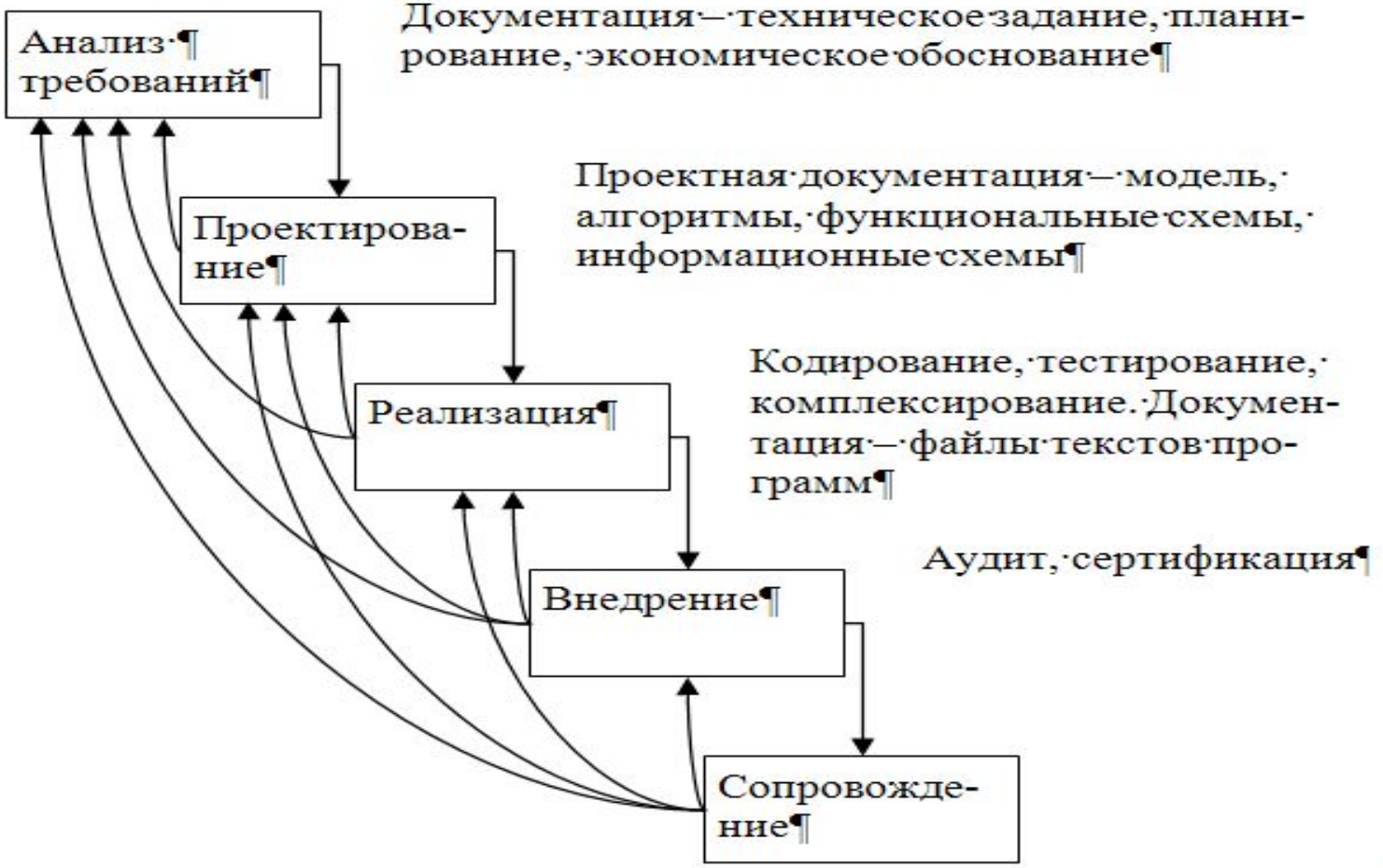
Лекция 2

Этапы разработки программного обеспечения

Жизненный цикл программ



Спецификации программы

- ***Спецификации*** – это описатели отдельных стадий ЖЦПО и проекта в целом. Согласно принятой терминологии в рамках учебного процесса полная документация программы содержит:
- – ***внешнюю*** спецификацию (анализ требований и разработка ТЗ);
- – ***внутреннюю*** спецификацию (проект программы);
- – спецификацию ***этапа реализации*** (код программы).

Международные стандарты при разработке ПО

- 1. ISO/IEC 12207:1995 – базовый стандарт, регламентирующий процессы ЖЦПО;
- 2. ISO/IEC 9126–1991 – базовый стандарт по показателям и метрикам характеристик качества ПО;
- 3. ISO/IEC 15504–98 – SPICE – стандарт оценки процессов ЖЦПО.

Стандарты Российской Федерации

- 1. Соответствующие ISO стандартам ГОСТ Р ИСО/МЭК – 12207, ГОСТ Р ИСО/МЭК 9126–93.
- 2. Группа стандартов ГОСТ 19.ххх. Из них широко применяются:
- – ГОСТ 19.701–90 ЕСПД – схемы алгоритмов, программ, данных, систем; условные обозначения и правила.
- – ГОСТ 19.102–77 – стадии разработки.
- 3. Группа стандартов ГОСТ 34.ххх. В определенной степени соответствует ISO/IEC 12207.

Качество программ

- ***Качество*** – объективная характеристика товара (продукта, услуги), показывающая *степень удовлетворенности потребителя*.
- Со своей стороны каждый товар имеет объективные, присущие ему свойства, или характеристики. Некоторые свойства могут иметь количественную оценку – показатель.
- ***Показатель*** – мера степени, в которой товару присуще свойство (характеристика).
- С точки зрения потребителя, некоторые свойства более значимы, другие – менее. Выделив значимые свойства (характеристики) и их показатели, потребитель формирует некоторый *комплексный показатель качества* или *метрику качества*.

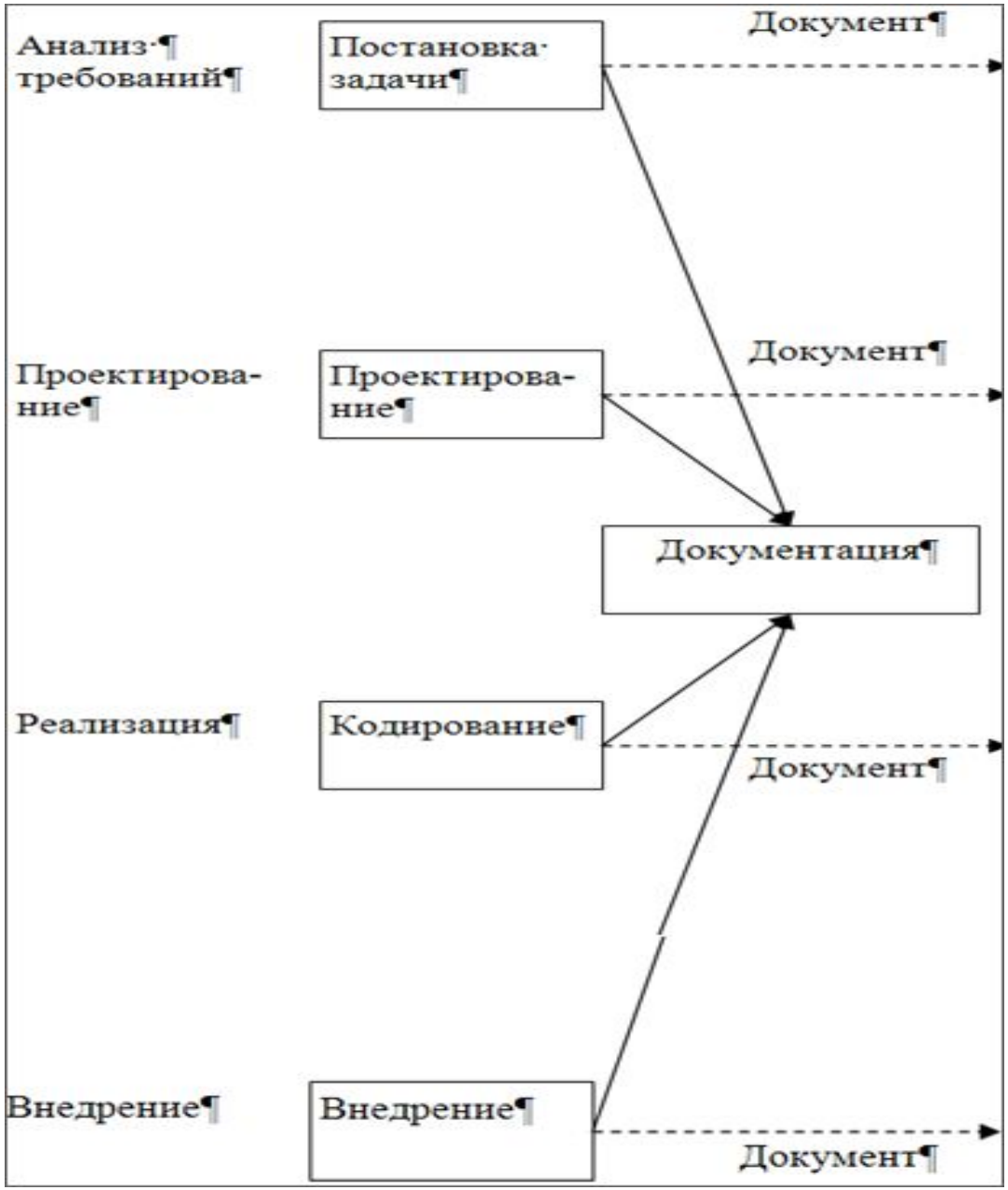
Показатели качества программ

- ***Функциональная полнота*** – возможно наиболее полная реализация внешних функций.
- ***Работоспособность*** – система работает и реализует требуемые функции.
- ***Надежность*** – система работает без отказов и сбоев.
- ***Робастность*** (восстанавливаемость) – способность системы восстанавливаться при возникновении ошибочных ситуаций как внешнего, так и внутреннего происхождения.
- ***Эффективность*** – система реализует свои функции наилучшим образом.

Показатели качества ПО

- ***Экономическая эффективность*** – минимальная стоимость конечного продукта при максимальной прибыли.
- ***Учет человеческого фактора*** – удобство эксплуатации, быстрота обучения работе с ПП, удобство сопровождения, внесения изменений.
- ***Переносимость*** (мобильность) – переносимость кода на другую платформу или систему.
- ***Точность вычисления*** – достижимая точность арифметических вычислений.

Модель ЖЗПО в учебном процессе



Постановка задачи

- На этапе *постановки задачи* осуществляется анализ требований и в результате формируется корректно сформулированное ***техническое задание*** (ТЗ). Техническое задание является словесным описанием и должно быть кратким, точным, четким и емким. ТЗ содержит:
- 1. Описание сути задачи.
- 2. Описание требуемого интерфейса.
- 3. Пример работающей модели задачи.

Документом являются *внешние спецификации* программы.

Внешние спецификации

- Внешними спецификациями называется документ, который отражает ТЗ и более подробно его описывает. **Все описания, представленные в этом документе, пишутся в терминах заказчика**, а сам документ становится *юридическим*, и впоследствии меняться уже не будет.

Связь ТЗ и внешних спецификаций

Техническое задание	Внешние спецификации
1. Суть задачи	1. Наименование задачи. 2. Словесное описание задачи. 3. Внешняя спецификация данных. 4. Функциональная спецификация
2. Требуемый интерфейс	5. Спецификация интерфейса
3. Пример работающей модели задачи	6. Спецификация внешнего тестирования

Разделы внешней спецификации

- **Именование задачи** - *краткое* и *информативное* название; отражает суть поставленной проблемы или назначение будущей программы.
- **Словесное описание задачи** -описание задачи в терминах заказчика. Отражает внешнюю модель решения задачи, связь между данными на входе программы и ее результатом.

Разделы внешней спецификации

- **Внешняя спецификация данных** содержит: описание данных программы как объектов внешнего мира;
- описание входных данных;
- описание выходных данных;
- внешнюю вычислительную модель – модель преобразования входных данных в выходные

Описание данных

Объект	Свойства объекта	Характеристики свойства	Связь между объектами и внутри объектов
Объекты внешнего мира	Свойства объектов, значимые с точки зрения решаемой задачи	Для каждого свойства указываются область определения и ограничения	Связь внутри объекта может быть аналитической или логической Связи между объектами являются вычислительными моделями задачи

Функциональные спецификации

- функции интерфейса;
- функции ввода исходных данных;
- функции обработки и вычисления результатов
- функции вывода

Спецификация интерфейса

- *Интерфейс* программы должен удовлетворять требованиям заказчика и должен отражать функциональную спецификацию. Внешний вид программы описывается отдельными *экранами* (экраны заставки, ввода данных, вывода результатов) и поясняется краткими, но информативными комментариями. Описание экранов должно быть наглядным

Спецификация внешнего тестирования

- Содержит данные для тестирования программы (*по данным*) и данные для тестирования внешней спецификации (*по функциям и интерфейсу*).

Проектирование

- Разрабатываются модели;
- Проектируются процедуры и соответствующие алгоритмы;

Документом являются внутренние спецификации: данные, модели, алгоритмы, данные для автономного тестирования.

Кодирование

- Выбор языка и среды программирования.
- Кодирование алгоритмов.
- Автономная отладка и тестирование.

Документом является отлаженный и протестированный код программы.

Внедрение

Изменения в программе по мере работы с ней.

Документом являются файлы отчета, содержащие ошибки, несоответствие спецификациям, изменения в кодах

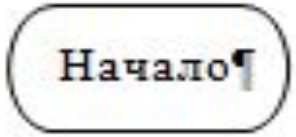
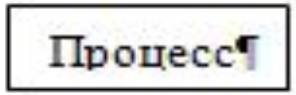
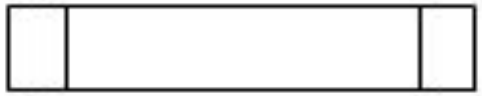
Способы описания алгоритма

- ***Все алгоритмы процедур и функций, а также обобщенный алгоритм будущей программы описываются на псевдокоде или при помощи блок-схем.***

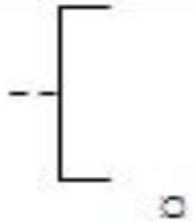
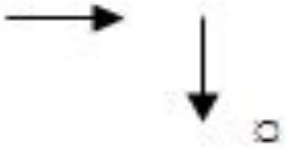
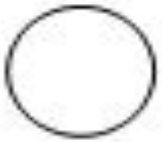
Блок-схема

- *Блок-схема* – это графическое изображение алгоритма в виде плоских геометрических фигур (блоков), соединенных линиями. Внутри блока записывается действие, которое нужно выполнить, или условие, которое необходимо проверить.
- . Существует государственный стандарт (ГОСТ 19.791–90 ЕСПД), содержащий перечень правил построения блок-схем.

Основные блоки

Символ	Графическое обозначение
Начало/конец	
Процесс	
Предопределенный процесс	
Условный (альтернативный) блок	

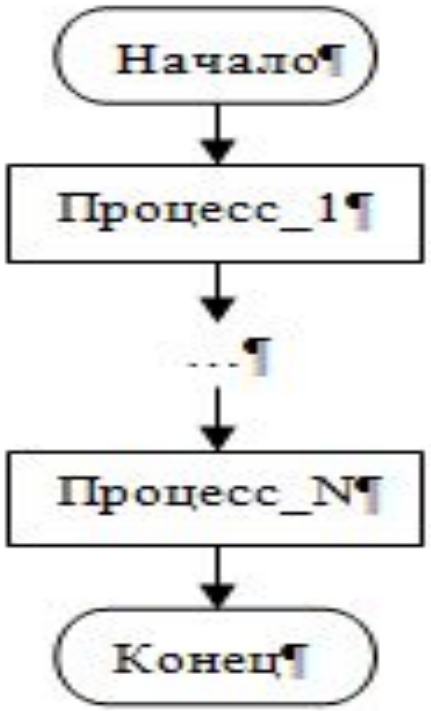
Основные блоки

Комментарий	
Ввод/вывод	
Линия связи	
Символ переноса	

Структурный подход к программированию

- Используются типовые алгоритмические структуры, имеющие один вход и один выход:
- Следование;
- Ветвление;
- Цикл

Линейный алгоритм

Название	Блок-схема	Псевдокод
Следование		нач ··Процесс_1 ····· ··Процесс_N кон
		Pascal

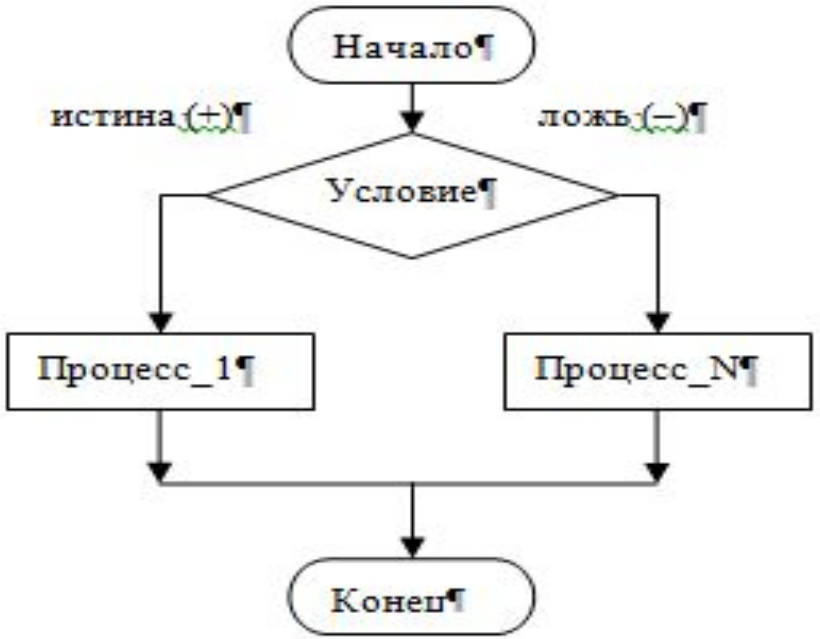
		Begin
		··Process_1;
		·····
		··Process_N;
		End;

		C

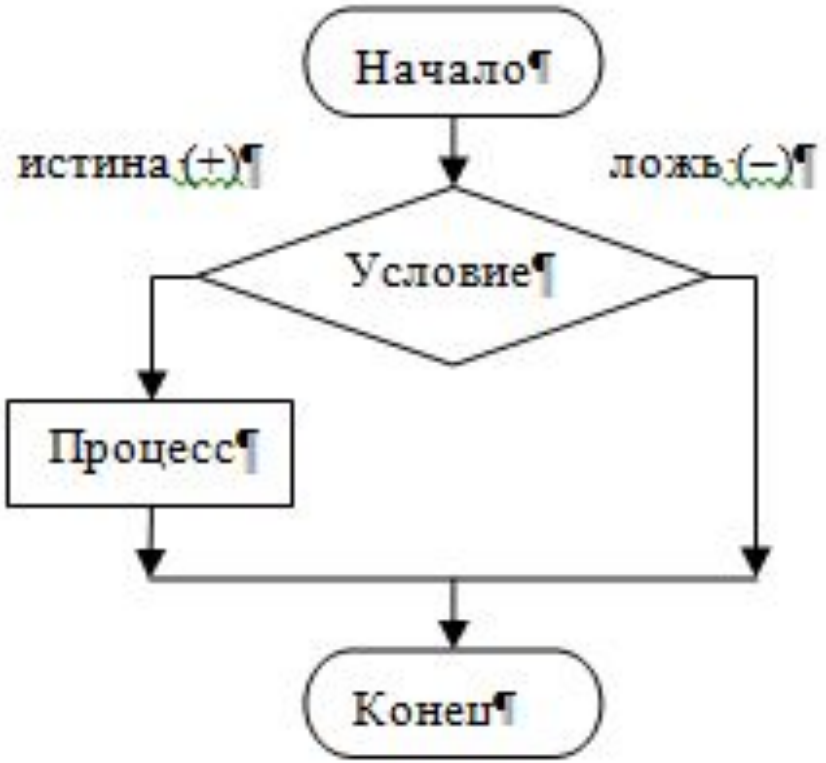
		{
		··Process_1();
		·····
		··Process_N();
		}

Ветвление

Название	Блок-схема	Псевдокод

Ветвление		нач
		··если ·<условие>
		··то
		····Процесс_1
		··иначе
		····Процесс_2
		··все
		кон
		Pascal
		Begin
		··If ·<условие>
		··Then
		····Process_1
		··Else
		····Process_2;
		End;
		C
		{
		··if ·(<условие>)
		····Process_1();
		··else
		····Process_2();
		}

Обход

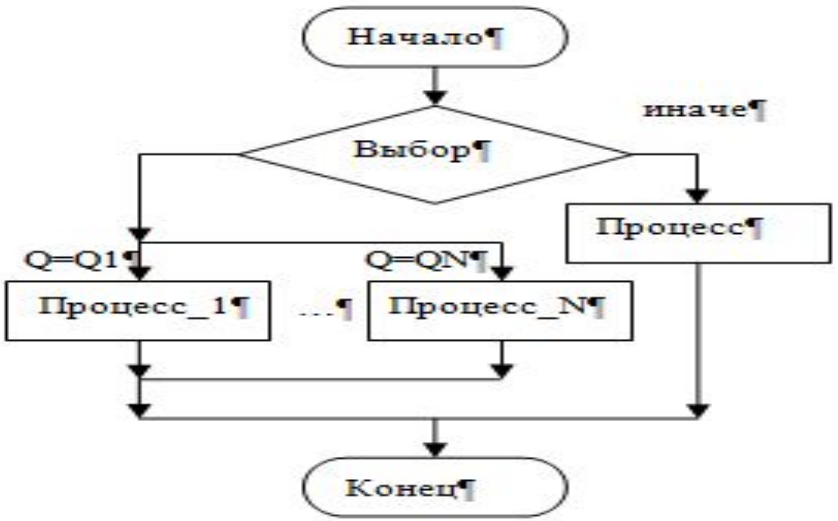
Название	Блок-схема	Псевдокод
Обход		**нач** ··**если** ·<условие> ··**то** ····Процесс ··**все** **кон**
		Pascal

		Begin
		··If ·<условие>
		··Then
		····Process;
		End;

		C

		{
		··if ·(<условие>)
		····Process();
		}

выбор

Название	Блок-схема	Псевдокод
Выбор		нач ··выбор по Q ····при Q1:Процесс_1 ····при Q2:Процесс_2 ·····... ····при QN:Процесс_N ····иначе Процесс ··кон выбора кон

Pascal

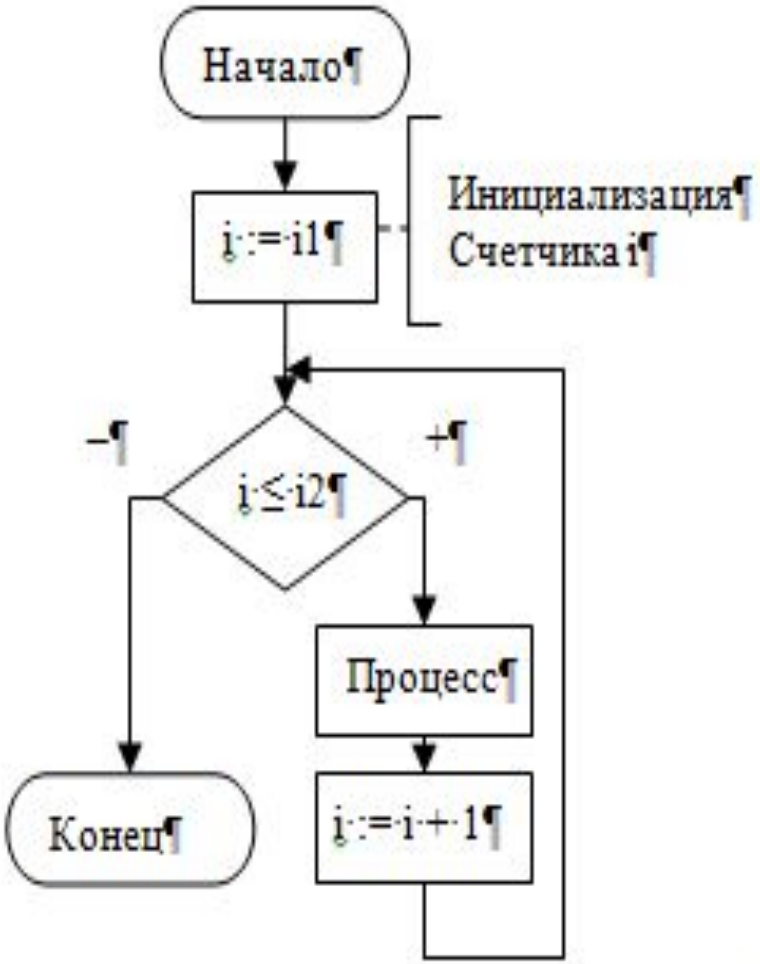
```
Begin
  Case Q of
    Q1:Process_1;
    Q2:Process_2;
    ...
    QN:Process_N
    Else Process;
  End;
End;
```

C

```
{
  switch (Q)
  {
    case°Q1:
      Process_1();
      break;
    case°Q2:
      Process_2();
      break;
    ...
    case°QN:
      Process_N();
      break;
    default:
      Process();
  }
}
```

Цикл с параметром

Название	Блок-схема	Псевдокод

Цикл с параметром		нач
		  нц для i от i1 до i2 шаг 1
		  Процесс
		  кц
		кон

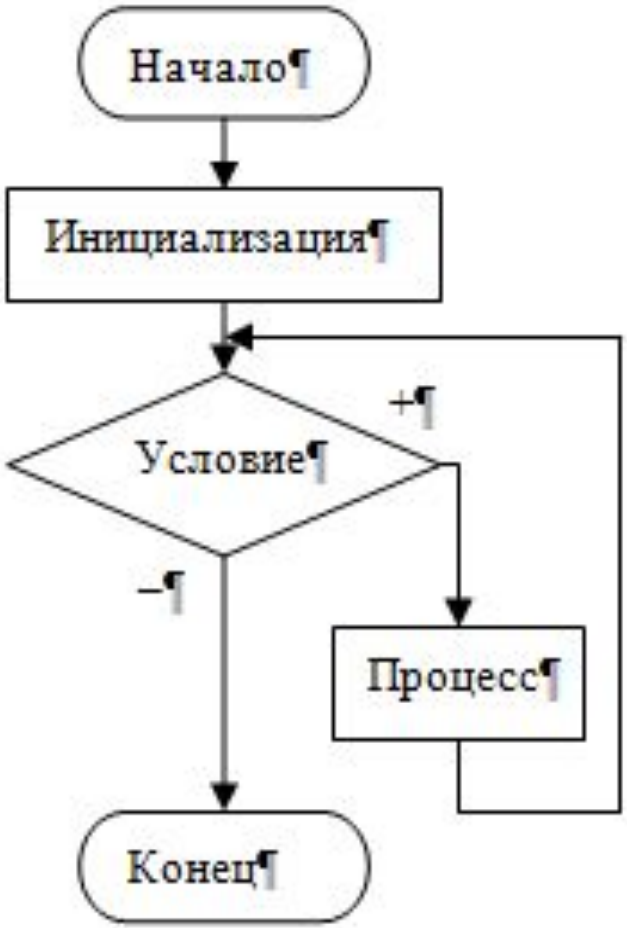
		Pascal

		Begin
		  For i := i1 To i2 Do
		  Process;
		End;

		C

		{
		  for (i = i1; i <= i2; i++)
		  Process();
		}

Цикл с предусловием

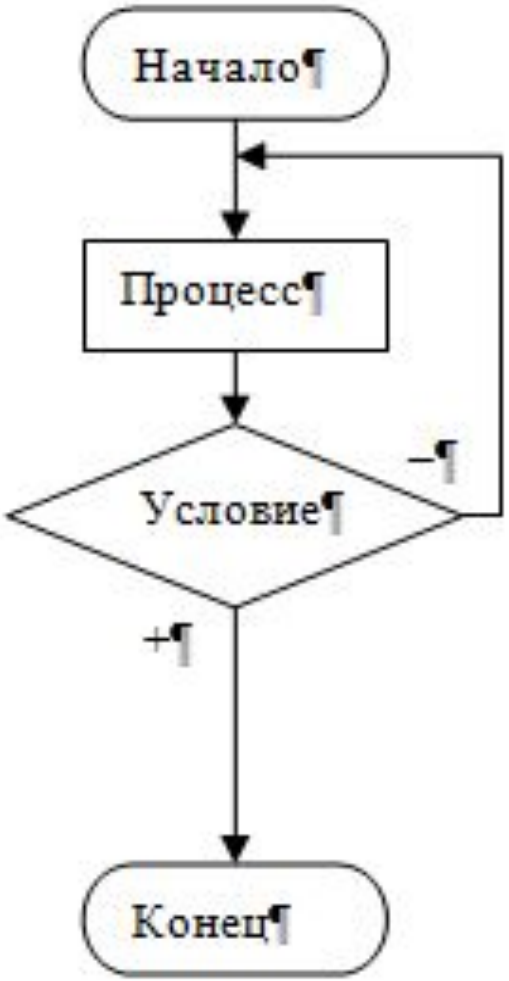
Название	Блок-схема	Псевдокод
Цикл с предусловием		нач ··Инициализация ··нц·пока·<условие> ····Процесс ··кц кон
		Pascal

		Begin
		··While·<условие>·Do
		····Process;
		End;

		C

		{
		··while·(<условие>)
		····Process();
		}

Цикл с постусловием

Название	Блок-схема	Псевдокод
Цикл с постусловием		нач ··нц ····Процесс ··кц при <условие> кон
		Pascal

		Begin
		··Repeat
		····Process;
		··Until <условие>;
		End;

		C

		{
		··do
		····Process();
		··while (!<условие>);
		}