

[컴퓨터 소프트웨어 설계]

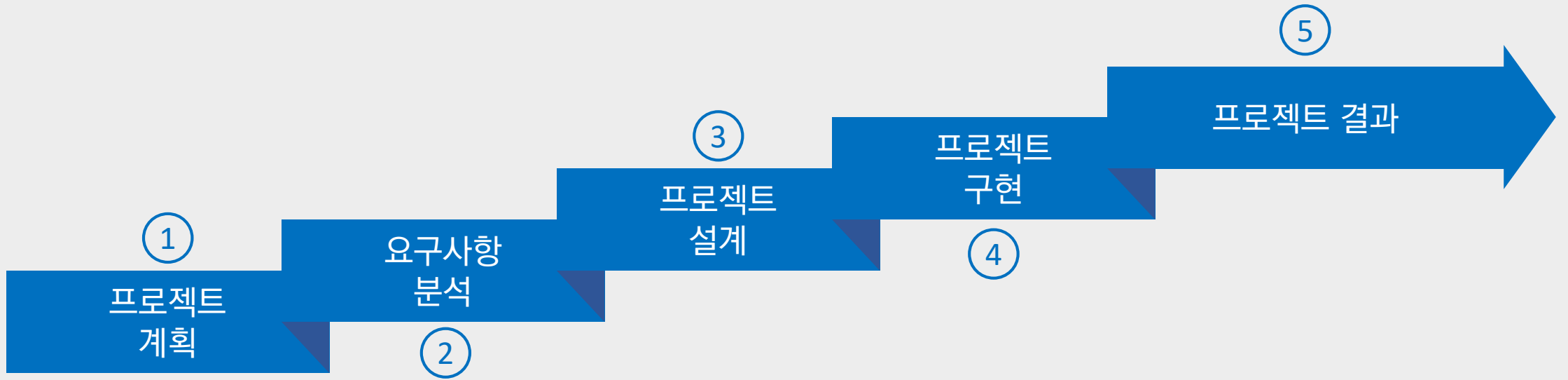
강남역 12번 출구 맥도날드 시뮬레이션 프로그램



2016.12.12
2조 강남역 버거킹



목 차



프로젝트 계획

프로젝트 목적



주 목적 : 맥도날드 매장/드라이브 스루 시뮬레이션 제작

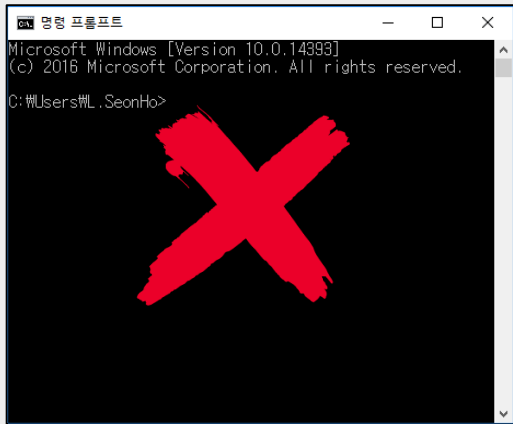
맥도날드에서 발생하는 상황(손님 입장, 주문, 식사 등)을 멀티 쓰레드와 세마포어, 큐를 이용해서 구현하고, GUI를 통해서 이를 시각적으로 확인하고 발생하는 데이터에 대해 AVL Tree 데이터베이스와 File I/O를 통해서 보관하고, 서버와 클라이언트 간의 네트워크 통신을 수행한다.

프로젝트 일정관리

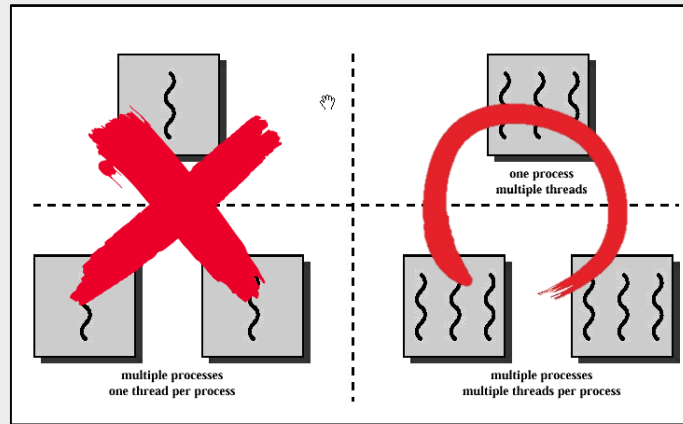
[illegible]

요구사항 분석

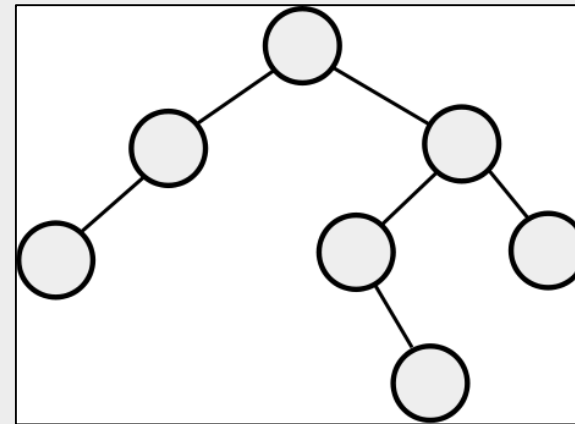
프로젝트 요구사항



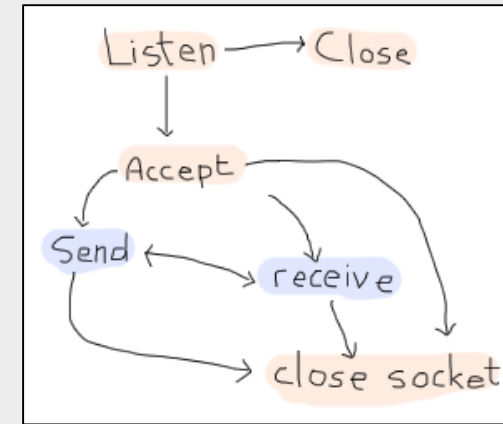
Graphical UI



Multi Thread
& Semaphore



Tree & DB

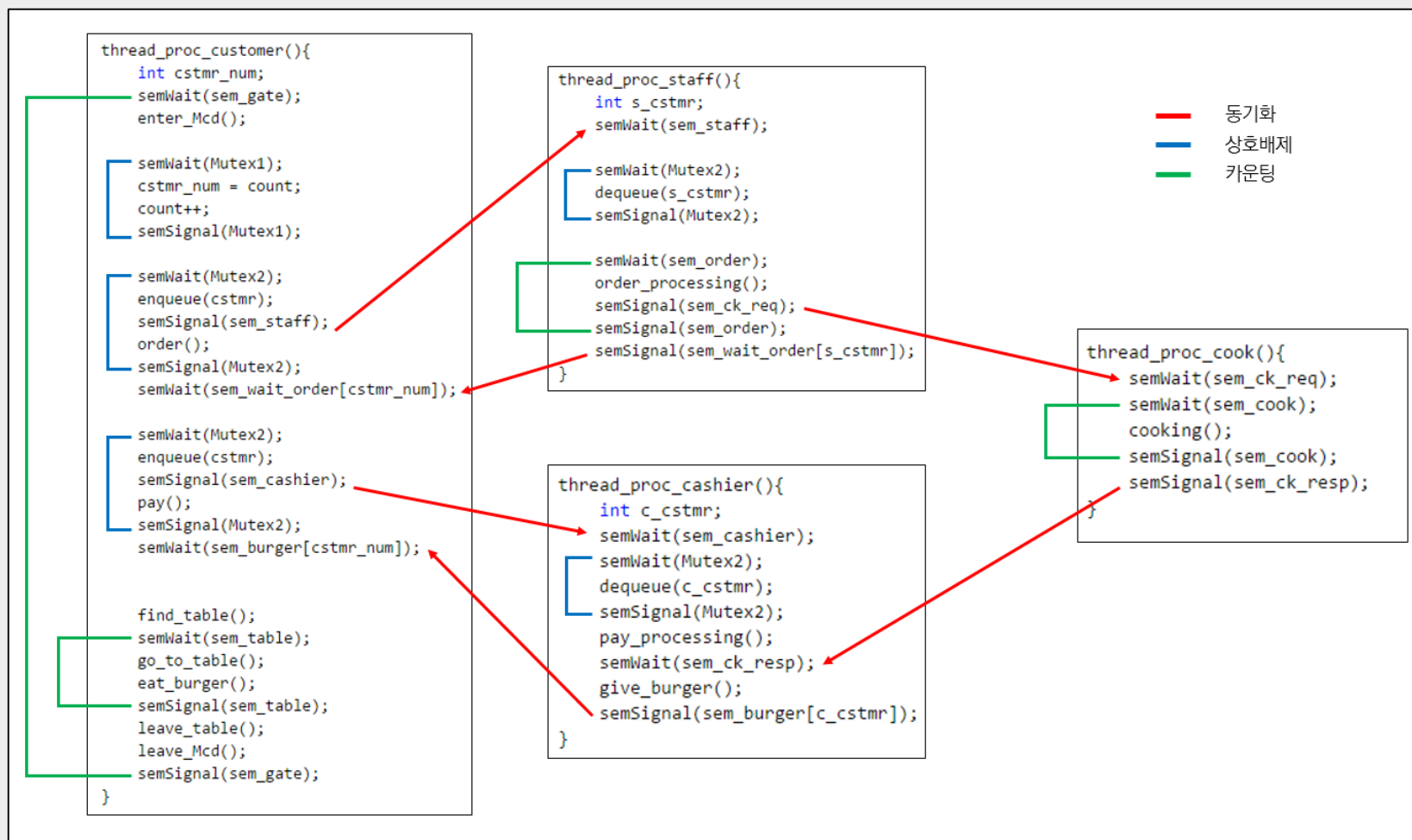


Network
(Server & Client)

프로젝트 설계

Thread & Semaphore

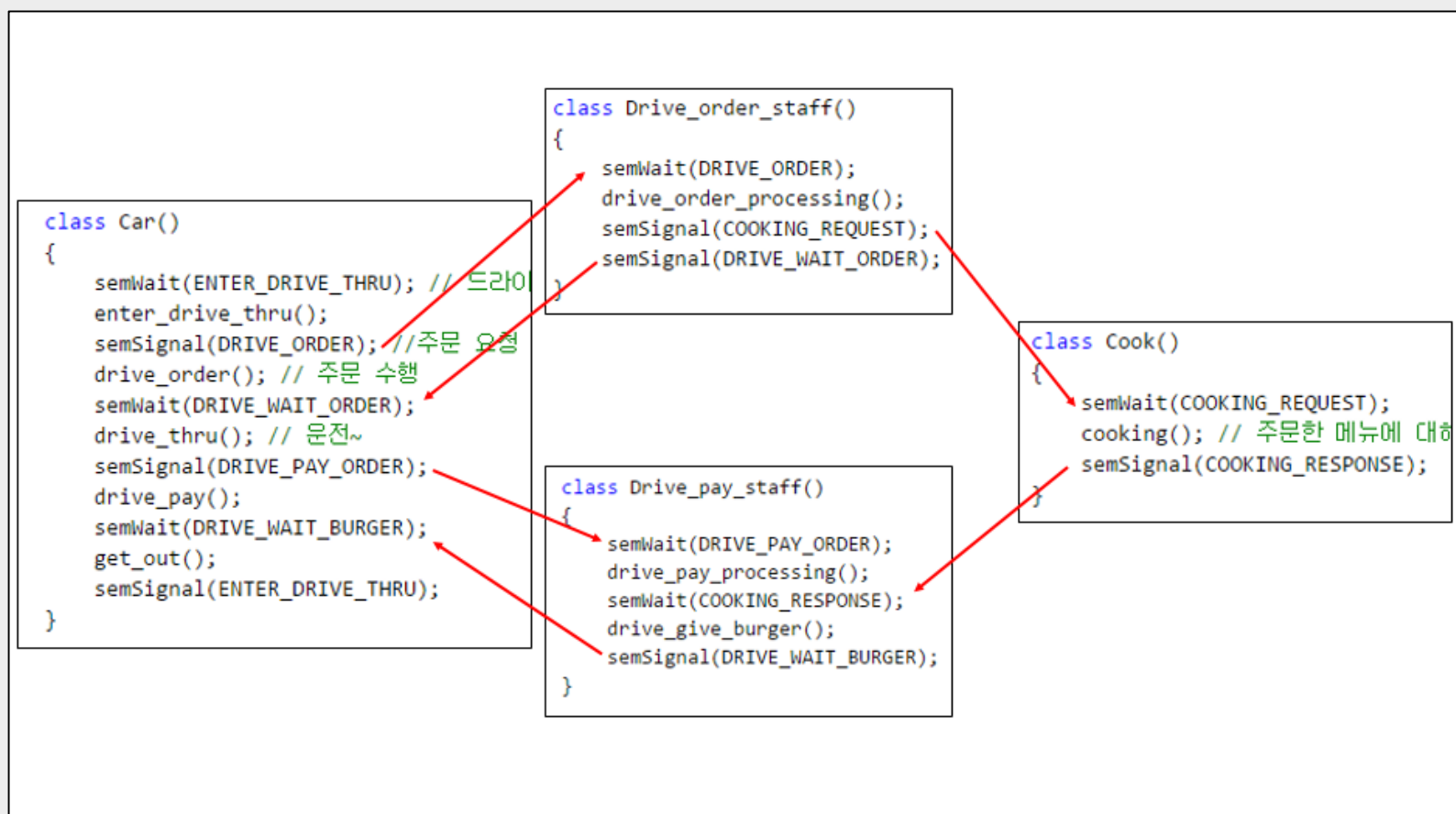
손님, 직원, 캐셔, 요리사 Thread pseudo code



프로젝트 설계

Thread & Semaphore

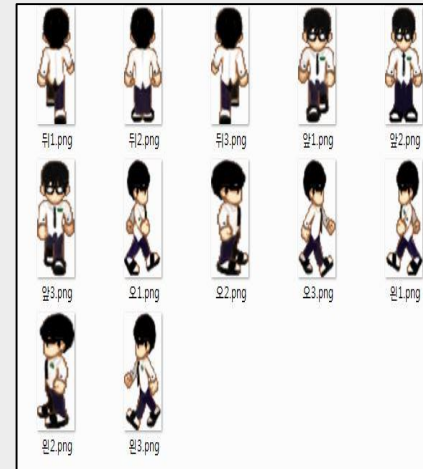
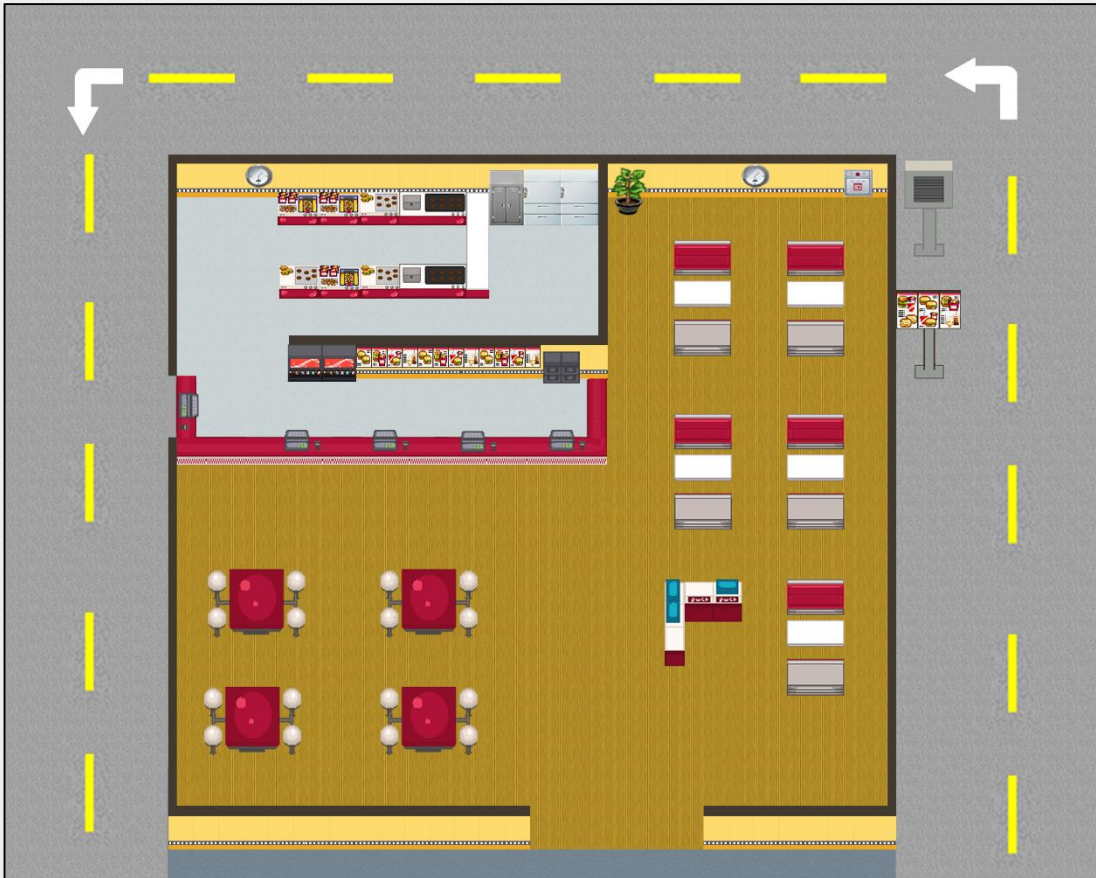
드라이브 스루 자동차 Thread pseudo code



프로젝트 설계

Graphical UI

GUI 화면 및 캐릭터 설계



프로젝트 설계

Tree & Database

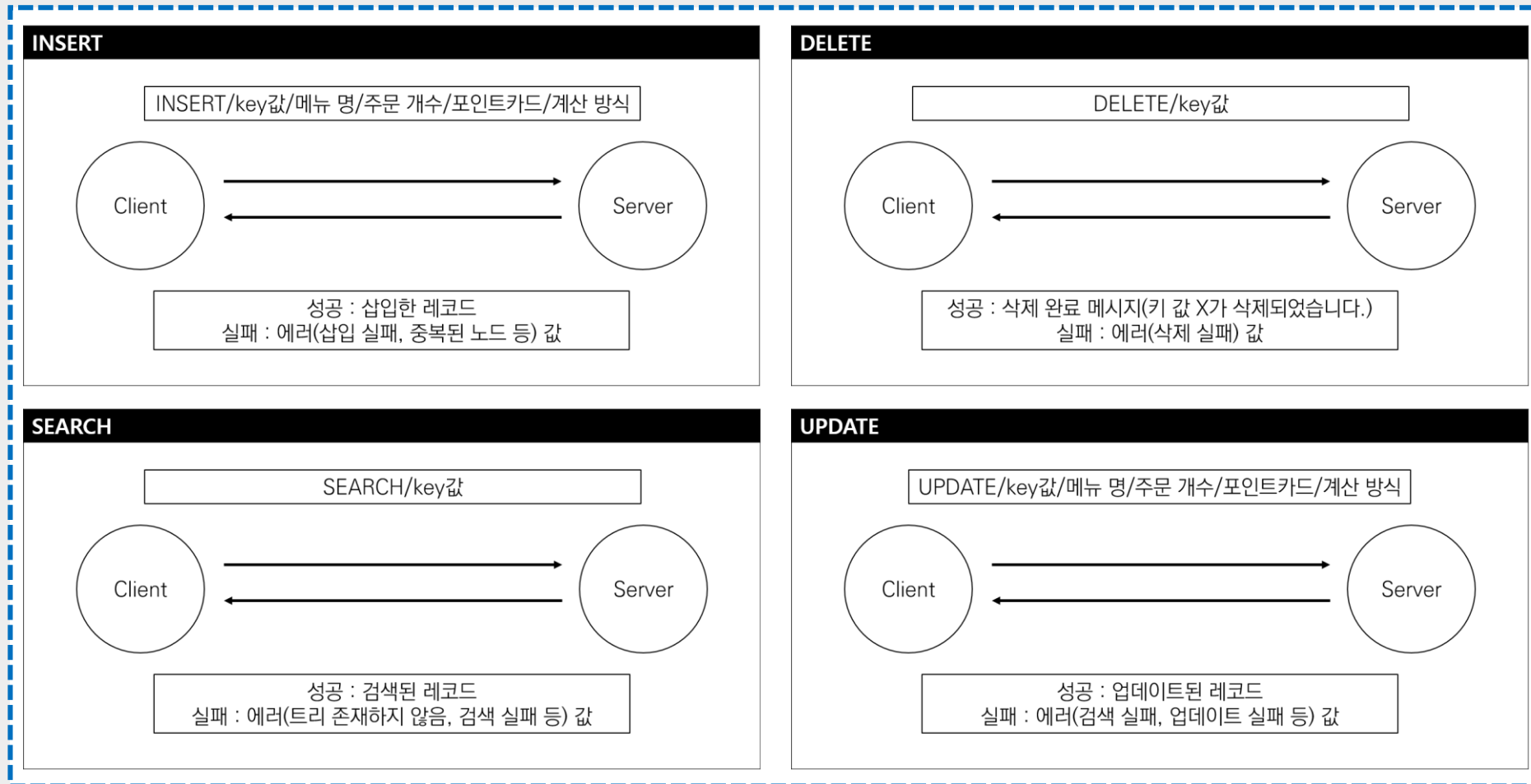
Database 레코드 설계

필드명	데이터 타입	내용
key	int	레코드 키(Primary Key) 값
point_card	bool	포인트 카드 유무
menu	string(char*)	메뉴 주문, 개수, 가격 정보
cal	string(char*)	계산 방식(현금/카드)
nowtime	string(char*)	주문 시각

프로젝트 설계

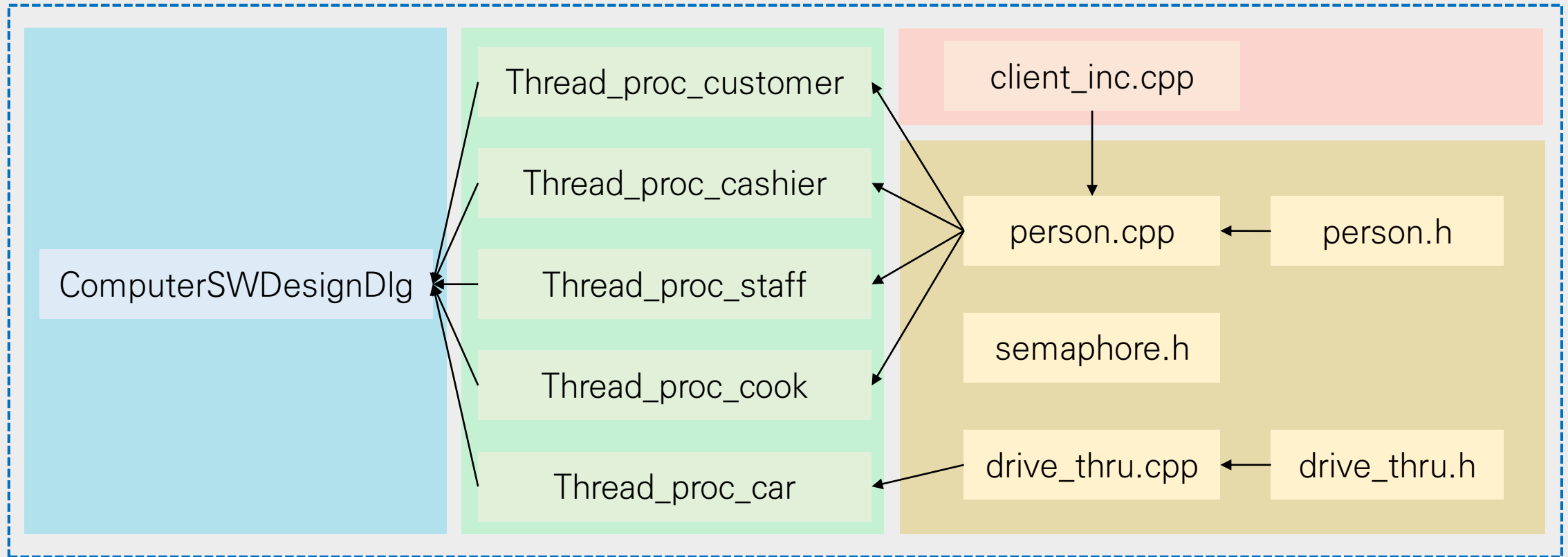
Network

네트워크 통신 프로토콜 설계



프로젝트 구현

전체 프로그램 구조



프로젝트 구현

Thread & Semaphore 생성

ComputerSWDesignDlg.cpp :: OnInitDialog()

```
obj_cstmr = new Customer[50];

/* Create Customer threads */
for (i = 0; i < THREAD_CNT; i++) {
    thread_customer[i] = CreateThread(NULL, 0,
    (LPTHREAD_START_ROUTINE)thread_proc_customer, &obj_cstmr[i],
    0, &customer_id);
    if (thread_customer[i] == NULL) {
        /* error */
        return 1;
    }
}
```

ComputerSWDesignDlg.cpp :: OnInitDialog()

```
smp_gate = CreateSemaphore(NULL,
    MAX_CUSTOMER_CNT, MAX_CUSTOMER_CNT, NULL);
if (smp_gate == NULL) {
    /* error */
    return 1;
}
```

프로젝트 구현

Thread & Semaphore 동작(1/2)

Thread_proc_customer.cpp

```
DWORD WINAPI thread_proc_customer(LPVOID param)
{
    Customer *obj_cstmr = (Customer*)param;
    int cstmr_num;

    do{
        /* 손님 객체의 메소드 호출 */
    } while (loop_flag);
    return TRUE;
}
```

Thread_proc_customer.cpp

```
/* [카운팅] 매장에 입장하는 손님의 수를 제한 */
wait_enter = WaitForSingleObject(smp_gate, INFINITE);
obj_cstmr->enter_Mcd(); // 매장에 입장
```

Thread_proc_customer.cpp

```
/* [상호배제] 손님들의 번호 부여를 위해 전역변수 접근
 * 해당 전역변수에 대한 상호배제 수행이 필요 */
mutex1 = WaitForSingleObject(smp_mutex1, INFINITE);
cstmr_num = customer_number;
customer_number++;
if (!ReleaseSemaphore(smp_mutex1, 1, NULL))
    /* error */
```

Thread_proc_customer.cpp

```
/* [동기화] 직원 앞에 도착했을 때 손님은 주문을 하고,
 * 직원은 손님의 주문을 받는다 */
if (!ReleaseSemaphore(smp_staff[cstmr_num], 1, NULL))
    /* error */
else
    // 주문 직원에게 가서 주문
    obj_cstmr->order(cstmr_num);
```


프로젝트 구현

Thread & Semaphore 동작(2/2)

Person.h

```
class Customer
{
    int num;
    int tcp_sock;
    PT cstmr_point;
    char char_buf[1000];

public:
    Customer();
    ~Customer();

    /* Customer function */
    void enter_Mcd();
    void order(int);
    void pay();
    void find_table();
    void go_to_table(int);
    void eat_burger();
    void leave_table();
    void leave_Mcd();
};
```

Person.cpp

```
void Customer::enter_Mcd(){
    for (int i = 0; i < 30; i++) {
        cstmr_point.y -= 10;
        motion += 1;
        if (motion == 3)
            motion = 1;
        Sleep(150);
    }
}
```

프로젝트 구현

GUI 동작

ComputerSWDesignDlg.cpp :: OnPaint()

```
CPaintDC dc(this);

CDC memDC;
CDC mdcOffScreen;
CBitmap bmpOffScreen;

CBitmap *oldbitmap;

/* ... Bitmap Load ... */

memDC.CreateCompatibleDC(&dc);
textDC.CreateCompatibleDC(&dc);
mdcOffScreen.CreateCompatibleDC(&dc);

bmpOffScreen.CreateCompatibleBitmap(&dc, m_Bitmap.bmWidth,
m_Bitmap.bmHeight);

oldbitmap = mdcOffScreen.SelectObject(&bmpOffScreen);
```

ComputerSWDesignDlg.cpp :: OnPaint()

```
memDC.SelectObject(&m_background);
mdcOffScreen.SetStretchBltMode(COLORONCOLOR);
mdcOffScreen.StretchBlt( ..., &memDC, ...);
```

ComputerSWDesignDlg.cpp :: OnPaint()

```
if (obj_cstmr[i].get_avata() == 1) {
    if (obj_cstmr[i].get_direction() == 1) {
        if (obj_cstmr[i].get_motion() == 1) {
            memDC.SelectObject(&A1_H1);
            mdcOffScreen.TransparentBlt(obj_cstmr[i].get_point_x(),
            obj_cstmr[i].get_point_y(), ..., &memDC, ...);
        }
    }
}
```

ComputerSWDesignDlg.cpp :: OnPaint()

```
dc.BitBlt(... , &mdcOffScreen, ...);
```

프로젝트 구현

Tree & DB

SERVER_AVL.h

```
class AvlNode
{
public:
    int key;
    char menu[MAX_MENU_LENGTH];
    AvlNode *left, *right;
    int height;
    bool point_card; // 포인트카드 유:1 , 무: 0
    char cal[20];    // 지불방식 현금,카드
    char nowtime[sizeof "2016-11-22T06:08:09Z"];

public:
    AvlNode(int key, ... )
    { ... }
    AvlNode(int key, ... )
    { ... }

};
```

AVL.cpp

```
...
/* Avl 트리의 삽입 함수 */
AvlNode* insert(AvlNode* rootNode, AvlNode* newNode);
/* Avl 트리의 삭제 함수 */
AvlNode* deleteNode(AvlNode* root, int key);
/* Avl 트리의 수정 함수 */
void updateNode(int x, AvlNode* node, Tcp_dat buf);
/* Avl 트리의 탐색 함수 */
AvlNode * searchNode(int x, AvlNode* &node);
...
```

ThreadTcpServer.cpp

```
DWORD WINAPI ThClient(LPVOID arg){
    ...
    AvlNode *newNode;
    ...
    while (1) {
        ...
        switch (Commend_str(buf.commend))
        {
        case(0):    ...
        case(1):
            root = insert(root, newNode);
        }
    }
}
```

프로젝트 구현

Network

Socket.cpp :: int Socket()

```
/*winsock 초기화*/  
...  
SOCKET tcp_sock =  
socket(AF_INET, SOCK_STREAM, 0);  
int sock = tcp_sock;  
tcp_sock == INVALID_SOCKET  
...  
//connect()  
retval = tcp_connect(tcp_sock);
```

S_OrderSend.cpp

```
//명령 별 문자열 생성  
int Com_Insert(int tcp_sock, char* char_buf,int key);  
int Com_Delete(int tcp_sock, char* char_buf, int key);  
int Com_Search(int tcp_sock, char* char_buf,int key);  
int Com_Update(int tcp_sock, char* char_buf,int key);  
int Com_randomNode(int tcp_sock, char* char_buf);  
int Com_Save(int tcp_sock, char* char_buf);  
int Com_Load(int tcp_sock, char* char_buf);  
int Com_First(int tcp_sock, char* char_buf);  
int Com_Next(int tcp_sock, char* char_buf, int key);
```

Client

connect

Socket.cpp :: oder_Send(...)

```
/*문자열 전송*/  
tcp_send(...)
```

Socket.cpp :: oder_Send(...)

```
/*문자열 수신대기*/  
tcp_recv(...)
```

Sever

Net_Server.cpp

```
//bind&listen  
int Bind_listen(SOCKET listen_sock);  
//accept  
int Accept_sock(SOCKET listen_sock);
```

ThreadTcpServer.cpp :: int main()

```
//Thread 생성  
CreateThread(...,ThClient, ...);
```

DWORD WINAPI ThClient(...)

```
/*문자열 수신*/  
recv(...)  
/*토큰화*/  
Buf = tokenize_tcpdat(buf, insert_buf);  
Switch(commend)  
{  
    /*client 요청 수행, avl트리*/  
    /*결과 전송*/  
    send(...);  
}
```

프로젝트 구현

File I/O

ThreadTcpServer.cpp :: save

```
fout.open("data.bin", ios::binary | ios::out);
if (fout.is_open()) {
    next_nodekey = root->key;
    do {
        cout << next_nodekey << endl;
        newNode = searchNode(next_nodekey, root);
        if (judge) {
            ...
            fout.write((char*)&file_buf, sizeof(file_buf));
            Next_Node(root, newNode->key);
        }
    } while (!(next_nodekey < 0));
    sprintf(insert_buf, "저장 완료");
}
```

ThreadTcpServer.cpp :: load

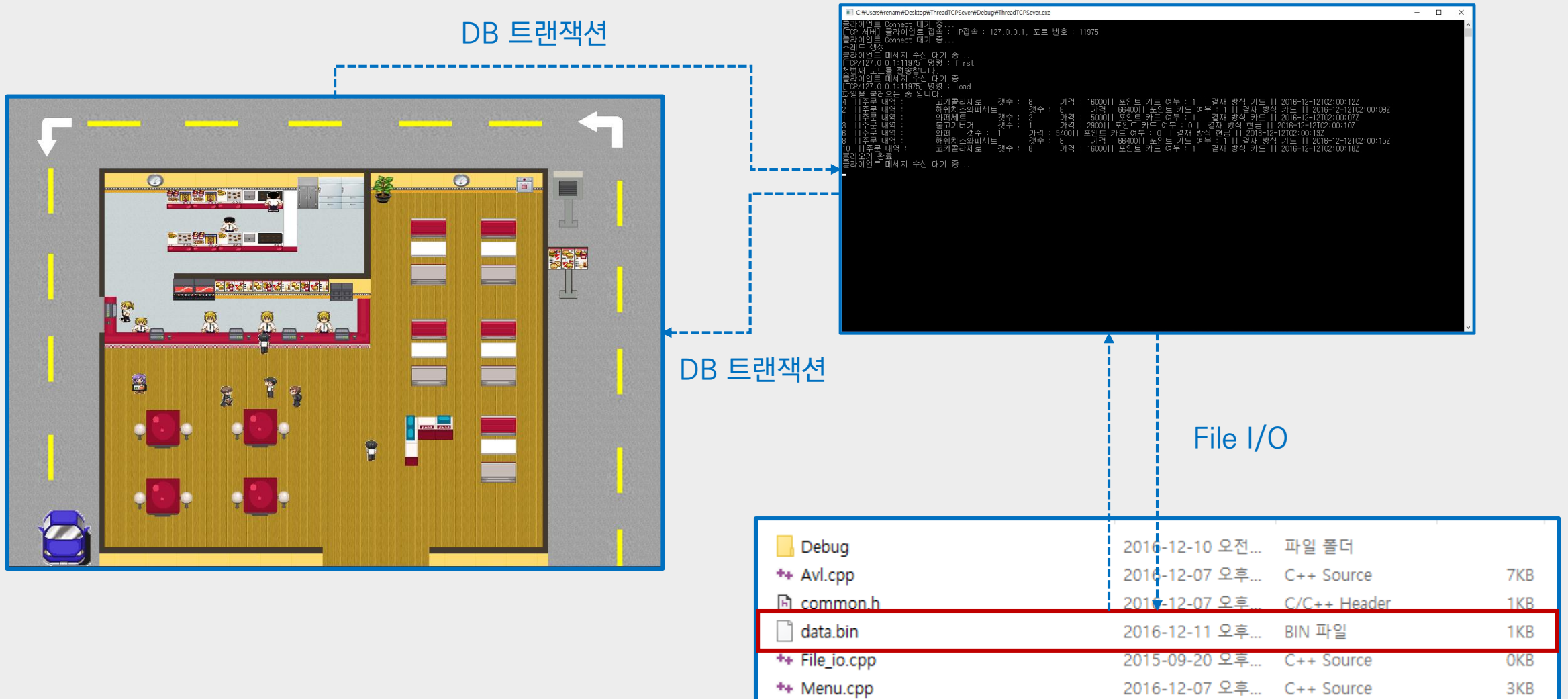
```
fin.open("data.bin", ios::in | ios::binary);
if (fin.is_open()) {
    while ((fin.read((char*)&file_buf, sizeof(file_buf)))) {
        ...
        newNode = new AvlNode(file_buf.key, ...);
        root = insert(root, newNode);
        nodekey.push_back(newNode->key);

        if (key <= file_buf.key)
        {
            key = file_buf.key + 1;
        }
    }
    sprintf(insert_buf, "불러오기 완료");
}
```

Debug	2016-12-10 오전...	파일 폴더	
Avl.cpp	2016-12-07 오후...	C++ Source	7KB
common.h	2016-12-07 오후...	C/C++ Header	1KB
data.bin	2016-12-11 오후...	BIN 파일	1KB
File_io.cpp	2015-09-20 오후...	C++ Source	0KB
Menu.cpp	2016-12-07 오후...	C++ Source	3KB

프로젝트 결과

프로그램 실행 화면



Demo

감사합니다