays a project structure with folders like '01\_Pengena...', '02\_Pengenalan\_C...', '03\_Abstract...', 'PRAKTIKU...', 'Guided(K...', 'Latihan(...', and 'mataPel...'. The main editor area shows the file 'pelajaran.h' with the following C++ code:

```
1  #ifndef PELAJARAN_H
2  #define PELAJARAN_H
3
4  #include <string> // Jangan Lupa untuk menyertakan string
5
6  struct pelajaran
7  {
8      std::string namaMapel;
9      std::string kodeMapel;
10 };
11
12 // Deklarasi fungsi
13 pelajaran create_pelajaran(std::string namaMapel, std::string kodeMapel);
14 void tampil_pelajaran(pelajaran pel);
15
16 #endif
17
```

Selanjutnya adalah 'pelajaran.cpp'. Ini adalah sub-class untuk memanggil variabel dan fungsi dari sub-class 'pelajaran.h'. Kemudian isi dari variabel ini adalah deklarasi variabel pelajaran dan deklarasi baru variabel namaMapel dan kodeMapel yang diinisialisasikan dengan definisi 'pel' kemudian dibuat outputnya adalah definisi 'pel'.

Kemudian buat method 'tampil\_pelajaran' yang parameternya adalah variabel pelajaran dengan output nama mapel dan kode mapelnya.



```
1  #include "pelajaran.h"
2  #include <iostream>
3
4  // Implementasi fungsi create_pelajaran
5  Qodo Gen: Options | Test this function
6  pelajaran create_pelajaran(std::string namaMapel, std::string kodeMapel)
7  {
8      pelajaran pel;
9      pel.namaMapel = namaMapel;
10     pel.kodeMapel = kodeMapel;
11     return pel;
12 }
13
14 // Implementasi fungsi tampil_pelajaran
15 Qodo Gen: Options | Test this function
16 void tampil_pelajaran(pelajaran pel)
17 {
18     std::cout << "Nama Pelajaran: " << pel.namaMapel << std::endl;
19     std::cout << "Kode Pelajaran: " << pel.kodeMapel << std::endl;
20 }
```

Terakhir adalah program utamanya, 'main.cpp'. Mulai sekarang hanya perlu memanggil variabel definisi 'pel' kemudian diintegrasikan dengan variabel baru namapel dan kodepel.

## 2.) Data Mahasiswa

Program kali ini cukup banyak dan rumit. **Kita harus buat constructor untuk deklarasi variabel nama, nim, nilai uts, nilai uas, nilai tugas, dan nilai tugas akhir. Buat rumusan untuk operasi hitung nilai akhir yang dimana kita memanggil variabel uts, uas, dan tugas untuk dioperasikan dan ditampilkan hasilnya sebagai outputnya.**

=====

Jangan lupa buat fungsi 'input\_mahasiswa' untuk menginput data-data dari nama hingga nilai akhir dari User sehingga program bisa memproses data yang diperoleh.

=====

Buat fungsi 'tampil\_mahasiswa' untuk membuat barisan output yang nantinya menampilkan keseluruhan data mahasiswa mulai dari nama sampai nilai akhir.

=====

Kita mulai memasuki operasi utama dari program ini, **kita buat deklarasi array yang dibatasi inputannya hanya 10 yang tujuannya untuk menampung total inputan yang kita masukkan.**



=====

Setelah itu buat perulangan 'while' yang akan tetap beroperasi apabila kondisi-kondisinya selalu bernilai benar. Didalamnya buat inputan untuk 10 data mahasiswa yang ingin diinput. Bila lebih dari 10 maka program akan mengeluarkan notifikasi untuk mengulangi program. **Untuk pengkondisian jika inputan gagal disini menggunakan fungsi cin.fail() dan cin.clear() untuk membersihkan kode operasi yang berantakan dan menavigasikan untuk mengulangi program.**

=====

Kemudian kita berlanjut ke perulangan berikutnya untuk fungsi 'input\_mahasiswa'. **Kita deklarasikan bahwa perulangan akan berhenti sesuai dengan jumlah inputan mahasiswa yang kita masukkan di inputan awal. Didalam perulangan ini ada output yang menampilkan data mahasiswa ke-sekian sesuai jumlah inputan. Agar program inputan untuk masing-masing tugas dapat terpenuhi, kita perlu memanggil fungsi 'input\_mahasiswa' dengan parameter mahasiswa[i] yang mengindikasikan jumlah mahasiswa yang akan diproses data-data inputannya.**

```

EXPLORE
O- dataMhs.cpp u x
O- MasfikaArray.cpp u
O- pelajaran.cpp u
STD JULIA MUSTARA AK...
> .vscode
> 01_Pengana...
> 02_Pengalan_C...
> 03_Abstract...
> .vscode
> PRAKTIKLI...
> .vscode
> GuidedY...
> LatihanU...
> mataPel...
> output
O- dataMhs... u
O- Masfika... u
> TugasPe...
> output
O- caritauA... u
O- kerucut... u
O- pakePol... u
- $ MODUL... u
TP MODUL... u

03_Abstract_Data_Type > PRAKTIKULUM_3 > Latihan(Unguided) > O- dataMhs.cpp > ...
1 #include <iostream>
2 #include <string>
3 #include <limits>
4 using namespace std;
5
6 struct Mahasiswa
7 {
8     string nama;
9     string nim;
10    float uts;
11    float uas;
12    float tugas;
13    float nilai_akhir;
14 };
15
16 // Qodo Gen: Options | test this function
17 float hitung_nilai_akhir(float uts, float uas, float tugas)
18 {
19     return 0.3 * uts + 0.4 * uas + 0.3 * tugas;
20 }
21
22 // Qodo Gen: Options | test this function
23 void input_mahasiswa(Mahasiswa &mhs)
24 {
25     cout << "Masukkan nama mahasiswa: ";
26     getline(cin, mhs.nama);
27     cout << "Masukkan NIM mahasiswa: ";
28     getline(cin, mhs.nim);
29     cout << "Masukkan nilai UTS: ";
30     cin >> mhs.uts;
31     cout << "Masukkan nilai UAS: ";
32     cin >> mhs.uas;
33     cout << "Masukkan nilai Tugas: ";
34     cin >> mhs.tugas;
35     cin.ignore();
36     mhs.nilai_akhir = hitung_nilai_akhir(mhs.uts, mhs.uas, mhs.tugas);
37 }
38
39 // Qodo Gen: Options | test this function
40 void tampil_mahasiswa(const Mahasiswa &mhs)
41 {
42     cout << "Nama          : " << mhs.nama << endl;
43     cout << "NIM           : " << mhs.nim << endl;
44     cout << "Nilai UTS      : " << mhs.uts << endl;
45     cout << "Nilai UAS      : " << mhs.uas << endl;
46     cout << "Nilai Tugas    : " << mhs.tugas << endl;
47     cout << "Nilai Akhir    : " << mhs.nilai_akhir << endl;
48     cout << "-----" << endl;
49 }
50
51 // Qodo Gen: Options | test this function
52 int main()
53 {
54     Mahasiswa mahasiswa[10];
55     int jumlah_mahasiswa;
56
57     while (true)
58     {
59         cout << "Masukkan jumlah mahasiswa (maks 10, tentukan angka): ";
60         cin >> jumlah_mahasiswa;
61
62         if (cin.fail() || jumlah_mahasiswa < 1 || jumlah_mahasiswa > 10)
63         {
64             cin.clear();
65             cin.ignore(numeric_limits<streamsize>::max(), '\n');
66             cout << "Bukan, maksud inputannya itu angka... dan gaboleh lebih dari 10" << endl;
67             cout << endl;
68         }
69         else
70         {
71             cin.ignore();
72             break;
73         }
74     }
75
76     for (int i = 0; i < jumlah_mahasiswa; i++)
77     {
78         cout << "Data mahasiswa ke-" << i + 1 << ": " << endl;
79         input_mahasiswa(mahasiswa[i]);
80     }
81
82     cout << "Data Mahasiswa yang Telah Dimasukkan:";
83     for (int i = 0; i < jumlah_mahasiswa; i++)
84     {
85         tampil_mahasiswa(mahasiswa[i]);
86     }
87
88     return 0;
89 }

```

Output :

exe'

Masukkan jumlah mahasiswa (maks 10, bentukannya angka): 3

Data mahasiswa ke-1:

Masukkan nama mahasiswa: Sung Shinhwa

Masukkan NIM mahasiswa: 2311104010

Masukkan nilai UTS: 85

Masukkan nilai UAS: 94

Masukkan nilai Tugas: 90

Data mahasiswa ke-2:

Masukkan nama mahasiswa: Sung Seoldam

Masukkan NIM mahasiswa: 2311104011

Masukkan nilai UTS: 92

Masukkan nilai UAS: 95

Masukkan nilai Tugas: 90

Data mahasiswa ke-3:

Masukkan nama mahasiswa: Nam In Hak

Masukkan NIM mahasiswa: 2311104012

Masukkan nilai UTS: 87

Masukkan nilai UAS: 91

Masukkan nilai Tugas: 84

Data Mahasiswa yang Telah Dimasukkan: Nama : Sung Shinhwa

NIM : 2311104010

Nilai UTS : 85

Nilai UAS : 94

Nilai Tugas : 90

Nilai Akhir : 90.1

-----  
Nama : Sung Seoldam

NIM : 2311104011

Nilai UTS : 92

Nilai UAS : 95

Nilai Tugas : 90

Nilai Akhir : 92.6

-----  
Nama : Nam In Hak

NIM : 2311104012

Nilai UTS : 87

Nilai UAS : 91

Nilai Tugas : 84

Nilai Akhir : 87.7  
-----

### 3.) Klasifikasi Array

Program kali ini juga sangat rumit. Ini adalah program untuk melakukan penukaran nilai antar variabel dalam array 2 dimensi. Pertama-tama kita buat beberapa method untuk mengoperasikan pertukaran nilai antar array. **Method pertama adalah method untuk menampilkan dua array 2 dimensi, yaitu 'tampilkanArray'.** **Disini kita inisialisasikan dimana dua array akan ditampilkan berderet kebawah secara teratur.**

Kemudian method selanjutnya adalah method untuk pengoperasian menukar keseluruhan nilai antar array. **Didalamnya, kita mulai buat parameter untuk array pertama dan array kedua dengan masing-masing dua [] yang melambangkan "dua dimensi".** Dalam masing-masing dimensi **diinisialisasikan lagi nilai** untuk membuat kolom dan baris, **artinya disini kita atur jadi tiga kolom dan tiga baris untuk masing-masing array-nya.**

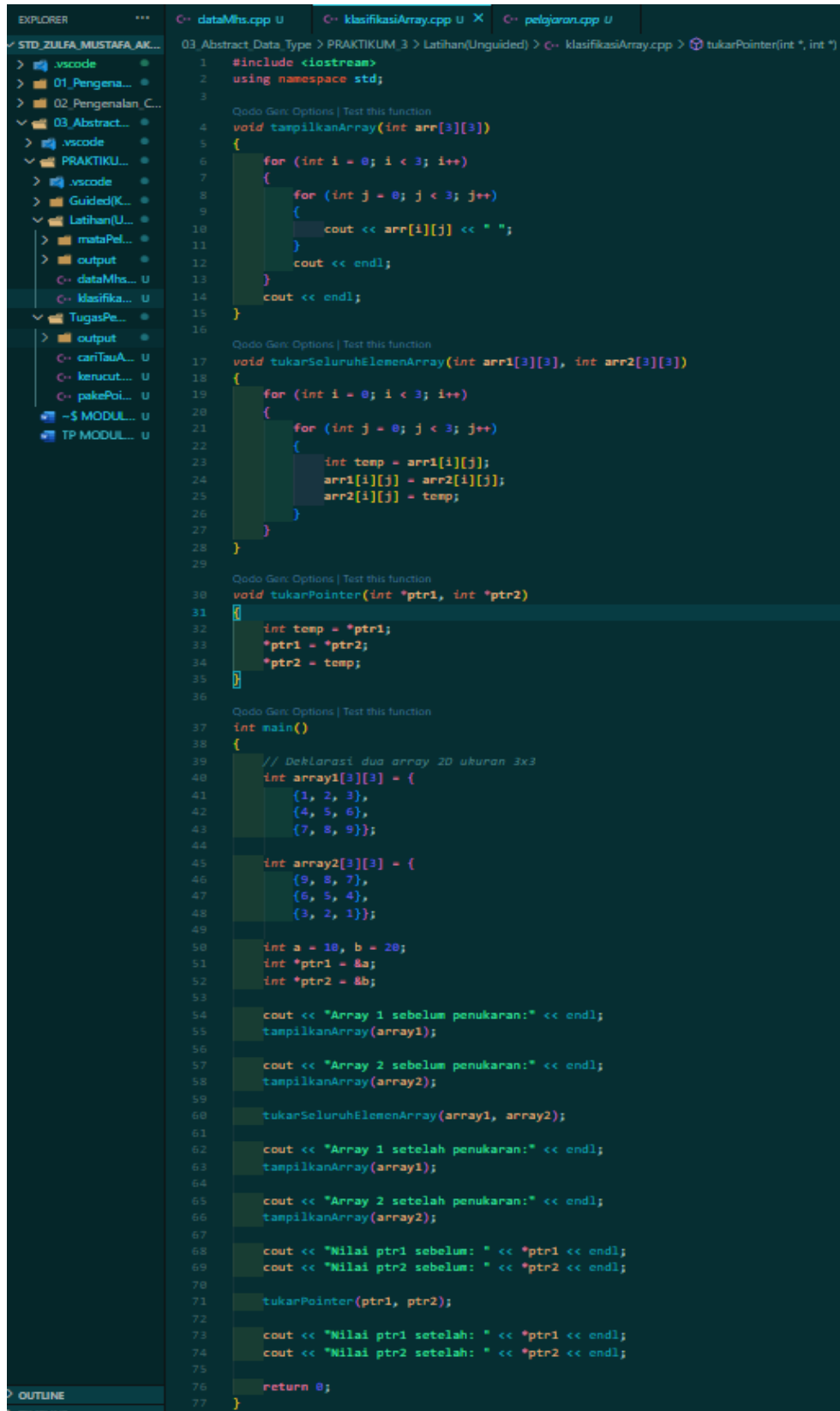
Selanjutnya adalah method untuk menukar pointer. **Kita inisiallisasikan pointer satu dan pointer dua dengan tipe data integer, lalu deklarasikan pointer satu ini sebagai variabel sementara serta pointer dua yang dijadikan variabel sementara juga.**

Kita mulai memasuki tahap operasi utama. Mulai deklarasikan dua array dua dimensi yang diatur jadi 3 kolom dan 3 baris. Untuk array1 kita ada {1,2,3,4,5,6,7,8,9}. Sedangkan untuk array2 memiliki nilai yang sama Cuma awalnya yang dibalik menjadi {9,8,7,6,5,4,3,2,1}. Jangan lupa untuk mengatur nilai pointer satu dan pointer dua dengan alamat nilai dari variabel a dan b yang sudah ada nilainya.

Buat output untuk menampilkan array1 dan array2 dengan fungsi 'tampilArray' **agar barisan data dalam array dapat dioperasikan sedemikian rupa agar tampil di interface sesuai fungsi perulangannya.**

**Selanjutnya, mulai operasi untuk menukar nilai masing-masing array agar dapat berganti posisi satu sama lain.**

Buat output hasil dari pertukaran array1 dan array2 lagi untuk mengecek apakah nilai-nilai dalam masing-masing array sudah tertukar atau belum. Buat juga output untuk nilai pointer yang sudah ditukar juga.



```
1 #include <iostream>
2 using namespace std;
3
4 void tampilkanArray(int arr[3][3])
5 {
6     for (int i = 0; i < 3; i++)
7     {
8         for (int j = 0; j < 3; j++)
9         {
10             cout << arr[i][j] << " ";
11         }
12         cout << endl;
13     }
14     cout << endl;
15 }
16
17 void tukarSeluruhElemenArray(int arr1[3][3], int arr2[3][3])
18 {
19     for (int i = 0; i < 3; i++)
20     {
21         for (int j = 0; j < 3; j++)
22         {
23             int temp = arr1[i][j];
24             arr1[i][j] = arr2[i][j];
25             arr2[i][j] = temp;
26         }
27     }
28 }
29
30 void tukarPointer(int *ptr1, int *ptr2)
31 {
32     int temp = *ptr1;
33     *ptr1 = *ptr2;
34     *ptr2 = temp;
35 }
36
37 int main()
38 {
39     // Deklarasi dua array 2D ukuran 3x3
40     int array1[3][3] = {
41         {1, 2, 3},
42         {4, 5, 6},
43         {7, 8, 9}};
44
45     int array2[3][3] = {
46         {9, 8, 7},
47         {6, 5, 4},
48         {3, 2, 1}};
49
50     int a = 10, b = 20;
51     int *ptr1 = &a;
52     int *ptr2 = &b;
53
54     cout << "Array 1 sebelum penukaran:" << endl;
55     tampilkanArray(array1);
56
57     cout << "Array 2 sebelum penukaran:" << endl;
58     tampilkanArray(array2);
59
60     tukarSeluruhElemenArray(array1, array2);
61
62     cout << "Array 1 setelah penukaran:" << endl;
63     tampilkanArray(array1);
64
65     cout << "Array 2 setelah penukaran:" << endl;
66     tampilkanArray(array2);
67
68     cout << "Nilai ptr1 sebelum: " << *ptr1 << endl;
69     cout << "Nilai ptr2 sebelum: " << *ptr2 << endl;
70
71     tukarPointer(ptr1, ptr2);
72
73     cout << "Nilai ptr1 setelah: " << *ptr1 << endl;
74     cout << "Nilai ptr2 setelah: " << *ptr2 << endl;
75
76     return 0;
77 }
```

Output :

```
Array 1 sebelum penukaran:  
1 2 3  
4 5 6  
7 8 9
```

```
Array 2 sebelum penukaran:  
9 8 7  
6 5 4  
3 2 1
```

```
Array 1 setelah penukaran:  
9 8 7  
6 5 4  
3 2 1
```

```
Array 2 setelah penukaran:  
1 2 3  
4 5 6  
7 8 9
```

```
Nilai ptr1 sebelum: 10  
Nilai ptr2 sebelum: 20  
Nilai ptr1 setelah: 20  
Nilai ptr2 setelah: 10
```

# KODINGAN BERSAMA ASPRAK (GUIDED)

```
adt.cpp

#include <iostream>
#include <conio.h>
#include <stdlib.h>

using namespace std;

struct mahasiswa
{
    char nim[10];
    int nilai1, nilai2;
};

void inputMhs(mahasiswa &m);
float ratarata(mahasiswa m);

int main()
{
    mahasiswa mhs;
    inputMhs(mhs);
    cout << "rata-rata = " << ratarata(mhs);
    return 0;
}

void inputMhs(mahasiswa &m)
{
    cout << "input nim = ";
    cin >> m.nim;
    cout << "input nilai1 = ";
    cin >> m.nilai1;
    cout << "input nilai2 = ";
    cin >> m.nilai2;
}

float ratarata(mahasiswa m)
{
    return (m.nilai1 + m.nilai2) / 2.0;
}
```



2311104010\_ZULFA MUSTAFA AKHYAR ISWAHYUDI

Han-Goo-Rang

**Semoga Selalu diberi kemudahan^^**