

LAPORAN PRAKTIKUM PEMROGRAMAN

PERTEMUAN 5

Ifham Syafwan Fikri

NIM(24/545184/PA/23161)

KOM B

Dosen Pengampu: Muhammad Husni Santriaji

Code: <https://github.com/FikriSyafwan/Penugasan-Praktikum-Pemrograman>

Problem 1: “FizzBuzz” with Vectors

Program ini menghasilkan nilai dari 1 hingga 100 dan menggantikan angka dengan kelipatan 3 berupa “Fizz”, angka kelipatan 5 berupa “Buzz”, serta angka kelipatan 3 dan 5 berupa “FizzBuzz”. Berikut merupakan struktur dan tugas dari program:

1. Menuliskan `#include <iostream>`, `<vector>`, dan `<string>`

```
1  #include <iostream>
2  #include <vector>
3  #include <string>
4  using namespace std;
```

- *iostream*: untuk input/output dari library.
- *vector*: untuk mengaktifkan penggunaan vector pada program.
- *string*: untuk menyimpan teks.

2. Fungsi main / *main function*

```
6  int main() {
7      vector<string> result;
```

Vector dengan tipe data string dibuat untuk menyimpan hasil dari program. Setiap entri dapat berupa angka ataupun string khusus(“Fizz”, “Buzz”, dan “FizzBuzz”).

3. Set perulangan dari angka 1 hingga 100

```
10      for (int i = 1; i <= 100; ++i) {
```

Perulangan ini akan mengulang operasi dimulai dengan angka 1 hingga menyentuh angka 100.

4. Menyeleksi menggunakan for, else if, dan else

```
9      for (int i = 1; i <= 100; ++i) {
0          if (i % 3 == 0 && i % 5 == 0) {
1              result.push_back("FizzBuzz");
2          } else if (i % 3 == 0) {
3              result.push_back("Fizz");
4          } else if (i % 5 == 0) {
5              result.push_back("Buzz");
6          } else {
7              result.push_back(to_string(i));
8          }
9      }
```

- Apabila angka i dapat dibagi atau kelipatan dari 3 dan 5, program akan mengeluarkan "FizzBuzz" kemudian menyimpannya pada hasil vector.
- Apabila angka i dapat dibagi atau kelipatan 3, program akan mengeluarkan "Fizz".
- Apabila angka i dapat dibagi atau kelipatan 5, program akan mengeluarkan "Buzz".
- Apabila angka i bukan merupakan salah satu dari angka di atas, program akan mengeluarkan angka i sendiri sebagai string (to_string(i)).

5. Print hasil dari program

```
21      for (const auto& entry : result) {
22          cout << entry << endl;
23      }
```

Perulangan ini akan berjalan untuk tiap entri dalam hasil vector dan mencetaknya ke console.

6. Fungsi return 0;

```
25      return 0;
26  }
27  }
```

"Return 0;" artinya program akan mengembalikan (return) nilai 0 ke operating system yang menjalankan program tersebut.

Berikut hasil dari program yang akan dikeluarkan:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

m1 } ; if ($?) { .\Problem1 }
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
```

```
FizzBuzz
91
92
Fizz
94
Buzz
Fizz
97
98
Fizz
Buzz
```

Problem 2: Reverse Array

Program ini akan membalikkan array dari input integer tertentu. Sebagai contoh dari input array [7, 5, 3, 2, 9], maka *output*-nya akan keluar menjadi [9, 2, 3, 5, 7].

1. Menuliskan `#include <iostream>` dan `<vector>`

```
1  #include <iostream>
2  #include <vector>
3  using namespace std;
```

- *iostream*: untuk input/output dari library
- *vector*: untuk mengaktifkan penggunaan vector pada program
- *using namespace std*: berarti seluruh baris dari program semua secara default akan menggunakan kelas, metode, atribut dari *namespace std*.

2. Fungsi untuk membalik vector

```
5  vector<int> reverseVector(const vector<int>& input) {
6      vector<int> reversed;
7
8      for (int i = input.size() - 1; i >= 0; --i) {
9          reversed.push_back(input[i]);
10     }
11
12     return reversed;
13 }
```

- *Vector<int> reverseVector()*: Fungsi ini mengubah vector yang telah diinput menjadi vector baru dengan elemennya yang telah dibalik.
- *Const vector<int>& input*: vector yang telah diinput akan menjadi constant reference supaya tidak di-copy ulang.
- *Vector<int> reversed*: vector kosong yang berfungsi untuk menyetor vector baru dengan elemen yang telah dibalik.
- Struktur perulangan:
Struktur perulangan “for” akan mulai dari elemen yang terakhir dan menuju elemen pertama menggunakan `i = input.size() - 1; i >= 0; i--`. Setelah itu, setiap elemen akan disimpan ke vector baru menggunakan `push_back`.

3. Fungsi main/ main function

```
15  int main() {
16      vector<int> input = {7, 5, 3, 2, 9};
17
18      vector<int> output = reverseVector(input);
19
20      cout << "Reversed vector: ";
21      for (int i : output) {
22          cout << i << " ";
23      }
24      cout << endl;
25
26      return 0;
27 }
```

- *Vector<int> input = {7, 5, 3, 2, 9}*: menghasilkan vector dengan input “int” {7, 5, 3, 2, 9}
- *Vector<int> output = reverseVector(input)*: mengeluarkan fungsi reverseVector, kemu dian disimpan dalam hasil output
- *For (int i : output)*: Perulangan ini menjalankan vector (output) dan mencetak tiap elemen diikuti dengan spasi.
- *Cout << endl*: mencetak baris baru.

Berikut merupakan hasil dari program yang akan dikeluarkan:

```
Reversed vector: 9 2 3 5 7
```