# Отчёт №6: «Динамическое программирование».

Динамическое программирование– способ решения сложных задач путём разбиения их на более простые подзадачи.

Он применим к задачам с оптимальной подструктурой, выглядящим как набор перекрывающихся подзадач, сложность которых чуть меньше

исходной. В этом случае время вычислений, по сравнению с «наивными» методами, можно значительно сократить.

Ключевая идея в динамическом программировании достаточно проста. Как правило, чтобы решить поставленную задачу, требуется решить отдельные части задачи, после чего объединить решения подзадач в одно общее

решение. Часто многие из этих подзадач одинаковы. Подход динамического программирования состоит в том, чтобы решить каждую подзадачу только один раз, сократив тем самым количество вычислений. Это особенно

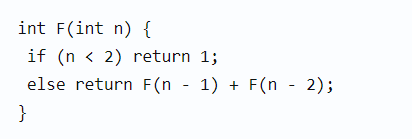
полезно в случаях, когда число повторяющихся подзадач экспоненциально велико.

**Метод динамического программирования сверху** – это простое запоминание результатов решения тех подзадач, которые могут повторно встретиться в дальнейшем.

**Динамическое программирование снизу** включает в себя переформулирование сложной задачи в виде рекурсивной последовательности более простых подзадач.

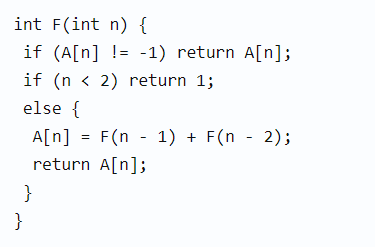
# Пример

Классической задачей на последовательности является задача на нахождение последовательности Фибоначчи. Одним из способов решения – с помощью рекурсии:

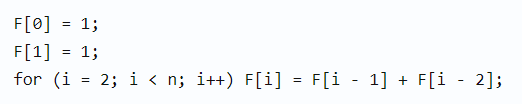


Но такая программа работает заметно дольше при n > 40. Это связано с тем, что одни и те же промежуточные данные вычисляются по несколько раз.

Один из выходов из данной ситуации – сохранение уже найденных промежуточных результатов с целью их повторного использования:



Или, более простое и эффективное решение:



Именно такое решение и является классическим для динамического

программирования: мы сначала решили все подзадачи, затем, зная решения подзадач, нашли ответ.

# Источники:

1. <https://ru.wikipedia.org/wiki/Динамическое_программирование>
2. <https://habr.com/ru/post/191498/>
3. https://tproger.ru/articles/dynprog-starters/