

ПРИМЕНЕНИЕ ОПЫТА ПОСТРОЕНИЯ МАТЕМАТИЧЕСКИХ МОДЕЛЕЙ ПРИ РАЗРАБОТКЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Е.А. Кондратьева
Витебск, ВГУ имени П.М. Машерова

На данный момент профессии, связанные с разработкой программного обеспечения являются популярными, востребованными и актуальными. Обучение разработке программного обеспечения происходит на всех специальностях математического факультета.

Математическое моделирование – главный среди методов и средств обучения, обеспечивающих совершенствование процесса подготовки будущих высококвалифицированных специалистов. Это основной из методов, активизирующих учебно-познавательную деятельность учащихся и позволяющий широко использовать информационно-коммуникационные технологии в обучении. Метод математического моделирования, применяемый при изучении математики в школе, позволяет усилить систематизацию знаний; находить опосредованный характер применения прикладной направленности курса математики для понимания процессов современного производства и сути научных теорий [1].

Цель работы – нахождение связи между построением математической модели и разработкой программного обеспечения, в том числе и на объектно-ориентированных языках.

Материал и методы. Проанализируем этапы построения математической модели и сравним их с этапами создания программного обеспечения.

На первом этапе построения математической модели происходит *обследование объекта моделирования и формулировка технического задания на разработку модели*. Данный опыт будет полезен при составлении технического задания, по которому в дальнейшем будет разрабатываться программное обеспечение. Корректно составленное техническое задание уменьшает время разработки за счет уменьшения его уточнений с последующими доработками.

Далее происходит *концептуальная и математическая постановка задачи* – реальная модель объекта упрощается, отбрасываются несущественные факторы и эффекты, формулируются основные вопросы и набор гипотез касательно свойств и поведения объекта моделирования. Аналогичные действия выполняются при реализации одного из принципов объектно-ориентированного программирования – абстракции, и последующем построении класса исследуемого объекта. Свойства объекта описываются как поля класса, поведение – как методы. Точная постановка задачи позволяет сократить время разработки программного обеспечения.

Качественный анализ и проверка корректности предполагают всесторонний контроль полученной модели. Полученный опыт может в дальнейшем использоваться на одном из этапов разработки программного обеспечения – при тестировании.

При *выборе и обосновании выбора методов решения задачи* выполняется сравнение результатов решения задачи любыми возможными методами. Получение результата каждым из методов может занимать различное время, давать избыточную или недостаточную для данной задачи точность, а также иметь некоторые особенности и ограничения, которые могут в конечном итоге повлиять на выбор. Похожим выбором является выбор языков программирования и шаблонов проектирования, которые будут использоваться при разработке. Подбор правильного инструмента для работы может сократить как время, затраченное на разработку, так и время, затраченное на доработку и расширение функционала программного обеспечения в случае возникновения необходимости в этом [2].

Результаты и их обсуждение. Для более наглядного результата рассмотрим пример построения математической модели для решения задачи о рюкзаке:

Дано несколько предметов, каждый предмет имеет положительную массу и стоимость. Необходимо выбрать из этих предметов такой набор, чтобы суммарная масса не превосходила вместимости рюкзака, а суммарная стоимость была максимальна.

Сформулируем техническое задание:

Дано N предметов, W – вместимость рюкзака, $w = \{w_1, w_2, \dots, w_N\}$ – соответствующий ему набор положительных целых весов, $p = \{p_1, p_2, \dots, p_N\}$ – соответствующий ему набор положительных целых стоимостей. Нужно найти набор бинарных величин $V = \{b_1, b_2, \dots, b_N\}$, где $b_i = 1$, если предмет p_i включен в набор, $b_i = 0$, если предмет p_i не включен, и такой что:

$b_{1w_1} + \dots + b_{Nw_N} \dots W$
 $b_{1p_1} + \dots + b_{Np_N}$ максимальна.

В случае ошибки при составлении технического задания для данной задачи полученная модель будет соответствовать обобщенному случаю, например когда любой предмет может быть взят некоторое количество раз (ограниченный рюкзак). При разработке программного обеспечения аналогичная ошибка в техническом задании привела бы к необходимости доработки существующей программы.

При постановке задачи объекты рюкзака и предметов упрощаются до набора необходимых свойств – веса и цены. По аналогии при построении классов поля бы соответствовали этим свойствами:

```
class Subject {  
    int weight;  
    int cost; }  
}
```

Задачу о рюкзаке можно решить несколькими способами: перебор всех подмножеств набора из N предметов и методом динамического программирования. Сложность в этих случаях будет равна $O(2^N)$ и $O(NW)$ соответственно. Для реализации указанных способов будет оптимальным использовать в первом случае программу для работы с электронными таблицами, когда во втором случае понадобится полноценная среда разработки [3]. Пример реализации на псевдоязыке:

```
for i = 0 to w  
    A[0][i] = 0  
for i = 0 to n  
    A[0][i] = 0 //первые элементы приравняем к 0  
for k = 1 to n  
    for s = 1 to w //перебираем для каждого k все вместимости  
        if s >= w[k] //если текущий предмет помещается в рюкзак  
            A[k][s] = max(A[k - 1][s], A[k - 1][s - w[k]] + p[k]) //выбираем класть ли его  
        else  
            A[k][s] = A[k - 1][s] //иначе не кладем
```

Затем найдем набор предметов рекурсивной функцией.

```
function findAns(int k, int s)  
if A[k][s] == 0  
return  
if A[k - 1][s] == A[k][s]  
findAns(k - 1, s)  
else  
findAns(k - 1, s - w[k])  
ans.push(k)
```

Заключение. Анализ этапов построения математической модели показал, что опыт каждого из этапов можно использовать при разработке программного обеспечения, причем на различных его этапах. Обучение математическому моделированию позволит более гибко подходить к разработке программного обеспечения, а также сэкономить время на возможных ошибках при разработке программного обеспечения.

1. Мамыкина, Л.А. Математическое моделирование как метод познания и обучения математике в профильной школе / Л.А. Мамыкина // Научная электронная библиотека [Электронный ресурс]. – Используемый источник: <https://cyberleninka.ru/article/n/matematiceskoe-modelirovanie-kak-metod-poznaniya-i-obucheniya-matematike-v-profilnoy-shkole/viewer>. – Дата использования: 01.01.2020.
2. Звонарев С.В. Основы математического моделирования: учебное пособие для вузов/ С.В. Звонарев. – Екатеринбург: Изд-во Урал. ун-та, 2019. – 112 с.
3. Университет ИТМО. Задача о рюкзаке [Электронный ресурс]. – Используемый источник: https://neerc.ifmo.ru/wiki/index.php?title=Задача_о_рюкзаке. – Дата использования: 01.01.2020.