

УДК 370.30+519.92

КОМПОНЕНТНОЕ ПРОГРАММИРОВАНИЕ В КУРСЕ ИНФОРМАТИКИ: ЛОГИКО-ПСИХОЛОГИЧЕСКИЙ АСПЕКТ

Варбанец С.В.

Рассмотрены связь между методами научного исследования и этапов вычислительного эксперимента при обучении программированию.

Человеческое познание в его историческом и индивидуальном становлении и развитии проходит ряд основных этапов. Первым является этап чувственно-образного познания на основе активного и непосредственного взаимодействия с миром. Ощущения – первичные формы чувственного познания, отражающие те или иные отдельные свойства объектов (объектом может выступать как предмет, так и явление). Оставшиеся в памяти представления, основанные на прошлых ощущениях – это первичная форма знаний, образующая фундамент. В отличие от ощущения, восприятие отражает объект в целом, но оно, как и ощущение непроизвольно, возникает в результате воздействия объекта на органы чувств. В процессе восприятия происходит опознание объекта и его оценка, в том числе на значимость для субъекта. При этом происходит не только синтез текущих ощущений, но и анализ воспринимаемого. В ходе умственного развития на базе конкретно-

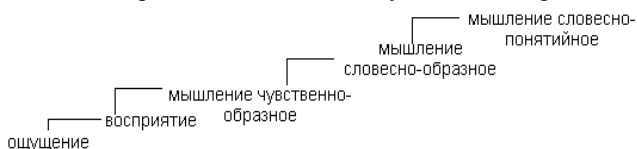


Рисунок 1 – Цепочка познания реального мира человеком

чувственного знания с помощью слов образуются понятия (рисунок 1).

Словесно-понятийное

мышление, в отличие от чувственно-образного и от словесно-образного, есть мышление абстракциями. И лишь в таком смысле оно является высшим и особым типом мышления [1].

Схема на рисунке 2 демонстрирует три аспекта процесса освоения и познания мира человеком [2].

В приложении к информатике данная схема может быть преобразована к виду (рисунок 3):

Отражение мира в соответствующей поня-

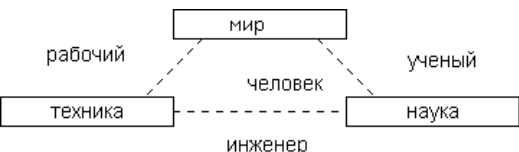


Рисунок 2 – Процесс освоения и познания мира

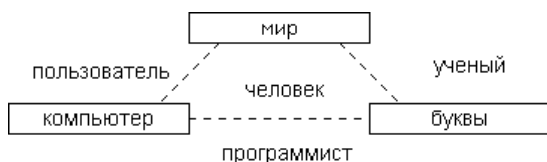


Рисунок 3 – Процесс освоения и познания мира: информационный аспект

тийной среде, т.е. собственно познание, – это прерогатива ученого, в школе и в институте обучаемые сталкиваются с готовыми моделями, и их задача заключается в осуществлении двух других бинарных отношений: программист, пользователь.

Как правило, после возникновения некоторой задачи в той или иной предметной области начинается ее решение при помощи моделирования. Моделирование – основной метод познания, заключается в воспроизведении характеристик некоторого объекта на другом объекте, специально созданном для их изучения. Оно включает в себя собственно построение модели и проведение вычислительного эксперимента. Согласно А.А. Самарскому, вычислительный эксперимент содержит цепочку [3] (рисунок 4):

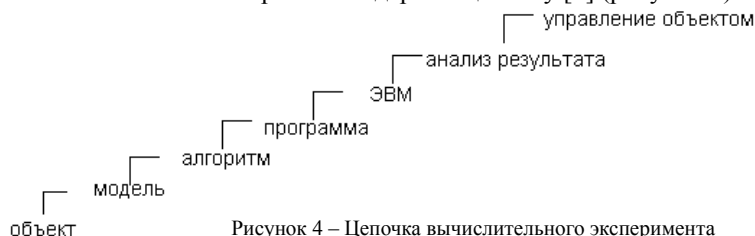


Рисунок 4 – Цепочка вычислительного эксперимента

Объект – то, на что направлена познавательная или иная деятельность. Объектом может выступать как предмет, так и какое-либо явление или процесс. Сначала строится предметная модель, а затем начинается ее исследование с помощью таких наук, как математика и информатика. Под предметной моделью принято понимать «некий объект-заменитель, который в определенных условиях может заменять оригинал, причем имеет существенные преимущества, удобства (наглядность, обозримость, доступность испытаний и пр.)».

Информационная модель – модель, выраженная в виде информации, доступной для использования человеком. Информационная модель включает три уровня:

- концептуальный, на котором содержательно записываются основные характеристики объекта, полученные в процессе отражения;
- логический – решаются задачи формализации информационной модели;
- физический – информационные массивы распределяются по физическим носителям.

Математическая модель – это выражение концептуальной информационной модели с помощью математических объектов. Существенной частью вычислительного эксперимента (часть, которая и рассматривается в учеб-

ном процессе), его ядром является триада «модель – алгоритм – программа», поэтому можно сказать, что вычислительный эксперимент содержит три основных этапа:

- построение математической модели, который включает в себя: постановку задачи, построение предметной модели, построение концептуальной информационной модели, а это приводит к построению математической модели;
- выделение из математической модели алгоритма;
- реализация алгоритма, производится построение логической информационной модели, т.е. выбор структур данных и алгоритмов над этими структурами данных. Здесь же происходит построение физической информационной модели.

Таким образом, можно сказать, что вычислительный эксперимент (основной метод информатики) является методологией решения задач с помощью ЭВМ и эту цепочку следует усвоить и ученикам и студентам (рассматривая на множестве разрешимых задач) [4].

Существуют различные взгляды на понятие «программирование»:

- исторически сложилось, что под «программированием» понимают не все этапы решения задач на ЭВМ, а часть, включающую кодировку и отладку;
- некоторые специалисты, придерживаются мнения, что это понятие включает в себя часть вычислительного эксперимента, начиная с логической информационной модели.

К методам научного познания относятся:

- философские методы, рассматривающие общие условия движения к истине;
- общенаучные методы (наблюдение, эксперимент, анализ, синтез, абстракция, обобщение и т.д.);
- методология конкретной науки (принципы) – это такие частнонаучные методы, систематическое применение которых формирует разделы науки;
- частнонаучные методы (уровень конкретной техники исследования).

Теперь становится ясно, что вычислительный эксперимент относится к принципам в научных методах, а программирование (в широком смысле) относится к частнонаучным методам информатики, так как оно отражает метод познания в самой информатике и во многом похоже на процесс познания реального мира человеком.

Как уже отмечалось выше, отражение мира в соответствующей понятийной среде, т.е. собственно познание – это прерогатива ученого, в школе и в институте обучаемые сталкиваются с готовыми моделями, и их задача заключается в осуществлении двух последних этапов вычислительного эксперимента, т.е. выделение алгоритма, выбор структур данных и алгоритмов над этими структурами. Таким образом, в учебном процессе понятие программирования сужается.

Из структуры вычислительного эксперимента вытекают этапы технологического цикла разработки программного обеспечения при структурном подходе:

анализ ► проектирование ► реализация ► тестирование ► сборка

Рисунок 5 – Технологический цикл при структурном подходе

Наиболее современной на данном этапе развития технологий программирования является компонентная технология. Нами была выдвинута гипотеза, что все технологии программирования можно изучить на базе компонентной, принципиальной стороной которой есть объектно-ориентированная. При объектно-ориентированном подходе технологический цикл, можно изобразить следующим образом (рисунок 6).

При изучении программирования используются общенаучные методы исследования, наиболее часто анализ, синтез, индукция, дедукция, обобщение, ограничение, абстрагирование, конкретизация, сравнение. Рассмотрим на каких этапах используются эти методы.

Итак, после выделения из математической модели алгоритма (т.е. некоторой модели работы программы) учащиеся должны выбрать структуры данных. Древним грекам принадлежит идея о том, что мир можно рассматривать как в терминах объектов, так и событий. В информатике существуют, соответственно, два противоположных подхода: первый процессно-ориентированный, другой ориентирован на данные. Именно ко второму подходу и относится объектно-ориентированное программирование. «... Мы интуитивно чувствуем, – писал Н. Вирт, – что данные предшествуют алгоритмам: необходимо иметь некоторые объекты, прежде чем выпол-

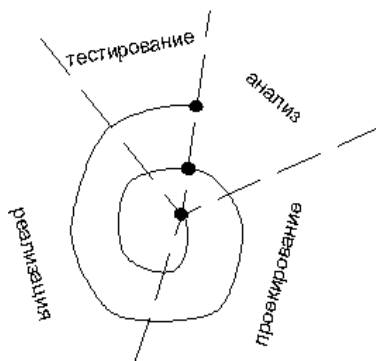


Рисунок 6 – Технологический цикл при объектно-ориентированном программировании

нять с ними действия». Основными конструкциями объектно-ориентированного программирования являются классы и объекты. Класс – конечная или бесконечная совокупность выделенных по некоторому признаку предметов, мыслимая как целая. В информатике под классом понимают структуру данных и методы их обработки, можно рассматривать как шаблон объектов, а объект – экземпляр класса, ранее понятие «объект» заменяло понятие класса.

Уже отмечались преимущества использования в обучении гибридных объектно-ориентированных систем перед чисто объектными. Поэтому будем исходить из того, что используется гибридная объектно-ориентированная система (Delphi, Visual Basic).

Центральным элементом объектно-ориентированного программирования является абстракция. Абстракция – один из основных процессов умственной деятельности человека, позволяющий мысленно выделить и превратить в самостоятельный объект рассмотрения отдельные свойства, стороны или состояния предмета. Итак, для информатики абстрагирование имеет особенное значение, так как каждое понятие в информатике – абстракция.

Таким образом, на этапе выбора структуры данных используется абстракция, но для того, чтобы абстрагироваться и отбрасывать несущественные признаки, необходимо выделить существенные, т.е. произвести анализ.

Анализ – в самом общем значении – процессы мысленного (наиболее используемый в информатике) или фактического разложения целого на составные части. Расчленение целого на составные части позволяет выявить строение исследуемого предмета, его структуру, расчленение сложного явления на более простые элементы, позволяющие отделить существенное от несущественного, сложное свести к простому. Анализ производят, чтобы глубже и детальнее исследовать тот или иной объект. Мысленно анализируя, человек почти всегда и синтезирует, и наоборот. Каждый анализ производится с какой-то целью и с соответствующей глубиной. Одной из форм анализа служит классификация предметов и явлений.

Большинству людей мир представляется состоящим из объектов, иерархически связанных друг с другом. Иерархическая классификация – это мощный способ создания абстракции. Она позволяет проанализировать семантику сложных систем, разбив их на более управляемые фрагменты.

Подбирая структуру данных программируемый склонен думать или мыслить только структурами, допускающими непосредственное выражение на языках, с которыми он знаком. Поскольку компонентная технология программирования в первую очередь предоставляет активные данные, то

мыслить необходимо именно объектно. Таким образом, на каждом этапе разделения происходит сравнение моделей объектов с компонентами, данная операция сходна с определением, входит ли данный объект в объем некоторого понятия или распознавание объекта. При этом компонент, который является идентификатором для объекта, должен воспроизводить не только свойства, но и функциональные требования, так, например, для ввода текста должен использоваться компонент Edit (Delphi), а не Image (Delphi).

До выбора структур данных студент должен был выделить алгоритм из математической модели. Ориентируясь на этот алгоритм, необходимо на следующем шаге описать алгоритмы взаимодействия структур данных. Объекты взаимодействуют между собой при помощи сообщений и обрабатываются эти сообщения соответствующими методами объекта-получателя. Эти методы на основании принципа объектно-ориентированного программирования инкапсуляции заключены внутри объекта. Поэтому алгоритмы взаимодействия нужно продумывать не как определенную последовательность действий, а как реакцию на событие, т.е. вносится стохастический характер в происходящее, что больше соответствует реальному миру. При написании методов уже вступает в силу структурное программирование, когда нужно отражать происходящее в виде строгой последовательности действий, т.е. применяется совершенно иная форма абстрагирования, при этом меняется и форма выражения абстракций и используется в большей мере декомпозиция.

Если необходимо применить некоторую конструкцию, непосредственно непредоставляемую языком, то программист должен сам осуществить ее реализацию с помощью примитивов этого языка. Это достаточно несложно при написании методов. Но системы компонентного программирования не могут предоставить совокупность компонент, при помощи которых можно решить любую поставленную задачу. Иногда возникают ситуации, когда необходимо создать собственный компонент. Итак, на этапе выбора структур данных ни один компонент не может являться моделью для выделенного объекта. В этом случае необходимо руководствоваться принципом наследования в объектно-ориентированном программировании. Компонент, который будет создаваться, должен быть наследником от какого-то имеющегося, при этом обладать теми свойствами, которые отличают его от родителя и не позволили заменить родителем-компонентом рассматриваемый объект. При этом используется обобщение – логический процесс перехода от единичного к общему, переход от видового понятия к родовому. При переходе отбрасываются признаки специфические для видового

понятия. Т.е. для того, чтобы выбрать родителя, необходимо отбросить специфические свойства, а при определении видового понятия, т.е. при написании собственного компонента их описать.

Более того, обучаемым иногда необходимо строить свою иерархию компонент, которые в крайнем случае будут наследниками Tobject (Delphi), в этом случае будет применяться весь арсенал общенаучных методов исследования: абстрагирование и выделение объектов, обобщение и выделение классов, и обобщение производится до наличия класса в системе программирования, который выступит родителем, далее происходит процесс конкретизации, когда каждый из выделенных объектов будет описан. На следующем этапе, исходя из технологической цепочки, происходит тестирование, и для совершенствования работы проекта все с первого этапа повторяется снова.

Таким образом, можно схематически изобразить соответствие используемых методов научного исследования (которые являясь логическими по своей природе и отражают познавательную сторону) и этапов вычислительного эксперимента, производимые студентами при обучении программированию:

дедукция анализ абстрагирование декомпозиция анализ	}	выделение из математической модели алгоритма;
абстрагирование индукция сравнение дедукция конкретизация ограничение синтез	}	выбор структур данных;
синтез конкретизация дедукция	}	описание алгоритмов взаимодействия структур данных.

Библиография

1 Психологический словарь. / Под ред. В.П. Зинченко, Б.Г. Мещерякова. — 2-е изд., перераб. и доп. — М.: Педагогика-Пресс, 1996. — 440 с.

2 Белошапка В.К. Три аспекта мироздания или мир как информационная структура / В.К. Белошапка // Информатика и образование. — 1988. — № 5. — С. 98–97.

3 Самарский А.А. Математическое моделирование и вычислительный эксперимент / А.А. Самарский // Вестник АН СССР. — 1975. — № 5. — С. 38–39.

4 Бороненко Т.А. Методика обучения информатике. Специальная методика / Т.А. Бороненко, Н.И. Рыжова. — СПб: Изд-во РГПУ им. А.И. Герцена, 1999. — 134 с.

Поступила в редакцию 2.12.2000 г.