Nama : Arie Farchan Fyrzatullah
NIM : 103032330094

Kelas : IT-47-KHS

1. Isi file list.h

main.cpp | list.h | list.cpp

```
#include <iostream>
#define first(L) L.first
#define next(P) P->next
#define info(P) P->info

using namespace std;
typedef int infotype;
typedef struct elmtList *address;

struct elmtList {
    infotype info;
    address next;
};

struct List {
    address first;
};

void createList(List &L);

address allocate(infotype X);

void insertFirst(List &L, address P);

void printInfo(List L);

```

2. Isi file list.cpp

main.cpp | list.h | list.cpp

```cpp
#include <iostream>
#include "list.h"

using namespace std;

void createList(List &L) {
    first(L) = NULL;
}


address allocate(infotype x) {

    address P = new elmtList;
    info(P) = x;
    next(P) = NULL;

    return P;
}

void insertFirst(List &L, address P) {
    next(P) = first(L);
    first(L)= P;
}

void printInfo(List L) {
    address P = L.first;
    while (P != NULL) {
        cout << info(P) << ", ";
        P = next(P);
    }
    cout << endl;
}
```

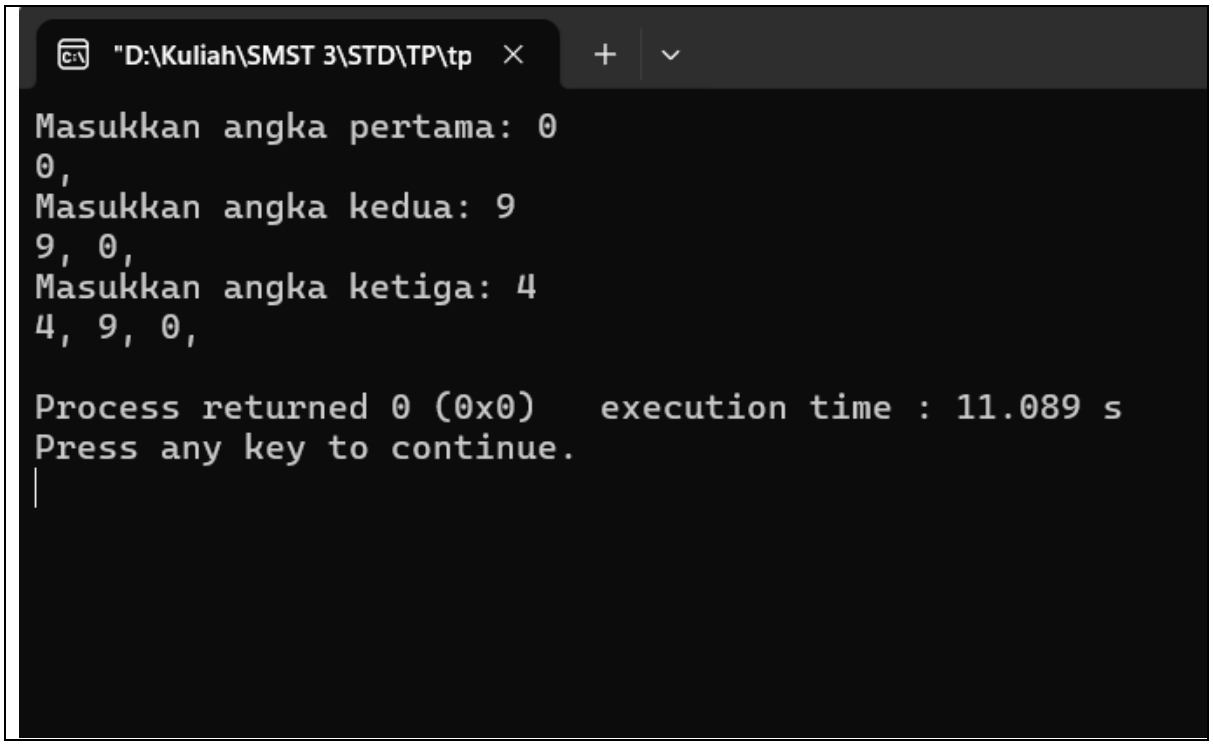
3. Isi file main.cpp

main.cpp | list.h | list.cpp

```cpp
#include <iostream>
#include "list.h"
using namespace std;

int main() {
List L;
    createList(L);
    int input;

    cout << "Masukkan angka pertama: ";
    cin >> input;
    address P1 = allocate(input);
    insertFirst(L, P1);
    printInfo(L);


    cout << "Masukkan angka kedua: ";
    cin >> input;
    address P2 = allocate(input);
    insertFirst(L, P2);
    printInfo(L);


    cout << "Masukkan angka ketiga: ";
    cin >> input;
    address P3 = allocate(input);
    insertFirst(L, P3);
    printInfo(L);

    return 0;
}
```

\

4. Hasil keluaran



7.

Isi list.cpp

main.cpp | list.h | list.cpp

```cpp
#include <iostream>
#include "list.h"

using namespace std;

void createList(List &L) {
    first(L) = NULL;
}


address allocate(infotype x) {

    address P = new elmtList;
    info(P) = x;
    next(P) = NULL;

    return P;
}

void insertFirst(List &L, address P) {
    next(P) = first(L);
    first(L)= P;
}

void insertLast(List &L, address P) {
    if (L.first == NULL) {
        L.first = P;
    } else {
        address last = L.first;
        while (next(last) != NULL) {
            last = next(last);
        }
        next(last) = P;
    }
}

void insertAfter(address Prec, address P) {
    if (Prec != NULL) {
        next(P) = next(Prec);
        next(Prec) = P;
    }
```

main.cpp | list.h | list.cpp

```
44
45  void deleteLast(List &L, address &P) {
46      if (L.first == NULL) {
47          P = NULL;
48      } else if (next(L.first) == NULL) {
49          P = L.first;
50          L.first = NULL;
51      } else {
52          address prev = NULL;
53          P = L.first;
54          while (next(P) != NULL) {
55              prev = P;
56              P = next(P);
57          }
58          next(prev) = NULL;
59      }
60  }
61
62
63  void deleteAfter(address Prec, address &P) {
64      if (Prec != NULL && next(Prec) != NULL) {
65          P = next(Prec);
66          next(Prec) = next(P);
67          next(P) = NULL;
68      }
69  }
70
71
72  address searchInfo(List L, infotype x) {
73      address P = L.first;
74      while (P != NULL) {
75          if (info(P) == x) {
76              return P;
77          }
78          P = next(P);
79      }
80      return NULL;
81  }
82
83  void printInfo(List L) {
84      address P = L.first;
85      while (P != NULL) {
86          cout << info(P) << "";
87          P = next(P);
88      }
89      cout << endl;
90  }
91
```

Isi list.h

main.cpp | list.h | list.cpp

```
1  #include <iostream>
2  #define first(L) L.first
3  #define next(P) P->next
4  #define info(P) P->info
5
6  using namespace std;
7  typedef int infotype;
8  typedef struct elmtList *address;
9
10 struct elmtList {
11     infotype info;
12     address next;
13 };
14
15 struct List {
16     address first;
17 };
18
19 void createList(List &L);
20
21 address allocate(infotype X);
22
23 void insertFirst(List &L, address P);
24
25 void insertLast(List &L, address P);
26
27 void insertAfter(address Prec, address P);
28
29 void deleteLast(List &L, address &P);
30
31 void deleteAfter(address Prec, address &P);
32
33 address searchInfo(List L, infotype x);
34
35 void printInfo(List L);
```

Isi main.cpp

main.cpp | list.h | list.cpp

```cpp
#include <iostream>
#include "list.h"
using namespace std;

int main() {
 List L;
    createList(L);
    int input;

    cout << "Masukkan NIM perdigit" << endl;

    for (int i = 1; i <= 10; i++) {
        cout << "Digit " << i << " : ";
        cin >> input;

        address P = allocate(input);
        insertLast(L, P);
    }

    cout << "Isi list : ";
    printInfo(L);

    return 0;
}
```

Hasil keluaran

```
"D:\Kuliah\SMST 3\STD\TP\tp
Masukkan NIM perdigit
Digit 1 : 1
Digit 2 : 0
Digit 3 : 3
Digit 4 : 0
Digit 5 : 3
Digit 6 : 2
Digit 7 : 3
Digit 8 : 3
Digit 9 : 0
Digit 10 : 0
Isi list : 1030323300

Process returned 0 (0x0)   execution time : 12.180 s
Press any key to continue.
```