

Nama : Muchamad Rofii

NPM : 242310030

Kelas : TI-24-PA

Matkul : Desain Analisis Algoritma

Nomor 1:

```
Class Mahasiswa.cpp
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5
6  class Mahasiswa {
7  protected:
8      string nama;
9      int umur;
10     string jurusan;
11     string fakultas;
12
13 public:
14     Mahasiswa(string nama, int umur, string jurusan, string fakultas)
15         : nama(nama), umur(umur), jurusan(jurusan), fakultas(fakultas) {}
16
17     virtual void tampilkanData() {
18         cout << "Nama      : " << nama << endl;
19         cout << "Umur       : " << umur << endl;
20         cout << "Jurusan    : " << jurusan << endl;
21         cout << "Fakultas   : " << fakultas << endl;
22     }
23
24     virtual ~Mahasiswa() {}
25 };
26
27
28 class MahasiswaAktif : public Mahasiswa {
29 public:
30     MahasiswaAktif(string nama, int umur, string jurusan, string fakultas)
31         : Mahasiswa(nama, umur, jurusan, fakultas) {}
32
33     void tampilkanData() override {
34         cout << "\n=== Mahasiswa Aktif ===\n";
35         Mahasiswa::tampilkanData();
36         cout << "Status    : Mahasiswa Aktif" << endl;
37     }
38 };
39
40
```

Class Mahasiswa.cpp

```

40
41 class Alumni : public Mahasiswa {
42 public:
43     Alumni(string nama, int umur, string jurusan, string fakultas)
44         : Mahasiswa(nama, umur, jurusan, fakultas) {}
45
46     void tampilkanData() override {
47         cout << "\n== Alumni ==\n";
48         Mahasiswa::tampilkanData();
49         cout << "Status : Alumni" << endl;
50     }
51 };
52
53
54 void tambahMahasiswa(vector<Mahasiswa*>& daftarMahasiswa) {
55     string nama, jurusan, fakultas, status;
56     int umur;
57
58     cin.ignore();
59     cout << "\nMasukkan Nama: ";
60     getline(cin, nama);
61
62     cout << "Masukkan Umur: ";
63     cin >> umur;
64     cin.ignore();
65
66     cout << "Masukkan Jurusan: ";
67     getline(cin, jurusan);
68
69     cout << "Masukkan Fakultas: ";
70     getline(cin, fakultas);
71
72     cout << "Status (Aktif/Alumni): ";
73     getline(cin, status);
74
75     if (status == "Aktif" || status == "aktif") {
76         daftarMahasiswa.push_back(new MahasiswaAktif(nama, umur, jurusan, fakultas));
77     } else {
78         daftarMahasiswa.push_back(new Alumni(nama, umur, jurusan, fakultas));
79     }
80
81     cout << "Mahasiswa berhasil ditambahkan!\n";
82

```

```

82
83
84
85 void tampilkanSemuaMahasiswa(vector<Mahasiswa*>& daftarMahasiswa) {
86     if (daftarMahasiswa.empty()) {
87         cout << "\nBelum ada data mahasiswa.\n";
88         return;
89     }
90
91     cout << "\n--- Daftar Mahasiswa ---\n";
92     for (size_t i = 0; i < daftarMahasiswa.size(); i++) {
93         daftarMahasiswa[i]->tampilkanData();
94         cout << "-----" << endl;
95     }
96 }
97
98

```

```

98
99 int main() {
100     vector<Mahasiswa> daftarMahasiswa;
101     int pilihan;
102
103     do {
104         cout << "\n== Menu ==";
105         cout << "\n1. Tambah Mahasiswa";
106         cout << "\n2. Tampilkan Semua Mahasiswa";
107         cout << "\n3. Keluar";
108         cout << "\nPilih opsi: ";
109         cin >> pilihan;
110
111         switch (pilihan) {
112             case 1:
113                 tambahMahasiswa(daftarMahasiswa);
114                 break;
115             case 2:
116                 tampilkanSemuaMahasiswa(daftarMahasiswa);
117                 break;
118             case 3:
119                 cout << "Keluar dari program.\n";
120                 break;
121             default:
122                 cout << "Pilihan tidak valid. Coba lagi.\n";
123         }
124     } while (pilihan != 3);
125
126
127     for (size_t i = 0; i < daftarMahasiswa.size(); i++) {
128         delete daftarMahasiswa[i];
129     }
130
131     return 0;
132 }

```

Select C:\Users\User\Documents\Algo smt 2\Per

=== Menu ===

1. Tambah Mahasiswa
 2. Tampilkan Semua Mahasiswa
 3. Keluar
- Pilih opsi: 2

Belum ada data mahasiswa.

=== Menu ===

1. Tambah Mahasiswa
 2. Tampilkan Semua Mahasiswa
 3. Keluar
- Pilih opsi: 1

Masukkan Nama: si
Masukkan Umur: 4
Masukkan Jurusan: TI
Masukkan Fakultas: PARI
Status (Aktif/Alumni): Aktif
Mahasiswa berhasil ditambahkan!

=== Menu ===

1. Tambah Mahasiswa
 2. Tampilkan Semua Mahasiswa
 3. Keluar
- Pilih opsi: 1

Masukkan Nama: di
Masukkan Umur: 8
Masukkan Jurusan: SI
Masukkan Fakultas: PARI
Status (Aktif/Alumni): AKTIF
Mahasiswa berhasil ditambahkan!

=== Menu ===

1. Tambah Mahasiswa
 2. Tampilkan Semua Mahasiswa
 3. Keluar
- Pilih opsi: 1

Masukkan Nama: Muchamad Rofii
Masukkan Umur: 18
Masukkan Jurusan: Teknologi Informasi
Masukkan Fakultas: Pariwisata
Status (Aktif/Alumni): Aktif
Mahasiswa berhasil ditambahkan!

=== Menu ===

1. Tambah Mahasiswa

Select C:\Users\User\Documents\Algo

=== Menu ===

1. Tambah Mahasiswa
2. Tampilkan Semua Mahasiswa
3. Keluar

Pilih opsi: 2

--- Daftar Mahasiswa ---

=== Mahasiswa Aktif ===

Nama : si
Umur : 4
Jurusan : TI
Fakultas : PARI
Status : Mahasiswa Aktif

=== Alumni ===

Jurusan : SI
Fakultas : PARI
Status : Alumni

=== Mahasiswa Aktif ===

Nama : Muchamad Rofii
Umur : 18
Jurusan : Teknologi Informasi
Fakultas : Pariwisata
Status : Mahasiswa Aktif

=== Menu ===

1. Tambah Mahasiswa
2. Tampilkan Semua Mahasiswa
3. Keluar

Pilih opsi:

Nomor 2:

Class Buku.cpp

```
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4
5  class Buku {
6  private:
7      string judul;
8      string penulis;
9      bool tersedia;
10
11 public:
12     Buku(string j, string p) : judul(j), penulis(p), tersedia(true) {}
13
14     void setJudul(string j) { judul = j; }
15     void setPenulis(string p) { penulis = p; }
16     void setTersedia(bool t) { tersedia = t; }
17
18     string getJudul() const { return judul; }
19     string getPenulis() const { return penulis; }
20     bool isTersedia() const { return tersedia; }
21
22     void pinjamBuku() {
23         if (tersedia) {
24             tersedia = false;
25             cout << "Buku " << judul << " berhasil dipinjam.\n";
26         } else {
27             cout << "Buku " << judul << " sedang tidak tersedia.\n";
28         }
29     }
30
31     void kembalikanBuku() {
32         tersedia = true;
33         cout << "Buku " << judul << " telah dikembalikan.\n";
34     }
35
36     void tampilkanInfo() const {
37         cout << "Judul: " << judul << ", Penulis: " << penulis << ", Status: " << (tersedia ? "Tersedia" : "Dipinjam") << endl;
38     }
39 };
40
```

Class Buku.cpp

```

40
41 int main() {
42     vector<Buku> perpustakaan;
43     int jumlahBuku;
44
45     cout << "Masukkan jumlah buku yang ingin ditambahkan: ";
46     cin >> jumlahBuku;
47     cin.ignore();
48
49     for (int i = 0; i < jumlahBuku; i++) {
50         string judul, penulis;
51         cout << "Masukkan judul buku ke-" << (i + 1) << ": ";
52         getline(cin, judul);
53         cout << "Masukkan penulis buku ke-" << (i + 1) << ": ";
54         getline(cin, penulis);
55         perpustakaan.push_back(Buku(judul, penulis));
56     }
57
58     int pilihan;
59     do {
60         cout << "\nMenu:\n1. Pinjam Buku\n2. Kembalikan Buku\n3. Tampilkan Daftar Buku\n4. Keluar\nPilihan: ";
61         cin >> pilihan;
62         cin.ignore();
63
64         if (pilihan == 1) {
65             if (perpustakaan.empty()) {
66                 cout << "Tidak ada buku di perpustakaan!\n";
67                 continue;
68             }
69             int index;
70             cout << "Masukkan nomor buku yang ingin dipinjam: ";
71             cin >> index;
72             if (index > 0 && index <= (int)perpustakaan.size()) {
73                 perpustakaan[index - 1].pinjamBuku();
74             } else {
75                 cout << "Nomor buku tidak valid!\n";
76             }
77         } else if (pilihan == 2) {
78             if (perpustakaan.empty()) {
79                 cout << "Tidak ada buku di perpustakaan!\n";
80                 continue;
81             }
82             int index;
83             cout << "Masukkan nomor buku yang ingin dikembalikan: ";
84             cin >> index;
85             if (index > 0 && index <= (int)perpustakaan.size()) {
86                 perpustakaan[index - 1].kembalikanBuku();
87             } else {
88                 cout << "Nomor buku tidak valid!\n";
89             }
90         } else if (pilihan == 3) {
91             if (perpustakaan.empty()) {
92                 cout << "Tidak ada buku di perpustakaan!\n";
93             } else {
94                 cout << "\nDaftar Buku di Perpustakaan:\n";
95                 for (size_t i = 0; i < perpustakaan.size(); i++) {
96                     cout << i + 1 << ". ";
97                     perpustakaan[i].tampilkanInfo();
98                 }
99             }
100         }
101     } while (pilihan != 4);
102
103     cout << "Program selesai. Terima kasih!\n";
104     return 0;
105 }

```

Select C:\Users\User\Documents\Algo smt 2\Pertemuan 3\Tugas\Class Buku.exe

Masukkan jumlah buku yang ingin ditambahkan: 3

Masukkan judul buku ke-1: 10 dosa besar

Masukkan penulis buku ke-1: nurdin

Masukkan judul buku ke-2: tere liyee

Masukkan penulis buku ke-2: random

Masukkan judul buku ke-3: atomic habits

Masukkan penulis buku ke-3: idk

Menu:

1. Pinjam Buku

2. Kembalikan Buku

3. Tampilkan Daftar Buku

4. Keluar

Pilihan: 1

Masukkan nomor buku yang ingin dipinjam: 3

Buku 'atomic habits' berhasil dipinjam.

Menu:

1. Pinjam Buku

2. Kembalikan Buku

3. Tampilkan Daftar Buku

4. Keluar

Pilihan: 3

Daftar Buku di Perpustakaan:

1. Judul: 10 dosa besar, Penulis: nurdin, Status: Tersedia

2. Judul: tere liyee, Penulis: random, Status: Tersedia

3. Judul: atomic habits, Penulis: idk, Status: Dipinjam

Menu:

1. Pinjam Buku

2. Kembalikan Buku

3. Tampilkan Daftar Buku

4. Keluar

Pilihan: 2

Masukkan nomor buku yang ingin dikembalikan: 3

Buku 'atomic habits' telah dikembalikan.

Menu:

1. Pinjam Buku

2. Kembalikan Buku

3. Tampilkan Daftar Buku

4. Keluar

Pilihan: 3

4. Keluar

Pilihan: 3

Daftar Buku di Perpustakaan:

1. Judul: 10 dosa besar, Penulis: nurdin, Status: Tersedia

2. Judul: tere liyee, Penulis: random, Status: Tersedia

3. Judul: atomic habits, Penulis: idk, Status: Tersedia

Menu:

1. Pinjam Buku

2. Kembalikan Buku

3. Tampilkan Daftar Buku

4. Keluar

Pilihan: