

Лекция. Этапы решения задач с помощью компьютера.

Вычислительные технологии в современном мире являются самыми перспективными средствами как для выполнения научных исследований, так и для повышения конкурентоспособности промышленности. Для решения сложных реальных задач объединяются силы учёных-исследователей и практиков, а также вычислительных мощностей разных компьютеров, иногда и по всему миру. Для синхронизации деятельности людей и вычислительной техники необходима чёткая организация, планирование, разделение творческих и вычислительных процессов. Конечно, не все задачи требуют такой сложной организации. Однако умение планировать сложную деятельность полезно осваивать на простых примерах.

Процесс решения задачи, разделив его на отдельные этапы.



Рис. 1. Этапы решения задачи

1. Постановка задачи. Это первый этап решения задачи, на котором определяется цель решения, собираются исходные данные, выясняются формат и область значений результатов. На этом этапе предполагается построение адекватных моделей объектов исследования с характеристиками, близкими к оригиналу. Здесь важно определить оптимальный набор исходных данных, которые позволят решить задачу, но не сделают её решение трудновыполнимым. На входе — проблема; на выходе — исходные данные и формат результатов.

2. Формализация задачи. На этом этапе осуществляется построение **математической модели задачи**. Взаимодействие моделей объектов, участвующих в реальном процессе, заменяется известными математическими соотношениями и уравнениями. Зачастую в определённой сфере человеческой деятельности уже накоплен набор математических уравнений, с помощью которых принято решать те или иные задачи. Умение предложить новые модели решения связано с широким кругозором исследователя, его неординарным подходом. Такие же качества требуются от человека и при решении новых задач, тех, которые ранее не решались в конкретной области. На этом этапе реализуется исключительно творческая деятельность человека, а успех решения задачи зависит от выбора и обоснования метода решения. На этом же этапе определяются допустимая точность, объём и время

вычислений, а также технические средства, с помощью которых будет реализовываться решение.

На входе — исходные данные и формат результатов; на выходе — математическая модель и технические средства для решения.

3. Создание алгоритма решения. На этом этапе реализуются методы, с помощью которых будет осуществляться решение. Здесь необходимо скоординировать работу всех участников проекта решения. Создаются блок-схемы алгоритмов, технологические карты загрузки оборудования, например распараллеливание вычислений. Определяется последовательность выполнения отдельных блоков вычислений.

На входе — математическая модель; на выходе — блок-схема или другой способ описания алгоритма.

4. Составление программы для решения задачи. Алгоритм реализуется на конкретном языке программирования. Уточняются формат и область значений результатов в зависимости от выбранного языка программирования.

На входе — блок-схема или другой способ описания алгоритма; на выходе — программа обработки исходных данных.

5. Тестирование и отладка программы. Составление программы обработки исходных данных — процесс трудоёмкий, особенно если необходимо соединить блоки разных разработчиков и параллельно работающих процессов. Отладка программы выполняется в несколько этапов: определяются синтаксические и семантические ошибки. Для обнаружения ошибок проводят тестирование на специально подготовленных данных. Внутри программы расставляют «контрольные точки», получая таким образом сообщение об ошибке в ходе выполнения программы до её завершения. Кроме компьютерного тестирования полученные данные должен проанализировать специалист в той области, в которой работает поставленная задача. Такой подход позволяет выявить ошибки на предыдущих этапах решения задачи. Например, анализ результатов может выявить недочёты, связанные с выбором алгоритма решения, постановкой задачи или со сбором исходных данных. В этом случае процесс решения возвращается на предыдущие этапы и проводится их корректировка.

На входе — программа обработки исходных данных; на выходе — результаты решения задачи.

Пример решения конкретной задачи

Задача

В некоторой диэлектрической среде на расстоянии r (см) помещены два точечных заряда. На каком расстоянии они должны находиться в воздухе, чтобы сила взаимодействия между ними осталась прежней?

Постановка задачи

Определим исходные данные, которые необходимо знать для решения задачи. Для учебной задачи обычно эти данные записывают под заголовком «Дано».

Нам необходимо знать расстояние r и диэлектрические проницаемости сред: ту, в которую помещены заряды, и воздуха. По условию задачи мы также можем

отметить, что силы взаимодействия в предложенной среде и в воздухе должны быть равными.

Формализация задачи

На этом этапе необходимо подобрать математическую модель, описывающую взаимодействие объектов, участвующих в реальном процессе. Для предложенной задачи это сила Кулона.

$$F = 14 \cdot \pi \cdot \epsilon \cdot \epsilon_0 \cdot q_1 \cdot q_2 / r^2.$$

Силу взаимодействия зарядов в предложенной среде F_1 и в воздухе F_2 можно уравнивать.

$$14 \cdot \pi \cdot \epsilon_1 \cdot \epsilon_0 \cdot q_1 \cdot q_2 / r_1^2 = 14 \cdot \pi \cdot \epsilon_2 \cdot \epsilon_0 \cdot q_1 \cdot q_2 / r_2^2.$$

После упрощения получим:

$$\epsilon_1 \cdot r_1^2 = \epsilon_2 \cdot r_2^2;$$

$$r_2 = r_1 \cdot \sqrt{\epsilon_1 / \epsilon_2};$$

учитывая, что $\epsilon_2 = 1$:

$$r_2 = r_1 \cdot \sqrt{\epsilon_1}.$$

Создание алгоритма решения

Представим алгоритм решения задачи в виде блок-схемы. При создании блок-схемы следует определиться с типом числовых данных. В условии задачи нам дано расстояние в сантиметрах, но ответ правильно представить в СИ, поэтому расстояние будет выражено вещественным числом. Диэлектрическая проницаемость по определению — отношение напряжённости в вакууме и в диэлектрике, следовательно, тоже вещественное число.

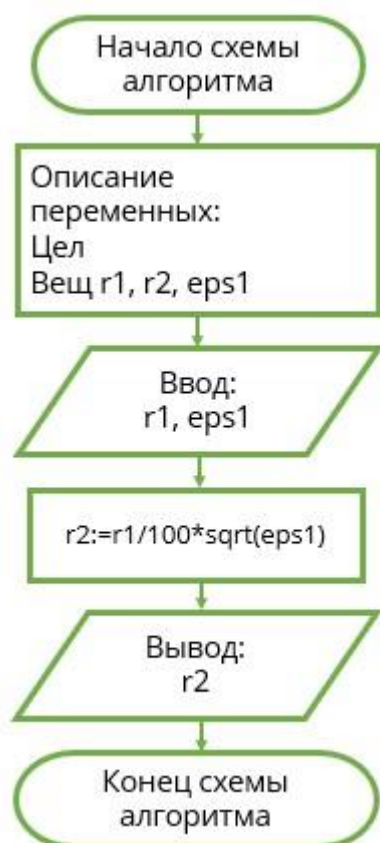


Рис. 1. Блок-схема № 1

Составление программы для решения задачи: алгоритм реализуется на конкретном языке программирования. Составим программу на языке Pascal.

```

var r1,r2,eps1: real;
begin
writeln ('введите расстояние в сантиметрах и eps1');
readln(r1,eps1);
r2:=r1/100*sqrt(eps1);
writeln ('расстояние между зарядами в воздухе = ',r2:3:3,' метра')
end.
  
```

Рис. 2. Решение задачи на Pascal

Тестирование и отладка программы

Тестирование проведём на следующих данных: $r1=45$ см; среда, в которой располагались заряды, — глицерин, её диэлектрическая проницаемость $=43$.

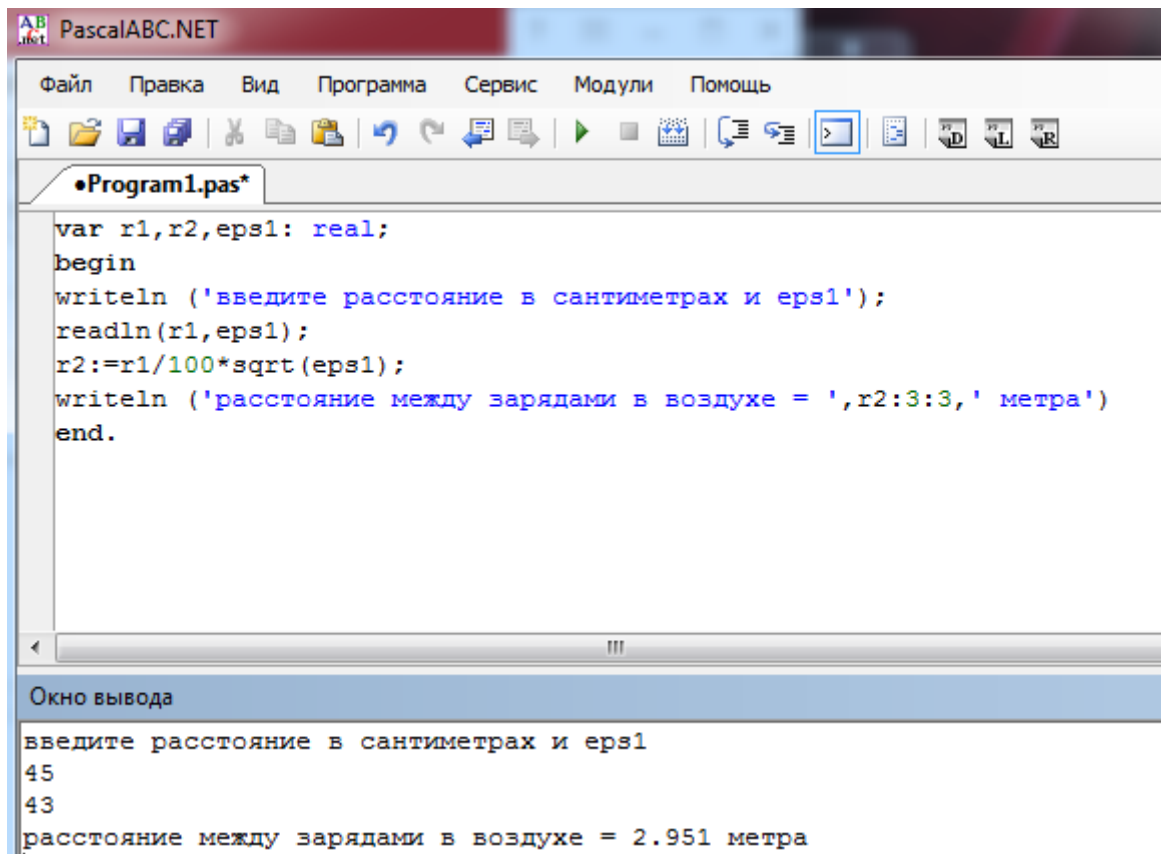


Рис. 3. Тестирование программы

При формализации задачи диэлектрическую проницаемость воздуха приняли за единицу и расстояние r_2 отформатировали до трёх знаков после запятой. Если такая точность недостаточна, можно внести уточнения.

На этапе формализации:

$$\epsilon_1 \cdot r_1 = \epsilon_2 \cdot r_2;$$

$$r_2 = r_1 \cdot \epsilon_1 \epsilon_2^{-1} \sqrt{\quad}$$

Учитывая, что $\epsilon_2 = 1,00057$:

$$r_2 = r_1 \cdot \epsilon_1 1,00057^{-1} \sqrt{\quad}$$

На этапе создания алгоритма решения:

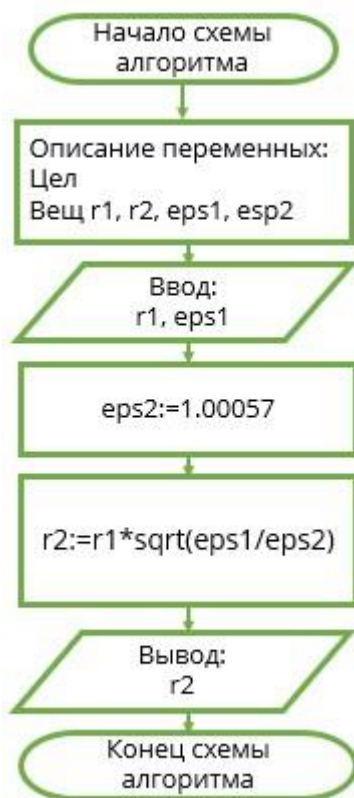


Рис. 4. Блок-схема № 2

На этапе составления программы для решения задачи:

```

var r1,r2,eps1: real;
begin
writeln ('введите расстояние в сантиметрах и eps1');
readln(r1,eps1);
r2:=r1/100*sqrt(eps1/1.00057);
writeln ('расстояние между зарядами в воздухе = ',r2:3:5,' метра')
end.
  
```

Окно вывода

```

введите расстояние в сантиметрах и eps1
45
43
расстояние между зарядами в воздухе = 2.95001 метра
  
```

Рис. 5. Тестирование программы 2

Относительное расхождение результатов — 0,001 м.

Процесс решения задачи на компьютере делится на отдельные этапы.



Рис. 1. Этапы решения задачи

Напиши по порядку номера этапов, к которым можно отнести следующие действия:

1. выбор языка программирования;
2. поиск коэффициентов по справочнику;
3. определение типов данных;
4. внесение уточнений в блок-схему