

## СЕМАНТИКА ПРОБЛЕМ ПРОГРАММИРОВАНИЯ

Развеева И.Ф., Кучма Н.А., старший преподаватель

*ФГБОУ ВО «Донской Государственный Технический Университет», (344000, г. Ростов-на-Дону, пл. Гагарина, 1) [reception@donstu.ru](mailto:reception@donstu.ru)*

*[razveevai@mail.ru](mailto:razveevai@mail.ru), [natalia.kuchma@gmail.com](mailto:natalia.kuchma@gmail.com)*

## SEMANTICS OF PROGRAMMING PROBLEMS

*Razheeva I.F., Kuchma N.A., senior lecturer*

*Don State Technical University (344000, Rostov-on-Don, Gagarin area, 1) [reception@donstu.ru](mailto:reception@donstu.ru)*

*[razveevai@mail.ru](mailto:razveevai@mail.ru), [natalia.kuchma@gmail.com](mailto:natalia.kuchma@gmail.com)*

Решение проблем программирования – постоянный процесс, выполняемый первоначально, сосредоточившись только на знании синтаксиса. Такой способ может безосновательно продолжаться неоднократно, надеясь получить верное решение.

Иногда, таким образом, можно удачно получить решение. Но это неэффективный способ решения проблем. И это огромная потеря не только времени. Лучший способ заключается в том, чтобы иметь основу предметной области и возможности практического применения.

Большинство работодателей в первую очередь отдают предпочтение навыкам решения проблем – самой важной характеристике квалификации. Способность вычислительного мышления, то есть способность разбивать большие, сложные проблемы столь же всегда актуальна, как и базовые технические навыки, необходимые для работы.

Программист, сталкиваясь с новой проблемой, выстраивает последовательность решения.

Выясняется понимание вопроса. Большинство сложных проблем трудны, потому что нет понимания. Показатель ясности проблемы, если возможно, объяснение простыми словами.

Объяснение решения проявляет все погрешности в логике, которые не видны были раньше.

Большинство программистов знают это свойство. Поэтому необходимо выполнить письменный, схемный или устный алгоритм.

Не начинать решение без плана. Ничто не поможет, если невозможно записать точные шаги.

В программировании это означает, что не надо сразу начинать взламывать. Необходимо проанализировать проблему и обработать информацию. План эффективен, если найден ответ на вопрос: учитывая вход  $X$ , какие шаги необходимы, чтобы выполнить выход  $Y$ ? У программистов есть отличный инструмент, чтобы помочь в этом – комментарии.

Разделить проблему на подзадачи. Это самый важный шаг из всех. Не пытаться решить одну большую проблему. Вместо этого разбить ее на подзадачи. Эти подзадачи гораздо проще решить. Затем решить каждую подзадачу одну за другой. Начать с самой простой. Нахождение самой простой означает, что есть знание ответа (или приближение к этому ответу). После этого самое простое означает, что эта решаемая подзадача не зависит от решаемых других. После того, как найдено решение каждой подзадачи можно соединить подзадачи. Соединение всех дополнительных решений дает решение исходной проблемы. Эта техника является ключевым вопросом решения проблем программистом.

Рассмотрим вариант – выполнены предыдущие шаги, но решение не получено. Надо отметить, что лучшие программисты больше интересуются ошибками, чем возмущаются. На самом деле, столкнувшись с отсутствием результата, необходимо пошагово пройти по решению, стараясь найти причину ошибки. Программисты называют это отладкой.

“Искусство отладки состоит в том, чтобы выяснить, что вы действительно сказали своей программе, а не то, что, как вы думали, что сказали”.

Посмотреть на проблему с другой точки зрения. Есть ли что-нибудь, что можно абстрагировать от более общего подхода?

“Иногда можно обращая внимание на детали проблемы, упустить общие принципы, которые могли бы решить проблему на более общем уровне.

Классическим примером этого является суммирование длинного ряда последовательных целых чисел,  $1 + 2 + 3 + \dots + n$ , который очень молодой Гаусс быстро признал как  $n(n + 1) / 2$ , таким образом, избегая действия того, чтобы сделать сложение”.

Еще один способ – удалить все и начать снова с чистого листа. Удивляет тем, насколько это эффективно.

Наиболее часто встречаемые ошибки обсуждаются с некоторыми решениями на специализированных форумах.

Решение проблем – это систематический навык, который нужно развивать. Все проблемы имеют сходные закономерности. Необходимо находить возможности для практики, что позволяет решать многие микро проблемы.

Список литературы:

1. Антон Спрол, Думай как программист, изд-во Эксмо, 2018.
2. Дэвид Аллен, Как привести дела в порядок, изд-во Манн, Иванов и Фарбер, 2016.