**Tugas Personal ke-1**

**Week 3**

**Frans Sebastian - 2502121162**

Jawablah soal berikut dengan benar!

1. Soal Essay
2. Sebutkan dan jelaskan mengenai:
3. Perbedaan *structured programming vs object-oriented programming*!

Jawab:

1. Structured Programming:
* Fokus pada konsep pemrograman prosedural, yang menekankan pada pemanggilan serangkaian prosedur atau fungsi.
* Program terstruktur secara linear, dengan fungsi-fungsi atau prosedur-prosedur yang dipanggil secara berurutan.
* Data dan fungsi dipisahkan secara jelas, dengan data disimpan di variabel yang diakses oleh fungsi-fungsi terpisah.
* Pengendalian aliran program bergantung pada struktur kontrol seperti loop, kondisional, dan pemanggilan fungsi.

Contoh:

*// Program C sederhana menggunakan pendekatan structured programming*

#include <stdio.h>

*// Deklarasi fungsi-fungsi*

int tambah(int *a*, int *b*) {

    return *a* + *b*;

}

int main() {

    int x = 5;

    int y = 3;

    int hasil = tambah(x, y);

    printf("Hasil penjumlahan: %d\n", hasil);

    return 0;

}

1. Object-Oriented Programming (OOP):
* Fokus pada konsep objek dan kelas, di mana program terdiri dari objek-objek yang merepresentasikan entitas dunia nyata.
* Setiap objek memiliki atribut (data) dan metode (fungsi) yang berkaitan dengan objek tersebut.
* Data dan fungsi yang berkaitan dengan suatu objek dikapsulasi bersama dalam objek itu sendiri, memungkinkan abstraksi yang lebih baik.
* Hubungan antara objek didefinisikan melalui konsep seperti pewarisan (inheritance), polimorfisme, dan enkapsulasi.

Contoh:

*// Program C++ sederhana menggunakan pendekatan Object-Oriented Programming*

#include <iostream>

using namespace std;

*// Deklarasi class*

class Bilangan {

private:

    int nilai;

public:

*// Konstruktor*

    Bilangan(int *n*) {

        nilai = *n*;

    }

*// Metode untuk menampilkan nilai*

    void tampilkanNilai() {

        cout << "Nilai: " << nilai << endl;

    }

};

int main() {

*// Membuat objek dari class Bilangan*

    Bilangan bilangan1(10);

    Bilangan bilangan2(20);

*// Memanggil metode untuk menampilkan nilai*

    bilangan1.tampilkanNilai();

    bilangan2.tampilkanNilai();

    return 0;

}

Dalam contoh di atas, class Bilangan merepresentasikan sebuah objek yang memiliki atribut nilai dan metode tampilkanNilai() untuk menampilkan nilai tersebut. Objek-objek bilangan1 dan bilangan2 dibuat berdasarkan class Bilangan.

1. Perbandingan antara Java, C#, C++, dan C !

Serta kelebihan dan kekurangan dari bahasa program tersebut

Jawab:

1. Java:
* Kelebihan:
	+ Portabilitas tinggi karena kode Java dapat dijalankan di berbagai platform yang mendukung Java Virtual Machine (JVM).
	+ Memiliki manajemen memori otomatis melalui pengumpulan sampah (garbage collection), yang mengurangi risiko kebocoran memori dan kesalahan alokasi memori.
* Kekurangan:
	+ Kinerja kadang kurang optimal karena program dijalankan pada mesin virtual Java (JVM), yang memperkenalkan overhead eksekusi.
	+ Terkadang diperlukan penanganan eksepsi yang memperlambat kinerja.
1. C#:
* Kelebihan:
	+ Integrasi yang baik dengan platform Windows dan kerangka .NET, memungkinkan pengembangan aplikasi yang kuat dan fleksibel di lingkungan Windows.
	+ Dukungan untuk pemrograman berorientasi objek yang kuat, termasuk fitur-fitur seperti pewarisan, polimorfisme, dan enkapsulasi.
* Kekurangan:
* Ketergantungan pada platform Windows, menyebabkan kurangnya portabilitas aplikasi yang dibangun dengan C#.
* Kurangnya dukungan untuk beberapa platform non-Windows, membatasi fleksibilitas pengembangan.
1. C++:
* Kelebihan:
* Kinerja tinggi karena mendekati bahasa mesin, memungkinkan pengembangan aplikasi yang efisien dalam penggunaan sumber daya.
* Dukungan untuk pemrograman prosedural dan berorientasi objek, memberikan fleksibilitas dalam pendekatan pengembangan.
* Kekurangan:
	+ Memiliki kurva belajar yang lebih tinggi dibandingkan bahasa lain, karena kompleksitas dan fleksibilitasnya yang tinggi.
	+ Rentan terhadap kesalahan memori dan kebocoran, karena pengelolaan memori manual yang memerlukan perhatian ekstra dari pengembang.
1. C:
* Kelebihan:
	+ Kinerja tinggi dan efisiensi memori, karena dekat dengan bahasa mesin dan kurangnya overhead yang terkait dengan abstraksi tinggi.
	+ Digunakan secara luas dalam pengembangan sistem operasi dan perangkat tertanam, karena kemampuannya untuk mengakses perangkat keras secara langsung.
* Kekurangan:
	+ Keterbatasan dalam pemrograman berorientasi objek, karena C lebih fokus pada paradigma pemrograman prosedural.
	+ Rentan terhadap kesalahan memori, karena pengelolaan memori manual yang memerlukan kehati-hatian dari pengembang.
1. *Stream input, stream output, stream manipulator* dan *stream error state*! Berikan contohnya!

Jawab:

* **Stream Input (istream):** Digunakan untuk membaca data dari sumber eksternal seperti keyboard, file, atau perangkat lain ke dalam program.

Contoh: **std::cin >> nilai**;

* **Stream Output (ostream):** Digunakan untuk menulis data dari program ke sumber eksternal seperti layar atau file.

Contoh: **std::cout << "Hello, world!" << std::endl;**

* **Stream Manipulator:** Digunakan untuk mengontrol format output dari data yang ditampilkan ke layar.

Contoh: **std::setw(10)** digunakan untuk menetapkan lebar output menjadi 10 karakter.

* **Stream Error State**: Mewakili status kesalahan pada operasi input/output. Ketika terjadi kesalahan pada operasi input/output, bit error pada objek stream diatur dan dapat diperiksa.

Contoh: **if(!std::cin)** digunakan untuk memeriksa apakah terjadi kesalahan pada operasi input.

1. Jelaskan dan berikan contoh mengenai:
2. *Class* dan *Object*!

Jawab:

* **Class:** Sebuah blueprint untuk menciptakan objek. Mendefinisikan atribut dan metode yang akan dimiliki oleh objek yang Object: Instance dari sebuah class. Merepresentasikan entitas konkret yang dibuat berdasarkan blueprint (class).

Contoh: dibuat berdasarkan class tersebut.

Contoh:

#include <string>

class Mobil {

public:

    string merek;

    string warna;

    int tahun;

};

* **Object:** Instance dari sebuah class. Merepresentasikan entitas konkret yang dibuat berdasarkan blueprint (class).

Contoh:

Mobil mobil1; *// Membuat objek mobil1 dari class Mobil*

mobil1.merek = "Toyota";

mobil1.warna = "Hitam";

mobil1.tahun = 2022;

1. *Data Member and Member Function*!

Jawab:

* Data Member: Variabel yang dimiliki oleh sebuah objek dari suatu class. Merepresentasikan atribut atau properti dari objek tersebut.

Contoh:

class Kotak {

public:

    double panjang;

    double lebar;

    double tinggi;

}

* Member Function: Fungsi yang dimiliki oleh sebuah class. Digunakan untuk melakukan operasi atau manipulasi terhadap data yang dimiliki oleh objek.

Contoh:

class Kotak {

    double volume() {

        return panjang \* lebar \* tinggi;

    }

};

1. *Namespace*!

Jawab:

* Namespace: Digunakan untuk mengelompokkan kode menjadi satu unit yang terisolasi. Mencegah konflik nama antara berbagai bagian dari program.

Contoh:

*// Deklarasi namespace*

namespace Matematika {

    int tambah(int *a*, int *b*) {

        return a + b;

    }

}

namespace Fisika {

    int tambah(int *a*, int *b*) {

        return a - b;

    }

}

int main() {

*// Memanggil fungsi tambah() dari namespace Matematika*

    Matematika::tambah(5, 3);

*// Memanggil fungsi tambah() dari namespace Fisika*

    Fisika::tambah(5, 3);

    return 0;

}

Referensi:

* Stroustrup, Bjarne. "The C++ Programming Language." Addison-Wesley Professional, 2013.
* Eckel, Bruce. "Thinking in Java." Prentice Hall, 2006.
* Albahari, Joseph, and Ben Albahari. "C# 5.0 in a Nutshell." O'Reilly Media, 2012.
* Kernighan, Brian W., and Dennis M. Ritchie. "The C Programming Language." Prentice Hall, 1988.
1. Soal Case.
2. Buatlah *code* untuk menentukan batas usia dalam pembuatan penerimaan bantuan dana ke masyarakat, adapun aturannya :
3. Input

Nama dan umur

1. Pilihan Umur
* Bayi = 0-1 tahun
* Batita = 1-3 tahun
* Balita = 3-5 tahun
* Anak-Anak = 5-12 tahun
* Remaja = 12-17 tahun
* Pemuda = 17-30 tahun
* Dewasa = 30-60 tahun

Lansia = >60 tahun

1. Dana yang diterima
* Bayi, Batita, Balita dan anak - anak = 1.000.000
* Remaja, Pemuda dan Dewasa = 1.500.000
* Lansia = 2.000.000

Jawab

Program: <https://onlinegdb.com/KoSHhdwYn>