

**TUGAS PENDAHULUAN**  
**PERTEMUAN 3**  
**ABSTRACT DATA TYPE (ADT)**



**Nama :**

Resita Istantia Purwanto (2311104037)

**Dosen :**

Yudha Islami Sulistya

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

## Soal Tugas Pendahuluan

1. Jelaskan apa yang dimaksud dengan pointer!

Jawab: Pointer adalah variabel yang menyimpan alamat memori dari variabel lain. Pemrograman C++ menggunakan pointer untuk mengakses dan memanipulasi data yang disimpan di berbagai lokasi memori, memungkinkan manipulasi data langsung pada tingkat memori.

2. Bagaimana cara menampilkan alamat memori dari suatu variabel dalam program C++? Berikan contoh!

Jawab: Dengan cara menggunakan operator **&** yang mengembalikan alamat memori dari variabel itu.

Contoh:

```
P > C:\soal2.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int x = 8;
6      cout << "Alamat memori dari variabel x adalah: " << &x << endl;
7      return 0;
8  }
9  |
```

Output:

```
PS C:\03_Abstract_Data_Type> & 'c:\Users\Resita\.vscode\
apters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-
Out-ici05i4f.hsb' '--stderr=Microsoft-MIEngine-Error-2y
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=r
Alamat memori dari variabel x adalah: 0x5ffe9c
PS C:\03_Abstract_Data_Type>
```

3. Bagaimana cara menggunakan pointer dalam program C++? Berikan contoh cara menampilkan nilai yang tersimpan pada suatu alamat melalui pointer!

Jawab:

Langkah awal yaitu dengan mendeklarasikan pointer dan variabel yang ditunjuknya. Kemudian gunakan operator dereferensi **\*** untuk mengakses nilai yang disimpan pada alamat yang ditunjuk oleh pointer.

Contoh:

Kode:

```

TP > G+ soal3.cpp > ...
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5      int x = 30;
6      int* ptr = &x; // pointer ptr menyimpan alamat variabel x
7
8      cout << "Nilai yang ditunjuk x: " << *ptr << endl;
9      return 0;
10 }
11

```

Output:

```

Nilai yang ditunjuk x: 30
PS C:\03_Abtract_Data_Type>

```

4. Jelaskan apa yang dimaksud dengan Abstract Data Type (ADT)!

Jawab:

Abstract Data Type (ADT) adalah model matematika untuk tipe data yang ditentukan oleh perilakunya, bukan implementasinya. ADT mendefinisikan apa yang dapat dilakukan oleh data, tetapi tidak menjelaskan bagaimana data diimplementasikan.

5. Berikan contoh ilustrasi sederhana di dalam dunia nyata, tetapi di luar konteks pemrograman!

Jawab:

Handphone melakukan banyak hal penting, seperti menelepon, mengirim pesan, dan menjalankan aplikasi. Ketika menekan tombol panggil atau membuka aplikasi, prosesor telepon memproses sinyal atau data.

Pengguna Handphone hanya berinteraksi dengan antarmuka sederhana, sementara proses internal yang kompleks disembunyikan. Oleh karena itu, telepon genggam, sebagai ADT, menyediakan berbagai fungsi yang dapat pengguna manfaatkan tanpa memahami mekanismenya.

6. Tuliskan ADT dari bangun ruang kerucut dalam bahasa C++!

Jawab:

Kode:

```

1 #include <iostream>
2 #include <cmath>
3
4 class Kerucut {
5 private:
6     double jari_jari;
7     double tinggi;
8
9 public:
10    // Konstruktor
11    Kerucut(double r, double t) : jari_jari(r), tinggi(t) {}
12
13    // Fungsi untuk menghitung volume kerucut
14    double volume() {
15        return (1.0 / 3) * M_PI * jari_jari * jari_jari * tinggi;
16    }
17
18    // Fungsi untuk menghitung Luas permukaan kerucut
19    double luasPermukaan() {
20        double sisi_miring = sqrt((jari_jari * jari_jari) + (tinggi * tinggi));
21        return M_PI * jari_jari * (jari_jari + sisi_miring);
22    }
23 };
24
25 int main() {
26    Kerucut kerucut(5.0, 10.0);
27    std::cout << "Volume kerucut: " << kerucut.volume() << std::endl;
28    std::cout << "Luas permukaan kerucut: " << kerucut.luasPermukaan() << std::endl;
29    return 0;
30 }
31
32

```

Output:

```

Out-zorz4bhv.szn' '--stderr=Microsoft-MIEngine-Error-yxnbvtd
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
PS C:\03_Abstract_Data_Type> & 'c:\Users\Resita\.vscode\ext
apters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEr
Out-zorz4bhv.szn' '--stderr=Microsoft-MIEngine-Error-yxnbvtd
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Out-zorz4bhv.szn' '--stderr=Microsoft-MIEngine-Error-yxnbvtd
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Volume kerucut: 261.799
Luas permukaan kerucut: 254.16
PS C:\03_Abstract_Data_Type>

```

## Unguided

1. Kode:  
file pelajaran.h

```

    #ifndef PELAJARAN_H
    #define PELAJARAN_H

    #include <string>

    struct pelajaran {
        std::string namaMapel;
        std::string kodeMapel;
    };

    pelajaran create_pelajaran(std::string nama, std::string kode);
    void tampil_pelajaran(const pelajaran& pel);

    #endif

```

file pelajaran.cpp

```

    #include "pelajaran.h"
    #include <iostream>

    pelajaran create_pelajaran(std::string nama, std::string kode) {
        pelajaran pel;
        pel.namaMapel = nama;
        pel.kodeMapel = kode;
        return pel;
    }

    void tampil_pelajaran(const pelajaran& pel) {
        std::cout << "nama pelajaran : " << pel.namaMapel << std::endl;
        std::cout << "nilai : " << pel.kodeMapel << std::endl;
    }

```

file main.cpp

```

    #include <iostream>
    #include "pelajaran.h"

    int main() {
        std::string namapel = "Struktur Data";
        std::string kodepel = "STD";
        pelajaran pel = create_pelajaran(namapel, kodepel);
        tampil_pelajaran(pel);

        return 0;
    }

```

Output:

```
nama pelajaran : Struktur Data
nilai : STD
```

Penjelasan: Kode menjelaskan dan menggunakan struktur pelajaran untuk menyimpan informasi tentang mata pelajaran. Terdapat dua fungsi:

1. create\_pelajaran: Membuat objek pelajaran dari nama dan kode mata pelajaran.
2. tampil\_pelajaran: Menampilkan nama dan kode mata pelajaran.\

Di fungsi main, objek pelajaran dibuat dengan nama "Struktur Data" dan kode "STD", lalu ditampilkan.

## 2. Kode:

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  struct Mahasiswa {
6      string nama;
7      string nim;
8      double uts;
9      double uas;
10     double tugas;
11     double nilaiAkhir;
12
13     // Fungsi untuk menghitung nilai akhir
14     double hitungNilaiAkhir() {
15         return 0.3 * uts + 0.4 * uas + 0.3 * tugas;
16     }
17 };
18
19 int main() {
20     Mahasiswa mahasiswaArray[10];
21     int jumlahMahasiswa;
22
23     cout << "Masukkan jumlah mahasiswa (max 10): ";
24     cin >> jumlahMahasiswa;
25
26     for (int i = 0; i < jumlahMahasiswa; i++) {
27         cout << "\nMahasiswa " << (i + 1) << endl;
28         cout << "Nama: ";
29         cin >> mahasiswaArray[i].nama;
30         cout << "NIM: ";
31         cin >> mahasiswaArray[i].nim;
32         cout << "UTS: ";
33         cin >> mahasiswaArray[i].uts;
```

```

34         cout << "UAS: ";
35         cin >> mahasiswaArray[i].uas;
36         cout << "Tugas: ";
37         cin >> mahasiswaArray[i].tugas;
38         mahasiswaArray[i].nilaiAkhir = mahasiswaArray[i].hitungNilaiAkhir();
39     }
40
41     cout << "\nData Mahasiswa:\n";
42     for (int i = 0; i < jumlahMahasiswa; i++) {
43         cout << "Nama: " << mahasiswaArray[i].nama
44             << ", NIM: " << mahasiswaArray[i].nim
45             << ", Nilai Akhir: " << mahasiswaArray[i].nilaiAkhir << endl;
46     }
47
48     return 0;
49 }
50

```

Output:

```

Masukkan jumlah mahasiswa (max 10): 3

Mahasiswa 1
Nama: Resi
NIM: 12345
UTS: 90
UAS: 100
Tugas: 97

Mahasiswa 2
Nama: Izal
NIM: 23456
UTS: 75
UAS: 80
Tugas: 90

Mahasiswa 3
Nama: Lisa
NIM: 34567
UTS: 79
UAS: 90
Tugas: 87

Data Mahasiswa:
Nama: Resi, NIM: 12345, Nilai Akhir: 96.1
Nama: Izal, NIM: 23456, Nilai Akhir: 81.5
Nama: Lisa, NIM: 34567, Nilai Akhir: 85.8
PS C:\03_Abstract_Data_Type>

```

Penjelasan:

Program akan berjalan bergantung pada inputan data yang dimasukan pengguna.

Di output saya memasukan jumlah siswa 3.

1. Nama : Resi, NIM : 12345, UTS: 90, UAS: 100, Tugas: 97
2. Nama : Izal, NIM : 23456, UTS: 75, UAS: 80, Tugas: 90
3. Nama : Lisa, NIM : 34567, UTS: 79, UAS: 90, Tugas: 87

Hasil nilai akhir:

1. **Mahasiswa Resi:** Nilai Akhir= $0.3 \times 90 + 0.4 \times 100 + 0.3 \times 97 = 96.1$
2. **Mahasiswa Izal:** Nilai Akhir= $0.3 \times 75 + 0.4 \times 80 + 0.3 \times 90 = 81.5$
3. **Mahasiswa Lisa:** Nilai Akhir= $0.3 \times 79 + 0.4 \times 90 + 0.3 \times 87 = 85.8$

3. Kode:

```
1  #include <iostream>
2  using namespace std;
3
4  // Fungsi untuk menampilkan isi array 2D
5  void tampilArray(int arr[3][3]) {
6      for (int i = 0; i < 3; i++) {
7          for (int j = 0; j < 3; j++) {
8              cout << arr[i][j] << " ";
9          }
10         cout << endl;
11     }
12 }
13
14 // Fungsi untuk menukar isi array 2D pada posisi tertentu
15 void tukarIsiArray(int arr1[3][3], int arr2[3][3], int row, int col) {
16     int temp = arr1[row][col];
17     arr1[row][col] = arr2[row][col];
18     arr2[row][col] = temp;
19 }
20
21 // Fungsi untuk menukar nilai yang ditunjuk oleh dua pointer
22 void tukarPointer(int *p1, int *p2) {
23     int temp = *p1;
24     *p1 = *p2;
25     *p2 = temp;
26 }
27
28 int main() {
29     int arr1[3][3] = { {1, 2, 3}, {4, 5, 6}, {7, 8, 9} };
30     int arr2[3][3] = { {9, 8, 7}, {6, 5, 4}, {3, 2, 1} };
31
32     cout << "Array 1:" << endl;
33     tampilArray(arr1);
```

```
34     cout << "\nArray 2:" << endl;
35     tampilArray(arr2);
36
37     // Menukar isi array di posisi tertentu
38     cout << "\nMenukar isi array pada posisi [1][1]: " << endl;
39     tukarIsiArray(arr1, arr2, 1, 1);
40     cout << "Array 1:" << endl;
41     tampilArray(arr1);
42     cout << "\nArray 2:" << endl;
43     tampilArray(arr2);
44
45     // Menukar nilai yang ditunjuk oleh dua pointer
46     int a = 10, b = 20;
47     int *p1 = &a, *p2 = &b;
48     cout << "\nSebelum tukar pointer:" << endl;
49     cout << "a = " << a << ", b = " << b << endl;
50
51     tukarPointer(p1, p2);
52     cout << "Setelah tukar pointer:" << endl;
53     cout << "a = " << a << ", b = " << b << endl;
54
55     return 0;
56 }
57
58
```



Output:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\03_Abstract_Data_Type> & 'c:\Users\Resita\.vscode\extensions\ms-vscode.cpptools-1.21.6-win
apters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-uhg3x1xy.xmu' '--stdout=Micro
Out-jcte3f3r.ltl' '--stderr=Microsoft-MIEngine-Error-agfrpqtfe1v' '--pid=Microsoft-MIEngine-Pid
'--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Array 1:
1 2 3
4 5 6
7 8 9

Array 2:
9 8 7
6 5 4
3 2 1

Menukar isi array pada posisi [1][1]:
Array 1:
1 2 3
4 5 6
7 8 9

Array 2:
9 8 7
6 5 4
3 2 1

Sebelum tukar pointer:
a = 10, b = 20
Setelah tukar pointer:
a = 20, b = 10
PS C:\03_Abstract_Data_Type>
```

Penjelasan: Program ini membuat dua array 2D, dua pointer integer, serta fungsi-fungsi untuk menampilkan isi array, menukar isi array di posisi tertentu, dan menukar nilai dari dua pointer.

- tampilArray: Menampilkan isi dari array 2D.
- tukarIsiArray: Menukar isi elemen pada posisi tertentu antara arr1 dan arr2.
- tukarPointer: Menukar nilai yang ditunjuk oleh dua pointer p1 dan p2