

Билет 10

1. Этапы решения задачи с использованием ЭВМ.

В конце 40-х – начале 50-х годов двадцатого века ошибки программирования не представляли серьезных проблем вследствие ограниченности возможностей ЭВМ и отсутствия опыта их применения. Однако, с развитием техники ситуация изменилась. И со временем встал вопрос о методах разработки программ, включающих в себя создание спецификаций, разработку алгоритмов, программ, а также методов доказательства их правильности. Главная идея созданных методик стоит в том, что процесс разработки программы представляет собой последовательность этапов. Успешность выполнения каждого из них зависит от полноты и правильности информации, предоставляемой предыдущем этапом.

Обычно выделяют следующие:

1) **Постановка задачи** – определяются цели и условия решения задачи, раскрывается ее содержание, выявляются факторы, оказывающие влияние на ход вычислений или конечный результат.

2) **Формализация** – по результатам выяснения сущности задачи определяется объект и специфика исходных данных, вводится система условных обозначений, устанавливается принадлежность поставленной задачи к одному из известных классов задач и выбирается математический аппарат описания.

3) **Выбор или разработка метода решения.** Построение на предыдущем этапе мат. модели приводит к необходимости поиска ее решения, т. е. устанавливается зависимость результатов от исходных данных, подбираются методы решения задачи, пригодные для реализации на ЭВМ.

4) **Составления алгоритма** – часто процесс решения задачи не удается получить в виде явной формулы. В этом случае метод решения задачи преобразовывается в последовательность действий ЭВМ, называемой алгоритмом. Главное назначение алгоритма – точное и детальное описание процесса переработки исходных данных в результат.

5) **Составление программы** – составленный алгоритм записывается на языке программирования, после чего программа компилируется в коды ЭВМ. Результат – рабочая программа.

6) **Отладка программы** – на любом из предшествующих этапов могли быть допущены ошибки, которые делают код программы неработоспособным. Отладка состоит в выявлении и устранении грубых ошибок.

7) **Тестирование** – предназначено для выявления грубых ошибок, которые не нарушают работоспособность программы, но не дают ей возможности выдавать правильный результат. Для выявления подобного несоответствия необходимо заготовить тесты, которые не только выявляют ошибочность функционирования, но и позволяют локализовать подозрительное на ошибку место программы.

2. Классификация центральных процессоров Intel и соответствующих локальных и системных шин ПЭВМ типа IBM PC.

Первым 16 разрядным процессором фирмы Intel стал выпущенный в 1978 году **i8086**. В нём применялась 16 разрядная внутренняя и внешняя шина данных и 20

разрядная адресная шина, что позволяло адресоваться к 1 Мб памяти. Процессор выпускался по 2 микронной CMOS технологии и содержал 29000 транзисторов. Работал на частотах от 4 до 12 МГц.

В 1979 году появился «младший брат» i8086 - процессор **i8088**. Это тот же процессор, но имеющий 8 разрядную внешнюю шину данных, что позволяло удешевить систему в целом.

В 1982 году был выпущен процессор **i80286**. Он имел 16 разрядную шину данных и 24 разрядную шину адреса и мог работать в 16 битном защищённом режиме. В этом режиме он мог адресоваться к 16 Мб физической памяти и 1Гб виртуальной. В реальном режиме был полностью совместим с i8086 плюс имел дополнительные инструкции. Процессор работал на частотах от 6 до 20 МГц и содержал 134 тысячи транзисторов.

В 1983 году появился процессор **i80186**. В реальном режиме он был совместим с i80286, но не имел защищённого режима. Процессор работал на частотах от 6 до 20 МГц. Так же как и i8086 имел модификацию i80188 с 8 разрядной внешней шиной данных. Не получил распространения на рынке персональных компьютеров и применялся только для построения различных контроллеров.

В 1985 году появился революционно новый процессор - **i80386**. Это первый 32 разрядный процессор фирмы Intel. Он имел 32 разрядные шины данных и адреса, что позволяло адресоваться к 4 Гб физической и 64 Тб виртуальной памяти. Процессор мог работать как в реальном режиме, так и в защищённом 32 битном режиме. Так же имелся и виртуальный режим 8086 процессора - аппаратная эмуляция нескольких 8086 процессоров для многозадачных ОС. Процессор был выполнен по 0.8 микронной технологии и содержал 275 тысяч транзисторов. Работал на частотах от 12 до 33 МГц. Все современные процессоры совместимы как с защищенным, так и виртуальным режимом процессора i80386.

В 1988 году был выпущен процессор **i80386SX**. Он был полностью совместим с i80386, но имел 16 разрядную внешнюю шину данных. Выпускались модификации с тактовыми частотами от 16 до 33 МГц. С этого времени процессор i80386 получил название i80386DX.

В 1989 году появился процессор **i80486DX**. Это совместимый с i80386 процессор с повышенным быстродействием, который имел в своём составе 8 кб кэша первого уровня со сквозной записью и выполненный на одном кристалле с процессором математический сопроцессор. В нём использовалось 1.2 миллиона транзисторов. Выпускались модели с частотами от 20 до 50 МГц.

Процессор **i80486SX** появился в 1991 году, был полностью совместим с i80486DX, но не имел на кристалле математического сопроцессора. Содержал 900 тысяч транзисторов и работал на частотах от 16 до 33 МГц.

В 1992 году был выпущен процессор с удвоением внутренней частоты кристалла - **i80486DX2**. При внешней частоте шины от 20 до 33 МГц внутренняя частота составляла от 40 до 66 МГц.

В 1993 году появился процессор 5 поколения - **Pentium**. В нём была применены суперскалярная архитектура, механизм предсказания переходов, поддержка мультипроцессорности. Кэш первого уровня построен по Гарвардской архитектуре (отдельные 8 кб кэша с обратной записью для хранения инструкций и команд). Разрядность внешней шины данных увеличена до 64 бит. Процессор содержал 3.1 миллиона транзисторов. Первые версии процессоров Pentium P5 выпускались по 0.8

микронной технологии для частот 60 и 66 МГц. В ранних версиях процессоров обнаружена ошибка в математическом сопроцессоре. В 1994 году был выпущен процессор **Pentium P54C** по 0.6 микронной технологии с частотой 75-100 МГц, в 1995 году производство перешло на 0.35 микронную технологию и тактовую частоту удалось поднять до 200 МГц. Процессоры Pentium P54C, как и i80486DX2, умножают внутреннюю тактовую частоту кристалла (при частоте шины от 50 до 66 МГц внутренняя частота составляет от 75 до 200 МГц, т.е. коэффициент умножения может быть от 1.5 до 3.0). На данный момент снят с производства.

В 1994 году появился процессор **i80486SX2** - версия процессора i80486SX с удвоением частоты.

В 1994 году был выпущен процессор **i80486DX4** - процессор i80486DX с утроением частоты кристалла, работающий на частотах 75 и 100 МГц. В его состав входил кэш первого уровня с обратной записью объёмом 16 кб. Процессор выполнялся по 0.6 микронной технологии и содержал 1.6 миллиона транзисторов.

1995 год - год рождения процессора 6 поколения - **Pentium Pro**. В одном корпусе с кристаллом процессора расположен кэш второго уровня объёмом от 256 кб до 1 Мб, работающий на внутренней частоте кристалла. Ускорена по сравнению с Pentium работа конвейера и степень параллелизма, улучшена система предсказания переходов. Внутренняя структура оптимизирована на выполнение 32 разрядных приложений. Выпускался по 0.6 и 0.35 микронной технологии (5.5 миллионов транзисторов) на частоты от 150 до 200 МГц. Позиционировался как процессор для серверов. В настоящее время снят с производства.

В 1997 году появился процессор **Pentium MMX P55C**. Фактически, это процессор Pentium P54C с блоком выполнения инструкций MMX (MultiMedia eXtention), предназначенных для работы с мультимедийными данными и увеличенным до 32 кб (16 кб данные + 16 кб инструкции) кэшем первого уровня. Выполнялся по 0.35 микронной технологии (4.5 миллиона транзисторов) для частот от 166 до 233 МГц. В настоящее время снят с производства.

В этом же году был представлен процессор **Pentium II**. Это процессор Pentium Pro с усовершенствованным блоком выполнения 16 битных инструкций и двумя блоками выполнения инструкций MMX. Производится по технологии 0.35 и 0.25 микрон (7.5 миллиона транзисторов) на частоты от 233 до 450 МГц с кэшем второго уровня, расположенном в картридже процессора ёмкостью 512 кб и работающего на половине частоты ядра процессора. Модели на 350 МГц и выше работают на частоте внешней шины 100 МГц (в отличие от младших моделей, работающих на частоте шины 66 МГц). Модели Pentium II 233-266 МГц сняты с производства.

В 1998 году был выпущен процессор **Celeron - Pentium II** без кэша второго уровня (модели 233-266 МГц) и с кэшем второго уровня 128 кб (модели 300-400 МГц). Изготавливаются по технологии 0.25 микрон.

В этом же году был представлен процессор **Pentium II Xeon**. Он предназначен для замены устаревших Pentium Pro и предназначен для использования в серверах и high-end рабочих станциях. В отличие от Pentium II этот процессор имеет кэш второго уровня от 512 кб до 1 Мб, работающий на частоте ядра процессора. Выполняется по технологии 0.25 микрон и работает на частотах 400-450 МГц.

В 1999 году был выпущен процессор **Pentium III** - процессор Pentium II плюс набор 70 инструкций KNI (Katmai New Instruction), предназначенных для работы с трёхмерной графикой. В отличие от Pentium II объём кэша первого уровня увеличен

до 64 кб (32+32 кб). Выпускается по технологии 0.25 микрон (в дальнейшем по 0.18 микрон).

В первом компьютере IBM PC в 1981 году была использована системная шина **PC/XT-bus**. Она была 8-разрядной и работала на частоте процессора 4.77 МГц. Теоретическая пропускная способность шины была 4.77 Мб/с.

В компьютерах PC/AT стала применяться 16-разрядная системная шина **ISA** (Industry Standart Architecture). Аппаратно она была совместима со старыми 8-разрядными картами расширения. Благодаря 24 адресным линиям шина могла напрямую обращаться к 16 Мб системной памяти. Количество линий аппаратных прерываний было увеличено с 7 до 15, а каналов DMA - с 4 до 7. Стандартная частота шины 8 МГц, теоретическая пропускная способность 16 Мб/с.

С появлением процессоров 80386, а затем и 80486 системная шина ISA стала тормозом на пути увеличения общей производительности системы. Новая шина должна была быть 32-разрядной, обеспечивать автоматическую конфигурацию системы и плат расширения. Такой шиной стала **EISA** (Extended Industry Standart Architecture). В пакетном режиме она обеспечивала теоретическую скорость 33 Мб/с. Однако, данная шина не получила большого распространения и использовалась преимущественно в мощных файл-серверах и рабочих станциях.

Шина **PCI** является надстройкой над локальной шиной процессора, т.к. связана с ней логически и электрически с помощью специального контроллера - «моста» (bridge), благодаря чему одновременно к шине может быть подключено до 10 устройств. Ещё одно преимущество PCI перед VLB в том что шина работает на одной частоте 33 МГц независимо от внешней частоты процессора. К тому же шина PCI обеспечивает автоконфигурирование системы и поддерживает стандарт Plug-n-Play. В настоящее время шина VLB полностью вытеснена шиной PCI.

3. Реалистическая графика. Обратная трассировка луча. Полная модель освещения Уиттеда.

С развитием вычислительной техники и ее возможностей все больший интерес вызывает задача построения по заданной модели сцены изображения, максимально приближенного к реальному. Построенное таким образом изображение должно восприниматься человеком как настоящее, реалистическое. Такие задачи возникают в самых разных областях человеческой деятельности - от рекламы и кинематографа до архитектурных сооружений и создания реалистических изображений новых только проектируемых машин и т.п.

Рассмотрим самые простейшие методы создания реалистических изображений.

Прямой трассировкой лучей называется процесс расчета освещения сцены с испусканием от всех источников лучей во всех направлениях. При попадании на какой-либо объект сцены луч света может преломившись уйти внутрь тела или отразившись далее продолжить прямолинейное распространение до попадания на следующий объект и так далее. Следовательно, каждая точка сцены может освещаться либо напрямую источником, либо отраженным светом. Часть лучей в конце концов попадет в глаз наблюдателя и сформирует в нем изображение сцены.

Понятно, что вычисления, необходимые для трассировки всех лучей для всех источников и поверхностей слишком объемисты. Причем существенный вклад в полученное изображение внесет лишь небольшая часть оттрассированных лучей.

Для избавления от излишних вычислений используется **обратная трассировка**, в которой вычисляются интенсивности только лучей, попавших в глаз наблюдателя. В простейшей реализации обратной трассировки отслеживаются лучи, проходящие из глаза наблюдателя через каждый пиксел экрана в сцену. На каждой поверхности сцены, на которую попадает луч, в общем случае формируются отраженный и преломленный лучи. Каждый из таких лучей отслеживается, чтобы определить пересекаемые поверхности. В результате для каждого пиксела строится дерево пересечений. Ветви такого дерева представляют распространение луча в сцене, а узлы - пересечения с поверхностями в сцене. Окончательная закраска определяется прохождением по дереву и вычислением вклада каждой пересеченной поверхности в соответствии с используемыми моделями отражения. При этом различают и обычно по-разному рассчитывают первичную освещенность, непосредственно получаемую от источников света, и вторичную освещенность, получаемую от других объектов.

Итак, идея обратной трассировки лучей. Для определения цвета пиксела экрана через него из камеры проводится луч, ищется его ближайшее пересечение со сценой и определяется освещенность точки пересечения. Эта освещенность складывается из отраженной и преломленной энергий, непосредственно полученных от источников света, а также отраженной и преломленной энергий, идущих от других объектов сцены. После определения освещенности этой точки учитывается ослабление света при прохождении через прозрачный материал и в результате получается цвет точки экрана.

Для определения освещенности, привносимой непосредственным освещением, из точки пересечения выпускаются лучи ко всем источникам света и определяется вклад всех источников, которые не заслонены другими объектами сцены. Для определения отраженной и преломленной освещенности из точки выпускаются отраженный и преломленный лучи и определяется привносимая ими освещенность.

Алгоритм работы функции трассировки луча с началом $o = (ox, oy, oz)$ и направлением $d = (dx, dy, dz)$, возвращающей, кстати, освещенность по всем трем цветовым компонентам, будет, таким образом, выглядеть примерно так:

1. ищем ближайшее пересечение луча со сценой (определяем точку p , где произошло пересечение, выясняем, с каким конкретно объектом оно произошло)
2. если не нашли, возвращаем 0
3. если пересечение есть, текущую освещенность полагаем равной фоновой освещенности (ambient)
4. определяем вектор n , нормаль к объекту в точке пересечения для каждого (точечного) источника света
5. считаем вектор l , соединяющий источник и точку p (начало в p)
6. ищем пересечение луча с началом в точке p и направлением d со сценой
7. если нашли, прекращаем обработку этого источника, так как его не видно
8. считаем симметричный l относительно n вектор r
9. считаем по уравнению Фонга (оно использует n , l , r , d) освещенность в точке за счет этого источника света, добавляем ее к текущей

10. считаем симметричный d относительно n вектор $refl$, то есть отраженный вектор

11. трассируем отраженный луч (с началом p и направлением $refl$), добавляем полученную освещенность к текущей

12. считаем направление преломленного луча (вектор $refr$) по d , n и коэффициентам преломления

13. трассируем преломленный луч (с началом p и направлением $refr$), добавляем полученную освещенность к текущей

14. покомпонентно умножаем текущую освещенность на цвет объекта

15. если для материала, заполняющего пройденную лучом от o до p трассу $\beta \neq 0$, умножаем текущую освещенность на коэффициент ослабления

16. возвращаем текущую освещенность

Простейшей моделью освещенности является **модель Уиттеда**, согласно которой световая энергия, покидающая точку в направлении вектора g , определяется формулой

$$I = k_a I_a + k_d C \sum_j I_j(n, l_j) + k_s \sum_j I_j(s, r_j)^p + k_r I_r e^{-\beta_r d_r} + k_t I_t e^{-\beta_t d_t}$$

где k_a, k_d, k_s - веса фонового, диффузного и зеркального освещения;

k_r, k_t - веса отраженного и преломленного лучей;

C - цвет объекта в точке P ;

n - вектор нормали в точке P ;

l_j - направление на j -е источник света;

I_j - цвет j -го источника света;

b_r, b_t - коэффициенты ослабления для сред, содержащих отраженный и преломленный лучи;

d_r, d_t - расстояния от точки P до ближайших точек пересечения отраженного и преломленного лучей с объектами сцены;

s - вектор наблюдения

r_j - вектор отражения

h_j - вектор, задаваемый формулой

p - коэффициент Фонга.