

Этапы решения задач на компьютере

Работа по решению любой задачи с использованием компьютера делится на следующие этапы.

Оглавление

Постановка задачи	1
Формализация задачи	2
Построение алгоритма.....	2
Составление программы на языке программирования	3
Отладка и тестирование программы.....	4
Проведение расчётов и анализ полученных результатов.	4

*Перед началом изучения данного вопроса необходимо определиться с понятием задача. Мы будем применять данное понятие не только к классическим школьным задачам по математике или физике. **Под задачей подразумевается проблемная ситуация с явно заданной целью, которую необходимо достичь.***

Постановка задачи



На данном этапе формулируется цель решения задачи, описывается её содержание.

Должны быть чётко определены исходные данные, необходимые для решения задачи, и результат.

Пример

Необходимо рассчитать стоимость покраски пола комнаты, если известны габариты комнаты (длина и ширина, расход краски на

1 м^2 и стоимость 1 кг краски.

Определим исходные данные и требуемый результат. Для удобства каждую исходную величину определим буквой.

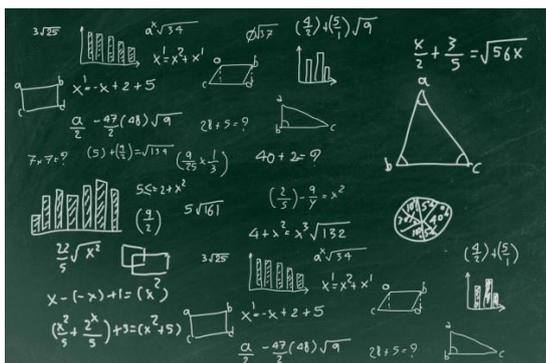
Исходные данные (какие величины нам известны?):

- Длина комнаты в метрах (a)
- Ширина комнаты в метрах (b)
- Расход краски 1 кг на 1 м^2 (r)
- Стоимость краски за 1 кг в руб. (c)

Результат (величина, которую мы должны определить):

- Стоимость всей краски (Z)

Формализация задачи



Компьютер "умеет работать только с числами". Все действия центрального процессора - это арифметические операции над числовыми данными. Поэтому, при решении задачи с помощью компьютера необходимо перевести задачу на язык математики. Чаще всего, решение задачи сводится к математическому описанию какого-то объекта, явления или процесса. Другими словами, **этап формализации сводится к получению математической модели.**

Пример

Необходимо определить как мы будем находить стоимость краски. Для этого нужно знать количество краски в кг и её стоимость за 1 кг. Определим новую величину - объём всей краски Q . Объём всей краски равен площади всего пола на расход краски.

$$Q = a * b * r \quad (1)$$

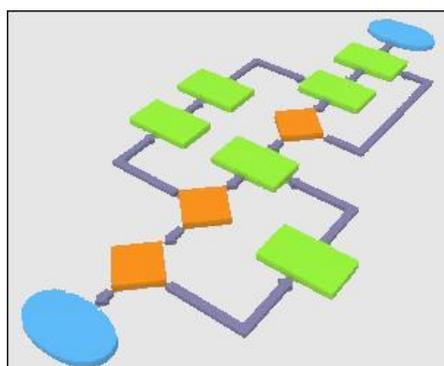
Теперь определим сколько стоит вся краска

$$Z = Q * c \quad (2)$$

Можно подставить (1) в (2) и получаем окончательную математическую модель задачи. Где в одной формуле связаны исходные данные и результат.

$$Z = a * b * r * c \quad (3)$$

Построение алгоритма

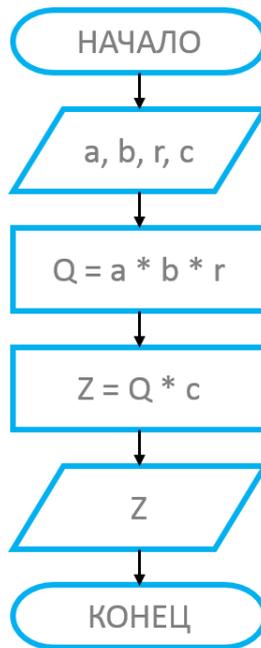


В очень простых задачах (например, как наша :) данный этап можно пропустить, и сразу писать код программы на языке программирования. Модель настолько простая, что последовательность действий здесь видна явно, и описывать алгоритм (используя блок-схемы или псевдокод) нет необходимости. Но в обучающих целях построим алгоритм для нашей задачи. Он будет включать ввод данных, их обработку и вывод данных.

Алгоритм - это последовательность команд управления каким-либо исполнителем. Алгоритм создаётся для выполнения исполнителем и набор команд, используемые для составления алгоритма зависит от его "возможностей" (Система команд исполнителя - СКИ).

В наших задачах исполнителем будет выступать компьютер, который работает с числами, хранящимися в памяти - величинами. Поэтому алгоритмы,

предназначенные для управления компьютером, называют *алгоритмами работы с величинами*.



Составление программы на языке программирования



Для выполнения первых трёх этапов, как видим, компьютер не нужен. Это очень важно понимать! Задача решается не на компьютере, задача решается в голове путём логических рассуждений. Получив математическую модель - мы, по сути, уже решили задачу. Компьютер - это инструмент, который позволяет ускорить данное решение задачи применимо уже к конкретным исходным данным.

Имея модель задачи и готовый алгоритм, можно переходить к этапу написания кода программы (часто употребляют название *компьютерная программа*, или просто *программа*).

Компьютерная программа - это текст (синтаксическая единица), который написан по правилам определённого языка программирования и состоит из определений, операторов и инструкций, необходимых для решения поставленной задачи.

Код программы на языке Python для нашей задачи может быть таким.

```
a = int(input())
b = int(input())
c = int(input())
r = float(input())
Q = a * b * r
Z = Q * c
print(Z)
```

Отладка и тестирование программы

Этот этап считается наиболее трудоёмким. Цель тестирования программы - это выявление ошибок, которые могли возникнуть на этапе формализации и разработки алгоритма, а также при написании самого кода программы.

На этапе тестирования выявляют два вида ошибок:

Синтаксические ошибки – это ошибки в записи конструкций языка программирования (чисел, переменных, функций, выражений, операторов, меток, подпрограмм). **Обнаружение большинства синтаксических ошибок автоматизировано в основных системах программирования. Они выявляются уже на этапе написания программы.**

Семантические ошибки – это ошибки, связанные с неправильным содержанием действий и использованием недопустимых значений величин.

Логические ошибки - это ошибки в самом алгоритме. Данные ошибки формируются на этапе разработки алгоритма при неправильном решении задачи. Выявление такой ошибки зачастую требует повторного решения задачи с самого первого этапа.

Если программа выполняется успешно, желательно завершить ее испытания тестированием при задании исходных данных, принимающих предельные для программы значения. а также выходящие за допустимые пределы значения на входе.

Контрольные примеры (тесты) – это специально подобранные задачи, результаты которых заранее известны или могут быть определены без существенных затрат.

Проведение расчётов и анализ полученных результатов.

Этот этап подразумевает использование готовой программы в практических целях для решение поставленной задачи. В нашем примере мы можем использовать нашу программу для вычисления стоимости краски для покраски пола во время проведения ремонта для любых допустимых исходных данных.