

1. Диаграммы Нанси-Шнейдермана

Диаграмма Нэсси–Шнейдермана (структурограмма) – это схема, иллюстрирующая структуру передачи управления внутри модуля с помощью вложенных друг в друга блоков. Каждый блок имеет форму прямоугольника и может быть вписан в любой внутренний прямоугольник любого другого блока. Запись внутри блока производится на естественном языке или с помощью математических обозначений.

При использовании структурограмм следует иметь в виду ряд обстоятельств:

- структурограмма, описывающая полную разработанную программу, сама представляет собой большой символ обработки, содержащий внутри себя другие символы;
- в полной программе, описанной системой вложенных блоков (структурограммой), управление начинает свой путь на верхней стороне внешнего прямоугольника, проходит через каждый прямоугольник и завершает путь на нижней стороне внешнего прямоугольника;
- структурограмма законченной программы или ее логически завершенной части должна размещаться на одной странице; в отличие от схем алгоритмов разрыв структурограмм и перенос их на другую страницу недопустим, так как здесь отсутствуют явные линии передачи управления от блока к блоку.

Диаграммы Нэсси-Шнейдермана являются развитием Flow-форм. Основное их отличие от Flow-форм заключается в том, что область обозначения условий и вариантов ветвления изображают в виде треугольников. Такое обозначение обеспечивает большую наглядность представления алгоритма.

а - следование; *б* - ветвление; *в* - выбор; *г* - цикл-пока; *д* - цикл-до; *е* - счетный цикл

Так же, как при использовании псевдокодов, описать неструктурный алгоритм, применяя Flow-формы или диаграммы Нэсси-Шнейдермана, невозможно (для неструктурных передач управления в этих нотациях просто отсутствуют условные обозначения). В то же время, являясь графическими, эти нотации лучше отображают вложенность конструкций, чем псевдокоды.

Общим недостатком Flow-форм и диаграмм Нэсси-Шнейдермана является сложность построения изображений символов, что усложняет практическое применение этих нотаций для описания больших алгоритмов.

2. Дракон схемы

Существуют визуальные языки для представления алгоритмов, в которых тоже есть порядок и структура, например ДРАКОН, BPMN и LML Action Diagrams. ДРАКОН не является самостоятельным языком программирования. Он работает в паре

с текстовым языком, например, с JavaScript, Python или C++. Вместе с текстовым языком, ДРАКОН образует гибридный язык: ДРАКОН-JavaScript, ДРАКОН-Python или ДРАКОН-C++.

Программирование на гибридном языке происходит следующим образом:

1. Рисуем ДРАКОН-схему.
2. Внутри икон помещаем небольшие кусочки кода на соответствующем языке программирования.
3. Программа-транслятор преобразует ДРАКОН-схему в текстовый файл с исходным кодом.
4. Этот текстовый файл включается в проект обычным образом.
Генерацию кода из диаграмм на сегодняшний день поддерживают несколько редакторов. Примеры в данной статье сделаны в DRAKON Editor.

3. Диаграммы деятельности UML

Для наглядного изображения процессов и явлений используют схемы, но под одним и тем же символом люди могут иметь в виду разное. Чтобы исключить разночтения, разработали стандарт UML. UML-диаграмма — это схема, нарисованная с применением символов UML. Она может содержать множество элементов и соединений между ними. Полное описание масштабного проекта может состоять из несколько UML-диаграмм, связанных или не связанных между собой.

Элементами диаграммы UML могут быть классы программного кода, страницы сайта, части механизма, зоны торгового зала — в зависимости от того, какой процесс или какую сущность описывает её создатель. Все диаграммы UML можно поделить на структурные и поведенческие. Первые описывают структуру сложных объектов и систем, вторые иллюстрируют взаимодействие с системой и процесс её работы. Внутри эти типы делятся на виды UML-диаграмм. Разберём наиболее популярные.

Типы диаграмм UML:

Структурные диаграммы:

- Диаграмма классов
- Диаграмма компонентов
- Диаграмма объектов

Поведенческие диаграммы:

- Диаграмма действий, или диаграмма активностей, активити-диаграмма
- Диаграмма сценариев использования
- Диаграмма последовательностей

4. Диаграммы потоков данных (DFD)

Диаграммы потоков данных (DFD) — это методология графического структурного анализа, описывающая внешние по отношению к системе, источники и адресаты данных, логические функции, потоки данных и хранилища данных, к которым осуществляется доступ.

DFD-диаграмма включает следующие компоненты:

- **Процесс** — функция или действия по обработке данных. Обозначается в виде круга или прямоугольника со скруглёнными краями и горизонтальной чертой внутри.
- **Внешняя сущность** — объект за пределами моделируемой системы, который является отправителем или получателем данных — человек, внешний сервис, носитель информации, сторонний источник данных и пр.. Обозначается квадратом.
- **Хранилище данных** — источник, приёмник или промежуточное хранилище данных внутри моделируемой системы — база данных, таблица, документ, список, файл и пр.. Обозначается в виде прямоугольника с незакрытым правым краем, может иметь вертикальную черту слева.
- **Поток данных** — непосредственно данные, которые входят в процессы и хранилища или выходят из них. Например, «ФИО клиента», «Номер договора», «Сведения по заявке», «Запрос» и т.д.. Обозначаются как сплошные стрелки с подписями.

DFD позволяет описать последовательность возникновения, изменения и преобразования данных через их движение между процессами и хранилищами.

5. Записи алгоритма с помощью Р-схемы

Р-схема — это нагруженный по дугам ориентированный граф, изображаемый вертикальными и горизонтальными линиями и состоящий из структур (подграфов), каждая из которых имеет только один вход и один выход. Записи над дугой Р-схемы имеют смысл условия прохождения по дуге, а записи под дугой — выполняемого при этом действия.

Графический язык Р-схем сконструирован так, чтобы осуществить плавный переход от алгоритма, представленного в виде двумерной картинке, к двумерной программе. При этом базис алгоритма и программы не изменяется. С помощью Р-схем можно легко и быстро решать любые задачи, в том числе создавать программные системы.

Источники

- https://itonboard.ru/analysis/664-diagramma_dejatelnosti_rukovodstvo_dlja_nachinajushhih/?ysclid=lsuz0pkgw1191638757
- <https://practicum.yandex.ru/blog/diagramma-potokov-dannyh-dfd/>
- <https://habr.com/ru/articles/345320/>
- <https://habr.com/ru/articles/541478/>
- https://practicum.yandex.ru/blog/uml-diagrammy/?utm_source=yandex&utm_medium=cpc&utm_campaign=Yan_Perfmax_RF_Data_Resk_b2c_Gener_Regular_Blog_460&utm_content=sty_search:s_none:cid_117839318:gid_5537384220:pid_54219252160:aid_16819193238:crd_0:rid_54219252160:p_1:pty_premium:mt_syn:mkw_:dty_desktop:cgcid_0:rn_Санкт-Петербург:rid_2&utm_term=диаграмма%20деятельности%20uml&yclid=16184414186580475903#diagrammy
- https://bstudy.net/724984/informatika/diagrammy_nassi_shneydermana?ysclid=lsv1vp5c5t336055479
- https://studopedia.su/1_15459_strukturogrammi.html
- <https://shareslide.ru/informatika/prezentatsiya-po-programmirovaniyu-graficheskoe-predstavlenie-algoritmov>