Лаб. 1 Начальное знакомство с API ОС

# Задание

Необходимо разработать приложение, которое при помощи чистого WinAPI будет создавать окно, обрабатывать клики мыши и нажатия клавиш, а также сохранять свое состояние и восстанавливать его при последующем запуске.

По умолчанию окно размером 320 на 240 px должно быть закрашено синим цветом с красной сеткой линий NxN.

Размер поля (переменная N) по умолчанию выбирается на усмотрение программиста. Однако, пользователь имеет возможность переопределить размер поля,

указав его в качестве аргумента командной строки при запуске приложения. Окно существует и обрабатывает события до тех пор, пока пользователь не закрыл его. При закрытии окна происходит завершение приложения.

При клике левой клавишей мыши на поле должен быть нарисован

круг определенного цвета в той ячейке поля, в которую произошел клик. При клике правой клавишей мыши на поле должен быть нарисован крестик определенного цвета в той ячейке поля, в которую произошел клик.

Цвет элементов поля (крестиков и кружков) выбирается на усмотрение программиста.

Если пользователь нажал на клавиатуре Ctrl+Q или ESC, окно должно закрыться. Если пользователь нажал на клавиатуре Shift+C, должен запуститься блокнот.

Если пользователь нажал ENTER, фоновый цвет поля должен измениться на случайный.

При прокрутке пользователем колеса мыши текущий цвет линий сетки должен плавно изменяться.

Смену цвета фона необходимо организовать при помощи подмены кисти окна. Использование сообщения WM\_PAINT *для данной подзадачи* не предполагается.

Приложение должно сохранять свое состояние путем реализации чтения/записи конфигурационного файла программы, в котором будут храниться следующие настройки:

Размер поля (N)

Размер окна (ширина и высота) Цвет фона и цвет линий сетки

При первом запуске (когда конфигурационного файла еще нет) программа должна корректно обработать эту ситуаццию,

выбрав для всех параметров указанные в задании умолчания.

Размер поля, указанный пользователем при запуске приложения в качестве аргумента командной строки, имеет приоритет

над значением, записанным в конфигурационном файле.

Формат файла (текстовый, бинарный) - на усмотрение программиста.

Лаб. 2: Межпроцессная коммуникация

# Задание

На основе ранее выполненных лабораторных работ, используя средства межпроцессного взаимодействия, предостав- ляемые ОС Windows, требуется реализовать приложение, синхронизирующее отображение игрового поля между несколькими своими экземплярами.

При запуске двух и более экземпляров программы изменения игрового поля (перемещение, появление или удаление элементов поля), производимые пользователем в любом из них, должны корректно и одновременно отображаться на интерфейсе всех экземпляров.

Вне зависимости от порядка запуска и закрытия экземпляров приложения, при запуске первого экземпляра поле должно корректно создаваться и очищаться; при закрытии последнего - также корректно удаляться.

Синхронизация цвета и размера (в пикселях) игрового поля не требуется, однако подразумевается, что размерность игрового поля (т.е. число клеток) фиксировано для всех экземпляров приложения и считывается из конфигурационного файла при запуске каждым экземпляром самостоятельно - таким образом, нет нужды синхронизировать размерность поля в явном виде.

Для выполнения лабораторной предполагается использование механизмов широковещательной отправки сообщений оконной подсистемы и разделения памяти, но допустимы и иные варианты по предварительному согласованию с преподавателем.

В данной лабораторной не учитываются эффекты, требующие реализации механизмов взаимного исключения.

Лаб. 3: Процессы и потоки

# Задание

Выполнение лабораторной работы предполагается на основании предыдущих лабораторных работ. Студенту необходимо вынести отрисовку поля в отдельный поток, анимировав отрисовку фона (например, при помощи переливающегося градиента). Должна быть реализована возможность устанавливать приоритет потока отрисовки путем

нажатия цифровых клавиш (например, 1 для установки фонового приоритета, 2 - низшего и т.д.). По нажатию клавиши "пробел" поток отрисовки должен приостанавливать и возобновлять свое выполнение.

Также необходимо добавить механизмы исключения хода одного из игроков (игроки должны ходить по очереди в строгой последовательности). При попытке сходить в чужой ход должно выдаваться сообщение об ошибке. После каждого хода должна выполняться проверка состояния игрового поля на предмет выполнения условий завершения (построение линии из N элементов для крестиков-ноликов, заполнение поля фишками для реверси) и определяться игрок-победитель, о чем выдаваться соответствующее сообщение. После закрытия этого сообщения процесс должен корректно завершаться, подчищая за собой все ресурсы . Последний из завершающихся процессов должен также очистить игровое поле и удалить все разделяемые ресурсы.