

**LAPORAN PRAKTIKUM STRUKTUR
DATA DAN ALGORITMA**

**MODUL I
TIPE DATA**



**Disusun Oleh :
WIDARI DWI HAYATI
2311102060**

**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 adalah adalah sebuah pengklasifikasian data berdasarkan jenis data tersebut. Tipe data dibutuhkan agar kompiler dapat mengetahui bagaimana sebuah data akan digunakan. Adapun tipe data yang akan dipelajari, sebagai berikut :

1. Tipe data Primitif
2. Tipe data Abstrak
3. Tipe data Koleksi

Tipe Data Primitif

Tipe data primitif adalah tipe data yang sudah ditentukan oleh sistem, tipe data primitif ini disediakan oleh banyak bahasa pemrograman, perbedaannya terletak pada jumlah bit yang dialokasikan untuk setiap bit pada tipe data primitif tergantung pada bahasa pemrograman, compiler dan sistem operasinya. Contoh tipe data primitif adalah :

- a. Int : adalah tipe data yang digunakan untuk menyimpan bilangan bulat seperti 12, 1, 4, dan sebagainya.
- b. Float : tipe data yang digunakan untuk menyimpan bilangan desimal seperti 1.5, 2.1, 3.14, dan sebagainya.
- c. Char : berfungsi untuk menyimpan data berupa sebuah huruf. Biasanya digunakan untuk simbol seperti A, B, C dan seterusnya.
- d. Boolean : tipe data ini digunakan untuk menyimpan nilai boolean yang hanya memiliki dua nilai yaitu true dan false.

Tipe Data Abstrak

Tipe data abstrak atau yang biasa disebut Abstrak Data Tipe (ADT) merupakan tipe data yang dibentuk oleh programer itu sendiri. Pada tipe data abstrak bisa berisi banyak tipe data, jadi nilainya bisa lebih dari satu dan beragam tipe data. Fitur Class adalah fitur Object Oriented Program (OPP) pada bahasa C++ yang mirip dengan fitur data structures Struct pada bahasa C. Keduanya berfungsi untuk membungkus tipe data di dalamnya sebagai anggota. menurut [learn.microsoft.com](https://learn.microsoft.com/en-us/cpp/faq/struct-vs-class) perbedaan antara Struct dan Class adalah pada akses defaultnya dimana Struct bersifat public dan Class bersifat private.

Tipe Data Koleksi

Tipe data koleksi (Collection Data Type) adalah tipe data yang digunakan untuk mengelompokkan dan menyimpan beberapa nilai atau objek secara bersamaan. Tipe data koleksi memungkinkan Anda menyimpan, mengelola, dan mengakses sejumlah besar data dengan cara yang terstruktur. Ada beberapa tipe data koleksi yang umum digunakan dalam pemrograman, dan di antaranya adalah:

- a. Array : Array adalah struktur data statis yang menyimpan elemen-elemen dengan tipe data yang sama. Elemen-elemen tersebut dapat diakses dengan menggunakan indeks. Array memiliki ukuran tetap yang ditentukan saat deklarasi.
- b. Vector : Vector adalah Standard Template Library (STL) jika di dalam C/C++ memiliki bentuk `std::vector`. Umumnya, vector mirip seperti array yang memiliki

kemampuan untuk menyimpan data dalam bentuk elemen-elemen yang alokasi memorinya dilakukan otomatis dan bersebelahan. Kemampuan vector bukan hanya pada jumlah elemen yang dinamis, vector pada C/C++ juga dilengkapi dengan fitur-fitur pelengkap seperti element access, iterators, capacity, modifiers.

- c. Map : Map terasa mirip dengan array namun dengan index yang memungkinkan untuk berupa tipe data selain integer. Pada map, indeks tersebut diberi nama “key”. Pada std::map digunakan Self-Balancing Tree khususnya Red-Black Tree.

B. Guided

Guided 1

Tipe Data Primitif

```
#include <iostream>
using namespace std;
// Main program
int main()
{
    char op;
    float num1, num2;

    // Meminta pengguna untuk memasukkan operator: +, -, *, /
    cout << "Masukkan operator (+, -, *, /): ";
    cin >> op;

    // Meminta pengguna untuk memasukkan operand
    cout << "Masukkan dua angka: ";
    cin >> num1 >> num2;

    // Switch statement begin
    switch (op)
    {
        // If user enter +
        case '+':
            cout << "Hasil penjumlahan: " << num1 + num2;
            break;
        // If user enter -
        case '-':
            cout << "Hasil pengurangan: " << num1 - num2;
            break;
        // If user enter *
        case '*':
            cout << "Hasil perkalian: " << num1 * num2;
            break;
        // If user enter /
        case '/':
            if (num2 != 0)
                cout << "Hasil pembagian: " << num1 / num2;
```

```

else
    cout << "Error! Tidak dapat melakukan pembagian dengan nol.";
    break;
// If the operator is other than +, -, * or /,
// error message will display
default:
    cout << "Error! operator is not correct";
} // switch statement ends
return 0;
}

```

Screenshots Output

```

Masukkan operator (+, -, *, /): +
Masukkan dua angka: 6 8
Hasil penjumlahan: 14
PS C:\Users\acer> & 'c:\Users\acer\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In
-efvtrwng.3sz' '--stdout=Microsoft-MIEngine-Out-rvegbf0m.wk4' '--stderr=Microsoft-MIEngine-Error-3xrjphxt.r0s' '--pid=Microsoft-MIEngine-Pid-zjsgrmts.wi4' '--dbgExe=
C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan operator (+, -, *, /): -
Masukkan dua angka: 78 54
Hasil pengurangan: 24
PS C:\Users\acer> & 'c:\Users\acer\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In
-phy3u2qs.phs' '--stdout=Microsoft-MIEngine-Out-5kbqbwzj.iil' '--stderr=Microsoft-MIEngine-Error-aufxtvde.02j' '--pid=Microsoft-MIEngine-Pid-ycjvyv05.bep' '--dbgExe=
C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan operator (+, -, *, /): *
Masukkan dua angka: 6 9
Hasil perkalian: 54
PS C:\Users\acer> & 'c:\Users\acer\.vscode\extensions\ms-vscode.cpptools-1.19.6-win32-x64\debugAdapters\bin\WindowsDebugLauncher.exe' '--stdin=Microsoft-MIEngine-In
-n251rjmh.52h' '--stdout=Microsoft-MIEngine-Out-qv1ypujh.qqa' '--stderr=Microsoft-MIEngine-Error-blvv4lhx.c4p' '--pid=Microsoft-MIEngine-Pid-mbaxhxcu.jos' '--dbgExe=
C:\msys64\ucrt64\bin\gdb.exe' '--interpreter=mi'
Masukkan operator (+, -, *, /): /
Masukkan dua angka: 81 6
Hasil pembagian: 13.5
PS C:\Users\acer>

```

*Untitled - Notepad
 File Edit Format View Help
 Nama : Widari Dwi Hayati
 NIM : 2311102060
 Kelas : IF-11-B

Deskripsi:

Program di atas adalah contoh program dengan tipe data primitif char dan float. Program di atas meminta pengguna untuk memasukkan operator (+, -, *, /), meminta pengguna memasukkan operand berupa dua angka. Dimana jika pengguna memasukkan operator '+', program akan menampilkan output hasil penjumlahan dengan rumus yang sudah ditambahkan dalam program. Begitu pula apabila pengguna memasukkan operator yang lainnya, program akan menampilkan output dengan rumusnya masing-masing.

Guided 2

Tipe Data Abstrak

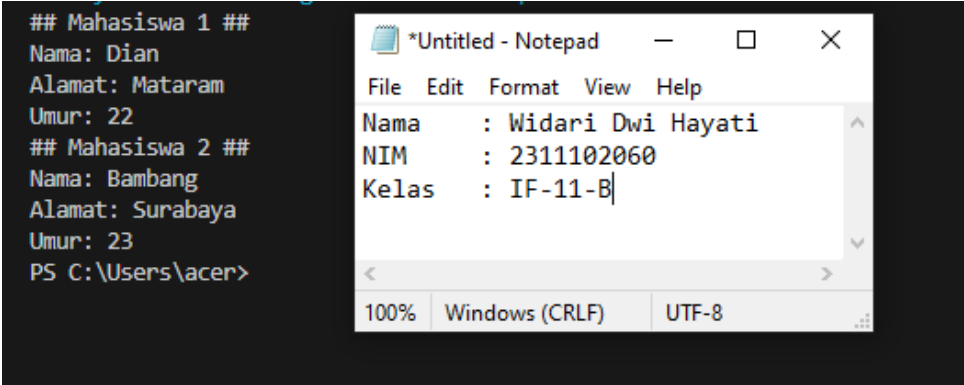
```
#include <stdio.h>

//Struct
struct Mahasiswa
{
    const char *name;
    const char *address;
    int age;
};

int main()
{
    // Menggunakan struct
    struct Mahasiswa mhs1, mhs2;
    // Mengisi nilai ke struct
    mhs1.name = "Dian";
    mhs1.address = "Mataram";
    mhs1.age = 22;
    mhs2.name = "Bambang";
    mhs2.address = "Surabaya";
    mhs2.age = 23;

    // Mencetak isi struct
    printf("## Mahasiswa 1 ##\n");
    printf("Nama: %s\n", mhs1.name);
    printf("Alamat: %s\n", mhs1.address);
    printf("Umur: %d\n", mhs1.age);
    printf("## Mahasiswa 2 ##\n");
    printf("Nama: %s\n", mhs2.name);
    printf("Alamat: %s\n", mhs2.address);
    printf("Umur: %d\n", mhs2.age);
    return 0;
}
```

Screenshots Output



The screenshot displays two windows. On the left is a terminal window with the following output:

```
## Mahasiswa 1 ##
Nama: Dian
Alamat: Mataram
Umur: 22
## Mahasiswa 2 ##
Nama: Bambang
Alamat: Surabaya
Umur: 23
PS C:\Users\acer>
```

On the right is a Notepad window titled '*Untitled - Notepad' with the following text:

```
Nama      : Widari Dwi Hayati
NIM       : 2311102060
Kelas    : IF-11-B
```

The Notepad window also shows a status bar at the bottom with '100%', 'Windows (CRLF)', and 'UTF-8'.

Deskripsi:

Program di atas adalah program yang menggunakan tipe data abstrak berupa struct. Dimana di dalam struct dengan variabel mahasiswa terdapat beberapa anggota seperti name, address, dan age. Dalam main() struct mahasiswa dideklarasikan sebagai mhs1, mhs2 dan struct diberi nilai dengan menambahkan tanda titik '.' kemudian program mencetak isi struct menggunakan printf.

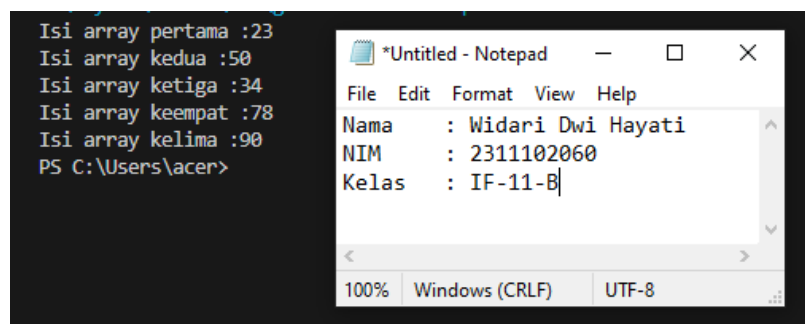
Guided 3

Tipe Data Koleksi

```
#include <iostream>
using namespace std;
int main()
{
    //deklarasi dan inisialisasi array
    int nilai[5];
    nilai[0] = 23;
    nilai[1] = 50;
    nilai[2] = 34;
    nilai[3] = 78;
    nilai[4] = 90;

    //mencetak array
    cout << "Isi array pertama :" << nilai[0] << endl;
    cout << "Isi array kedua :" << nilai[1] << endl;
    cout << "Isi array ketiga :" << nilai[2] << endl;
    cout << "Isi array keempat :" << nilai[3] << endl;
    cout << "Isi array kelima :" << nilai[4] << endl;
    return 0;
}
```

Screenshots Output



Deskripsi:

Program di atas menggunakan tipe data koleksi yaitu array. Dimana program diawali dengan mendeklarasikan array dengan variabel nilai sejumlah 5 elemen sebagai integer dan menginisialisasi setiap elemen array nilai, kemudian program mencetak array menggunakan cout.

C. Unguided/Tugas

Unguided 1

Buatlah program menggunakan tipe data primitif minimal dua fungsi dan bebas. Menampilkan program, jelaskan program tersebut dan ambil kesimpulan dari materi tipe data primitif!

```
#include <iostream>
using namespace std;

int sisi, pilih;
float alas, tinggi, jari2;
char huruf;

int main()
{
    do
    {
        cout << "\n1. Menghitung Luas Persegi";
        cout << "\n2. Menghitung Luas Segitiga";
        cout << "\n3. Menghitung Luas Lingkaran";
        cout << "\nMasukkan pilihan (1,2,3): ";
        cin >> pilih;
        switch (pilih)
        {
            case 1:
                cout << "\nLuas Persegi" << endl;
                cout << "=====" << endl;
                cout << "Masukkan sisi persegi: ";
                cin >> sisi;
                cout << "Luas Persegi: " << sisi*sisi << endl;
                cout << endl;
                break;
            case 2:
                cout << "\nLuas Segitiga" << endl;
                cout << "=====" << endl;
                cout << "Masukkan alas segitiga: ";
                cin >> alas;
                cout << "Masukkan tinggi segitiga: ";
                cin >> tinggi;
                cout << "Luas segitiga: " << alas*tinggi/2 << endl;
                cout << endl;
                break;
            case 3:
                cout << "\nLuas Lingkaran" << endl;
                cout << "=====" << endl;
                cout << "Masukkan jari-jari lingkaran: ";
                cin >> jari2;
                cout << "Luas lingkaran: " << 3.14*jari2*jari2 << endl;
                cout << endl;
        }
    }
}
```

```

        break;
    default:
        cout << "pilihan tidak valid";
    }
    cout << "Apakah Anda ingin mengulangi operasi (y,n): ";
    cin >> huruf;
} while (huruf == 'y');

if (huruf != 'y')
{
    cout << "Terima Kasih";
}

return 0;
}

```

Screenshots Output

The screenshot shows the output of a C++ program that calculates the area of a square, triangle, or circle. The program prompts the user to choose an option (1, 2, or 3). In the first run, option 1 is chosen, and the user enters a side length of 5, resulting in a square area of 25. The program then asks if the user wants to repeat the operation, and 'y' is entered. In the second run, option 2 is chosen, and the user enters a base of 7 and a height of 5, resulting in a triangle area of 17.5. The program again asks if the user wants to repeat, and 'y' is entered. In the third run, option 3 is chosen, and the user enters a radius of 7, resulting in a circle area of 153.86. Finally, the user enters 'n' when asked to repeat, and the program displays "Terima Kasih".

```

1. Menghitung Luas Persegi
2. Menghitung Luas Segitiga
3. Menghitung Luas Lingkaran
Masukkan pilihan (1,2,3): 1

Luas Persegi
=====
Masukkan sisi persegi: 5
Luas Persegi: 25

Apakah Anda ingin mengulangi operasi (y,n): y

1. Menghitung Luas Persegi
2. Menghitung Luas Segitiga
3. Menghitung Luas Lingkaran
Masukkan pilihan (1,2,3): 2

Luas Segitiga
=====
Masukkan alas segitiga: 7
Masukkan tinggi segitiga: 5
Luas segitiga: 17.5

Apakah Anda ingin mengulangi operasi (y,n): y

1. Menghitung Luas Persegi
2. Menghitung Luas Segitiga
3. Menghitung Luas Lingkaran
Masukkan pilihan (1,2,3): 3

Luas Lingkaran
=====
Masukkan jari-jari lingkaran: 7
Luas lingkaran: 153.86

Apakah Anda ingin mengulangi operasi (y,n): n
Terima Kasih

```

The Notepad window shows the following text:

```

*Untitled - Notepad
File Edit Format View Help
Nama : Widari Dwi Hayati
NIM : 2311102060
Kelas : IF-11-B

```


Deskripsi:

Program di atas menggunakan tipe data primitif berupa integer, float, dan character. Dimana program menampilkan tiga opsi dan meminta pengguna memasukkan pilihan berupa integer. Jika pengguna memasukkan pilihan 1 program akan menampilkan operasi penghitungan luas persegi, berlaku juga untuk opsi yang lainnya. Program akan mengulang ke pemilihan opsi jika pengguna menginputkan 'y' di "Apakah Anda ingin mengulangi operasi (y,n): ", program akan berakhir jika pengguna menginputkan 'n' dan menampilkan "Terima Kasih". Dapat disimpulkan bahwa tipe data primitif digunakan di berbagai bahasa pemrograman untuk menyimpan data sesuai jumlah bit masing-masing.

Unguided 2

Jelaskan fungsi dari class dan struct secara detail dan berikan contoh programnya.

```
#include <iostream>
using namespace std;

class lampu
{
public:
    string task;
    double jam;
};

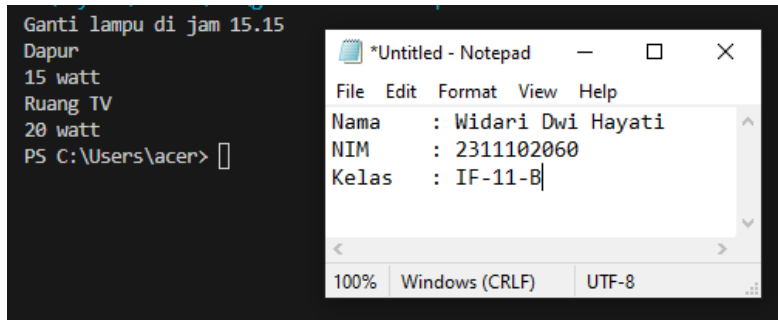
struct
{
    string tempat;
    double watt;
}lampu1, lampu2;

int main()
{
    lampu tugas;
    tugas.task = "Ganti lampu";
    tugas.jam = 15.15;
    lampu1.tempat = "Dapur";
    lampu1.watt = 15;
    lampu2.tempat = "Ruang TV";
    lampu2.watt = 20;

    cout << tugas.task << " " << "di jam " << tugas.jam << endl;
    cout << lampu1.tempat << "\n" << lampu1.watt << " watt" << endl;
    cout << lampu2.tempat << "\n" << lampu2.watt << " watt";

    return 0;
}
```

Screenshots Output



Deskripsi:

Program di atas menggunakan tipe data abstrak berupa struct dan class. Dimana struct dan class memiliki fungsi untuk mendefinisikan tipe data dan dapat menggabungkan beberapa variabel menjadi satu. Keduanya memiliki perbedaan dimana struct secara default, semua anggota dapat diakses secara langsung dari luar struct sedangkan class secara default, semua anggota adalah private yang berarti mereka tidak dapat diakses secara langsung dari luar class dan untuk membaca dan menulis nilai anggota harus menyediakan metode public.

Unguided 3

Buat dan jelaskan program menggunakan fungsi map dan jelaskan perbedaan dari array dengan map.

```
#include <iostream>
#include <map>
using namespace std;

int main()
{
    map <int, string> mahasiswa;

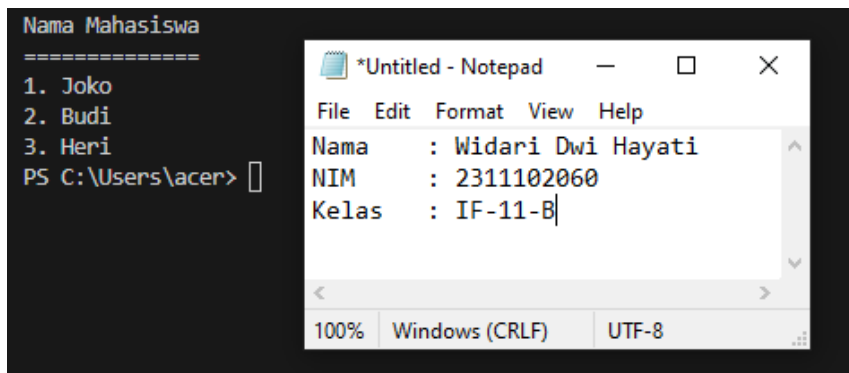
    mahasiswa.insert(make_pair(1, "Joko"));
    mahasiswa.insert(make_pair(2, "Budi"));
    mahasiswa.insert(make_pair(3, "Heri"));

    cout << "Nama Mahasiswa" << endl;
    cout << "=====" << endl;

    for (map <int, string>::iterator it = mahasiswa.begin(); it != mahasiswa.end(); ++it)
    {
        cout << it->first << ". " << it->second << endl;
    }

    return 0;
}
```

Screenshots Output

The screenshot shows a Java program's output in a console window. The output lists three student names: 1. Joko, 2. Budi, and 3. Heri. Below the list, the command prompt shows the current directory as C:\Users\acer>. Overlaid on the console is a Notepad window titled '*Untitled - Notepad'. The Notepad window contains the following text: Nama : Widari Dwi Hayati, NIM : 2311102060, and Kelas : IF-11-B. The Notepad window also shows a menu bar with File, Edit, Format, View, and Help, and a status bar at the bottom indicating 100% zoom, Windows (CRLF) line endings, and UTF-8 encoding.

Deskripsi:

Program di atas menggunakan tipe data koleksi berupa map. Map adalah kontainer yang menyimpan pasangan key-value. Dalam program di atas, map digunakan untuk menyimpan pasangan antara nomor mahasiswa (key) dan nama mahasiswa (value), mendeklarasikan variabel mahasiswa dengan tipe data map dan key-nya adalah int dan value-nya adalah string, memasukkan data mahasiswa ke dalam kontainer mahasiswa menggunakan fungsi insert, kemudian program mencetak nama-nama mahasiswa yang telah disimpan dan menggunakan iterator untuk mengiterasi seluruh elemen pada kontainer mahasiswa.

Perbedaan antara array dan map adalah sebagai berikut:

- Array adalah kontainer yang menyimpan elemen-elemen dengan tipe data yang sama dan indexnya dimulai dari 0. Sedangkan map adalah kontainer yang menyimpan pasangan key-value.
- Array tidak dapat menambah atau menghapus elemen saat runtime, sedangkan map dapat menambah atau menghapus elemen saat runtime.
- Array memiliki efisiensi yang lebih tinggi daripada map karena tidak memerlukan alokasi memori dinamis. Namun, map lebih fleksibel karena dapat menyimpan pasangan key-value.
- Array hanya dapat diakses menggunakan indeks, sedangkan map dapat diakses menggunakan key

D. Kesimpulan

Tipe data adalah pengklasifikasi data berdasarkan jenis data tersebut, yang dibutuhkan agar kompiler dapat mengetahui bagaimana sebuah data akan digunakan. Ada tiga jenis tipe data yang akan dipelajari, yaitu tipe data primitif, tipe data abstrak, dan tipe data koleksi.

Tipe data primitif adalah tipe data yang sudah ditentukan oleh sistem, seperti int, float, char, dan boolean. Tipe data abstrak atau ADT adalah tipe data yang dibentuk oleh programmer itu sendiri, yang dapat berisi banyak tipe data dan nilainya bisa lebih dari satu

dan beragam tipe data. Tipe data koleksi adalah tipe data yang digunakan untuk mengelompokkan dan menyimpan beberapa nilai atau objek secara bersamaan, seperti array, vector, dan map. Array dan map memiliki beberapa perbedaan.

Dalam C++, tipe data abstrak dapat didefinisikan menggunakan struct dan class, yang memiliki perbedaan dalam akses defaultnya. Struct bersifat public, sedangkan class bersifat private. Tipe data primitif digunakan di berbagai bahasa pemrograman untuk menyimpan data sesuai jumlah bit masing-masing. Tipe data koleksi, seperti map, dapat digunakan untuk menyimpan pasangan key-value dan dapat diakses menggunakan key.

E. Referensi (APA)

Stroustrup, B. (2021). A Tour of C++ (2nd ed.). Addison-Wesley Professional.

Lippman, S. L., Lajoie, J., & Moo, B. (2020). C++ Primer (6th ed.). Addison-Wesley Professional.

Malik, D. S. (2014). C++ Programming: From Problem Analysis to Program Design (7th ed.). Cengage Learning.