

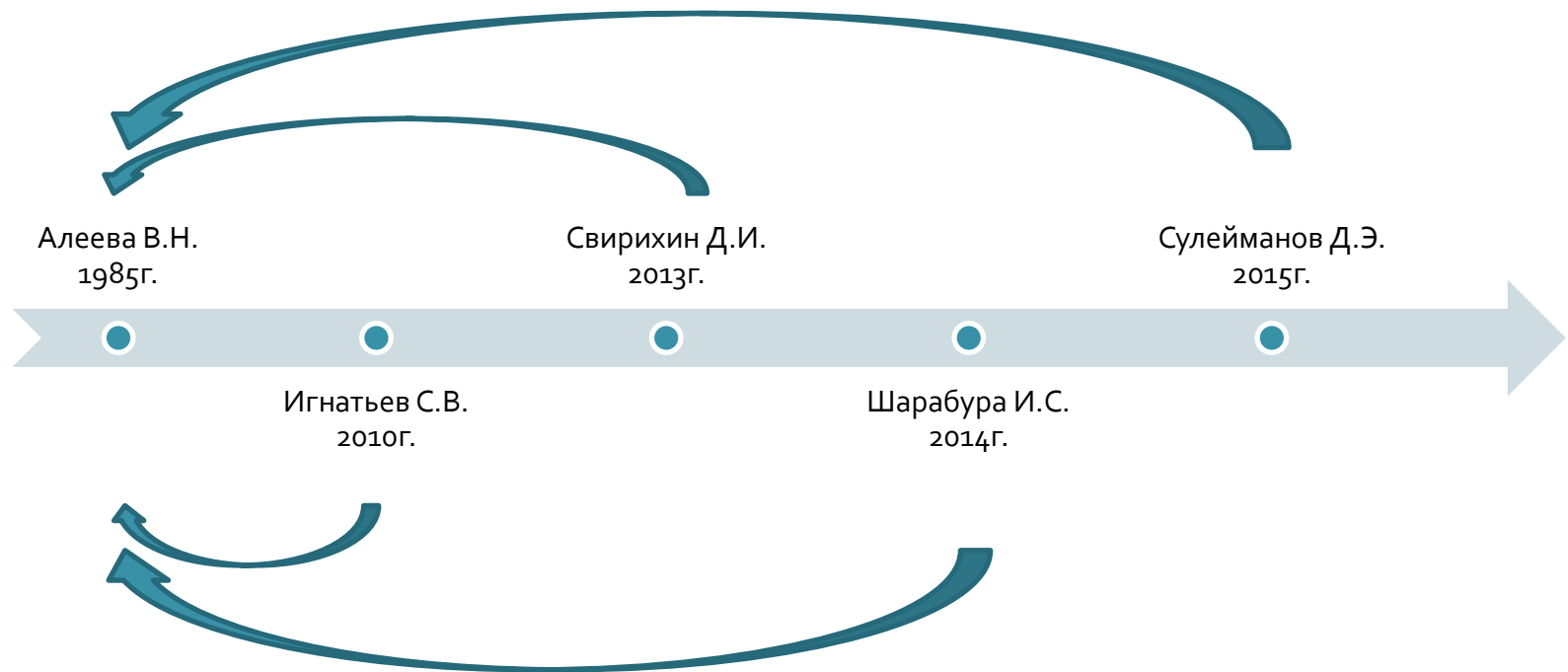


Программное обеспечение QStudio

Докладчики: Сулейманов Денис (ВМИ-211)

Шарабура Илья (ВМИ-301)

История развития



Проблемы концепции



QStudio

Программное обеспечение, представляющее способ представления алгоритмов в форме Q-детерминанта с возможностью построения плана выполнения максимально быстрой реализации.

Цели разработки

Развитие концепции Q-детерминанта

Создание абстракции для обработки алгоритма

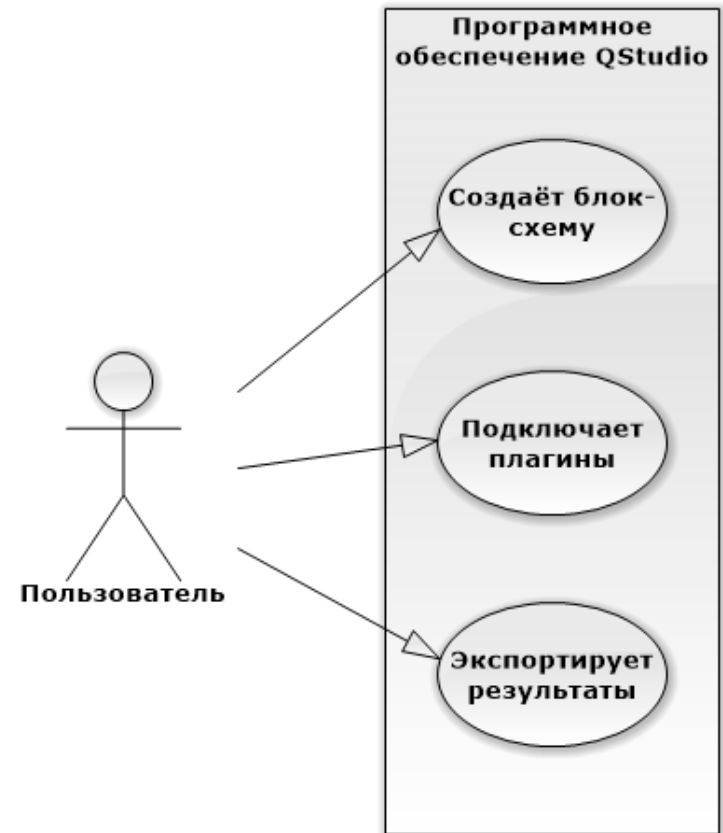
Привлечение новых людей к изучению и работе с концепцией Q-детерминанта



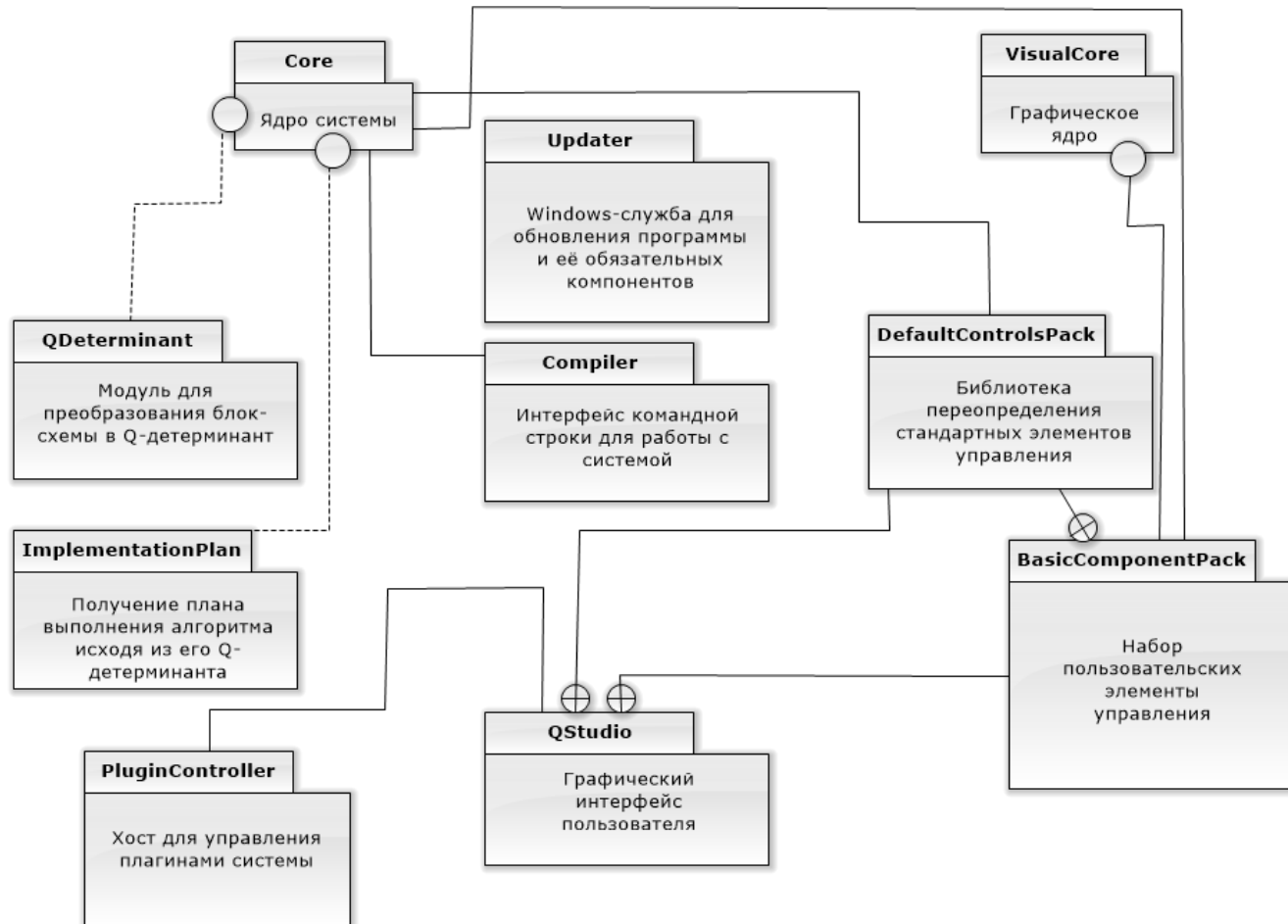
АРХИТЕКТУРА

Со стороны пользователя

- Создание проектов
- Ввод блок-схем
- «Компиляция»
- Сохранение общего плана выполнения



Модульность



Плагины

Просмотр плана выполнения алгоритма

Кодогенерация

Обработка блок-схем

Получение сравнительных характеристик сложности алгоритма

Сохранение информации в базе данных

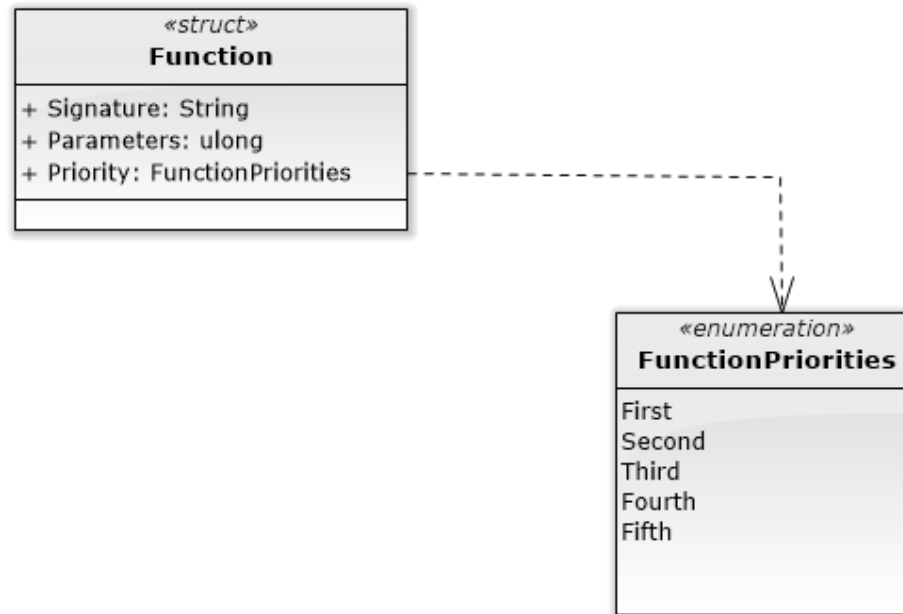
Наложение плана выполнения на архитектуру ВС



РАЗВИТИЕ КОНЦЕПЦИИ

Множество функций

a+b	+(a,b)
sin(a)	sin(a)
MyFunction(a,b,c,....,z)	MyFunction(a,b,c,....,z)

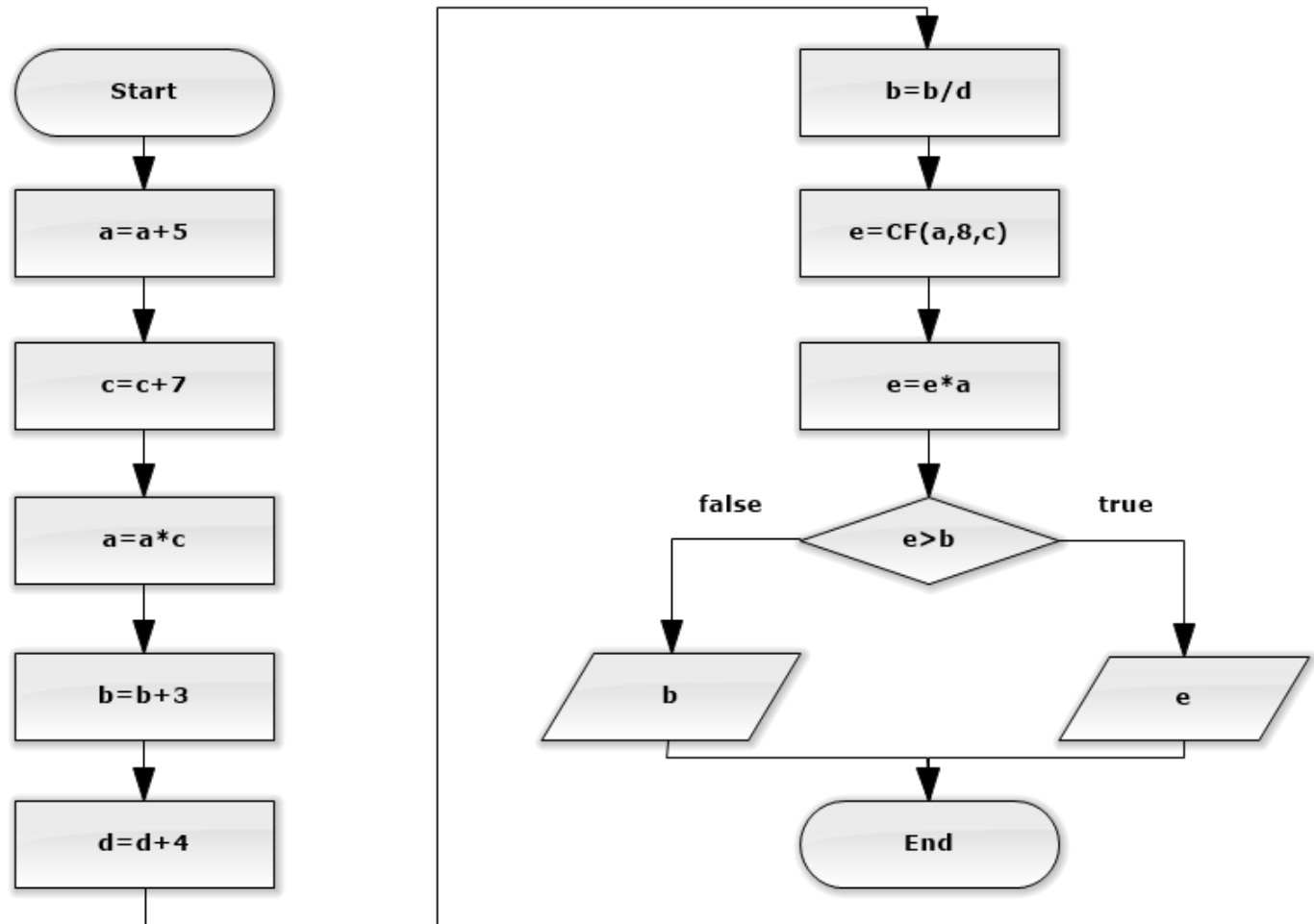




ПРИМЕР

Пример 1

- Блок-схема с условием

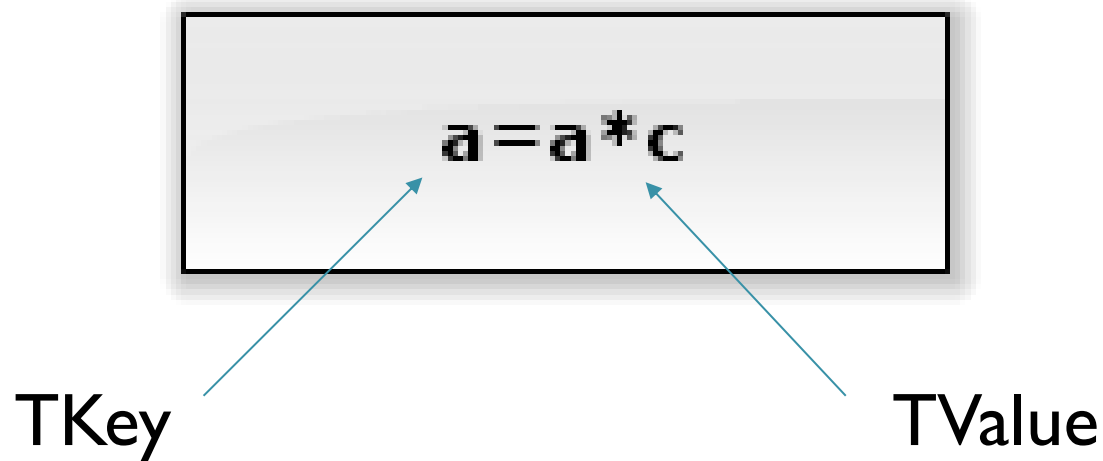


Получение Q-детерминанта

- Обработка блок-схемы начинается с блока “Begin”
- Находится связь, исходящая из обрабатываемого блока
- Следующим обрабатывается блок, в который приходит данная связь
- Обработка заканчивается блоком “End”

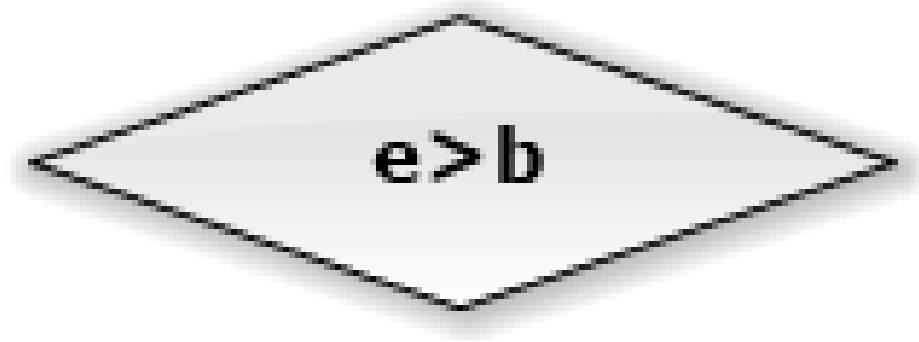
Обработка блок-схемы

- Для каждой переменной создается элемент словаря



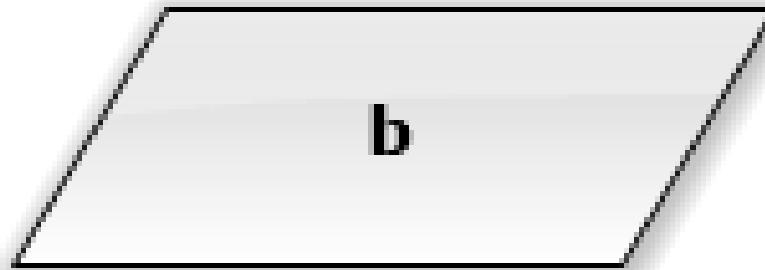
Обработка блок-схемы

- Выражение из блока условия попадает в условную часть Q-терма



Обработка блок-схемы

- Выражение из TValue элемента словаря, ключ TKey которого соответствует переменной из блока вывода, добавляется в Q-терм



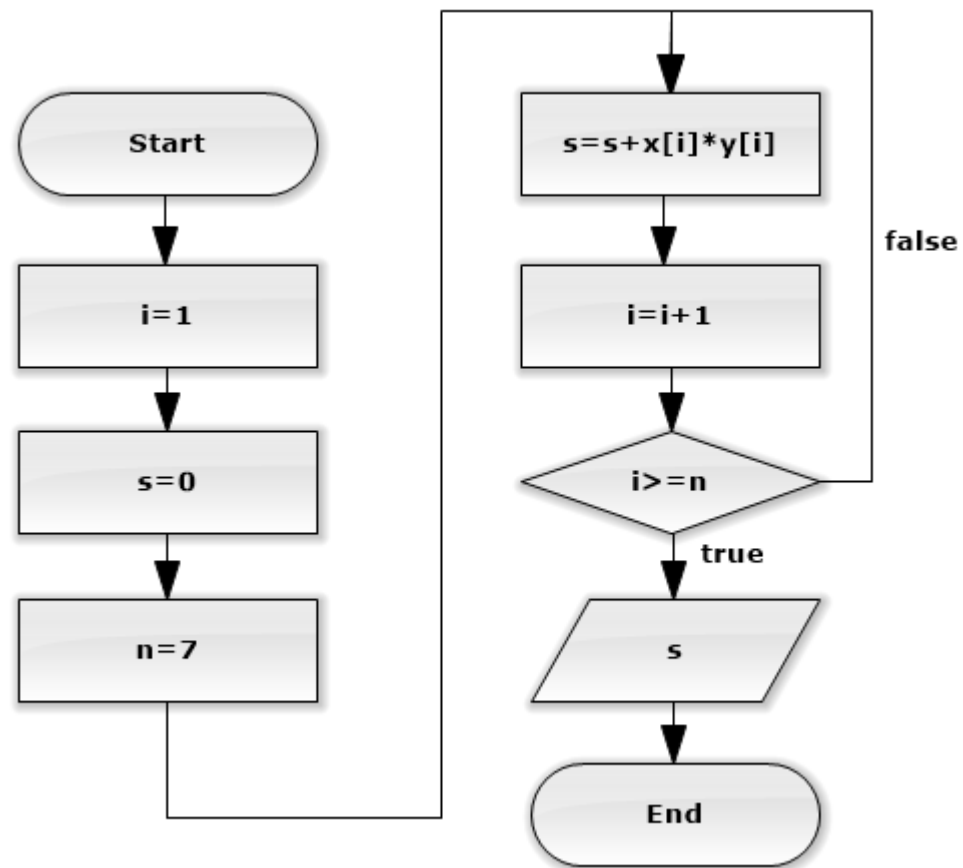
Пример 1

- Q-детерминант данного алгоритма имеет вид:

```
{(  
  >(  
    *(  
      *(5,7),  
      CF(a,8,c)  
    ),  
    /(3,4)  
  )  
);  
(  
  *(  
    *(5,7),  
    CF(a,8,c)  
  )  
),  
(  
  <=(  
    *(  
      *(5,7),  
      CF(a,8,c)  
    ),  
    /(3,4)  
  )  
);  
(  
  /(3,4)  
)}  
}}
```

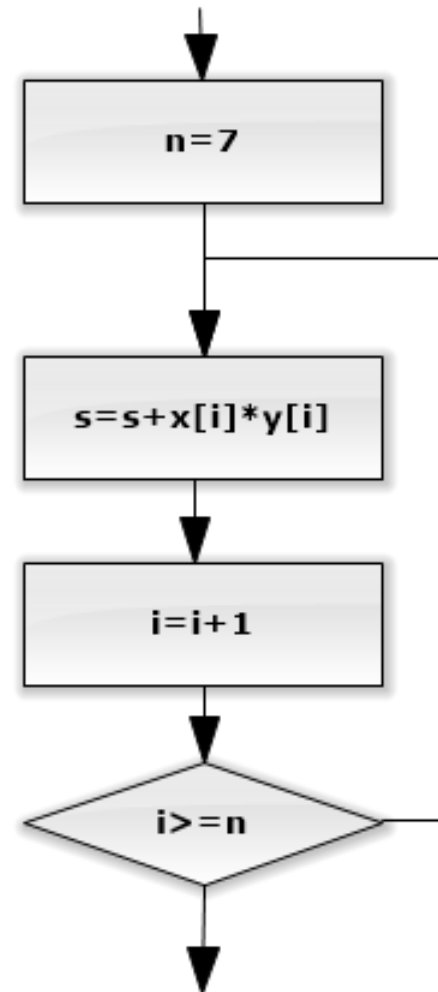
Пример 2

- Скалярное произведение векторов



Обработка блок-схемы

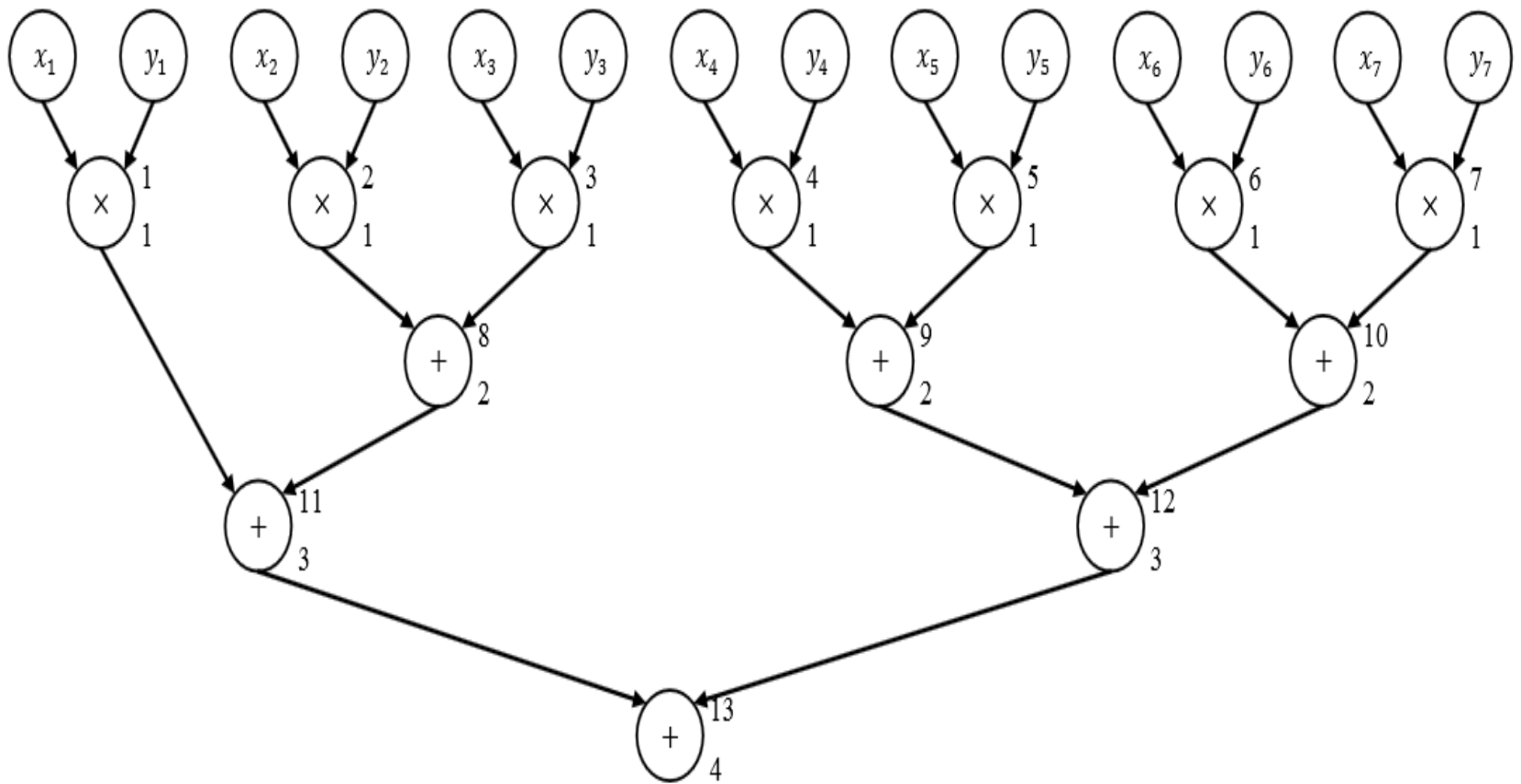
- Циклы



Пример 2

- Q-детерминант алгоритма скалярного произведения векторов имеет следующий вид:

```
+(  
  +(  
    +(  
      *(x[4],y[4]),*(x[5],y[5])  
    ),  
    +(  
      *(x[6],y[6]),*(x[7],y[7])  
    )  
  ),  
  +(  
    +(  
      *(x[3],y[3]),*(x[2],y[2])  
    ),  
    *(x[1],y[1])  
  )  
)
```





ИТОГИ И ПЛАНЫ

Проблемы концепции



Цели разработки

Развитие концепции Q-детерминанта

Создание абстракции для обработки алгоритма

Привлечение новых людей к изучению и работе с концепцией Q-детерминанта



ЗАКЛЮЧЕНИЕ