**Лабораторная работа №1**

**«Системная динамика»**

**Основная часть**

**1. Модель реализации продукта по Бассу**

Модель реализации продукции описывается системой уравнений:

Начальное значение:

В модели приняты следующие условные обозначения для накопителей:

* *PotentialAdopters* (Потенциальные потребители продукции);
* *Adopters* (Потребители, которые уже купили продукт).

Поток, моделирующий процесс потребления обозначен как *AdoptionRate*.

В модели используются переменные:

* *AdoptionFromAd* - число потребителей продукта, которые его приобрели под влиянием рекламы;
* *AdoptionFromWOM* - число потребителей продукта, которые его приобрели под влиянием общения с потребителями, которые уже купили продукт.

Интенсивность процесса, приобретения продукта моделируется потоком *AdoptionRate*.

Константы-параметры модели:

* *TotalPopulation* (Численность населения);
* *ContactRate* (Число контактов);
* *AdEffectivenes* (Эффективность рекламы);
* *AdoptionFraction* (Сила убеждения);

Модель создается с «нуля». Построение модели начинается с создания накопителей *PotentialAdopters* и *Adopters*, соединенных потоком *AdoptionRate*. Для создания модели нужно использовать палитру «Системная динамика».

Что бы создать поток, соединяющий накопители нужно:

Разместить накопители, задать им имена (см. рисунок 1).

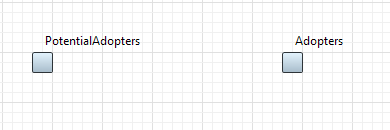


Рис. 1 Размещение накопителей

Выделите накопитель *PotentialAdopters*, затем выполните на нем двойной щелчок левой кнопкой мыши и соедините его с помощью стрелки потока с накопителем *Adopters*, выполнив на нем щелчок мышью .

Затем нужно присвоить потоку имя *AdoptionRate*, так как это показано на рисунке 2.

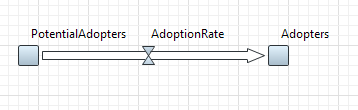


Рис. 2. Накопители, соединенные потоком.

Затем следует разместить параметры модели и переменные в соответствии с уравнениями модели.

В таблице 1. приводятся значения параметров модели.

Таблица 1. Параметры динамической модели

|  |  |  |
| --- | --- | --- |
| **№** | **Параметр** | **Значение** |
| 1 | TotalPopulation | 100000 |
| 2 | ContactRate | 100 |
| 3 | AdEffectivenes | 0.011 |
| 4 | AdoptionFraction | 0.015 |

Вид модели показан на рисунке 3.

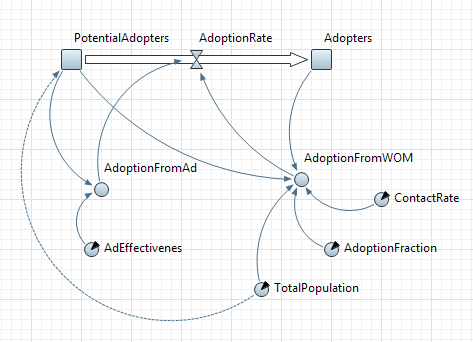


Рис. 3. Модель реализации продукта

После установления всех связей, параметров и переменных, как показано на рисунке 3, необходимо задать начальные значения параметров из таблицы 1 и прописать для потока и динамических переменных формулы, приведенные ранее. Пример на рисунке 4.

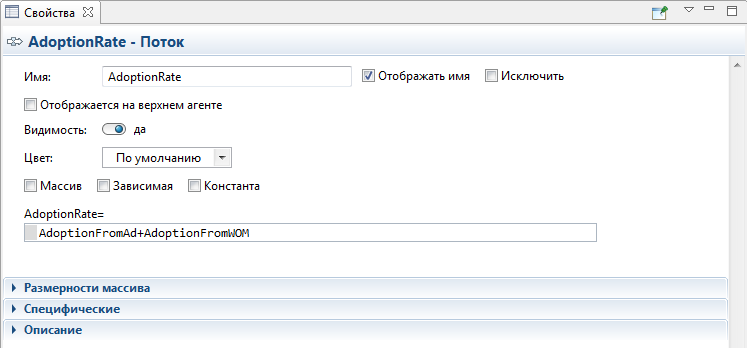


Рис. 4. Задание свойств потока

Чтобы проследить за процессами, протекающими в модели, следует разместить временные графики.

Первый график показывает динамику изменения числа потенциальных потребителей продукта и числа лиц, которые приобрели продукт.

Для получения наглядных графиков при моделировании следует период обновления (в разделе Обновление данных) выбрать равным 0.1, а количество отображаемых точек задать равным 100.

На вкладке проекта в разделе Simulation установить в подразделе Модельное время следующие параметры:

Остановить -> В заданное время

Конечное время -> 8

Протестируйте модель. Вид графиков работающей модели должен соответствовать рисунку 5.

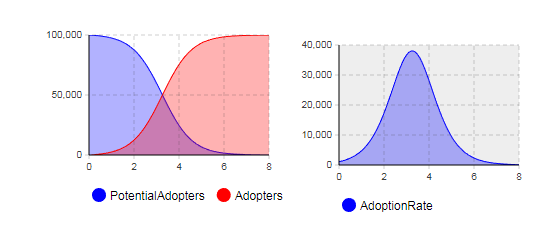


Рис. 5. Динамика процессов модели Басса

**2. Учет повторных покупок**

Создайте модель, которая моделирует процесс реализации продукции по Бассу с учетом, того что продукт, купленный потребителем со временем приходит в негодность и потребитель покупает новый продукт на его замену.

Методические указания:

Модель реализации продукта примет вид:

Начальное значение:

Здесь *DiscardRate* – новый поток, отражающий повторные покупки. Для его определения в модели Басса вводится формула:

Где: *ProductLifeTime*=2(года) – параметр, который задает время годности продукта в процессе его использования пользователем.

Функция f представляет собой задержку и вычисляется в зависимости от двух аргументов: основного потока и константы, которая характеризует задержку – время жизни продукта.

Чтобы получить такую функцию в AnyLogic нужно использовать встроенную функцию delay.

Модель реализации продукта примет вид, показанный на рисунке 6.

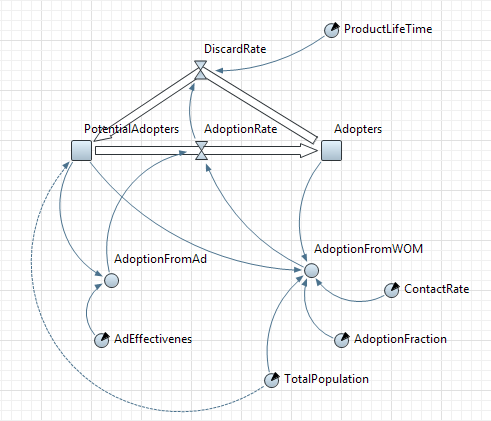


Рис. 6. Модель, созданная с учетом совершения повторных покупок

Протестируйте модель. Динамика процессов модели должна соответствовать рисунку 7.

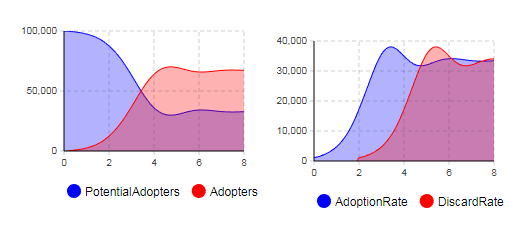


Рис. 7. Реализация продукта, с учетом повторных покупок

На втором графике показана также динамика изменения интенсивности повторных покупок Discard rate.

**3. Модель распространения эпидемии**

Постройте модель распространения эпидемии. Модель описывается системой уравнений:

Здесь *get\_stick* – интенсивность протекания заболевания, *get\_well* – интенсивность выздоровления. Параметры *infectionRate*=0.00218 и *recoveryRate*=0.5 – факторы,

влияющие на процесс заболевания и выздоровления.

Накопители:

* *susceptible* – не заболевшие;
* *infected* – инфицированные;
* *recovered* – выздоровевшие.

Постройте временные графики для трех накопителей. Динамика изменения процесса эпидемии показана на рисунке 8.

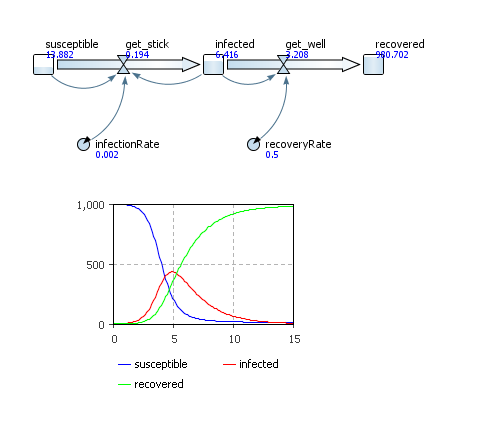


Рис. 8. Динамические процессы протекания заболевания

**4. Взаимодействие активных классов**

Разработать модель изменения динамики численности городского населения.

Данная модель должна учитывать динамику роста населения в зависимости от жилищных условий.

Построение такой модели следует выполнить на базе двух взаимодействующих активных классов (см. рисунок 9).

populationSector

housingSector

Рис. 9. Взаимодействующие классы

**Активный класс сектора населения**

Активный класс *populationSector* (сектор населения) моделирует динамку роста численности населения, а класс *housingSector* (жилищный сектор) динамику роста жилья.

Для построения модели сектора населения нужно использовать параметры, показанные в таблице 2.

Таблица 2. Параметры сектора населения

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Параметр** | **Описание** | **Значение** |
| 1 | birthRate | Уровень рождаемости | 0.03 |
| 2 | imigrationNormal | Коэффициент миграции | 0.1 |
| 3 | populationInitial | Начальная численность населения | 50000 |
| 4 | averageLifetime | Средняя продолжительность жизни | 64 |
| 5 | householdSize | Среднее количество человек в составе семьи | 4 |
| 6 | emigrationNormal | Доля эмиграции | 0.07 |

Уравнение системной динамики численности населения примет вид:

Где:

* *births* – уровень рождаемости;
* *imigration* – уровень миграции;
* *deaths* – уровень смертности;
* *emigration* – уровень эмиграции.

Переменные модели:

Переменные: houses(число домов, которые построены в городе) и *householdsToHousesRatio* (заселенность города) образуют интерфейс активного класса сектора населения, *attractionDueToHousing* (спрос на жилье).

Переименуйте текущий класс **main** в **populationSector**.

Потоки:

В поле активного класса постройте по уравнению системной динамики, с учетом переменных, параметров и потоков модель, используя палитру «Системная динамика». Вид модели должен соответствовать рисунку 10.

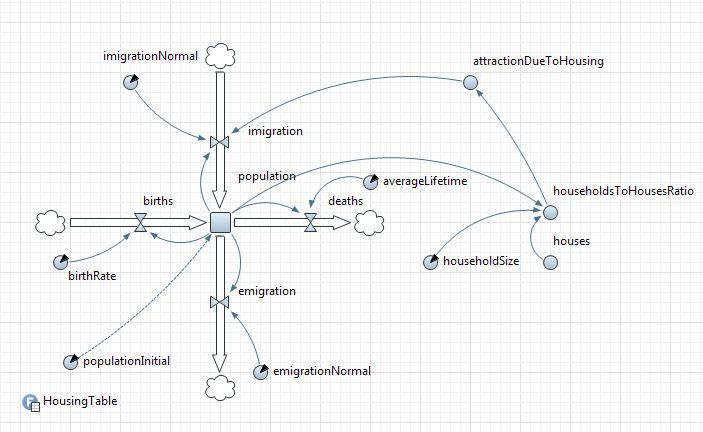


Рис. 10. Активный класс сектора населения

Примечание: Если в качестве источника в накопитель входит поток, то он создается из динамической переменной палитры «Системная динамика».

Аналогично создается поток, если данные передаются от накопителя потоку.

При построении активного класса сектора населения нужно в его поле добавить функцию *HousingTable*. Эта функция создается с помощью инструмента «Табличная функция» панели системная динамика. После размещения функции в класс нужно на вкладке «Основные» ее свойств задать ей имя *HousingTable*, тип интерполяции должен быть линейный, если аргумент функции выходит за пределы табличных значений, то выбирается ближайший. Затем следует сформировать таблицу значение функции так, как это показано на рисунке 11.



Рис.4.11. Значения функции

Для возможности взаимодействия активного класса с другим активным классом следует создать интерфейс. Для активного класса сектора населения интерфейс будет образован двумя переменными *houses* и *householdsToHousesRatio*.

Для создания интерфейса класса разместите в поле класса из палитры «Презентация» элемент «Скругленный прямоугольник» и активизируйте его свойство «Значок». В созданный прямоугольник поместите фигуру человека. Для этого откройте палитру «Картинки» и разместите изображение человека с помощью одноименного элемента. У изображения активизируйте свойство «Значок».

Создайте копии переменных *houses* и *householdsToHousesRatio*. Для этого выделите нужную переменную, вызовите контекстное меню и выполните команду «Создать копию» (см. рисунок 12).



Рис. 12. Контекстное меню динамической переменной

Оставьте копии переменных на модели, а сами переменные расположите на контуре значка активного класса, так как это показано на рисунке 13. Все связи, которые вели к оригинальным переменным, необходимо провести к копиям.



Рис. 13. Значок активного класса

Для переменной houses установите свойства «Отображается на верхнем уровне» и «Зависимая» в состояние активности. Такая настройка позволяет создать внешнюю зависимую переменную, которая может принимать значение из другого активного класса.

Для переменной *householdsToHousesRatio* установите свойство «На верхнем уровне» в состояние активности. Такая переменная может передавать значение в другой активный класс.

**Активный класс сектора жилищного строительства**

Создайте новый активный класс *housingSector*. Для этого во вкладке проекты нажмите правой кнопкой мыши на имя Вашей модели. В открывшемся меню выберите «Создать -> Тип агента». Имя нового типа задайте «housingSector» и пункт «Создать новый тип с нуля». После описанных шагов нажмите «Готово». Активный класс обладает набором параметров, которые приводятся в таблице 3.

Таблица 3 . Параметры сектора жилищного строительства

|  |  |  |  |
| --- | --- | --- | --- |
| **№** | **Параметр** | **Описание** | **Значение** |
| 1 | constructionNormal | Норма возведения нового жилья | 0.07 |
| 2 | landPerHouse | Доля городской земли на строение | 0.1 |
| 3 | area | Городская площадь | 8000 |
| 4 | housesInitial | Начальное число домов в городе | 14000 |
| 5 | demolitionNormal | Норма сноса ветхого жилья | 0.015 |

Уравнение системной динамики жилищного строительства примет вид:

Переменные модели:

Описание переменных приводится в таблице 4.

Таблица 4. Переменные сектора жилья

|  |  |  |
| --- | --- | --- |
| **№** | **Переменная** | **Описание** |
| 1 | constrMultiplier | Интенсивность строительства |
| 2 | constrDueToHousingAv | Реальная потребность в строительстве |
| 3 | constrDueToLandAv | Земельные участки, имеющиеся для строительства |
| 4 | fractionOfOccupiedLand | Доля использованной под застройку земли |
| 5 | housesExport | Число возведенных домов |
| 6 | householdsToHousesRatio | Заселенность города |

Вид модели приводится на рисунке 14.

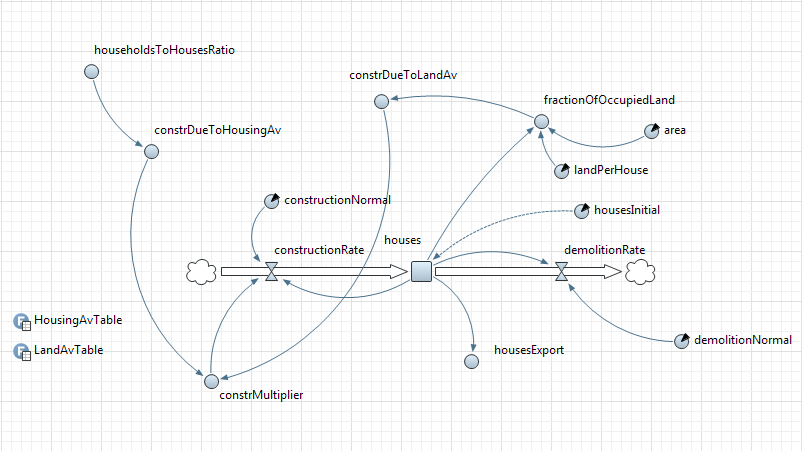


Рис. 14. Активный класс сектора жилищного строительства

В модель нужно разместить две табличные функции *HousingAvTable* и *LandAvTable*. Значение таблицы первой функции приводятся на рисунке 15, а значения второй функции на рисунке 16.



Рис. 15. Таблица функции *HousingAvTable*

Обе функции должны реализовывать линейный метод интерполяции, при отсутствии значения берется ближайшее значение.



Рис. 16. Таблица функции *LandAvTable*

Создайте значок для представления активного класса. В качестве элементов значка используйте «Скругленный прямоугольник» и изображение здания из палитры «Картинки» элемент «Дом». Вид значка показан на рисунке 17.



Рис. 17. Значок активного класса сектора жилья

Создайте интерфейс активного класса. Получите копии переменных *householdsToHousesRatio* и *housesExport*. Копии оставьте на модели, переменные поместите на значок, так как это показано на рисунке 17. Настройте свойства переменных таким образом, чтобы переменная *householdsToHousesRatio* могла принимать значение из активного класса, а переменная *housesExport* передавать значение в активный класс.

**Настройка корневого объекта модели**

Создайте новый класс main и перетащить в его рабочую область по одному экземпляру классов populationSector и housingSector. Настройте интерфейсные переменные значков таким образом, чтобы не отображались их имена.



Рис. 18. Соединение активных классов

Используйте элемент «Соединитель» палитры «Агент» и соедините классы с помощью их интерфейсов, так как это показано на рисунке 18.

Для запуска симуляции класса main необходимо будет переключить на вкладке «Проекты» у сущности Simulation:\*\*\* в свойствах, что агентом верхнего уровня будет выступать класс main.

Данные можно передавать внешней переменной класса от переменной интерфейса класса источника. Внешние переменные очерчены не сплошной линией. При правильном соединении стрелка должна быть направлена от переменной класса источника к внешней переменной класса приемника.



Рис. 19. График динамики роста населения

Разметите временной график для отображения роста численности городского населения - *population*. Для обращения к переменной population требуется использовать форму записи с использованием имени класса: populationSector.population.

Настройте эксперимент модели. На вкладке «Модельное время» задайте, что модель должна останавливаться в заданное время, конечное время задайте равным 100 единиц.

Протестируйте модель, вид графика изменения численности населения должен соответствовать рисунку 19.

**Визуализация модели динамики численности населения**

Создайте дополнительный временной график, для вывода динамики изменения городской площади, отведенной под жилищное строительство. Вид графика показан на рисунке 20.



Рис. 20. Изменение площади под застройку



Рис. 21. Распределение населения

Создайте столбиковую диаграмму, которая отображает количество родившихся жителей, умерших, покинувших город, прибывших в город (см. рисунок 21).

Протестируйте модель, вид графиков должен соответствовать контрольным рисункам.

**Дополнительная часть**

Доработать модель эпидемии по следующим критериям:

1. Ввести естественный прирост населения;
2. Ввести фактор смертности;
3. Добавить вероятность повторного заражения спустя некоторый период после выздоровления;
4. Отразить влияние мер противодействия развитию эпидемии на динамику протекания процесса.