

**LAPORAN PRAKTIKUM STRUKTUR
DATA DAN ALGORITMA**

**MODUL I
TIPE DATA**



Disusun Oleh :

Muhammad Rusdiyanto Asatman
2311102053

Dosen :

Wahyu Andi Saputra, S.Pd., M.Eng.

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2024**

A. Dasar Teori

Tipe data merupakan klasifikasi data yang memberitahu *compiler* atau *interpreter* mengenai bagaimana data tersebut akan digunakan. Tipe data memiliki beberapa jenis yang dapat dicontohkan sebagai berikut,

1. Tipe data primitif

Tipe data primitif adalah tipe data yang sudah terdefinisi secara bawaan dalam bahasa pemrograman C++. Tipe data ini digunakan untuk menyimpan nilai yang sederhana dan tidak kompleks. Berikut adalah beberapa tipe data primitif yang umum digunakan:

- `int`: Digunakan untuk menyimpan bilangan bulat seperti 1, 2, 3, dan seterusnya.
- `float`: Digunakan untuk menyimpan bilangan pecahan seperti 2.5, 3.14, dan seterusnya.
- `char`: Digunakan untuk menyimpan karakter seperti 'a', 'b', 'c', dan seterusnya.
- `bool`: Digunakan untuk menyimpan nilai boolean yang hanya memiliki dua nilai yaitu `true` dan `false`.

2. Tipe data koleksi

Tipe data koleksi adalah tipe data yang digunakan untuk mengelompokkan sekumpulan nilai. Dalam bahasa pemrograman C++, terdapat beberapa tipe data yang digunakan untuk mengelola kumpulan data. Berikut adalah beberapa tipe data koleksi yang umum digunakan:

- `Array`, adalah tipe data yang memungkinkan kita menyimpan sejumlah elemen dengan tipe data yang sama. Elemen-elemen dalam array diakses menggunakan indeks.
- `Vector`, adalah tipe data yang mirip dengan array, tetapi ukurannya dapat berubah secara dinamis. Didefinisikan menggunakan `std::vector` dari STL (Standard Template Library).
- `Map`, adalah tipe data yang menyimpan pasangan `key-value`. Setiap `key` hanya muncul sekali dalam `map`. Didefinisikan menggunakan `std::map` dari STL.

3. Tipe data abstrak

Tipe data abstrak adalah tipe data yang tidak terdefinisi secara bawaan dalam bahasa pemrograman C++. Dalam kata lain, tipe data ini bisa dideklarasikan dan diubah - ubah sesuai dengan kebutuhan. Tipe data ini digunakan untuk menyimpan data dengan struktur yang lebih kompleks, misalnya suatu kumpulan data dengan tipe data lebih dari 1.

Beberapa tipe data abstrak di C++ meliputi:

- Class merupakan blueprint atau cetak biru yang digunakan untuk menciptakan instance dari objek. Dalam class, kita dapat mendefinisikan variabel (data member) dan fungsi (method) yang terkait dengan objek tersebut. Secara default, atribut dan method dalam class adalah private.
- struct: Terdiri dari kumpulan tipe data dasar dan bisa lebih dari satu jenis. Kurang lebih sama dengan class, hanya saja, atribut dan method dalam struct bersifat public secara default.

B. Guided

Guided 1

```
#include <iostream> // memanggil library iostream untuk input
output

using namespace std; // default library std

// fungsi main, start program dari sini
int main() {
    char op; // variabel op untuk menyimpan operasi yang akan
    dilakukan pengguna
    float num1, num2; // tempat menyimpan angka satu dan angka dua

    cout << "Masukkan operator (+, -, *, /) : "; // meminta
    operator aritmatika
    cin >> op; // input operator
    cout << "Masukkan dua angka : "; // meminta dua angka
    cin >> num1 >> num2; // input dua angka
    // percabangan dari input operasi pengguna
    switch (op)
    {
        // jika operator yang diinputkan adalah +, maka perintah
        di dalam case akan dijalankan
        case '+':
            cout << "Hasil penjumlahan: " << num1 + num2; //
            tampil hasil penjumlahan
            break; // memberhentikan case supaya case lain tidak
            dijalankan
            // jika operator adalah -
        case '-':
            cout << "Hasil pengurangan: " << num1 - num2; //
            tampil hasil pengurangan
            break; //memberhentikan case
            // jika operator adalah *
        case '*':
```

```

        cout << "Hasil perkalian: " << num1 * num2; // tampil
hasil perkalian
        break; // stop case
        // jika operator adalah /
        case '/':
            // cek apakah angka kedua tidak sama dengan 0
            if (num2 != 0) {
                cout << "Hasil pembagian: " << num1 / num2; //
tampil hasil pembagian
            }
            // jika angka kedua sama dengan 0
            else {
                cout << "Error! Tidak dapat melakukan pembagian
dengan nol."; // tampil error
            }
            break; // stop case
        // selain keempat operator di atas, default akan
dieksekusi
        default:
            cout << "Error! operator is not correct."; // tampil
error
        }
        return 0; // program berhenti
    }
}

```

Screenshots Output

```

PS D:\Codes\strukdat\2> & 'c:\Users\Roesdi\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\
bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-m1f3x4i5.rfe' '--stdout=Microsoft-MIEngine-Out-fqkfa
tug.wqv' '--stderr=Microsoft-MIEngine-Error-qdf2biuk.bvk' '--pid=Microsoft-MIEngine-Pid-vbevxa1m.bvt' '--dbgExe=C
:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan operator (+, -, *, /) : -
Masukkan dua angka : 12
3
Hasil pengurangan: 9
PS D:\Codes\strukdat\2>

```

Deskripsi

Program di atas adalah program kalkulator sederhana yang dapat melakukan 4 operasi aritmatika yaitu penjumlahan, pengurangan, perkalian, dan pembagian. Program dimulai dengan meminta input operator (+, -, *, atau /) dan 2 angka dalam console. Setelah itu, program akan menjalankan kode sesuai dengan case dan operator yang diinputkan. Jika '+' maka 2 angka akan dijumlahkan. Jika '-' maka angka 1 akan dikurangi angka 2. Jika '*' maka 2 angka akan dikalikan. Jika '/' maka angka 1 akan dibagi angka 2. Sebelum angka dibagi, program akan mengecek apakah angka kedua sama dengan 0. Hal ini dilakukan untuk menghindari error division by zero. Jika operator yang dimasukan adalah selain keempat operator yang tersedia, maka program

akan menampilkan pesan error.

Guided 2

```
#include <stdio.h> // include library stdio.h, untuk struct

// membuat struct mahasiswa
struct Mahasiswa
{
    const char *name; // untuk menyimpan nama
    const char *address; // menyimpan alamat
    int age; // menyimpan umur
};

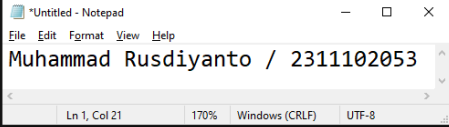
// fungsi utama, program mulai dari sini
int main() {
    struct Mahasiswa mhs1, mhs2; // membuat koleksi data mhs1 dan mhs2

    mhs1.name = "Dian"; // nama mhs1 adalah Dian
    mhs1.address = "Mataram"; // alamat mhs1 di Mataram
    mhs1.age = 22; // umur mhs1 adalah 22 tahun
    mhs2.name = "Bambang"; // nama mhs2 adalah Bambang
    mhs2.address = "Surabaya"; // alamat di Surabaya
    mhs2.age = 23; // umur 23 tahun

    // output informasi mahasiswa 1
    printf("## Mahasiswa 1 ##\n"); // header
    printf("Nama: %s\n", mhs1.name); // nama
    printf("Alamat: %s\n", mhs1.address); // alamat
    printf("Umur: %d\n", mhs1.age); // umur
    // output informasi mahasiswa 2
    printf("## Mahasiswa 2 ##\n"); // header
    printf("Nama: %s\n", mhs2.name); // nama
    printf("Alamat: %s\n", mhs2.address); // alamat
    printf("Umur: %d\n", mhs2.age); // umur
    return 0; // program selesai
}
```

Screenshots Output

```
PS D:\Codes\strukdat\2> & 'c:\Users\Roesdi\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-ufnoarqn.wvy' '--stdout=Microsoft-MIEngine-Out-2qyxbt1m.bph' '--stderr=Microsoft-MIEngine-Error-hgbssan1.hdt' '--pid=Microsoft-MIEngine-Pid-1n4c4olg.fwf' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS D:\Codes\strukdat\2>
```



Deskripsi

Program di atas adalah program sederhana yang mendemonstrasikan penggunaan Struct di C++. Dalam konteks ini, struct digunakan untuk menyimpan informasi terkait mahasiswa seperti nama, alamat, dan umur. Di dalam struct mahasiswa, terdapat tipe data `const char*` yang digunakan sebagai pengganti tipe data string. Sebagai catatan tambahan, tipe data `const char*` sebaiknya digunakan sebagai pengganti string yang tidak akan pernah diubah (immutable). Ketika eksekusi, program akan membuat 2 struct mahasiswa yaitu mhs1 dan mhs2. Nantinya, mhs1 dan mhs2 ini akan diisi (via script) sesuai dengan tipe data yang tertera di struct mahasiswa. Setelah itu, program akan menampilkan semua data yang telah diinputkan ke dalam console. Sebagai catatan tambahan, proses output program menggunakan fungsi `printf`. Fungsi tersebut adalah `cout` versi bahasa C. Fungsi tersebut memiliki sintaks yang lebih ringkas terutama untuk menampilkan output string yang berformat (formatted string).

Guided 3

```
#include <iostream> // memanggil library iostream, input output

using namespace std; // set default library ke std

// titik awal eksekusi program
int main()
{
    int nilai[5]; // buat array baru dengan 5 slot item
    nilai[0] = 23; // item index-0 bernilai 23
    nilai[1] = 50; // item index-1 bernilai 50
    nilai[2] = 34; // item index-2 bernilai 34
    nilai[3] = 78; // item index-3 bernilai 78
    nilai[4] = 90; // item index-4 bernilai 90

    // output isi array
    cout << "Isi array pertama : " << nilai[0] << endl; // output
index-0
    cout << "Isi array kedua : " << nilai[1] << endl; // output
index-1
    cout << "Isi array ketiga : " << nilai[2] << endl; // output
```

```

index-2
    cout << "Isi array keempat : " << nilai[3] << endl; // output
index-3
    cout << "Isi array kelima : " << nilai[4] << endl; // output
index-4
    return 0; // program selesai
}

```

Screenshots Output

```

PS D:\Codes\strukdat\2> & 'c:\Users\Roesdi\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\
bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-m01fztqx.qd1' '--stdout=Microsoft-MIEngine-Out-mlqhh
wii.yiw' '--stderr=Microsoft-MIEngine-Error-queych1x.lq4' '--pid=Microsoft-MIEngine-Pid-dusim1px.t54' '--dbgExe=C
:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Isi array pertama : 23
Isi array kedua : 50
Isi array ketiga : 34
Isi array keempat : 78
Isi array kelima : 90
PS D:\Codes\strukdat\2>

```

Deskripsi

Program di atas adalah program yang mendemonstrasikan penggunaan array. Dalam program ini, array digunakan untuk menyimpan 5 nilai dalam bentuk angka (integer). Pengisian semua nilai array dilakukan melalui program (via code/script). Nantinya nilai - nilai tersebut akan ditampilkan satu per satu ke terminal/console. Perlu dicatat bahwa proses input (memasukkan nilai array) dan output (menampilkan nilai array) dilakukan tanpa menggunakan looping, alias manual.

D. Unguided/Tugas

Unguided 1

```

#include <iostream> //Memanggil library iostream untuk
input/output

using namespace std; //Set default library ke std

// fungsi boolean untuk autentikasi pengguna
// parameter berupa username dan pin akun
// akan mengembalikan nilai boolean
// true jika sesuai, false jika salah
bool checkAuth(string username, int pin) {
    // cek username dan pin
    // jika username bukan bakso dan pin bukan 17701369
    // maka jalankan perintah
    if (username != "bakso" && pin != 17701369) {
        return false; // mengembalikan nilai false
    }
}

```

```

    }
    // jika username adalah bakso dan pin adalah 17701369
    // maka jalankan perintah
    else {
        return true; // mengembalikan nilai true
    }
}

// deklarasi fungsi main, program pertama kali di eksekusi dari
fungsi ini.
int main() {
    string username; // untuk menyimpan nama pengguna berupa huruf
dan angka
    int pin; // untuk menyimpan pin pengguna, hanya untuk angka

    // menampilkan interface
    cout << "=====[ Bank Abang ]=====" << endl;
    cout << "Masukkan username: "; // meminta username
    cin >> username; // input username
    cout << "Masukkan 8 angka pin: "; // meminta pin
    cin >> pin; // memasukkan pin

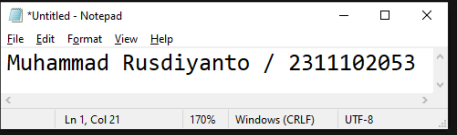
    // mengecek autentikasi pengguna dengan memanggil fungsi
checkAuth
    // jika return fungsi adalah true, maka pengguna akan disambut
    if (checkAuth(username, pin)) {
        cout << "Selamat datang, bakso!" << endl; // sambut
pengguna
    }
    // jika tidak, maka akan muncul pesan
    else {
        cout << "Kredensial salah, memberhentikan program..." <<
endl; // pesan kredensial salah
    }

    return 0; // memberhentikan program
}

```

Screenshots Output


```
PS D:\Codes\strukdat\2> & 'c:\Users\Roesdi\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\5gp.2mw' '--stderr=Microsoft-MIEngine-Error-ty2424nm.0xb' '--pid=Microsoft-MIEngine-Pid-tuieyfgn.wut' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
===== [ Bank Abang ] =====
Masukkan username: bakso
Masukkan 8 angka pin: 17701369
Selamat datang, bakso!
PS D:\Codes\strukdat\2>
```



Deskripsi:

Program di atas adalah program login sederhana. Program ini menerima username dengan tipe data string dan pin dengan tipe data integer, Kedua data tersebut akan dicek berdasarkan kredensial yang telah di-set dalam program yaitu 'bakso' untuk username dan 17701369 untuk pin. Jika input tidak sesuai (tidak sama dengan kredensial yang telah di-set), maka program akan menampilkan pesan error. Jika input sesuai dengan kredensial, maka program akan menyambut pengguna.

Tipe data primitif merupakan tipe data yang paling basic dalam pemrograman. Hal ini bisa diibaratkan sebagai pilar yang menopang dan memperkuat bangunan, dimana pilar disini adalah tipe data primitif dan bangunan adalah tipe data abstrak dan koleksi. Tanpa adanya tipe data primitif, tipe data lain tidak bisa dijalankan.

Unguided 2

```
#include <iostream> // memanggil library iostream, input output
#include <stdio.h> // memanggil library stdio, supaya bisa
menggunakan struct
#include <string> // memanggil library string, untuk menggunakan
tipe data string

using namespace std; // set default library ke std (standard
library)

// Membuat class baru dengan nama player
class player {
    // semacam flag atau identifier (tanda) yang memberitahu
    compiler bahwa
    // kode di bawah kata kunci 'PUBLIC' dapat diakses oleh semua
    fungsi.
    // ini ada hubungannya dengan OOP.
public:
    // set attribut class
    int id; // attribut id
    string name; // attribut nama pemain
    int clanId; // attribut id clan pemain

    // deklarasi method dalam class
    // untuk method di bawah, digunakan untuk menampilkan info
```

```

terkait player
    void playerInfo() {
        cout << "ID: " << id << endl; // output id pemain
        cout << "Name: " << name << endl; // output nama
pemain
        cout << "Clan ID: " << clanId << endl; // output id
clain pemain
    }
};

// Membuat struct baru dengan nama clan
struct clan {
    // set attribut dari struct
    int id; // attribut id
    string name; // attribut nama

    // set method di dalam struct
    // method ini hanya digunakan untuk mengembalikan nilai name
    string getName() {
        return name; // mengembalikan nama klan
    }
};

// deklarasi fungsi utama
int main() {
    player player1; // deklarasi player1
    clan clan1; // deklarasi clan1

    player1.id = 1; // mengisi attribut id pemain
    player1.name = "John"; // mengisi attribut nama pemain
    player1.clanId = 1; // mengisi attribut id clan pemain

    clan1.id = 1; // mengisi attribut id klan
    clan1.name = "John Clan"; // mengisi attribut nama klan

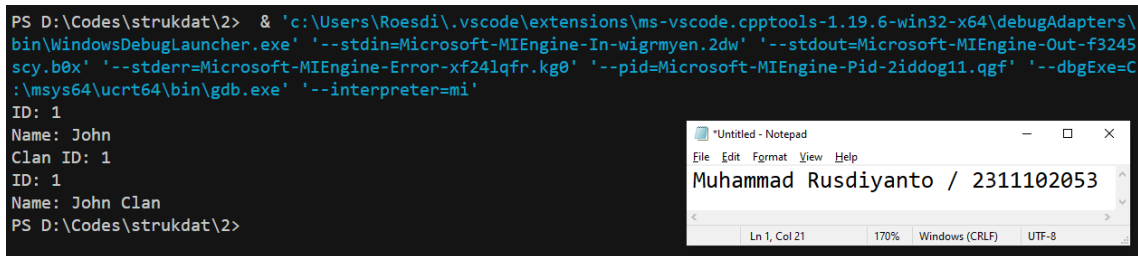
    player1.playerInfo(); // memanggil fungsi untuk menampilkan
info dari attribut player1

    cout << "ID: " << clan1.id << endl; // output id klan
    cout << "Name: " << clan1.name << endl; // output nama klan

    return 0; // memberhentikan program
}

```

Screenshots Output



```
PS D:\Codes\strukdat\2> & 'c:\Users\Roesdi\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-wigrymen.2dw' '--stdout=Microsoft-MIEngine-Out-f3245scy.b0x' '--stderr=Microsoft-MIEngine-Error-xf24lqfr.kg0' '--pid=Microsoft-MIEngine-Pid-2iddog11.qgf' '--dbgExe=C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
```

ID: 1
Name: John
Clan ID: 1
ID: 1
Name: John Clan
PS D:\Codes\strukdat\2>

Untitled - Notepad
File Edit Format View Help
Muhammad Rusdiyanto / 2311102053
Ln 1, Col 21 170% Windows (CRLF) UTF-8

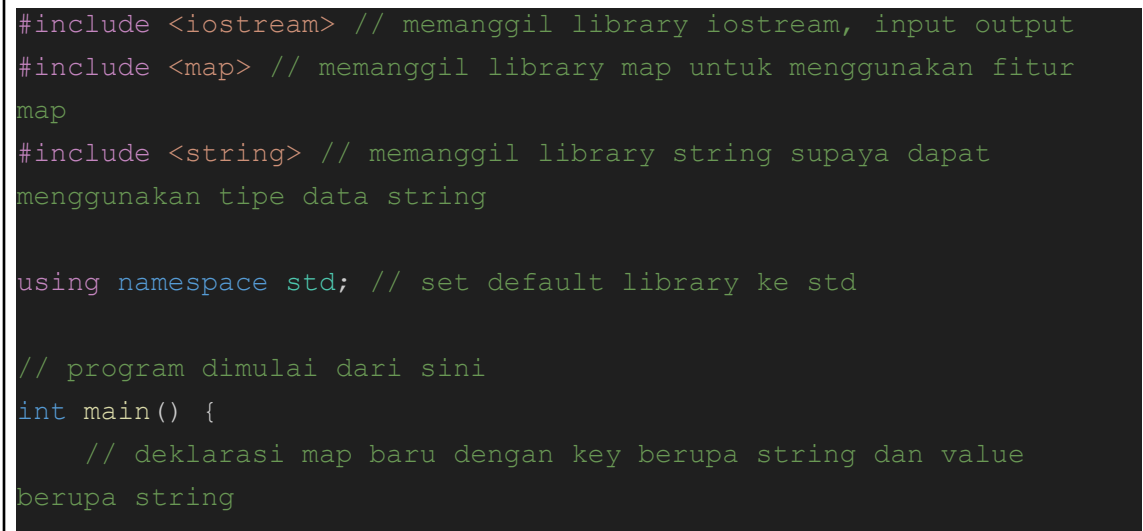
Deskripsi:

Program di atas adalah program yang menampilkan informasi player (pemain) dan klan. Informasi player disimpan sebagai class dan informasi clan disimpan sebagai struct. Data yang dapat diisi ke dalam player berupa id (int), name (string), dan clanId (int). Sementara data yang dapat ditambahkan ke dalam klan adalah id (int) dan name (string). Eksekusi program ini dimulai dari deklarasi player1 dan clan1, lalu pengisian informasi terkait keduanya (by script), dan output informasi. Untuk output informasi player1 menggunakan method yang telah disertakan di dalam class, sementara clan1 di-outputkan secara manual (memanggil atribut satu per satu).

Pada intinya, class dan struct memiliki fungsionalitas yang serupa. Namun dari keserupaan tersebut terdapat 2 perbedaan yang paling menonjol, yaitu:

1. Class bersifat private secara default, sementara Struct bersifat public secara default. Private disini berarti bahwa class/struct tidak bisa diakses di luar dari class/struct itu sendiri. Jadi ketika suatu class memiliki data nama dan sifatnya private, maka program tidak bisa mengubah atau mengisi nilai tersebut. Sementara public adalah kebalikan dari private. Jadi semua atribut dan metode dari struct/class bisa diakses secara leluasa di luar struct/class itu sendiri.
2. Class bisa menyimpan nilai NULL, sementara struct tidak bisa. Hal ini dikarenakan struct menyimpan data dalam bentuk pointer, dan pointer tidak bisa NULL.

Unguided 3



```
#include <iostream> // memanggil library iostream, input output
#include <map> // memanggil library map untuk menggunakan fitur
map
#include <string> // memanggil library string supaya dapat
menggunakan tipe data string

using namespace std; // set default library ke std

// program dimulai dari sini
int main() {
    // deklarasi map baru dengan key berupa string dan value
    berupa string
```

```

// key disini bisa diibaratkan sebagai index dalam array
// layaknya index, setiap nilai key harus unik, beda dari key
lain
map<string, string> companies;

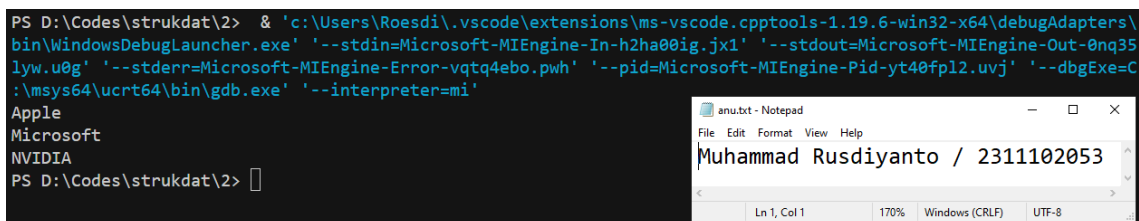
// input value ke map
companies["AAPL"] = "Apple"; // memasukkan value Apple dengan
key AAPL
companies["MSFT"] = "Microsoft"; // memasukkan value microsoft
dengan key MSFT
companies["NVDA"] = "NVIDIA"; // memasukkan value NVIDIA
dengan key NVDA

// output map value ke console
cout << companies["AAPL"] << endl; // menampilkan value map
dengan key AAPL
cout << companies["MSFT"] << endl; // menampilkan value map
dengan key MSFT
cout << companies["NVDA"] << endl; // menampilkan value map
dengan key NVDA

return 0; // memberhentikan program
}

```

Screenshots Output



```

PS D:\Codes\strukdat\2> & 'c:\Users\Roesdi\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\
bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In-h2ha00ig.jx1' '--stdout=Microsoft-MIEngine-Out-0nq35
lyw.u0g' '--stderr=Microsoft-MIEngine-Error-vqtq4ebo.pwh' '--pid=Microsoft-MIEngine-Pid-yt40fp12.uvj' '--dbgExe=C
:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Apple
Microsoft
NVIDIA
PS D:\Codes\strukdat\2>

```

Deskripsi:

Program di atas adalah program yang menampilkan nama perusahaan berdasarkan kode nasdaq perusahaan tersebut. Program ini menggunakan map dengan key berupa string dan value berupa string. Program ini dimulai dengan deklarasi map, lalu pengisian map, dan output value map. Proses pengisian (input) map dan output dilaksanakan secara manual (satu per satu via script).

Array dan map adalah dua struktur data yang sering digunakan dalam C++. Array adalah kumpulan elemen dengan tipe data yang sama, diakses melalui indeks, dan disimpan pada lokasi memori berdekatan. Di sisi lain, map adalah struktur data asosiatif yang menyimpan pasangan key-value. Setiap elemen dalam map memiliki key yang unik. Dengan map, kita dapat mencari nilai berdasarkan key tertentu. Sementara

array digunakan untuk menyimpan data terindeks, sedangkan map digunakan untuk menghubungkan key dengan nilai yang sesuai. Satu hal lagi yang membedakan map dengan array adalah map memiliki size atau ukuran yang dinamis, sementara array ukurannya statis.

E. Kesimpulan

Tipe data dalam pemrograman C++ dapat dikelompokkan menjadi tipe data primitif, tipe data koleksi, dan tipe data abstrak. Tipe data primitif mencakup int (bilangan bulat), float (bilangan pecahan), char (karakter), dan bool (nilai boolean). Tipe data koleksi melibatkan Array (kumpulan elemen dengan tipe data yang sama), Vector (mirip dengan array, ukurannya dapat berubah), dan Map (pasangan key-value). Tipe data abstrak termasuk Class (blueprint untuk menciptakan objek dengan atribut dan method) dan struct (kumpulan tipe data dasar dengan atribut dan method bersifat public secara default). Setiap tipe data memiliki peran dan fungsinya masing - masing.

F. Referensi

Asisten Praktikum, "Modul 1 Tipe Data"

Stack Overflow. (2011). "When to use const char* and when to use const char[]?", diakses dari <https://stackoverflow.com/questions/7903551/when-to-use-const-char-and-when-to-use-const-char>

Stack Overflow. (2010). "printf vs cout in C++", diakses dari <https://stackoverflow.com/questions/2872543/printf-vs-cout-in-c>

Quora. "In the C language, is it true that a struct itself, as opposed to the individual items comprising the struct, cannot be assigned to NULL? If so, why?" diakses dari <https://www.quora.com/In-the-C-language-is-it-true-that-a-struct-itself-as-opposed-to-the-individual-items-comprising-the-struct-cannot-be-assigned-to-NULL-If-so-why#:~:text=Only%20pointer%20types%20can%20be,it%20cannot%20be%20assigned%20NULL%20>.

GeeksforGeeks. "Structure vs Class in C++", diakses dari <https://www.geeksforgeeks.org/structure-vs-class-in-cpp/>

GeeksforGeeks. "Map in C++ Standard Template Library (STL)", diakses dari <https://www.geeksforgeeks.org/map-associative-containers-the-c-standard-template-library-stl/>

GeeksforGeeks. "Difference between Array and Map". diakses dari: <https://www.geeksforgeeks.org/difference-between-array-and-map/>