

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 1**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Apa itu Variabel?  
Variabel adalah sebuah tempat/lokasi memori yang digunakan untuk menyimpan sebuah data yang bisa diubah, diakses, ataupun dihapus.
2. Apa perbedaan dari initialization dan defining untuk sebuah variabel?  
Initialization adalah proses pengisian/assignment isi sebuah variable, sedangkan defining adalah proses pendefinisian nama dan typedata variabel
3. Apa perbedaan antara sebuah #define constant dan const?  
Constant yang dideklarasikan dengan `#define` akan disubstitusi text biasa dengan nilai aslinya pada saat compilation. Jadi, tipe data dari konstan akan mengikuti aturan implicit casting.  
Sedangkan jika menggunakan `const` tipe datanya dapat diatur secara eksplisit pada saat definisi konstanta.
4. Apa yang dimaksud konstanta literal dan simbolis?  
Konstanta literal adalah konstanta yang hardcoded/diketik langsung di dalam program. Sedangkan konstanta simbolis adalah konstanta yang direpresentasikan menggunakan nama.
5. Apakah variabel DOG, dog, Dog, doG sama? Jelaskan!  
Keempat variabel tersebut merupakan variabel berbeda, karena bahasa c++ menerapkan aturan case sensitive. Case sensitive maksudnya adalah huruf lowercase dianggap berbeda dengan huruf uppercase.

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 2**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Apa yang dimaksud dengan array.  
Array adalah deretan data dengan tipe yang sama yang dikelompokkan dalam satu nama dan dapat diakses menggunakan indeks masing-masing elemennya
2. Jelaskan perbedaan arti dari `int a[5]`; dan arti dari `a[4]`. Apa arti `[5]` dan `[4]` dalam setiap kasus?  
Perbedaannya adalah pada ukuran array-nya. Pada pendefinisian array, angka didalam kurung siku ( `[ ]` ) menandakan ukuran dari array. `[4]` berarti array tersebut memiliki 4 elemen. `[5]` berarti array tersebut memiliki 5 elemen.
3. Jelaskan Perbedaan Array dengan variabel biasa.  
Variabel biasa hanya dapat menyimpan 1 nilai, sedangkan array dapat menyimpan banyak nilai sekaligus.
4. Jelaskan perbedaan Array satu dimensi dengan multi dimensi.  
Array satu dimensi adalah array yang hanya memiliki satu indeks, sedangkan array multidimensi dapat memiliki 2 atau lebih indeks.
5. Buatlah program sederhana menggunakan Array dua dimensi.
  - a. Source code

```
lat2_1.cpp M X
pertemuan2 > lat2_1.cpp > main()
You, 18 minutes ago | 2 authors (Fuad Mustamirrul Ishlah and others)
1 #include <iostream>
2 using namespace std;
3 int main()
4
5     int nilai[10] = {8, 12, 20, 15, 7, 5, 6, 4, 32, 3};
6     int i, jumlah = 0;
7     float rata_rata;
8     // Menghitung jumlah
9     for (i = 0; i < 10; i++)
10         jumlah += nilai[i];
11     rata_rata = (float)jumlah / 10;
12     // Mencetak Elemen Array
13     cout << "\n\nDeretan Bilangan = ";
14     for (i = 0; i < 10; i++)
15         cout << nilai[i] << " ";
16     // Mencetak Harga Jumlah
17     cout << "\nJumlah Bilangan = " << jumlah;
18     cout << "\nRata-Rata Bilangan = " << rata_rata;
19     cout << "\n";
20 Fuad Mustamirrul Ishlah, 2 days ago • initial commit
```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan2
>c++ lat2_1.cpp -o lat2_1.exe

C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan2
>lat2_1.exe

Deretan Bilangan = 8 12 20 15 7 5 6 4 32 3
Jumlah Bilangan =112
Rata-Rata Bilangan = 11.2

C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan2
>|
```

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 3**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Apa yang dimaksud dengan pointer?  
Pointer adalah sebuah tipe data yang digunakan untuk menyimpan alamat memory.
2. Sebutkan dan jelaskan jenis-jenis operator pointer!
  - a. Operator Deference (&  
Digunakan untuk mendapatkan alamat memory dari suatu variable
  - b. Operator Reference (\*)  
Digunakan untuk mendapatkan nilai yang ditunjuk oleh pointer
3. Apa yang dimaksud dengan variabel pointer?  
Variable Pointer adalah variable yang menyimpan sebuah alamat memory
4. Buatlah program sederhana menggunakan pointer!
  - i. Source Code

```
lat4_5.cpp ×
pertemuan4 > lat4_5.cpp > main()
You, 5 days ago | 1 author (You)
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      const char *x = "UNPAM";
7      while(*x){
8          cout << *x++;
9      }
10 }
```

#### ii. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan4
>c++ lat4_5.cpp -o lat4_5.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan4
>lat4_5.exe
UNPAM
```

5. Buatlah program dengan menggunakan kombinasi antara pointer dengan array!
  - a. Source Code

```
tugas_akhir.cpp U ×
pertemuan4 > tugas_akhir.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int main()
5  {
6      int x[2][2] = {
7          {1, 2},
8          {3, 4}};
9      for (int i = 0; i < 2; i++)
10         for (int j = 0; j < 2; j++)
11             cout << "x[" << i << "][" << j << "]: "
12             << "x[" << i << "][" << j << "]: "
13             << "x[" << i << "][" << j << "]: " << endl;
```

#### b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan4
>c++ tugas_akhir.cpp -o tugas_akhir.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan4
>tugas_akhir.exe
x[0][0]: 1
x[0][1]: 2
x[1][0]: 3
x[1][1]: 4
```

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 4**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Jelaskan menurut anda apa itu struktur!  
Struktur adalah kumpulan elemen dengan tipe data berbeda yang di kategorikan menjadi satu.
2. Jelaskan perbedaan array dengan struktur!  
Array hanya dapat menyimpan sekumpulan data dengan tipe yang sama, sedangkan struktur dapat menyimpan sekumpulan data dengan tipe berbeda-beda
3. Jelaskan menurut anda manfaat menggunakan struktur!  
Program menjadi lebih terorganisir dan lebih mudah dibaca karena data datanya mudah divisualisasikan keterkaitannya.
4. Buatlah program struktur dalam struktur yang terdiri dari data mahasiswa!
  - a. Source Code

```

lat3_1.cpp U  lat3_2.cpp U ×
pertemuan3 > lat3_2.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  struct Alamat
5  {
6      char jalan[40], kota[15], pos[6];
7  };
8  struct Tanggal
9  {
10     int tanggal, bulan, tahun;
11 };
12 struct Mahasiswa
13 {
14     char nim[13], nama[25];
15     Alamat alamat;
16     Tanggal tanggalLahir;
17 };
18 int main()
19 {
20     Mahasiswa Mhs;
21     cout << "NIM : "; cin.getline(Mhs.nim, 13);
22     cout << "Nama : "; cin.getline(Mhs.nama, 25);
23     cout << "Alamat : \n";
24     cout << "\tJalan : "; cin.getline(Mhs.alamat.jalan, 40);
25     cout << "\tKota : "; cin.getline(Mhs.alamat.kota, 15);
26     cout << "\tKode pos : "; cin.getline(Mhs.alamat.pos, 6);
27     cout << "Tanggal Lahir : \n";
28     cout << "\tTanggal : "; cin >> Mhs.tanggalLahir.tanggal;
29     cout << "\tBulan : "; cin >> Mhs.tanggalLahir.bulan;
30     cout << "\tTahun : "; cin >> Mhs.tanggalLahir.tahun;
31
32     cout << "\n\nMencetak Kembali Nilai Anggota\n\n";
33     cout << "NIM : " << Mhs.nim;
34     cout << "\nNama : " << Mhs.nama;
35     cout << "\nAlamat : \n";
36     cout << "\n\tJalan : " << Mhs.alamat.jalan;
37     cout << "\n\tKota : " << Mhs.alamat.kota;
38     cout << "\n\tKode Pos : " << Mhs.alamat.pos;
39     cout << "\nTanggal Lahir : " << Mhs.tanggalLahir.tanggal << "-";
40     cout << Mhs.tanggalLahir.bulan << "-" << Mhs.tanggalLahir.tahun;
41 }
42

```



## b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan3
>c++ lat3_2.cpp -o lat3_2.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan3
>lat3_2.exe
```

NIM : 201011400093

Nama : Fuad Mustamirrul I.

Alamat :

Jalan : Banyumas KM 3

Kota : Wonosobo

Kode pos : 56313

Tanggal Lahir :

Tanggal : 21

Bulan : 2

Tahun : 2001

Mencetak Kembali Nilai Anggota

NIM : 201011400093

Nama : Fuad Mustamirrul I.

Alamat :

Jalan : Banyumas KM 3

Kota : Wonosobo

Kode Pos : 56313

Tanggal Lahir : 21-2-2001

## 5. Buatlah Program menghitung luas dan keliling segitiga dengan menggunakan struktur!

### a. Source code

```
p4_no5.cpp U X
pertemuan4 > elearning > p4_no5.cpp > main()
1  #include <iostream>
2  using namespace std;
3  struct Segitiga {
4      int a, t, m;
5      double getLuas(){
6          return (double)(a * t) / 2;
7      }
8      int getKeliling(){
9          return a + t + m;
10     }
11 };
12 int main()
13 {
14     Segitiga s;
15     cout << "Program Menghitung Luas & Keliling Segitiga" << endl;
16     cout << "===== " << endl;
17     cout << "masukkan sisi alas segitiga: ";
18     cin >> s.a;
19     cout << "masukkan sisi tinggi segitiga: ";
20     cin >> s.t;
21     cout << "masukkan sisi miring segitiga: ";
22     cin >> s.m;
23     cout << "===== " << endl;
24     cout << "luas segitiga    : " << s.getLuas() << endl;
25     cout << "keliling segitiga: " << s.getKeliling() << endl;
26 }
```

### b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan4\learning
>c++ p4_no5.cpp -o p4_no5.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan4\learning
>p4_no5.exe
```

Program Menghitung Luas & Keliling Segitiga

=====

masukkan sisi alas segitiga: 20

masukkan sisi tinggi segitiga: 30

masukkan sisi miring segitiga: 50

=====

luas segitiga : 300

keliling segitiga: 100

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 5**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Jelaskan menurut anda apa itu fungsi pada c++  
Fungsi adalah sebuah potongan program yang mengerjakan suatu hal yang spesifik sesuai input yang diberikan pada saat dipanggil.
2. Sebutkan dan jelaskan manfaat menggunakan fungsi  
Fungsi sangat bermanfaat untuk mengabstraksi kode dan membuatnya mudah dibaca dan dipahami oleh programmer lain. Selain itu fungsi juga sangat bermanfaat untuk menghindari duplikasi kode.
3. Jelaskan pemanggilan fungsi dari call by value dan call by reference  
Pada fungsi dengan call by value, input yang diberikan ke fungsi adalah isi/nilai/value dari variable dari sisi pemanggil. Nilai tersebut akan disalin dan disimpan pada variabel baru di dalam fungsi sehingga isi dari variable yang digunakan untuk memanggil fungsi tidak akan terpengaruh oleh proses yang ada di dalam fungsi.  
Sedangkan pada fungsi call by reference, yang diberikan ke fungsi adalah pointer dari variable input, sehingga nilai dari variable yang digunakan untuk memanggil fungsi akan terpengaruh oleh proses yang ada di dalam fungsi.
4. Buat program menggunakan overloading fungsi
  - a. Source code

```

Release Notes: 1.69.0  p5_no4.cpp U X
pertemuan5 > elearning > p5_no4.cpp > main()
1  #include <iostream>
2  using namespace std;
3
4  int maksimum(int x, int y)
5  {
6      return x > y ? x : y;
7  }
8  int maksimum(int x, int y, int z)
9  {
10     return maksimum(x, y) ? maksimum(x, z) : maksimum(y, z);
11 }
12 int main()
13 {
14     int a = 8, b = -4, c = 12;
15     cout << "Nilai a = " << a;
16     cout << "\nNilai b = " << b;
17     cout << "\nNilai Terbesar : " << maksimum(a, b);
18
19     cout << "\nNilai c = " << c;
20     cout << "\nNilai Terbesar : " << maksimum(a, b, c);
21 }

```

b. Hasil

```

C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan5\elearning
>c++ p5_no4.cpp -o p5_no4.exe

```

```

C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan5\elearning
>p5_no4.exe
Nilai a = 8
Nilai b = -4
Nilai Terbesar : 8
Nilai c = 12
Nilai Terbesar : 12

```

5. Buat program fungsi menggunakan parameter array, struktur dan pointer

a. Source code

```
p5_no5.cpp U X
pertemuan5 > elearning > p5_no5.cpp > main()
1  #include <iostream>
2  using namespace std;
3  struct Item { int value; };
4  int findIndex(Item stack[], int* length, Item key)
5  {
6      for (int i = 0; i < *length; i++)
7          if (stack[i].value == key.value)
8              return i;
9      return -1;
10 }
11 int main()
12 {
13     int length = 10;
14     Item stack[] = {{1},{2},{3},{4},{5},{6},{7},{8},{9},{10}};
15     cout << "findIndex key 4: " << findIndex(stack, &length, {4});
16 }
```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan5\elearning
>c++ p5_no5.cpp -o p5_no5.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan5\elearning
>p5_no5.exe
findIndex key 4: 3
```

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 6**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Jelaskan pengertian, fungsi dan jenisnya dari searching!

Searching adalah sebuah proses yang berfungsi untuk mencari suatu data dalam sekumpulan data berdasarkan sebuah kunci tertentu. Searching ada beberapa jenis tergantung dari urutan kumpulan data sumbernya. Apabila datanya tidak terurut maka hanya bisa digunakan sequential search. Apabila datanya terurut maka dapat menggunakan binary search atau interpolation search.

2. Buatlah program menggunakan sequential search!

- a. Source code

```
p6_no2.cpp U x tugas_akhir.cpp
pertemuan6 > elearning > p6_no2.cpp > printVector<T>(vector<T>&)
1  #include <iostream>
2  #include <string>
3  #include <vector>
4  using namespace std;
5
6  template <typename T>
7  vector<int> sequentialSearch(vector<T>& arr, T x){
8      vector<T> foundIndexes = {};
9      for (int i = 0; i < arr.size(); i++){
10         if (arr[i] == x)
11             foundIndexes.push_back(i);
12     }
13     return foundIndexes;
14 }
15 template <typename T>
16 void insertionSort(vector<T>& arr){
17     for (int i = 1; i < arr.size() - 1; i++){
18         int k = i;
19         while (k > 0 && arr[k] < arr[k-1]){
20             T temp = arr[k-1];
21             arr[k-1] = arr[k];
22             arr[k] = temp;
23             k--;
24         }
25     }
26 }
27 template <typename T>
28 void printVector(vector<T>& arr){
29     for (int i = 0; i < arr.size(); i++){
30         cout << arr[i] << (i < arr.size()-1 ? ", " : "");
31     }
32 }
33 int main(){
34     vector<int> arr = {
35         34, 8, 50, 74, 87,
36         90, 12, 25, 20, 30,
37         35, 45, 40, 22, 29,
38         72, 60, 55, 53, 12,
39         32, 33, 12, 41, 12
40     };
41     int key = 12;
42     cout << "array: ";
43     printVector(arr);
44     cout << endl << "mencari " << key << endl;
45     vector<int> findings = sequentialSearch(arr, key);
46     if (findings.size() == 0)
47         cout << "tidak ditemukan" << endl;
48     else {
49         cout << "ditemukan sebanyak " << findings.size() << " kali pada index: ";
50         printVector(findings);
51     }
52 }
```

- b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\elearning
>c++ p6_no2.cpp -o p6_no2.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\elearning
>p6_no2.exe
array: 34,8,50,74,87,90,12,25,20,30,35,45,40,22,29,72,60,55,53,12,32,33,12,41,12
mencari 12
ditemukan sebanyak 4 kali pada index: 6,19,22,24
```

### 3. Buatlah program menggunakan Binary search!

#### a. Source code

```
p6_no2.cpp U p6_no3.cpp U tugas_akhir.cpp
pertemuan6 > elearning > p6_no3.cpp > binarySearch<T>(vector<T>&, T)
1  #include <iostream>
2  #include <string>
3  #include <vector>
4  using namespace std;
5
6  template <typename T>
7  int binarySearch(vector<T>& arr, T x){
8      int start = 0, end = arr.size() - 1;
9      int i = arr.size()/2;
10     while (start != end && end - start != 1){
11         if (arr[i] == x)
12             return i;
13         if (arr[i] > x)
14             end = i;
15         else
16             start = i;
17         i = (start + end) / 2;
18     }
19     return -1;
20 }
21
22 template <typename T>
23 void insertionSort(vector<T>& arr){
24     for (int i = 1; i < arr.size(); i++){
25         int k = i;
26         while (k > 0 && arr[k] < arr[k-1]){
27             T temp = arr[k-1];
28             arr[k-1] = arr[k];
29             arr[k] = temp;
30             k--;
31         }
32     }
33 }
34
35 template <typename T>
36 void printVector(vector<T>& arr){
37     for (int i = 0; i < arr.size(); i++)
38         cout << arr[i] << (i < arr.size()-1 ? ", " : "");
39 }
40
41 int main(){
42     vector<int> arr = {
43         34, 8, 50, 74, 87,
44         90, 12, 25, 20, 30,
45         35, 45, 40, 22, 29,
46         72, 60, 55, 53, 12,
47         32, 33, 12, 41, 12
48     };
49     int key = 25;
50     cout << "array: ";
51     printVector(arr);
52     insertionSort(arr);
53     cout << endl << "array di urutkan terlebih dahulu menjadi: ";
54     printVector(arr);
55     cout << endl << "mencari " << key << endl;
56     int finding = binarySearch(arr, key);
57     if (finding < 0)
58         cout << "tidak ditemukan" << endl;
59     else {
60         cout << "ditemukan pada index ke-" << finding;
61     }
62 }
```

#### b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\elearning
>c++ p6_no3.cpp -o p6_no3.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\elearning
>p6_no3.exe
array: 34,8,50,74,87,90,12,25,20,30,35,45,40,22,29,72,60,55,53,12,32,33,12,41,12
array di urutkan terlebih dahulu menjadi: 8,12,12,12,12,20,22,25,29,30,32,33,34,35
,40,41,45,50,53,55,60,72,74,87,90
mencari 25
ditemukan pada index ke-7
```

#### 4. Buatlah program menggunakan interpolation search!

##### a. Source code

```
p6_no2.cpp U p6_no3.cpp U p6_no4.cpp U X p6_no3.exe U
pertemuan6 > elearning > p6_no4.cpp > main()
1  #include <iostream>
2  #include <string>
3  #include <vector>
4  using namespace std;
5  template <typename T>
6  int interpolationSearch(vector<T>& arr, int start, int end, T x)
7  {
8      int pos;
9      if (start <= end && x >= arr[start] && x <= arr[end]) {
10         pos = start
11             + (((double)(end - start) / (arr[end] - arr[start]))
12               * (x - arr[start]));
13         if (arr[pos] == x)
14             return pos;
15         if (arr[pos] < x)
16             return interpolationSearch(arr, pos + 1, end, x);
17         if (arr[pos] > x)
18             return interpolationSearch(arr, start, pos - 1, x);
19     }
20     return -1;
21 }
22 template <typename T>
23 void insertionSort(vector<T>& arr){
24     for (int i = 1; i < arr.size(); i++){
25         int k = i;
26         while (k > 0 && arr[k] < arr[k-1]){
27             T temp = arr[k-1];
28             arr[k-1] = arr[k];
29             arr[k] = temp;
30             k--;
31         }
32     }
33 }
34 template <typename T>
35 void printVector(vector<T>& arr){
36     for (int i = 0; i < arr.size(); i++)
37         cout << arr[i] << (i < arr.size()-1 ? ", " : "");
38 }
39 int main(){
40     vector<int> arr = {
41         34, 8, 50, 74, 87,
42         90, 12, 25, 20, 30,
43         35, 45, 40, 22, 29,
44         72, 60, 55, 53, 12,
45         32, 33, 12, 41, 12
46     };
47     int key = 25;
48     cout << "array: ";
49     printVector(arr);
50     insertionSort(arr);
51     cout << endl << "array di urutkan terlebih dahulu menjadi: ";
52     printVector(arr);
53     cout << endl << "mencari " << key << endl;
54     int finding = interpolationSearch(arr, 0, arr.size()-1, key);
55     if (finding < 0)
56         cout << "tidak ditemukan" << endl;
57     else {
58         cout << "ditemukan pada index ke-" << finding;
59     }
60 }
```

##### b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\elearning
>c++ p6_no4.cpp -o p6_no4.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\elearning
>p6_no4.exe
array: 34,8,50,74,87,90,12,25,20,30,35,45,40,22,29,72,60,55,53,12,32,33,12,41,12
array di urutkan terlebih dahulu menjadi: 8,12,12,12,12,12,20,22,25,29,30,32,33,34,35
,40,41,45,50,53,55,60,72,74,87,90
mencari 25
ditemukan pada index ke-7
```



5. Buatlah program searching mengecek apakah sebuah huruf ada dalam karakter yang telah diinput, kemudian hurufnya urutkan!

a. Source code

```
p6_no5.cpp U X
pertemuan6 > elearning > p6_no5.cpp > main()
1  #include <iostream>
2  #include <string>
3  #include <vector>
4  using namespace std;
5  template <typename T>
6  int interpolationSearch(vector<T>& arr, int start, int end, T x) {
7      int pos;
8      if (start <= end && x >= arr[start] && x <= arr[end]) {
9          pos = start
10             + (((double)(end - start) / (arr[end] - arr[start]))
11               * (x - arr[start]));
12         if (arr[pos] == x)
13             return pos;
14         if (arr[pos] < x)
15             return interpolationSearch(arr, pos + 1, end, x);
16         if (arr[pos] > x)
17             return interpolationSearch(arr, start, pos - 1, x);
18     }
19     return -1;
20 }
21 template <typename T>
22 void insertionSort(vector<T>& arr){
23     for (int i = 1; i < arr.size(); i++){
24         int k = i;
25         while (k > 0 && arr[k] < arr[k-1]){
26             T temp = arr[k-1];
27             arr[k-1] = arr[k];
28             arr[k] = temp;
29             k--;
30         }
31     }
32 }
33 template <typename T>
34 void printVector(vector<T>& arr){
35     for (int i = 0; i < arr.size(); i++)
36         cout << arr[i] << (i < arr.size()-1 ? ", " : "");
37 }
38 int main(){
39     vector<char> arr;
40     char key;
41     int n;
42     cout << "banyak karakter: ";
43     cin >> n;
44     for(int i = 0; i < n; i++) {
45         char in;
46         cout << "karakter ke-" << i+1 << ": ";
47         cin >> in;
48         arr.push_back(in);
49     }
50     cout << "array karakter: ";
51     printVector(arr);
52     insertionSort(arr);
53     cout << endl << "array di urutkan terlebih dahulu menjadi: ";
54     printVector(arr);
55     cout << endl << "mencari: ";
56     cin >> key;
57     int finding = interpolationSearch(arr, 0, arr.size()-1, key);
58     if (finding < 0)
59         cout << "tidak ditemukan" << endl;
60     else
61         cout << "ditemukan pada index ke-" << finding;
62 }
```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\learning
>c++ p6_no5.cpp -o p6_no5.exe

C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan6\learning
>p6_no5.exe
banyak karakter: 5
karakter ke-1: h
karakter ke-2: s
karakter ke-3: c
karakter ke-4: r
karakter ke-5: f
array karakter: h,s,c,r,f
array di urutkan terlebih dahulu menjadi: c,f,h,r,s
mencari: r
ditemukan pada index ke-3
```

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 7**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Apa yang dimaksud dengan Sorting?  
Sorting adalah proses pengurutan kumpulan data berdasarkan urutan tertentu
2. Algoritma mana yang memiliki kinerja yang baik?  
Algoritma mana yang memiliki kinerja yang baik adalah Quick Sort karena memakai strategi divide and conquer sehingga rata-rata kompleksitasnya  $O(n) = n \cdot \log(n)$
3. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data dengan menggunakan “BubbleSort”?
  - a. 12 7 9 10 13 15 16 1
  - b. 7 9 10 12 13 15 1 16
  - c. 7 9 10 12 13 1 15 16
  - d. 7 9 10 12 1 13 15 16
  - e. 7 9 10 1 12 13 15 16
  - f. 7 9 1 10 12 13 15 16
  - g. 7 1 9 10 12 13 15 16
  - h. 1 7 9 10 12 13 15 16
4. Buatlah program sorting berdasarkan inputan user dengan menghasilkan output data dengan bilangan genap saja menggunakan bubble sort
  - a. Source code

```

p7_no4.cpp U X
pertemuan7 > elearning > p7_no4.cpp > main()
1  #include <iostream>
2  #include <vector>
3  using namespace std;
4  void swap(int& a, int& b){
5      int t = a;
6      a = b;
7      b = t;
8  }
9  void printArrayOnlyEven(vector<int>& arr)
10 {
11     for (int i = 0; i < arr.size(); i++)
12         if (arr[i] % 2 == 0)
13             cout << arr[i] << " ";
14     cout << endl;
15 }
16
17 void bubbleSort(vector<int>& arr)
18 {
19     for (int i = 0; i < arr.size() - 1; i++)
20         for (int j = 0; j < arr.size() - i - 1; j++)
21             if (arr[j] > arr[j + 1])
22                 swap(arr[j], arr[j + 1]);
23 }
24
25 int main()
26 {
27     vector<int> arr;
28     int n;
29     cout << "Masukkan jumlah array: ";
30     cin >> n;
31     for(int i = 0; i < n; i++){
32         int v;
33         cout << "Masukkan data ke-" << i + 1 << ": ";
34         cin >> v;
35         arr.push_back(v);
36     }
37     bubbleSort(arr);
38     cout << "Sorted array: \n";
39     printArrayOnlyEven(arr);
40     return 0;
41 }

```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan7\elearning  
>c++ p7_no4.cpp -o p7_no4.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan7\elearning  
>p7_no4.exe
```

```
Masukkan jumlah array: 6
```

```
Masukkan data ke-1: 2
```

```
Masukkan data ke-2: 9
```

```
Masukkan data ke-3: 6
```

```
Masukkan data ke-4: 7
```

```
Masukkan data ke-5: 4
```

```
Masukkan data ke-6: 8
```

```
Sorted array:
```

```
2 4 6 8
```

5. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data dengan menggunakan “Quick Sort”?

- a. 7, 9, 10, 1                      12              13, 15, 16
- b. 1      7, 9, 10                  12              13      15, 16
- c. 1      7      9, 10              12              13      15              16
- d. 1      7      9      10      12              13      15              16
- e. 1, 7, 9, 10, 12, 13, 15, 16

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 8**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Jelaskan Konsep dari Metode Maximum dan Minimum Sort?  
Maximum adalah pengurutan dengan mencari elemen dengan nilai tertinggi kemudian menempatkannya ke elemen pertama apabila urutan menurun, atau menempatkannya ke elemen terakhir apabila urutan menaik.  
Sedangkan Minimum sort merupakan proses yang sama tapi yang dicari adalah elemen dengan nilai terendah.
2. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data secara menaik Dengan menggunakan Metode Maximum Sort?
  - a. 12 7 9 10 13 15 1 16
  - b. 12 7 9 10 13 1 15 16
  - c. 12 7 9 10 1 13 15 16
  - d. 1 7 9 10 12 13 15 16
3. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data secara menurun Dengan menggunakan Metode Maximum Sort?
  - a. 16 7 9 10 13 15 12 1
  - b. 16 15 9 10 13 7 12 1
  - c. 16 15 13 10 9 7 12 1
  - d. 16 15 13 12 9 7 10 1
  - e. 16 15 13 12 10 7 9 1
  - f. 16 15 13 12 10 9 7 1
4. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data secara menaik Dengan menggunakan Metode Minimum Sort?
  - a. 1 7 9 10 13 15 16 12
  - b. 1 7 9 10 13 15 16 12
  - c. 1 7 9 10 13 15 16 12
  - d. 1 7 9 10 13 15 16 12
  - e. 1 7 9 10 12 15 16 13
  - f. 1 7 9 10 12 13 16 15
  - g. 1 7 9 10 12 13 15 16
5. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data secara menurun Dengan menggunakan Metode Minimum Sort?
  - a. 12 7 9 10 13 15 16 1
  - b. 12 16 9 10 13 15 7 1
  - c. 12 16 15 10 13 9 7 1
  - d. 12 16 15 13 10 9 7 1
  - e. 13 16 15 12 10 9 7 1
  - f. 15 16 13 12 10 9 7 1
  - g. 16 15 13 12 10 9 7 1

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 9**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Jelaskan Konsep Metode Shell Sort, Merge Sort, dan Insertion Sort?
2. Sebutkan dan Jelaskan Kelebihan dan Kekurangan dari Metode Shell Sort, Merge Sort dan Insertion Sort?
3. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data Dengan menggunakan “Shell Sort”?
4. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data Dengan menggunakan “Merge Sort”?
5. Terdapat urutan data berikut: 12 7 9 10 13 15 16 1 tulislah urutan proses untuk mengurutkan data Dengan menggunakan “Insertion Sort”?



# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 10**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Apa perbedaan single Linked List dan Double Linked List?

Single Linked List hanya memiliki nilai dan satu pointer ke node berikutnya, sedangkan double linked list memiliki nilai dan dua pointer yang menunjuk ke node sebelum dan node berikutnya.

2. Buat fungsi insert terakhir suatu struktur data berbentuk single Linked List?

```
146 void sisipBelakang(simpul &l, char elemen)
147 {
148     simpul bantu, baru;
149     baru = (simpul)malloc(sizeof(simpul));
150     baru->Isi = elemen;
151     baru->Next = NULL;
152     if (l == NULL)
153         l = baru;
154     else
155     {
156         bantu = l;
157         while (bantu->Next != NULL)
158             bantu = bantu->Next;
159         bantu->Next = baru;
160     }
161 }
```

3. Buat fungsi delete akhir struktur data single Linked List?

```
195 void hapusBelakang(simpul &l)
196 {
197     simpul bantu, hapus;
198     if (l == NULL)
199         cout << "Linked List Kosong.....";
200     else
201     {
202         bantu = l;
203         while (bantu->Next->Next != NULL)
204             bantu = bantu->Next;
205         hapus = bantu->Next;
206         bantu->Next = NULL;
207         free(hapus);
208     }
209 }
```

4. Buat fungsi insert dalam Double Linked List?

```
149 void sisipBelakang(simpul &dl, char elemen)
150 {
151     simpul bantu, baru;
152     baru = (simpul)malloc(sizeof(simpul));
153     baru->isi = elemen;
154     baru->kanan = NULL;
155     baru->kiri = NULL;
156     if (dl == NULL)
157         dl = baru;
158     else
159     {
160         bantu = dl;
161         while (bantu->kanan != NULL)
162             bantu = bantu->kanan;
163         bantu->kanan = baru;
164         baru->kiri = bantu;
165     }
166 }
```

5. Buat program lengkap dengan double Linked List?

a. Source Code

```
lat10_1.cpp U lat10_2.cpp U X Settings
pertemuan10 > lat10_2.cpp > ...
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  typedef struct node *simpul;
5  struct node
6  {
7      char isi;
8      simpul kanan;
9      simpul kiri;
10 };
11
12 void sisipDepan(simpul &dl, char elemen);
13 void sisipBelakang(simpul &dl, char elemen);
14 void sisipTengah(simpul &dl, char elemen1, char elemen2);
15 void sisipTengah2(simpul &dl, char elemen1, char elemen2);
16 void hapusDepan(simpul &dl);
17 void hapusBelakang(simpul &dl);
18 void hapusTengah(simpul &dl, char elemen);
19 void cetak(simpul dl);
20
21 int main()
22 {
23     char huruf, huruf2;
24     simpul dl = NULL; // Pastikan Bahwa dl kosong
25     cout << "==" OPERASI PADA DOUBLY LINKED LIST ==> << endl;
26     << endl;
27
28     cout << "Penyisipan simpul di depan" << endl;
29     for (int i = 0; i < 4; i++)
30     {
31         cout << "Masukan Huruf : ";
32         cin >> huruf;
33         sisipDepan(dl, huruf);
34     }
35     cetak(dl);
36     cout << endl;
37
38     cout << endl;
39     << "Penyisipan simpul di belakang" << endl;
40     for (int i = 0; i < 4; i++)
41     {
42         cout << "Masukan Huruf : ";
43         cin >> huruf;
44         sisipBelakang(dl, huruf);
45     }
46     cetak(dl);
47     cout << endl;
48
49     cout << endl;
50     << "Penyisipan simpul di tengah setelah huruf lain" << endl;
51     cout << "Masukan Huruf: ";
52     cin >> huruf;
53     cout << "Disisip Setelah Huruf: ";
54     cin >> huruf2;
55     cout << huruf << " Disisip Setelah " << huruf2 << endl;
56     sisipTengah(dl, huruf, huruf2);
57     cetak(dl);
58     cout << endl;
```

```

59
60     cout << endl;
61     | << "Penyisipan simpul di tengah sebelum huruf lain" << endl;
62     cout << "Masukan Huruf: ";
63     cin >> huruf;
64     cout << "Disisip Sebelum Huruf: ";
65     cin >> huruf2;
66     cout << huruf << " Disisip Sebelum " << huruf2 << endl;
67     sisipTengah2(d1, huruf, huruf2);
68     cetak(d1);
69     cout << endl;
70
71     cout << endl;
72     | << "Setelah Hapus Simpul Depan " << endl;
73     hapusDepan(d1);
74     cetak(d1);
75     cout << endl;
76
77     cout << endl;
78     | << "Setelah Hapus Simpul Belakang " << endl;
79     hapusBelakang(d1);
80     cetak(d1);
81     cout << endl;
82
83     cout << endl;
84     | << "Masukkan Huruf Tengah Yang akan dihapus: ";
85     cin >> huruf;
86     hapusTengah(d1, huruf);
87     cetak(d1);
88 }

90 void sisipDepan(simpul &d1, char elemen)
91 {
92     simpul baru;
93     baru = (simpul)malloc(sizeof(simpul));
94     baru->isi = elemen;
95     baru->kanan = NULL;
96     baru->kiri = NULL;
97     if (d1 == NULL)
98         d1 = baru;
99     else
100     {
101         baru->kanan = d1;
102         d1->kiri = baru;
103         d1 = baru;
104     }
105 }

107 void sisipTengah(simpul &d1, char elemen1, char elemen2)
108 {
109     simpul bantu, baru;
110     baru = (simpul)malloc(sizeof(simpul));
111     baru->isi = elemen1;
112     baru->kanan = NULL;
113     baru->kiri = NULL;
114     if (d1 == NULL)
115         cout << "List Kosong ....." << endl;
116     else
117     {
118         bantu = d1;
119         while (bantu->isi != elemen2)
120             bantu = bantu->kanan;
121         baru->kanan = bantu->kanan;
122         baru->kiri = bantu;
123         bantu->kanan->kiri = baru;
124         bantu->kanan = baru;
125     }
126 }

```

```

128 void sisipTengah2(simpul &d1, char elemen1, char elemen2)
129 {
130     simpul bantu, baru;
131     baru = (simpul)malloc(sizeof(simpul));
132     baru->isi = elemen1;
133     baru->kanan = NULL;
134     baru->kiri = NULL;
135     if (d1 == NULL)
136     |     cout << "List Kosong....." << endl;
137     else
138     {
139         bantu = d1;
140         while (bantu->kanan->isi != elemen2)
141         |     bantu = bantu->kanan;
142         baru->kanan = bantu->kanan;
143         baru->kiri = bantu;
144         bantu->kanan->kiri = baru;
145         bantu->kanan = baru;
146     }
147 }

```

```

149 void sisipBelakang(simpul &d1, char elemen)
150 {
151     simpul bantu, baru;
152     baru = (simpul)malloc(sizeof(simpul));
153     baru->isi = elemen;
154     baru->kanan = NULL;
155     baru->kiri = NULL;
156     if (d1 == NULL)
157     |     d1 = baru;
158     else
159     {
160         bantu = d1;
161         while (bantu->kanan != NULL)
162         |     bantu = bantu->kanan;
163         bantu->kanan = baru;
164         baru->kiri = bantu;
165     }
166 }

```

```

168 void cetak(simpul d1)
169 {
170     simpul bantu;
171     if (d1 == NULL)
172     |     cout << "Linked List Kosong ....." << endl;
173     else
174     {
175         bantu = d1;
176         cout << "isi Linked List : ";
177         while (bantu->kanan != NULL)
178         {
179             cout << bantu->isi << "<= >";
180             bantu = bantu->kanan;
181         }
182         cout << bantu->isi;
183     }
184 }

```

```

186 void hapusDepan(simpul &d1)
187 {
188     simpul Hapus;
189     if (d1 == NULL)
190         cout << "Linked List Kosong.....";
191     else
192     {
193         Hapus = d1;
194         d1 = d1->kanan;
195         d1->kiri = NULL;
196         Hapus->kanan = NULL;
197         free(Hapus);
198     }
199 }
200
201 void hapusBelakang(simpul &d1)
202 {
203     simpul bantu, hapus;
204     if (d1 == NULL)
205         cout << "Linked List Kosong.....";
206     else
207     {
208         bantu = d1;
209         while (bantu->kanan->kanan != NULL)
210             bantu = bantu->kanan;
211         hapus = bantu->kanan;
212         bantu->kanan = NULL;
213         hapus->kiri = NULL;
214         free(hapus);
215     }
216 }
217
218 void hapusTengah(simpul &d1, char elemen)
219 {
220     simpul bantu, hapus;
221     if (d1 == NULL)
222         cout << "Linked List Kosong.....";
223     else
224     {
225         bantu = d1;
226         while (bantu->kanan->isi != elemen)
227             bantu = bantu->kanan;
228         hapus = bantu->kanan;
229         bantu->kanan->kanan->kiri = bantu;
230         bantu->kanan = bantu->kanan->kanan;
231         hapus->kanan = NULL;
232         hapus->kiri = NULL;
233         free(hapus);
234     }
235 }

```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan10
>c++ lat10_2.cpp -o lat10_2.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan10
>lat10_2.exe
```

```
== OPERASI PADA DOUBLY LINKED LIST ==
```

Penyisipan simpul di depan

Masukan Huruf : d

Masukan Huruf : a

Masukan Huruf : u

Masukan Huruf : f

isi Linked List : f<=>u<=>a<=>d

Penyisipan simpul di belakang

Masukan Huruf : 0

Masukan Huruf : 0

Masukan Huruf : 1

Masukan Huruf : 0

isi Linked List : f<=>u<=>a<=>d<=>0<=>0<=>1<=>0

Penyisipan simpul di tengah setelah huruf lain

Masukan Huruf: m

Disisip Setelah Huruf: d

m Disisip Setelah d

isi Linked List : f<=>u<=>a<=>d<=>m<=>0<=>0<=>1<=>0

Penyisipan simpul di tengah sebelum huruf lain

Masukan Huruf: u

Disisip Sebelum Huruf: 0

u Disisip Sebelum 0

isi Linked List : f<=>u<=>a<=>d<=>m<=>u<=>0<=>0<=>1<=>0

Setelah Hapus Simpul Depan

isi Linked List : u<=>a<=>d<=>m<=>u<=>0<=>0<=>1<=>0

Setelah Hapus Simpul Belakang

isi Linked List : u<=>a<=>d<=>m<=>u<=>0<=>0<=>1

Masukkan Huruf Tengah Yang akan dihapus: 0

isi Linked List : u<=>a<=>d<=>m<=>u<=>0<=>1

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 11**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten



1. Buat algoritma dasar push dan pop?
  - a. Push
    - i. Cek stack penuh atau tidak
    - ii. Jika penuh, maka
      1. Tampilkan "Stack penuh"
      2. Keluar
    - iii. Increment Top
    - iv. Tambah elemen pada elemen stack yang ditunjuk top
  - b. Pop
    - i. Cek stack kosong atau tidak
    - ii. Jika kosong
      1. Tampilkan "Stack kosong"
      2. Keluar
    - iii. Ambil nilai pada elemen yang ditunjuk oleh Top
    - iv. Decrement Top
2. Sebutkan ciri-ciri stack kosong dan penuh?
  - a. Kosong:  $Top == 0$
  - b. Penuh:  $Top == MaxStack - 1$
3. Buat program input data satu persatu dan push data sampai stack penuh?
  - a. Source code

```

pertemuan11 > elearning > p11_no3.cpp > main()
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  #define MaxS 10
5  struct Stack {
6      char isi[MaxS];
7      unsigned int Top;
8  };
9  void initStack(Stack &s) { s.Top = 0; }
10 bool push(Stack &s, char Data) {
11     if (s.Top < MaxS) {
12         s.isi[s.Top] = Data;
13         s.Top++;
14         return true;
15     }
16     cout << "Stack penuh.....";
17     return false;
18 }
19 int main() {
20     char huruf;
21     Stack s;
22     initStack(s);
23     cout << "isi stack hingga penuh: " << endl;
24     bool belumPenuh;
25     do {
26         cout << "Masukan karakter: ";
27         cin >> huruf;
28         belumPenuh = push(s, huruf);
29     } while(belumPenuh);
30 }

```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan11\learning
>c++ p11_no3.cpp -o p11_no3.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan11\learning
>p11_no3.exe
```

isi stack hingga penuh:

```
Masukan karakter: a
Masukan karakter: d
Masukan karakter: g
Masukan karakter: s
Masukan karakter: r
Masukan karakter: v
Masukan karakter: z
Masukan karakter: c
Masukan karakter: g
Masukan karakter: d
Masukan karakter: g
Stack penuh.....
```

4. Buat program hapus satu satu sampai stack kosong?

a. Source code

```
pertemuan11 > learning > p11_no4.cpp > main()
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  #define MaxS 10
5  struct Stack {
6      char isi[MaxS];
7      unsigned int Top;
8  };
9  void initStack(Stack &s) { s.Top = 0; }
10 bool push(Stack &s, char Data) {
11     if (s.Top < MaxS) {
12         s.isi[s.Top] = Data;
13         s.Top++;
14         return true;
15     }
16     cout << "Stack penuh....."<<endl;
17     return false;
18 }
19 char pop(Stack &s){
20     if (s.Top != 0){
21         s.Top--;
22         return s.isi[s.Top];
23     }
24     cout << "Stack Kosong ...";
25     return '\0';
26 }
27
28 int main() {
29     char huruf;
30     Stack s;
31     initStack(s);
32     cout << "isi stack hingga penuh: " << endl;
33     bool belumPenuh;
34     do {
35         cout << "Masukan karakter: ";
36         cin >> huruf;
37         belumPenuh = push(s, huruf);
38     } while(belumPenuh);
39     huruf = pop(s);
40     do {
41         cout << "Pop: " << huruf << endl;
42         huruf = pop(s);
43     } while(huruf != '\0');
44 }
```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan11\learning
>c++ p11_no4.cpp -o p11_no4.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan11\learning
>p11_no4.exe
```

isi stack hingga penuh:

Masukan karakter: s

Masukan karakter: d

Masukan karakter: g

Masukan karakter: x

Masukan karakter: h

Masukan karakter: d

Masukan karakter: y

Masukan karakter: c

Masukan karakter: b

Masukan karakter: m

Masukan karakter: s

Stack penuh.....

Pop: m

Pop: b

Pop: c

Pop: y

Pop: d

Pop: h

Pop: x

Pop: g

Pop: d

Pop: s

Stack Kosong ...

5. Menambah increment nilai top setiap ada penambahan elemen stack selama belum penuh, adalah langkah-langkah awal pada operasi stack?

Push

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 12**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Bagaimana prinsip kerja stack

- a. FIFO
- b. LIFO?

Stack memiliki prinsip LIFO (Last In First Out)

2. Apakah stack termasuk

- a. Linear List atau
- b. nonlinear?

Stack termasuk linear list

3. Sebutkan operasi di dalam stack?

- a. Push
- b. Pop

4. Jelaskan perbedaan stack dengan array dengan stack dengan linked list?

Apabila menggunakan array, kita harus membuat variable top untuk melacak data terakhir dalam stack, ini juga dapat dimanfaatkan untuk mengetahui berapa banyak data dalam stack. Sedangkan apabila menggunakan Linked list maka kita tidak perlu membuat variable baru, namun perlu menelusuri setiap simpul dalam linked list.

5. Ubah notasi infix  $A + B * - D$  ke notasi post fix?

$A B D - * +$

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 13**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

1. Apa yang dimaksud dengan Queue?  
Queue adalah struktur data yang menyimpan sekumpulan data layaknya antrian, menggunakan asas First In First Out (FIFO).
2. Jelaskan apa Operasi add Queue, front, back, dan delete Queue?
  - a. Add Queue: operasi menambahkan nilai baru pada antrian, nilai baru akan masuk dari belakan queue
  - b. Front: sisi depan dari queue (nilai yang pertama masuk)
  - c. Back: sisi belakan dari queue (nilai yang terakhir masuk)
  - d. Delete Queue: menghapus nilai pada queue, nilai yang dihapus adalah nilai yang paling depan
3. Jelaskan apa itu Empty Dan Full Queue?  
Empty queue adalah antrian yang kosong, sedangkan full queue adalah yang penuh
4. Buatlah program sederhana menggunakan queue?
  - a. Source Code

```

lat13_1.cpp 1, U X Settings lat10_1.cpp
pertemuan13 > lat13_1.cpp > main()
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  typedef struct node *simpul;
5  struct node { char Isi; simpul Next; };
6  void sisipBelakang(simpul &l, char elemen);
7  void hapusDepan(simpul &l);
8  void cetak(simpul l);
9  int main() {
10     char huruf, huruf2;
11     simpul l = NULL; // Pastikan Bahwa l kosong
12     cout << "== OPERASI PADA SINGLE LINKED LIST ==" << endl << endl;
13     cout << "Penyisipan simpul" << endl;
14     for (int i = 0; i < 3; i++) {
15         cout << "Masukan Huruf: ";
16         cin >> huruf;
17         sisipBelakang(l, huruf);
18     }
19     cetak(l);
20     cout << endl;
21     for (int i = 0; i < 3; i++) {
22         hapusDepan(l);
23         cout << "Setelah hapus simpul: ";
24         cetak(l);
25         cout << endl;
26     }
27     cout << "Penyisipan simpul" << endl;
28     for (int i = 0; i < 3; i++) {
29         cout << "Masukan Huruf: ";
30         cin >> huruf;
31         sisipBelakang(l, huruf);
32     }
33     cetak(l);
34     cout << endl;
35     for (int i = 0; i < 2; i++) {
36         hapusDepan(l);
37         cout << "Setelah hapus simpul: ";
38         cetak(l);
39         cout << endl;
40     }
41 }

```

b. Hasil

```
E:\kuliah\struktur data\praktikum-struktur-data\pertemuan13
>c++ lat13_1.cpp -o lat13_1.exe

E:\kuliah\struktur data\praktikum-struktur-data\pertemuan13
>lat13_1.exe
== OPERASI PADA SINGLE LINKED LIST ==

Penyisipan simpul
Masukan Huruf: 1
Masukan Huruf: 2
Masukan Huruf: 3
Isi Linked List : 1-->2-->3
Setelah hapus simpul: Isi Linked List : 2-->3
Setelah hapus simpul: Isi Linked List : 3
Setelah hapus simpul: Linked List Kosong .....

Penyisipan simpul
Masukan Huruf: 4
Masukan Huruf: 5
Masukan Huruf: 6
Isi Linked List : 4-->5-->6
Setelah hapus simpul: Isi Linked List : 5-->6
Setelah hapus simpul: Isi Linked List : 6
```

5. Buatlah program sederhana queue dengan menggunakan singly linked list?

a. Source Code

```
lat13_1.cpp 1, U X Settings lat10_1.cpp
pertemuan13 > lat13_1.cpp > main()
1  #include <iostream>
2  #include <iomanip>
3  using namespace std;
4  typedef struct node *simpul;
5  struct node { char Isi; simpul Next; };
6  void sisipBelakang(simpul &l, char elemen);
7  void hapusDepan(simpul &l);
8  void cetak(simpul l);
9  int main() {
10     char huruf, huruf2;
11     simpul l = NULL; // Pastikan Bahwa l kosong
12     cout << "== OPERASI PADA SINGLE LINKED LIST ==" << endl << endl;
13     cout << "Penyisipan simpul" << endl;
14     for (int i = 0; i < 3; i++) {
15         cout << "Masukan Huruf: ";
16         cin >> huruf;
17         sisipBelakang(l, huruf);
18     }
19     cetak(l);
20     cout << endl;
21     for (int i = 0; i < 3; i++) {
22         hapusDepan(l);
23         cout << "Setelah hapus simpul: ";
24         cetak(l);
25         cout << endl;
26     }
27     cout << "Penyisipan simpul" << endl;
28     for (int i = 0; i < 3; i++) {
29         cout << "Masukan Huruf: ";
30         cin >> huruf;
31         sisipBelakang(l, huruf);
32     }
33     cetak(l);
34     cout << endl;
35     for (int i = 0; i < 2; i++) {
36         hapusDepan(l);
37         cout << "Setelah hapus simpul: ";
38         cetak(l);
39         cout << endl;
40     }
41 }
```



b. Hasil

```
E:\kuliah\struktur data\praktikum-struktur-data\pertemuan13
>c++ lat13_1.cpp -o lat13_1.exe

E:\kuliah\struktur data\praktikum-struktur-data\pertemuan13
>lat13_1.exe
== OPERASI PADA SINGLE LINKED LIST ==

Penyisipan simpul
Masukan Huruf: 1
Masukan Huruf: 2
Masukan Huruf: 3
Isi Linked List : 1-->2-->3
Setelah hapus simpul: Isi Linked List : 2-->3
Setelah hapus simpul: Isi Linked List : 3
Setelah hapus simpul: Linked List Kosong .....

Penyisipan simpul
Masukan Huruf: 4
Masukan Huruf: 5
Masukan Huruf: 6
Isi Linked List : 4-->5-->6
Setelah hapus simpul: Isi Linked List : 5-->6
Setelah hapus simpul: Isi Linked List : 6
```

# **LAPORAN PRAKTIKUM**

## **STRUKTUR DATA**

**LAPORAN KE – 14**



**Disusun Oleh :**

Nama : Fuad Mustamirrul Ishlah

NIM : 201011400093

Kelas : 04TPLE005

**TEKNIK INFORMATIKA**  
**FAKULTAS TEKNIK**  
**UNIVERSITAS PAMULANG**

Jl. Surya Kencana No.1 Pamulang Telp (021)7412566, Fax. (021)7412566  
Tangerang Selatan – Banten

# 1. Buat contoh simulasi antrian di bank menggunakan linked list?

## a. Source code

```
pertemuan14 > elearning > C++ p14_no1.cpp > main()
1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5  typedef struct node *simpul;
6  struct node{ string nama; int noAntrian; simpul Next; };
7  void enqueue(simpul &l, string orangBaru, int noAntrian) {
8      simpul bantu, baru;
9      baru = new node;
10     baru->nama = orangBaru;
11     baru->noAntrian = noAntrian;
12     baru->Next = NULL;
13     if (l == NULL)
14         l = baru;
15     else {
16         bantu = l;
17         while (bantu->Next != NULL)
18             bantu = bantu->Next;
19         bantu->Next = baru;
20     }
21 }
22 void dequeue(simpul &l) {
23     simpul Hapus;
24     if (l == NULL)
25         cout << "Antrian Kosong.....";
26     else {
27         Hapus = l;
28         l = l->Next;
29         Hapus->Next = NULL;
30         free(Hapus);
31     }
32 }
33 void cetak(simpul l) {
34     simpul bantu;
35     if (l == NULL)
36         cout << "Antrian Kosong ....." << endl;
37     else {
38         bantu = l;
39         cout << "Isi Antrian: ";
40         while (bantu->Next != NULL) {
41             cout << bantu->nama << " (" << bantu->noAntrian << ") -->";
42             bantu = bantu->Next;
43         }
44         cout << bantu->nama << " (" << bantu->noAntrian << ") ";
45     }
46 }
47 int main() {
48     string nama;
49     int pilihan, noAntrian = 0;
50     bool running = true;
51     simpul l = NULL;
52     do {
53         cout << "== ANTRIAN BANK IRIT ADJA ==" << endl << endl;
54         cetak(l);
55         cout << endl << "Menu: " << endl;
56         cout << "1. Tambah antrian" << endl;
57         cout << "2. Kurangi antrian" << endl;
58         cout << "3. Keluar" << endl;
59         cout << "Pilihan: "; cin >> pilihan;
60         if (pilihan == 1) {
61             cout << "Masukan nama orang baru: ";
62             cin.ignore(); getline(cin, nama);
63             enqueue(l, nama, ++noAntrian);
64         } else if (pilihan == 2) {
65             cout << "selesai melayani: " << l->nama << "(nomor antrian " << l->noAntrian << ")" << endl;
66             dequeue(l);
67             cetak(l);
68             cout << endl;
69         } else if (pilihan == 3)
70             running = false;
71         else
72             cout << "Maaf pilihan anda salah";
73         cout << endl << endl;
74     } while (running);
75 }
```

b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan14\elarning
>c++ p14_no1.cpp -o p14_no1.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan14\elarning
>p14_no1.exe
== ANTRIAN BANK IRIT ADJA ==
```

```
Antrian Kosong .....
```

```
Menu:
```

1. Tambah antrian
2. Kurangi antrian
3. Keluar

```
Pilihan: 1
```

```
Masukan nama orang baru: adi
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Isi Antrian: adi (1)
```

```
Menu:
```

1. Tambah antrian
2. Kurangi antrian
3. Keluar

```
Pilihan: 1
```

```
Masukan nama orang baru: tina
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Isi Antrian: adi (1) -->tina (2)
```

```
Menu:
```

1. Tambah antrian
2. Kurangi antrian
3. Keluar

```
Pilihan: 2
```

```
selesai melayani: adi(nomor antrian 1)
```

```
Isi Antrian: tina (2)
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Isi Antrian: tina (2)
```

```
Menu:
```

1. Tambah antrian
2. Kurangi antrian
3. Keluar

```
Pilihan: 3
```

```
Press any key to continue . . . |
```

## 2. Buat 2 menu berbeda untuk nasabah dan teller di bank?

### a. Source code

```
pertemuan14 > elearning > p14_no2.cpp > enqueue(simpul &, string, int)
1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5  typedef struct node *simpul;
6  struct node{ string nama; int noAntrian; simpul Next; };
7  > void enqueue(simpul &l, string orangBaru, int noAntrian) { ...
22 > void dequeue(simpul &l) { ...
33 > void cetak(simpul l) { ...
47 int main() {
48     string nama;
49     int pilihan, noAntrian = 0;;
50     bool running = true, tellerMode = false;
51     simpul l = NULL;
52     do {
53         cout << "== ANTRIAN BANK IRIT ADJA ==" << endl << endl;
54         cetak(l);
55         cout << endl << "Menu " << (tellerMode ? "Teller:" : "Nasabah: ") << endl;
56         cout << "1. Tambah antrian" << endl;
57         if (tellerMode)
58             cout << "2. Kurangi antrian" << endl;
59         else
60             cout << "2. Teller Mode" << endl;
61         cout << "99. Keluar" << endl;
62         cout << "Pilihan: "; cin >> pilihan;
63         if (pilihan == 1) {
64             cout << "Masukan nama orang baru: ";
65             cin.ignore(); getline(cin, nama);
66             enqueue(l, nama, ++noAntrian);
67         } else if (pilihan == 2 && tellerMode) {
68             cout << "selesai melayani: " << l->nama << "(nomor antrian " << l->noAntrian << ")" << endl;
69             dequeue(l);
70             cetak(l);
71             cout << endl;
72         } else if (pilihan == 2 && !tellerMode)
73             tellerMode = true;
74         else if (pilihan == 99)
75             running = false;
76         else
77             cout << "Maaf pilihan anda salah";
78         cout << endl << endl;
79     } while (running);
80 }
```

### b. Hasil

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan14\learning
>c++ p14_no2.cpp -o p14_no2.exe
```

```
C:\Users\indbr\Documents\kuliah\SMT 4\praktikum-struktur-data\pertemuan14\learning
>p14_no2.exe
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Antrian Kosong .....
```

```
Menu Nasabah:
```

```
1. Tambah antrian
```

```
2. Teller Mode
```

```
99. Keluar
```

```
Pilihan: 1
```

```
Masukan nama orang baru: dana
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Isi Antrian: dana (1)
```

```
Menu Nasabah:
```

```
1. Tambah antrian
```

```
2. Teller Mode
```

```
99. Keluar
```

```
Pilihan: 2
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Isi Antrian: dana (1)
```

```
Menu Teller:
```

```
1. Tambah antrian
```

```
2. Kurangi antrian
```

```
99. Keluar
```

```
Pilihan: 1
```

```
Masukan nama orang baru: andi
```

```
== ANTRIAN BANK IRIT ADJA ==
```

```
Isi Antrian: dana (1) -->andi (2)
```

```
Menu Teller:
```

```
1. Tambah antrian
```

```
2. Kurangi antrian
```

```
99. Keluar
```

```
Pilihan: 2
```

```
selesai melayani: dana(nomor antrian 1)
```

```
Isi Antrian: andi (2)
```

### 3. Buat program menghitung waktu tunggu pada saat mengantri?

#### a. Source code

```
pertemuan14 > elearning > p14_no3.cpp > main()
1  #include <iostream>
2  #include <iomanip>
3  #include <string>
4  using namespace std;
5  typedef struct node *simpul;
6  struct node{ string nama; int noAntrian; simpul Next; };
7  > void enqueue(simpul &l, string orangBaru, int noAntrian) { ...
22 > void dequeue(simpul &l) { ...
33 > void cetak(simpul l) { ...
47 int estimasiWaktuTunggu(simpul &l, int noAntrian){
48     simpul bantu;
49     int estWaktuPerAntrian = 5;
50     int hasilEst = 0;
51     if (l == NULL)
52         cout << "Antrian Kosong ....." << endl;
53     else {
54         bantu = l;
55         while (bantu->noAntrian != noAntrian && bantu->Next != NULL) {
56             bantu = bantu->Next;
57             hasilEst += estWaktuPerAntrian;
58         }
59     }
60     return hasilEst;
61 }
62 int main() {
63     string nama;
64     int pilihan, noAntrian = 0;
65     bool running = true, tellerMode = false;
66     simpul l = NULL;
67     do {
68         cout << "== ANTRIAN BANK IRIT ADJA ==" << endl << endl;
69         cetak(l);
70         cout << endl << "Menu " << (tellerMode ? "Teller:" : "Nasabah: ") << endl;
71         cout << "1. Tambah antrian" << endl;
72         if (tellerMode)
73             cout << "2. Kurangi antrian" << endl;
74         else
75             cout << "2. Teller Mode" << endl;
76         cout << "3. Estimasi waktu antrian" << endl;
77         cout << "99. Keluar" << endl;
78         cout << "Pilihan: "; cin >> pilihan;
79         if (pilihan == 1) {
80             cout << "Masukan nama orang baru: ";
81             cin.ignore(); getline(cin, nama);
82             enqueue(l, nama, ++noAntrian);
83         } else if (pilihan == 2 && tellerMode) {
84             cout << "selesai melayani: " << l->nama << "(nomor antrian " << l->noAntrian << ")" << endl;
85             dequeue(l);
86             cetak(l);
87             cout << endl;
88         } else if (pilihan == 2 && !tellerMode)
89             tellerMode = true;
90         else if (pilihan == 3) {
91             cout << "Masukan nomor antrian: "; cin >> noAntrian;
92             cout << "Estimasi waktu antrian anda: " << estimasiWaktuTunggu(l, noAntrian);
93         } else if (pilihan == 99)
94             running = false;
95         else
96             cout << "Maaf pilihan anda salah";
97         cout << endl << endl;
98     } while (running);
99 }
```

b. Hasil

== ANTRIAN BANK IRIT ADJA ==

Antrian Kosong .....

Menu Nasabah:

1. Tambah antrian
2. Teller Mode
3. Estimasi waktu antrian
99. Keluar

Pilihan: 1

Masukan nama orang baru: adi

== ANTRIAN BANK IRIT ADJA ==

Isi Antrian: adi (1)

Menu Nasabah:

1. Tambah antrian
2. Teller Mode
3. Estimasi waktu antrian
99. Keluar

Pilihan: 1

Masukan nama orang baru: dani

== ANTRIAN BANK IRIT ADJA ==

Isi Antrian: adi (1) -->dani (2)

Menu Nasabah:

1. Tambah antrian
2. Teller Mode
3. Estimasi waktu antrian
99. Keluar

Pilihan: 1

Masukan nama orang baru: retno

== ANTRIAN BANK IRIT ADJA ==

Isi Antrian: adi (1) -->dani (2) -->retno (3)

Menu Nasabah:

1. Tambah antrian
2. Teller Mode
3. Estimasi waktu antrian
99. Keluar

Pilihan: 3

Masukan nomor antrian: 2

Estimasi waktu antrian anda: 5

== ANTRIAN BANK IRIT ADJA ==

Isi Antrian: adi (1) -->dani (2) -->retno (3)

Menu Nasabah:

1. Tambah antrian
2. Teller Mode
3. Estimasi waktu antrian
99. Keluar

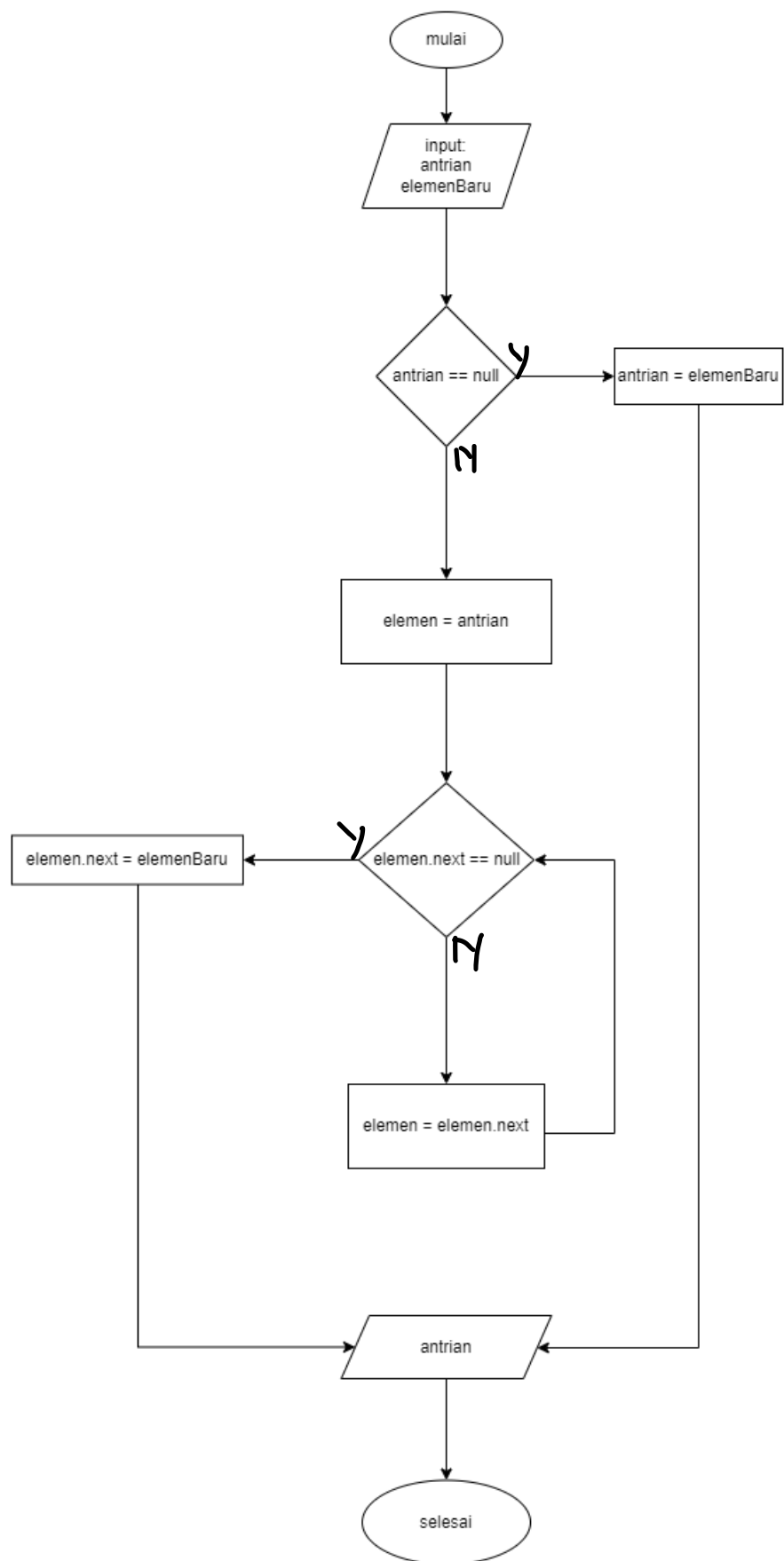
Pilihan: 3

Masukan nomor antrian: 3

Estimasi waktu antrian anda: 10



4. Buat flowchart penambahan pada antrian menggunakan linked list?



5. Buat flowchart penghapusan pada antrian menggunakan linked list?

