

**JURNAL PRAKTIKUM MODUL 3**  
**STUKTUR DATA 2021/2022 - 1**  
**“Abstract Data Typed dan C++ Dasar”**

**A. Terbimbing (Max: 60 Point)**

1. Buatlah Code::Block Console C++ Project baru.
2. Buatlah file header baru (\*.h), nama file: “catalog.h”
3. Buatlah file implementasi baru (\*.cpp), nama file: “catalog.cpp”
4. Di dalam catalog.h, buatlah ADT untuk menyimpan data array integer, dengan bentuk:

(Nilai: 5 poin)

```
Const nMax = 50

Type catalog <    data :    integer[nMax],
                  max  :    integer,
                  num   :    integer >
```

Catatan:

- Max adalah panjang maksimal dari array
- Num menunjukkan banyaknya data yang disimpan di dalam array. Ini juga menunjukkan (di indeks ke berapa) data terakhir di dalam array.  
Contoh: num: -1 menunjukkan array kosong.

5. Di dalam catalog.h, tambahkan nama fungsi dan prosedur berikut:

(Nilai: 5 poin)

```
void add_data( catalog &C, int x);
void view_data ( catalog C);
int search_data( catalog C, int x);
```

6. Ketentuan **procedure add\_data()**

Prosedur menerima bilangan bulat x, dan memasukkannya ke dalam data catalog pada indeks: 0, 1, 2, ..., dan seterusnya sampai mencapai nilai maksimal. Prosedur ini tidak mengubah catalog jika catalog sudah penuh. Perhatikan ilustrasi contoh berikut:

<pre>catalog storage; storage.max = 8; storage.num = -1;</pre>		
<b>Storage:</b> <div style="border: 1px solid black; display: flex; width: 100px; height: 20px; margin: 5px 0;"></div>	<b>Max:</b> <div style="border: 1px solid black; display: flex; align-items: center; justify-content: center; width: 40px; height: 30px; margin: 5px auto;">8</div>	<b>Num:</b> <div style="border: 1px solid black; display: flex; align-items: center; justify-content: center; width: 40px; height: 30px; margin: 5px auto;">-1</div>
<b>Add_data( storage, 6 )</b> <b>Storage:</b> <div style="border: 1px solid black; display: flex; width: 100px; height: 20px; margin: 5px 0; position: relative;"> <span style="position: absolute; left: 5px; top: 5px;">6</span> </div>	<b>Max:</b> <div style="border: 1px solid black; display: flex; align-items: center; justify-content: center; width: 40px; height: 30px; margin: 5px auto;">8</div>	<b>Num:</b> <div style="border: 1px solid black; display: flex; align-items: center; justify-content: center; width: 40px; height: 30px; margin: 5px auto;">0</div>
<b>Add_data( storage, 5 )</b> <b>Storage:</b> <div style="border: 1px solid black; display: flex; width: 100px; height: 20px; margin: 5px 0; position: relative;"> <span style="position: absolute; left: 5px; top: 5px;">6</span> <span style="position: absolute; left: 25px; top: 5px;">5</span> </div>	<b>Max:</b> <div style="border: 1px solid black; display: flex; align-items: center; justify-content: center; width: 40px; height: 30px; margin: 5px auto;">8</div>	<b>Num:</b> <div style="border: 1px solid black; display: flex; align-items: center; justify-content: center; width: 40px; height: 30px; margin: 5px auto;">1</div>

<div>Add_data( storage, 10 )</div> <div>Storage:</div> <div><div>6</div><div>5</div><div>10</div><div></div><div></div><div></div><div></div><div></div></div>	<div>Max:</div> <div><div>8</div></div>	<div>Num:</div> <div><div>2</div></div>
--	---	---

7. Ketentuan `function search_data()`

Berfungsi untuk menemukan nilai x yang pertama kali ditemukan di dalam penyimpanan data. Fungsi akan mengembalikan lokasi indeks di dalam data catalog. Jika nilai x tidak ditemukan, fungsi akan mengembalikan -1. Lihat contoh di bawah ini:

<div>Storage:</div> <div><div>4</div><div>3</div><div>2</div><div>5</div><div></div><div></div><div></div><div></div></div>	<div>Max:</div> <div><div>8</div></div>	<div>Num:</div> <div><div>3</div></div>	
<div>i = search_data(storage, 4)</div> <div>Storage:</div> <div><div>4</div><div>3</div><div>2</div><div>5</div><div></div><div></div><div></div><div></div></div>	<div>Max:</div> <div><div>8</div></div>	<div>Num:</div> <div><div>3</div></div>	<div>i:</div> <div><div>0</div></div>
<div>i = search_data(storage, 5)</div> <div>Storage:</div> <div><div>4</div><div>3</div><div>2</div><div>5</div><div></div><div></div><div></div><div></div></div>	<div>Max:</div> <div><div>8</div></div>	<div>Num:</div> <div><div>3</div></div>	<div>i:</div> <div><div>3</div></div>
<div>i = search_data(storage, 7)</div> <div>Storage:</div> <div><div>4</div><div>3</div><div>2</div><div>5</div><div></div><div></div><div></div><div></div></div>	<div>Max:</div> <div><div>8</div></div>	<div>Num:</div> <div><div>3</div></div>	<div>i:</div> <div><div>-1</div></div>

8. Di dalam `catalog.cpp` buatlah implementasi dari:  
(Nilai: 40 poin)

```
Procedure add_data(Input/Output C : Catalog, Input x: integer)
{
  IS. -
  FS: X ditambahkan ke dalam array C.Data
}

Procedure view_data(Input C : Catalog)
{
  IS. -
  FS: memberikan output semua data yang ada di dalam array C.Data
}
```

```
Function search_data(C : Catalog, x: integer) → Integer
{
    IS. -
    FS: mengembalikan indeks dari data jika ditemukan atau
        mengembalikan -1 jika data tidak ditemukan
}
```

9. Di dalam main.cpp, tambahkan baris code berikut:

(Nilai: 10 poin)

```
//Inisiasi Varabel
catalog storage;
storage.max = 8;
storage.num = -1;

//Menambahkan data & Output
add_data(storage, 6);
view_data(storage);
add_data(storage, 4);
view_data(storage);
add_data(storage, 8);
view_data(storage);
add_data(storage, 2);
view_data(storage);

cout<<endl;

//mencari index
int i;
i = search_data(storage, 8);
cout<<i<<endl;
i = search_data(storage, 5);
cout<<i<<endl;
```

10. Jalankan program. Output akan berbentuk seperti ini:

```
6
6 4
6 4 8
6 4 8 2

2
-1
```

## B. Mandiri (Max: 30 Point)

1. Buatlah prosedur dibawah ini:

(Nilai: 20 poin)

```
Procedure reversed_view(Input C : Catalog)
{
    IS. Tersedia data catalog, data mungkin kosong
    FS: menampilkan keluaran semua data mulai dari indeks paling belakang
}
Hint: Gunakan perulangan dan variabel decreament
```

(Nilai: 10 poin)

```
Procedure delete_data(Input/Output C : Catalog)
{
    IS. Tersedia data catalog, data mungkin kosong
    FS: data terakhir pada data catalog terhapus
}
Hint: untuk menghapus data, cukup geser indeks num ke kiri.
```

Tambahkan baris kode berikut d idalam main.cpp

```
reversed_view(storage);  
add_data(storage, 5);  
reversed_view(storage);  
  
cout << endl;  
  
view_data(storage);  
delete_data(storage);  
view_data(storage);
```

2. Jalankan program dan program akan menghasilkan keluaran:

```
2 8 4 6  
5 2 8 4 6  
  
6 4 8 2 5  
6 4 8 2
```

~Selamat Mengerjakan^^