Nama : Abiyoso Danar Panji Yudhanto

NIM : 103012300006

Kelas : IF 47 05

Main:

list.h | list.cpp | main.cpp

```
3
4   using namespace std;
5
6   int main()
7   {
8       List L;
9       createList(L);
10      int a, b, c;
11
12      cout << "Masukkan digit NIM pertama: ";
13      cin >> a;
14      address NIM1 = allocate(a);
15      insertFirst(L, NIM1);
16
17      cout << "Masukkan digit NIM kedua: ";
18      cin >> b;
19      address NIM2 = allocate(b);
20      insertFirst(L, NIM2);
21
22      cout << "Masukkan digit NIM ketiga: ";
23      cin >> c;
24      address NIM3 = allocate(c);
25      insertFirst(L, NIM3);
26
27      printinfo(L);
28
29      return 0;
30  }
31
```

List.cpp:

list.h | list.cpp | main.cpp

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

using namespace std;

void createList(List &L){
    first(L) = NULL;
}
address allocate(infotype x) {
    address P = new elmlist;
    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 = first(L);
    while (p!= NULL) {
        cout << p -> info << ", ";
        p = p -> next;
    }
    cout << endl;
}

```

Header:

list.h | list.cpp | main.cpp

```
2   #define LIST_H_INCLUDED
3
4   #include <iostream>
5   #define first(L) L.first
6   #define next(P) P -> next
7   #define info(P) P -> info
8
9   using namespace std;
10
11  typedef int infotype;
12
13  typedef struct elmlist *address;
14
15  struct elmlist {
16      infotype info;
17      address next;
18  };
19
20  struct List {
21      address first;
22  };
23
24  void createList(List &L);
25  address allocate(infotype x);
26  void insertFirst(List &L, address P);
27  void printinfo(List L);
28
29  #endif // LIST_H_INCLUDED
30
```

Output: