

# Основы алгоритмизации и программирования

# Базовые понятия программирования

- алгоритм
  - последовательность действия для достижения поставленной цели
- данные
  - представление фактов, понятий или инструкций в формализованном виде, пригодном для интерпретации, обработки и передачи
- исполнитель
  - человек, компьютер, механическое устройство и т.п.
- программа
  - последовательность инструкций, записанных строго в соответствии с правилами синтаксиса какого-либо языка программирования
  - определяет процедуру решения конкретной задачи в ЭВМ

# Алгоритм

- Слово "алгоритм" - от имени автора научного трактата по алгебре / персидского учебника по математике, средневекового ученого Аль-Хорезми (Abu Abd Allah Muhammad ibn Musa al-Khwarizmi)



al-  
Khwarizmi

Algorismus

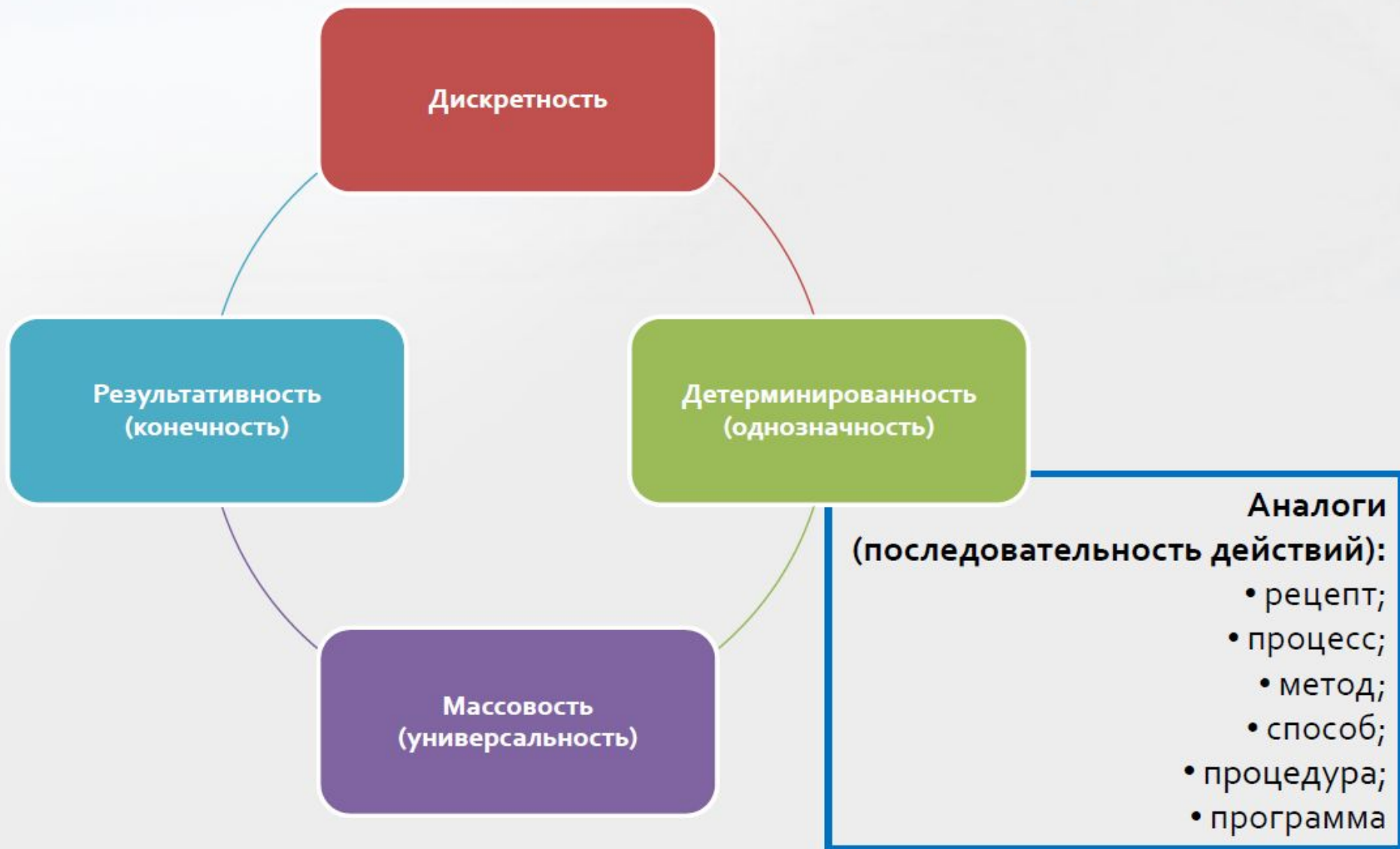
Algorithmi

# Алгоритм

*Варианты определения алгоритма:*

- "Четкое описание последовательности действий, которое необходимо выполнить для получения результата"
- "Понятная и точная последовательность действий, описывающая процесс преобразования объекта из начального состояния в конечное"
- "Последовательность команд, позволяющих получить из корректных входных данных (input) требующиеся выходные данные (output) за ограниченный промежуток времени"
- "Реализация функции, которая каждому набору входных данных ставит в соответствие некоторый"

Алгоритм = последовательность действий + особые свойства (требования к алгоритму)



# Этапы решения задачи на ЭВМ



# Этапы решения задачи на ЭВМ (пример)

- Каким уровням (этапам) соответствуют примеры описания решения задачи?

Сравнить 2 числа. Если первое число больше второго, то результатом считать единицу, иначе – ноль.

$$f(x,y) = \begin{cases} 1, & \text{если } x > y; \\ 0, & \text{если } x \leq y \end{cases}$$

```
целое x, y, z;  
ввести (x, y)  
если x > y то z=1;  
иначе z=0;  
вывести (z);
```

```
#include <stdio.h>  
int x,y;  
int z;  
int main()  
{  
    scanf("%d %d", &x, &y);  
    if (x>y) z = 1;  
    else z = 0;  
    printf("%d", z);  
}
```