**ИТМО Кафедра Информатики и прикладной математики**

Отчет по лабораторной работе №6

«Перехват прерываний в ОС Windows»

**Выполнил: студент группы P3217**

**Плюхин Дмитрий**

**Преподаватель: Зыков А. Г.**

**2017 год**

1. **Задание**

Разработать программу, по функциональности альтернативную 6 лабораторной работе из прошлого семестра (по резидентным программам), выполняющую обработку требуемого события и осуществляющую при этом необходимое действие по крайней мере двумя способами.

1. **Листинг основной части программы**

В лабораторной работе первым способом (с использованием функции SetWindowsHookEx) происходит регистрация обработчика событий нажатия Left Alt, Right Alt, по которым производится подача звукового сигнала и выгрузка обработчика соответственно. Вторым способом (с использованием RegisterHotKey) происходит привязка обработчика нажатия Left Alt и подачи звука.

void releaseHook(){

if (UnhookWindowsHookEx(\_hook)){

showHookReleaseSuccessful();

} else {

showHookReleaseFailed();

}

}

LRESULT \_\_stdcall hookCallback(int hookHandlingCode, WPARAM actionIdentifier, LPARAM eventInfo){

if (hookHandlingCode >= 0){

\_eventInfo = \*((KBDLLHOOKSTRUCT\*)eventInfo);

if (((actionIdentifier == WM\_KEYDOWN) || (actionIdentifier == WM\_SYSKEYDOWN)) && (GetAsyncKeyState(164) != 0)){

if (\_eventInfo.vkCode == 164){

Beep(1000,500);

} else if (\_eventInfo.vkCode == 160){

releaseHook();

}

}

}

return CallNextHookEx(\_hook,hookHandlingCode,actionIdentifier,eventInfo);

}

void setHook(){

if (!(\_hook = SetWindowsHookEx(WH\_KEYBOARD\_LL,hookCallback,NULL,0))){

showHookSetupFailed();

} else {

showHookSetupSuccessful();

}

}

void registerHotKey(){

if (RegisterHotKey(NULL,1,MOD\_ALT | MOD\_NOREPEAT,0)){

showHotkeySetupSuccessful();

} else {

showHotkeySetupFailed();

}

}

int main(int argc, char\* argv[]){

char\* hook = "hook";

char\* hotkey = "hotkey";

setlocale(LC\_ALL,"Russian");

if (argc != 2){

showErrorMessage();

return 1;

} else if (strcmp(argv[1],hook) == 0){

cout << "Hook selected" << endl;

setHook();

} else if (strcmp(argv[1],hotkey) == 0){

cout << "Hotkey selected" << endl;

registerHotKey();

} else {

showErrorMessage();

return 1;

}

MSG msg;

while(GetMessage(&msg, NULL, 0, 0)){

if (msg.message == WM\_HOTKEY){

Beep(1000,500);

}

}

return 0;

}

**3. Результаты работы программы**

|  |  |
| --- | --- |
|  |  |

1. **Вывод**

Таким образом, были исследованы простейшие способы регистрации событий для выполнения действий по нажатии горячих клавиш в Windows. Использование SetHookWindowsEx позволяет делегировать некоторой функции обработку сразу нескольких сочетаний горячих клавиш, применение же RegisterHotKey сопряжено с некоторыми ограничения, основное из которых – невозможность определения нескольких комбинаций клавиш в рамках конкретного процесса, в связи с чем появляются основания считать использование SetHookWindowsEx предпочтительным в общем случае, хотя и несколько более сложным.