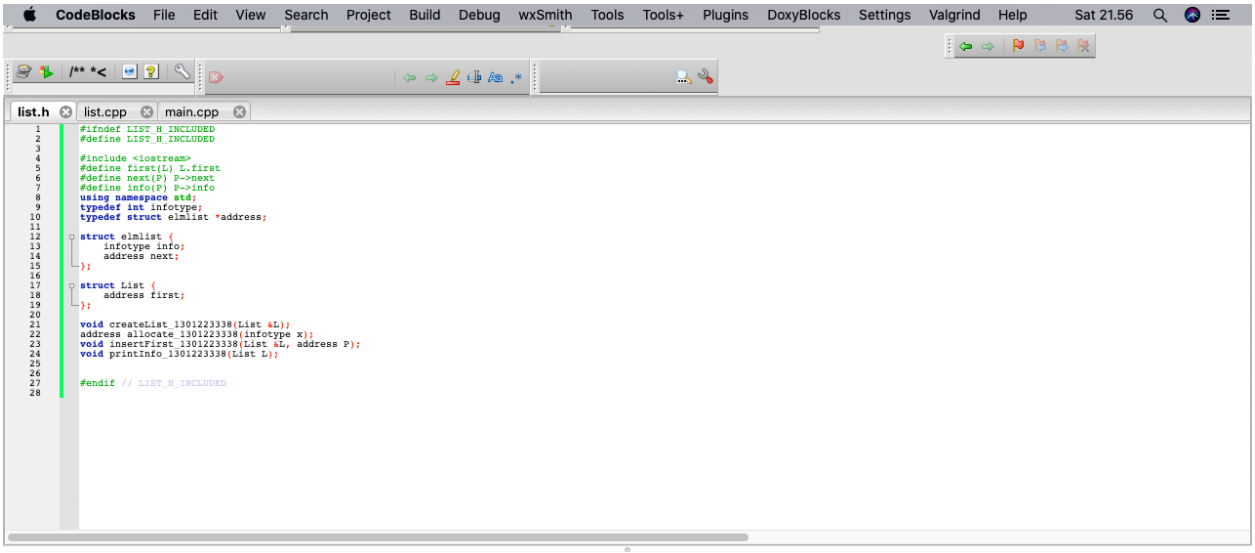
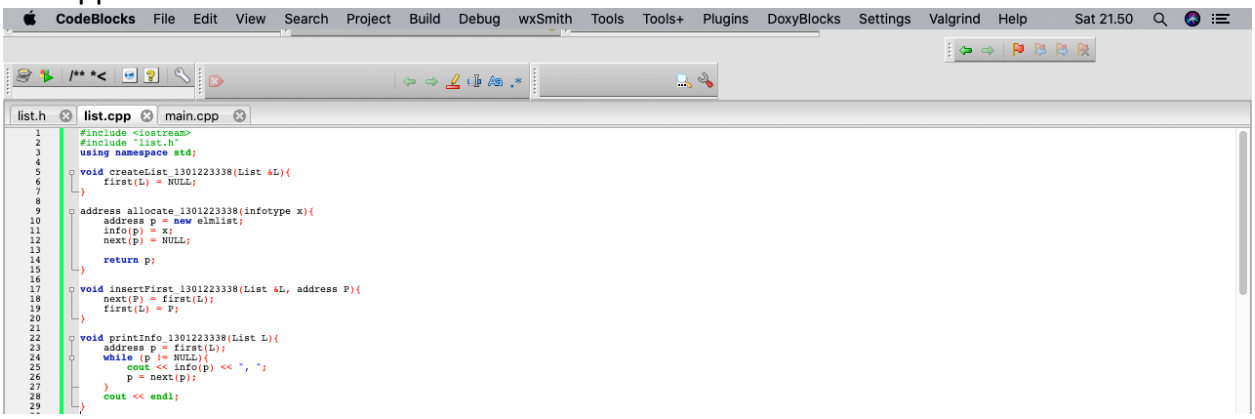
Nama : Helmi Efendi Lubis
NIM : 1301223338

TP MODUL 4

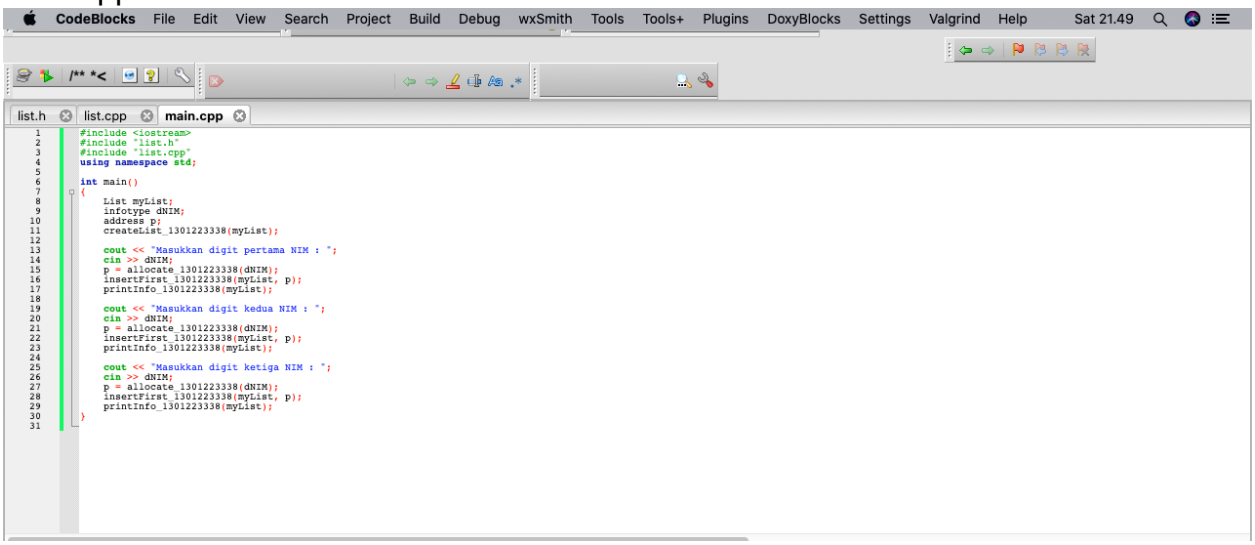
List.h



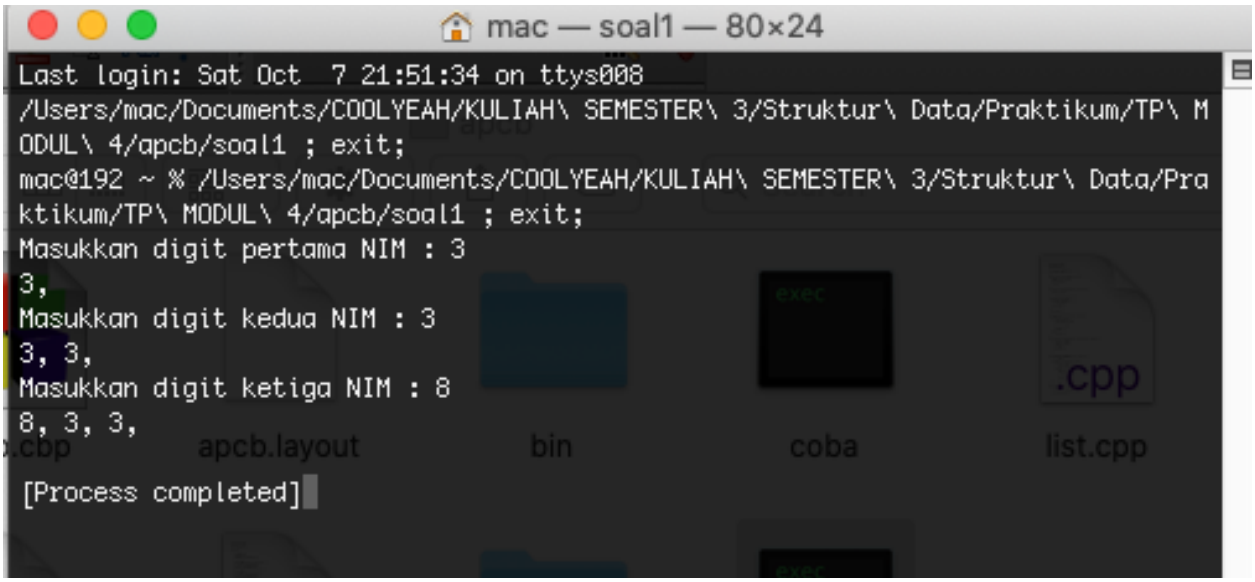
List.cpp



Main.cpp

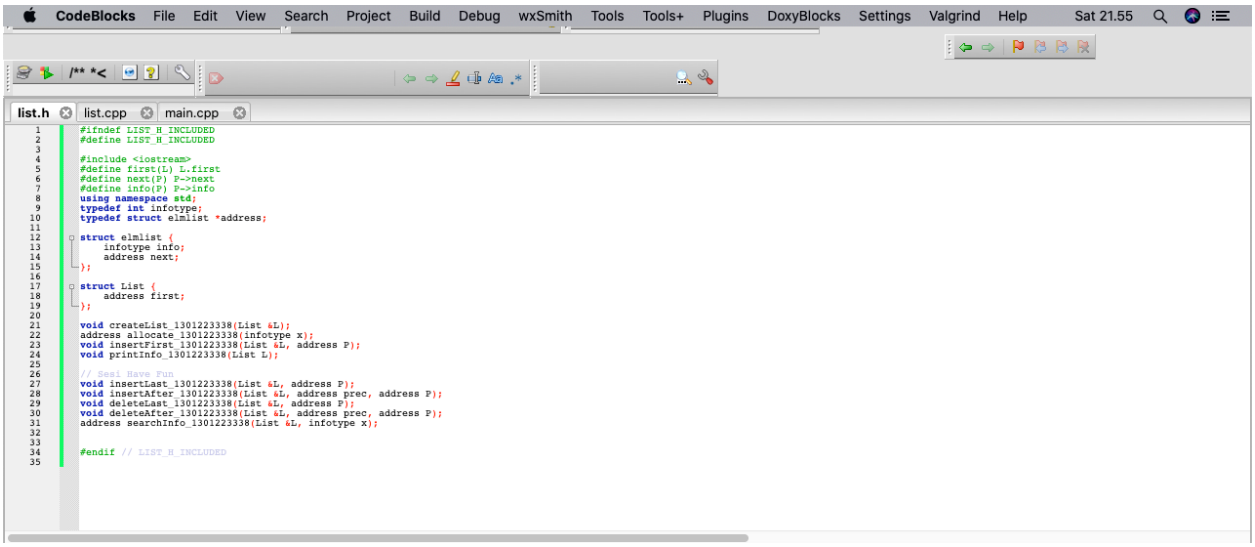


Hasil:



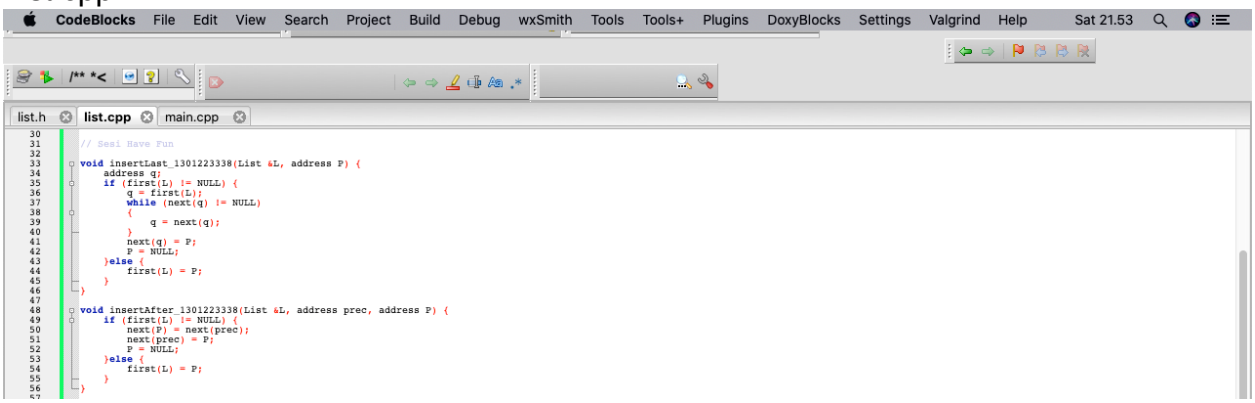
SESI HAVE FUN

List.h

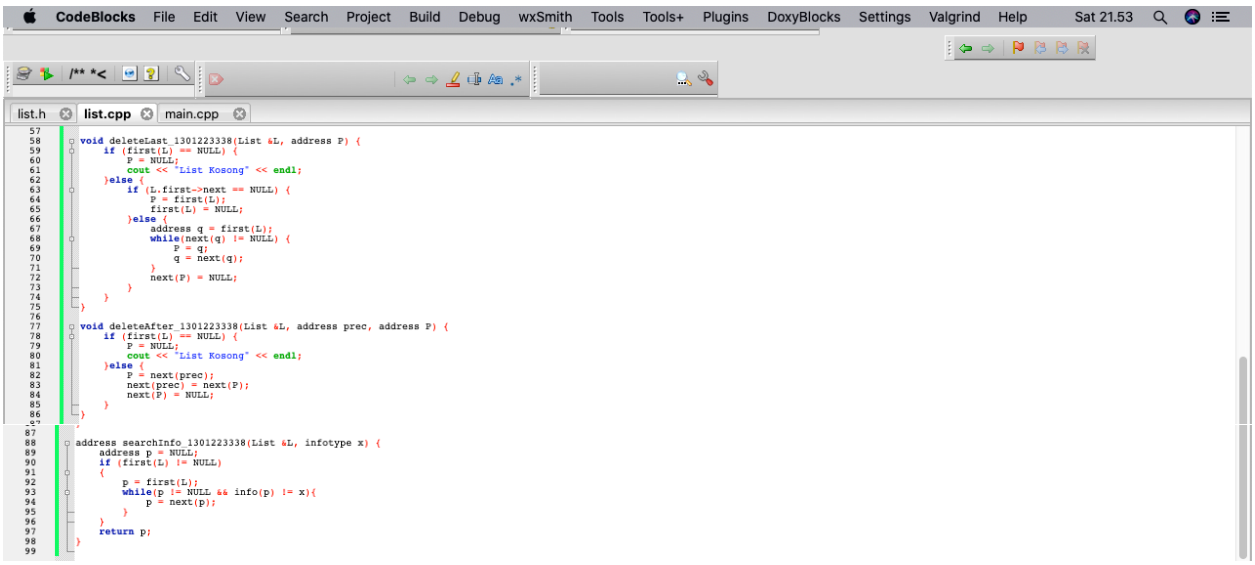


```h
#ifndef LIST_H_INCLUDED
#define LIST_H_INCLUDED

#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 elmlist *address;

struct elmlist {
    infotype info;
    address next;
};

struct List {
    address first;
};

void createList_1301223338(List &L);
address allocate_1301223338(infotype x);
void insertFirst_1301223338(List &L, address P);
void printInfo_1301223338(List L);

// Sesi Have Fun
void insertLast_1301223338(List &L, address P);
void insertAfter_1301223338(List &L, address prec, address P);
void deleteLast_1301223338(List &L, address P);
void deleteAfter_1301223338(List &L, address prec, address P);
address searchInfo_1301223338(List &L, infotype x);


#endif // LIST_H_INCLUDED
```

List.cpp

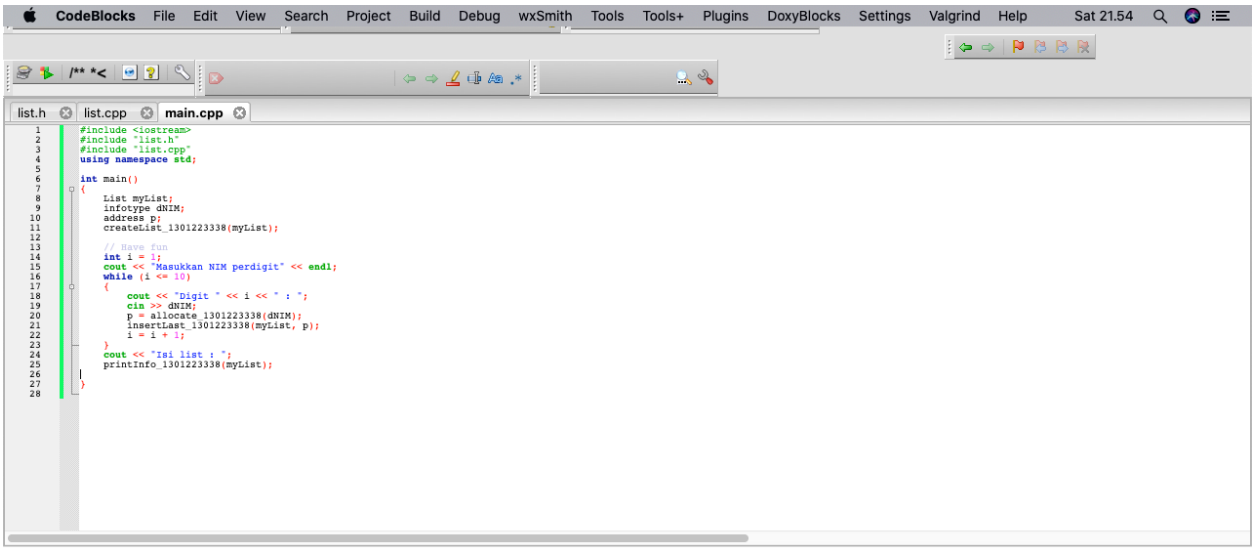


```cpp
// Sesi Have Fun

void insertLast_1301223338(List &L, address P) {
    address q;
    if (first(L) != NULL) {
        q = first(L);
        while (next(q) != NULL)
        {
            q = next(q);
        }
        next(q) = P;
        P = NULL;
    }else {
        first(L) = P;
    }
}

void insertAfter_1301223338(List &L, address prec, address P) {
    if (first(L) != NULL) {
        next(P) = next(prec);
        next(prec) = P;
        P = NULL;
    }else {
        first(L) = P;
    }
}
```



```cpp
void deleteLast_1301223338(List &L, address P) {
    if (first(L) == NULL) {
        P = NULL;
        cout << "List Kosong" << endl;
    }else {
        if (L.first->next == NULL) {
            P = first(L);
            first(L) = NULL;
        }else {
            address q = first(L);
            while(next(q) != NULL) {
                P = q;
                q = next(q);
            }
            next(P) = NULL;
        }
    }
}

void deleteAfter_1301223338(List &L, address prec, address P) {
    if (first(L) == NULL) {
        P = NULL;
        cout << "List Kosong" << endl;
    }else {
        P = next(prec);
        next(prec) = next(P);
        next(P) = NULL;
    }
}

address searchInfo_1301223338(List &L, infotype x) {
    address p = NULL;
    if (first(L) != NULL)
    {
        p = first(L);
        while(p != NULL && info(p) != x){
            p = next(p);
        }
    }
    return p;
}
```

Main.cpp



Hasil: