**23. Общая характеристика СМО**

Состояние СМО характеризуется состояниями составляющих ее объектов.

Состояние ОА выражается булевыми величинами, значение которых интерпретируется как true (занято) и false (свободно), и длинами очередей на входах ОА, принимающими неотрицательные целочисленные значения, называемые переменными состояния или фазовыми переменными.

Хранилища (памяти) характеризуются общим объемом, занятым (свободным) объемом и длинами очередей.

Правило, согласно которому заявки выбирают из очередей, называют дисциплинами обслуживания, а величину, выражающую преимущественное право на обслуживание – приоритетом. Приоритеты бывают:

1. абсолютный приоритет (прерывает);

2. относительный (ждет окончания);

3. динамический (может меняться).

В приоритетных дисциплинах для заявок на входе ОА выделяется своя очередь. Заявка из очереди с низким приоритетом поступает на обслуживание, если пусты очереди с более высокими приоритетами.

Исследование поведения СМО, т. е. определение временных зависимостей переменных, характеризирующих состояние СМО, при подаче на входы любых, требуемых в соответствии с заданием на эксперимент потоков заявок, называют имитационным моделированием СМО. Имитационное моделирование проводят путем воспроизведения событий, происходящих одновременно или последовательно в модельном (системном) времени. При этом под событием понимают факт изменения значения любой фазовой переменной.

Подход, альтернативный имитационному моделированию, называют аналитическим исследованием СМО. Оно заключается в получении формул для расчета выходных параметров СМО с последующей подстановкой значений аргументов в эти формулы в каждом отдельном эксперименте.

Соответствующие модели СМО называются имитационными и аналитическими.

Аналитические модели (АМ) удобны в использовании, поскольку для них не требуются значительные затраты вычислительных ресурсов. Часто без постановки специальных вычислительных экспериментов можно оценить характер влияния аргументов на выходные параметры, выявить те или иные общие закономерности в поведении системы. Но АМ удается получить только для частных случаев сравнительно не сложных СМО. Для сложных СМО АМ если и удается получить, то только при принятии упрощающих допущений, ставящих под сомнение адекватность модели.

Поэтому основным подходом на системном уровне считают имитационное моделирование, а АМ используют при предварительной оценке различных предлагаемых вариантов систем.

**24. Аналитическая модель СМО**

При *аналитическом моделировании* математическая модель реализуется в виде такой системы уравнений относительно искомых величин, которая допускает получение нужного результата аналитически (в явном виде) или численным методом. Аналитическое решение, если оно возможно, дает более полную и наглядную картину, позволяющую получать зависимость результатов моделирования от совокупности исходных данных. С точки зрения задач оптимизации параметров и стратегии управления в системе аналитическое моделирование практически незаменимо. Аналитическое моделирование обладает определенными недостатками, которые становятся особенно существенными на заключительных этапах создания системы, когда уже приняты стратегические решения, и речь идет об оценке характеристик некоторого сравнительно небольшого числа вариантов окончательного построения системы, причем на этих этапах при моделировании требуется учитывать достаточно детальное описание системы. В этом случае аналитическое описание системы становится чрезмерно сложным, что затрудняет получение требуемых результатов. В данной ситуации следует переходить к использованию имитационных моделей.

**25. Имитационная модель СМО**

Имитационная модель предназначена отображать структуру и внутренние связи в моделируемой системе. Правильность теории построения модели может быть проверена только на практике. Так как при построении моделей используется упрощения и аппроксимация, то модель не является абсолютно точной в смысле однозначности соответствия реальной системе, следовательно, существуют различные степени корректности и точности моделей. Имитационная модель не занимается поиском истины. Она позволяет получить лишь серию приближений к истинной модели.

При оценке правильности построения имитационной модели необходимо рассматривать всю цепочку ее преобразования, начиная от входных данных, формального представления и заканчивая результатами моделирования. Поэтому оценку адекватности обычно рассматривают на уровне концепции, на уровне реализации имитационной модели и уровне элементов, связанных с прогоном модели. Разработка концептуальной модели связана с мыслительным процессом. Поэтому такую проверку начинают с постановки задачи и заканчивают проверкой концептуальной модели. Проверку начинают с анализа концептуальной модели, возвращаются к принятым аппроксимациям и затем рассматривают реальный процесс. Другой метод проверки концептуальной модели связан с привлечением экспертов, которые не принимают участие в разработке моделей.

Проверка достоверности модели на уровне ее реализации связана с рассмотрением следующей цепочки ее преобразований. Эта логическая схема (алгоритм работы моделей), схема программы и сама программа модели. Фактически, необходимо на этом этапе ответить на следующие вопросы:

1) дает ли возможность замысел модели решить поставленную задачу;

2) точно ли отражен этот замысел в логической схеме;

3) полная ли предложенная логическая схема и обладает ли она необходимой последовательностью;

4) правильно ли использованы математические уравнения;

***Имитационная модель*** в принципе позволяет воспроизвести весь процесс функционирования с сохранением логической структуры, связи между явлениями и последовательность протекания их во времени. Имитационное моделирование позволяет учесть большое количество реальных деталей функционирования моделируемого объекта и является незаменимым на финальных стадиях создания системы, когда все .стратегические. вопросы уже решены, и речь идет лишь об отборе одного из конкретных вариантов реализации , причем описание этих вариантов должно быть весьма детальным (вплоть до описания конкретного оборудования и конкретных протоколов, реализуемых в системе). Отметим, что имитационное моделирование предназначено для решения задач расчета системных характеристик. Количество вариантов, подлежащих оценке, должно быть относительно небольшим, поскольку осуществление имитационного моделирования для каждого варианта построения требует значительных вычислительных ресурсов. Дело в том, что принципиальной особенностью имитационного моделирования является тот факт, что для получения содержательных результатов необходимо использовать статистические методы. Данный подход требует многократного повторения имитируемого процесса при изменяющихся значениях случайных факторов с последующим статистическим усреднением (обработкой) результатов отдельных однократных расчетов. Применение статистических методов, неизбежное при имитационном моделировании, требует больших затрат машинного времени и вычислительных ресурсов. Другим недостатком метода имитационного моделирования является тот факт, что для создания достаточно содержательных моделей (а на тех этапах создания , когда применяется имитационное моделирование, нужны весьма детальные и содержательные модели) требуются значительные концептуальные и программистские усилия.

**28. Языки моделирования ориентированы на описание событий**

(Ориентированные на описание событий)

GASP, Симскрипт, SMPL

Сущность событийного метода заключается в отслеживании на модели последовательности событий в том же порядке, в котором они происходили бы в реальной системе. Вычисления выполняют только для тех моментов t и тех частей (процедур) модели, к которым относится совершаемое событие.

Моделирование начинается с просмотра операторов генерирования заявок. Для каждого независимого источника это позволяет рассчитать момент генерации первой заявки. Этот момент с именем-ссылкой на заявку заносится в СБС, а сведения о генерируемой заявке в список заявок – СЗ. Запись в СЗ включает в себя имя заявки, значение ее атрибутов, место, занимаемое в данный момент СИМ. События в СБС упорядочиваются по мере увеличения моментов наступления.

Затем из СБС выбирается событие, относящееся к наиболее раннему моменту времени и переносится в СТС. Обращение к СЗ позволяет определить все сведения о заявке, в том числе, место, где она находится в СИМ. Далее программа моделирования выполняет следующие действия:

Интерпретация

Продвижение

заявки А в СИМ

Выбор заявки В на обслуживание

Накопление статистики

Прогноз будущих событий

СБС

СТС

1. изменяет параметры состояния устройства, к которому заявка обратилась. Например, если заявка А освобождает устройство Х, а очередь к нему не была пуста, то в соответствие с заданной дисциплиной обслуживания, из очереди к Х выбирается заявка В и поступает на обслуживание к Х;

2. прогнозируется время наступления следующего события, связанного с заявкой В, путем обращения к модели устройства Х, в которой рассчитывается продолжительность обслуживания заявки В, сведения об этом будущем событии заносятся в СБС и СЗ.

3. происходит имитация движения заявки А в СИМ по маршруту, определенному данной программой, пока заявка либо задерживается в очереди, либо путем обращения к модели ОА прогнозируется наступление будущего события, сведения о котором заносятся в СБС и СЗ;

4. в файл статистики заносятся необходимые данные.

После обработки всех событий, относящихся к моменту ТК, происходит увеличение модельного времени до значения, соответствующему ближайшему будущему событию и процесс имитации повторяется.

**29. Языки моделирования ориентированы на представление процессов**

SOL, aspol, GPSS

Каждая процедура описывающая систему содержит ряд процедур связанных с выполнением процесса.

Каждая процедура описывающая процесс является описанием класса процесса и в любой момент времени это описание может выполняться в разных стадиях выполнения для различных заявок.

Любой процесс может находиться в одном из 4х состояний:

1) готовности;

2) выполнения;

3) задержки;

4) ожидания.

**ЯЗЫКИ МОДЕЛИРОВАНИЯ ориентиров.. на …-** совокупность языковых и программных средств, обеспечивающих простое и эффективное моделирование всех процессов проектирования, формализацию и воспроизведение динамических свойств моделируемых объектов, как сложных систем, а также имитацию стохастических процессов, генерацию и анализ случайных величин и временных рядов. Я.м. в АПКС представляют из себя систему моделирования, которая включает специализированные средства автоматизации моделирования всех проектных процедур и операций и состоит из следующих компонентов: язык описания, объект моделирования, средства обработки языковых конструкций, управляющую программу моделирования, осуществляющую имитацию во времени и набор стандартных программных средств, реализующих дополнительные возможности по организации модельных экспериментов на вычислительной технике. **Языки имитационного моделирования** делятся на две группы, соответствующие двум видам имитации: для непрерывных и дискретных систем. Группа языков **непрерывного моделирования** делится на языки аналогового моделирования и языки, которые применяются для решения систем дифференциальных уравнений, описывающих детерминированные замкнутые непрерывные системы. **Языки моделирования дискретных систем** в зависимости от подхода к описанию объектов имитации и метода организации квазипараллелизма делятся на следующие виды: ориентированные на просмотр **активностей**. В этом случае алгоритм системы представляется последовательным выполнением активностей; составляющие расписания событий, в котором программа модели организована в виде совокупности так называемых процедур обслуживания событий; управляющие процессами, в которых моделирующая программа организуется в виде набора описаний процессов, каждое из которых соответствует одному классу процессов," организующие взаимодействие транзактов, в них при описании программы модели используются транзакты (заявки на обслуживание); ориентированные на взаимодействие агрегатов, из которых скомпонована сложная система, в этом языке моделирования в качестве динамических объектов выступают агрегаты. В последние годы широкое применение получили языки моделирования, обеспечивающие имитацию непрерывно-дискретных систем. В этом случае все компоненты сложных систем, у которых состояния изменяются дискретным образом, описываются аналогично процессному способу имитации, а для описания остальных компонент используются такие же уравнения, как для языков непрерывного моделирования.

**Системы Simula и SmallTalk как первые средства объектного моделирования**

**(так и не понял нахера Данс это сюда втулил???)**

Рассмотрим языки программирования Simula и SmallTalk, которые являются родоначальниками объектного подхода, с целью выявления тенденций, которые они заложили в развитие объектного моделирования систем реального физического мира.

Алгоритмический язык Simula 67 (от “simulation” – моделирование) является первым полноценным объектно-ориентированных языком, который представлял собой все основные концепции объектного моделирования (инкапсуляция, полиморфизм, наследование). Simula был разработан в 1967 году для моделирования дискретных событий в реальном физическом мире.

Для инкапсуляции (сокрытие реализации, принцип “черный ящика”) в Simula было введено строгое контролирование к доступу информации, однако “черный ящик” в объектной модели является в открытым к изменениям. Создавая свою модель на подобием существующей (наследуясь от нее) можно сделать изменения в базовой модели для своего объекта. Это достигается за счет техники полиморфизма.

В Simula имеется поддержка моделирования физических процессов реального времени. Для этого Simula может оперировать понятиями Модель (Simulation) и Процесс (Process).

На базе объектной модели Simula, в 1970г Алан Кей и Ден Ингалс, создали язык SmallTalk, соединив объектно-ориентированное моделирование и графическое отображение. SmallTalk является первым языком использующим моделирование поведения объектов с помощью модели состояний. В SmallTalk объекты взаимодействуют за счет посылаемых друг другу сообщений, так называемых “событий”. Подобная концепция описания взаимодействия объектов является одним из основополагающих моментов объектного моделирования.

SmallTalk является графически-ориентированным языком и графическая среда разработки одновременно. Его логические схемы можно строить только на графическом дисплее. Таким образом, SmallTalk разрушает представление о языке программирования, как о исключительно символьной информации.

Ингал формулирует цель создания SmallTalk так:"…сделать мир информации доступным для детей любого возраста". Т. Бадд и А. Кей отмечают что данная цель была достигнута, т.к. обучать SmallTalk детей легче чем профессиональных программистов с устоявшимися взглядами

Сформулируем тенденции заложенные Simula и SmallTalk

* 1. В обучении физике эффективными языками объектного моделирования являются гибридные языки программирования, сочетающие в себе полноценную поддержку объектных концепций и возможность алгоритмических предписаний
  2. Объектное моделирование физических систем обретает простую и наглядную для понимания форму при графической интерпретации объектных моделей
  3. Эффективным средством обучения детей объектному моделированию физических систем является среды графического объектного моделирования
  4. Эффективность объектного моделирования физических систем повышается, если имеется среда моделирования, содержащая в себе все необходимое для построения моделей