

### Задача 3. Сапер.

Написать аналог игры «Сапер» (“Minesweeper”) из состава стандартных программ для Windows OS. Архитектура программы должна быть основана на паттерне MVP (Mode-View-Presenter).

#### Требования

Поведение должно быть такое же, как у оригинальной игры:

Игровое поле состоит из клеток. Вначале все клетки закрыты. В каких-то клетках спрятаны мины.

1. При нажатии левой кнопкой мыши на закрытую клетку она открывается.
  - a) Если там мина, конец игры (проигрыш).
  - b) Если мины нет, но вокруг клетки есть мины, отображается цифра, обозначающая количество мин вокруг
  - c) Если мин вокруг нет, отображается пустая открытая клетка и открываются все клетки вокруг нее. Если вокруг нее есть другие пустые клетки, открываем все соседние для каждой пустой и так далее.
2. При нажатии правой кнопкой мыши на закрытую клетку, она помечается флажком. Если на ней уже есть флажок, он убирается.
3. При нажатии левой кнопкой мыши на клетку с флажком ничего не происходит.
4. Если открыты все клетки без мин, конец игры (выигрыш)
5. В интерфейсе отображается количество оставшихся мин (рассчитывается как реальное количество мин минус количество поставленных флажков)
6. При нажатии колесиком на открытую клетку с цифрой открываются все клетки вокруг нее, при условии, что вокруг этой клетки стоит столько флажков, какая цифра на клетке. Алгоритм открытия соседних клеток такой же, как в пункте 1. Вместо колесика такой же эффект должно производить нажатие на левую и правую кнопку одновременно – в качестве усложнения, если останется время.
7. Размер поля и количество мин можно изменить. По умолчанию поле размером 9x9, количество мин 10.
8. Игра должна поддерживать таблицу рекордов, которая сохраняется между запусками программы. Для этого необходимо запускать секундомер при первом нажатии на поле.
9. Пользователю должно быть доступно меню с командами: *Exit, About, New Game, High Scores*.



#### Реализация графического UI

1. Для реализации пользовательского интерфейса использовать библиотеку Swing (javax.swing.\*).
2. Мины и флажки отображать с помощью картинок.
3. Для расположения элементов на игровой панели рекомендуется использовать класс GridBagLayout.
4. Для расположения ячеек поля рекомендуется использовать класс GridLayout.