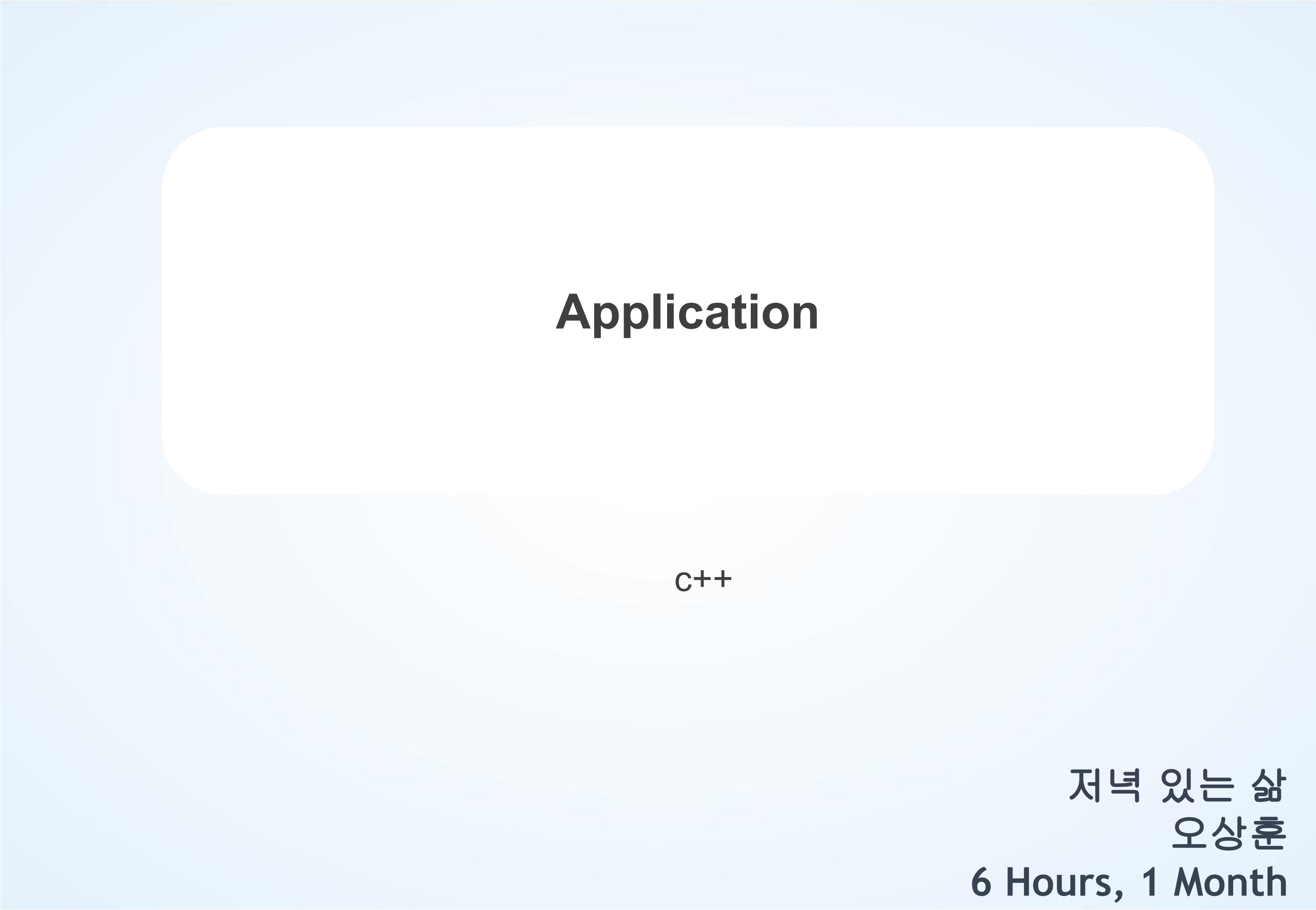


Application

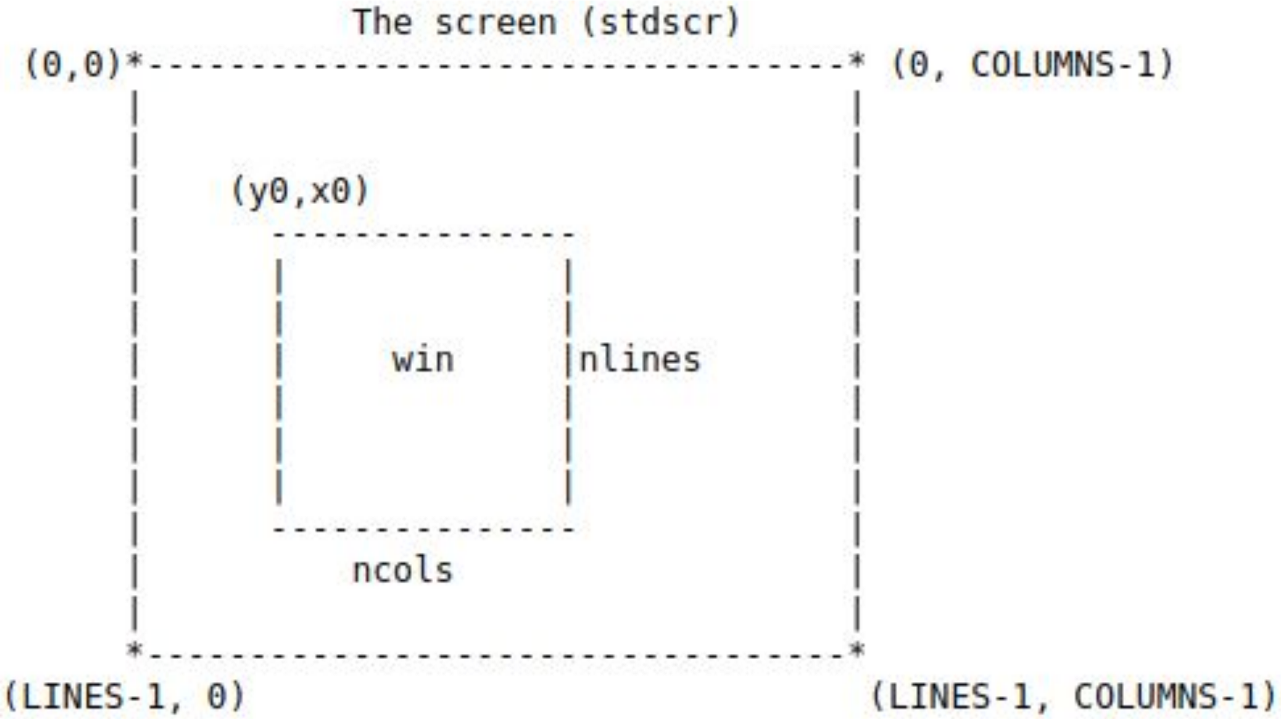
c++

저녁 있는 삶
오상훈
6 Hours, 1 Month

curses(cursor motion optimization)

- refer : http://www.cs.ukzn.ac.za/~hughm/os/notes/ncurses.html
- Terminal control library for Unix-like systems, Manage an application's display on character-cell terminals (e.g., VT100)
  - initscr() : TERM 변수 통해 단말기 종류 결정
  - endwin() : 모든 자원 반납

```
~$ sudo apt install libncurses5-dev
~$ vi helloworldwithcurses.cpp
#include <curses.h>
int main() {
   WINDOW *w;
   w = initscr();
   curs_set(0);        // visible cursor
// start main
   printw("Hello Ncurse!");
   getch();         // Wait keyboard
// end main
   endwin();
   return 0;
}
~$ c++ helloworldwithcurses.cpp -l curses
~$ ./a.out
```



curses - Display with Point

```
~$ vi DisplayWithPoint.cpp
…
#include <string>
int main() {
   …
   using namespace std;                    // start main
   int max_y = LINES-1, max_x = COLS-COLS;                   // define in curses.h
   printw("Start Hello Ncurse!");          // start main
   move(3, 4);   addch('3');   mvaddch(5, 4, '5');
   string str_var = "max_y,max_x:" + to_string(int(LINES)) + ", " + to_string(int(COLS));
   mvaddstr(6, 4, str_var.c_str());
   mvprintw(max_y / 2, max_x / 2, "Middle Hello Ncurse!");
   mvaddch(max_y - 1, max_x - 1, 'E');
   getch();                               // end main
   ...
}
```

❖ 알아가기

- ➢ setup debug mode with curses on VScode

Try - curses - Display with Point

- ❖ 해 보기
  - ➢ Game Map 구성해 보기
- ❖ 실행 결과

```
*********************************************************************************************
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                      Game Start                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*                                                                                           *
*********************************************************************************************
```

curses - Move with keyboard

```
~$ vi MoveWithKeyboard.cpp
...
#define PLAYER '*'  #define WALL 'W'  #define OPPONENT 'O'  #define E_TRACE ' '
int main(void) {
    ...
  keypad(stdscr, TRUE);           timeout(30);      // control game speed
  int y = LINES-1, x = COLS-COLS, move_x = 0, move_y = 0;      // start main
  int ch = 'y';             // define integer for compare with define
  while ((ch != 'q') && (ch != 'Q'))  {
     mvaddch(++move_y, ++move_x, OPPONENT);  // auto moving
     mvaddch(y, x, PLAYER); mvaddch(10, 5, WALL); mvaddch(11, 5, OPPONENT);
     ch = getch();
     switch (ch)      {
     case KEY_UP:
     case 'w':                y = y - 1;    break;
     case KEY_DOWN:      y = y + 1;   break;
     case KEY_LEFT:       x = x - 1;    break;
     case KEY_RIGHT:     x = x + 1;   mvaddch(y, x - 1, E_TRACE); break;
     }
  }                           // end main
    ...
}
```

curses - Collision With Opponent

```
~$vi CollisionWithOpponent.cpp
...
int is_move_okay(int y, int x) {
  int campare_ch;
  campare_ch = mvinch(y, x); // 주어진 위치 문자 return
  return !((campare_ch == 'W'));
}
int main(void) {
      ...
  while ((ch != 'q') && (ch != 'Q'))  {
      ...
      switch (ch) {
      case KEY_UP:
          if (is_move_okay(y - 1, x)) {  y = y - 1;  };      break;
      case KEY_DOWN:
          if (is_move_okay(y + 1, x)) { y = y + 1; };      break;
      case KEY_LEFT:
              if (is_move_okay(y, x - 1)) { x = x - 1; };   break;
      ...
}
```

curses - Move With Struct Player

```
~$vi MoveWithStructPlayer.cpp
...
struct Player {
   static const int shape_size_y = 2, shape_size_x = 2;
   void appear(const int &move_y, const int &move_x) {
       mvaddstr(move_y, move_x, "@@");
       mvaddstr(move_y + this->shape_size_y - 1, move_x, "||");
   }
};
int main(void) {
      …
   Player player01;
   while ((ch != 'q') && (ch != 'Q'))  {
      …
       player01.appear(y, x);
       ch_var=to_string(player01.shape_size_y)+","+to_string(player01.shape_size_x);
       mvprintw(20, 30, ch_var.c_str());
       switch (ch) {
      ...
}
```

- reference
  - https://www.linuxjournal.com/content/creating-adventure-game-terminal-ncurses
  -