**Домашнее задание № 4**

**Программирование GUI**

В домашнем задании требуется реализовать графическую версию домашнего задания №3 (лебедь, рак и щука), а именно, программу симулятор-визуализатор синхронизированного перемещения тележки под воздействием сил нескольких существ.

Предполагается, что графический интерфейс должен иметь как минимум три основных компонента:

1. Меню, с помощью которого можно задавать параметры симуляции.
2. Панель, визуализирующая движение тележки на плоскости.
3. Панель запуска и остановки симуляции.

Необходимо реализовать поддержку следующих параметров симуляции:

1. Угол направления движения каждого из существ (градусы 0-360).
2. Интервал распределения непрерывной равномерной случайной величины, задающей сдвиг для каждого из существ (переменная *si*).
3. Интервал распределения дискретной равномерной случайной величины, задающей время сна для каждого из существ в миллисекундах.
4. Продолжительность симуляции в секундах.
5. Начальные координаты тележки.

Все параметры должны проверяться на корректность. В случае некорректного ввода необходимо информировать пользователя об этом и сообщать о возможных значениях параметра (N.B.: сообщение исключения с трассировкой стека не является информацией о некорректном вводе для пользователя). В качестве значений по умолчанию, необходимо использовать значения из ДЗ №3.

Помимо установки значений параметров симуляции, меню также должно:

1. Обладать функционалом корректного выхода из приложения (пункт меню “Exit” или “Выход).
2. Показывать информацию о приложении и его авторе (пункта меню “About” или “О программе”).

Панель, визуализирующая движение тележки на плоскости, должна отображать:

1. Тележку и её направление, от которого отмеряются углы направления существ.
2. Существ и их направление движения.
3. Траекторию движение тележки.

Панель запуска и остановки симуляции должна иметь три кнопки:

1. Start (Начать симуляцию) – начинает симуляцию используя заданные параметры симуляции визуализируя процесс на панели визуализации. Должна становиться недоступной после того, как симуляция началась и до нажатия на кнопку Reset.
2. Stop (Остановить симуляцию) – останавливает симуляцию, если она запущена. Недоступна, если симуляция не запущена.
3. Reset (Сброс) – сбрасывает состояние симуляции на начальное.

Для реализации необходимо использовать одну из следующих технологий: AWT, Swing, JavaFX. Выбор остаётся за автором приложения.

Оценивание:

При оценке работ учитывается стиль и качество кодирования (инкапсуляция, наличие пакетов, Javadocs и т.п.).

За отсутствие потоков - максимальная оценка 2.   
При некорректной синхронизации потоков, представляющих существ и тележку – максимальная оценка 6.

Оценка 8: реализация базовой функциональности с корректной синхронизацией потоков.  
Оценка 9 -10: ставится за реализацию дополнительного функционала или за “качественный” код.

Архив должен загрузиться в moodle до дедлайна в **zip** архиве.

Имя архива должно выглядеть следующим образом:

{**group\_number**}\_{**surname**}\_{**first\_name\_letter**}.zip, где

**group\_number** - номер группы

**surname** - фамилия транслитом

**first\_name\_letter** - первая буква имени (транслитом)

например: 171\_Ivanov\_I.zip

Если архив назван не по гайдлайну: минус 2 балла от оценки.

В архиве должно быть:

1. проект в IDEA. Проект должен компилироваться на JDK 8.
2. исполняемый jar файл, который должен запускаться в JRE 8.
3. README.txt (опционально)

Исходный код должен соответствовать Java Code Convention: <https://www.oracle.com/technetwork/java/codeconventions-150003.pdf>

За грубые нарушения допустимо снижение оценки (максимальное снижение 3 балла)