

Екзамен
з теорії алгоритмів
Білет №10.

1. Нехай задані часткові ^{числові} функції $g^n: N^{(n)} \rightarrow N$,
 $h^{n+2}: N^{(n+2)} \rightarrow N$.

Оператор, який за формулами

$$\begin{cases} f(x_1, \dots, x_n, 0) = g(x_1, \dots, x_n) \\ f(x_1, \dots, x_n, y+1) = h(x_1, \dots, x_n, y, f(x_1, \dots, x_n, y)) \end{cases} \text{ або}$$

$$\begin{cases} f(0) = a \\ f(y+1) = h(y, f(y)) \end{cases}$$

з функції g і h дає змогу побудувати функцію f , називають оператором примітивної рекурсії і позначають R :

$$f = R(g, h).$$

2.

Якщо задача є розв'язною, то вдається побудувати декілька різних алгоритмів для її розв'язання. Для того, щоб алгоритм давав найкращий розв'язок, потрібно пам'ятати про ефективні методи розробки алгоритмів.

До таких методів належать:

- 1) метод часткових цілей (пов'язаний зі зведенням великої задачі до послідовності простіших)
 - 2) метод підйому (починається із прийняття початкового припущення або обчислення початкової розв'язку)
 - 3) метод відпрацювання назад (починається з цілі чи розв'язку і рухається назад до початкового формулювання задачі)
 - 4) рекурсія (процес, який прямо чи опосередковано звертається до себе)
 - 5) динамічне програмування
- застосовують до задач оптимізації, у яких для того, щоб отримати оптимальний розв'язок, необхідно зробити певну множину виборів. Після того, як вибір зроблено, часто виникають доповнісні підзадачі.

Динамічне програмування ефективне

коли певна допоміжна підзадача виникає
внаслідок декількох варіантів виборів.

Головний метод розв'язування таких задач
полягає у збереженні розв'язку кожної підза-
дачі, яка може виникнути повторно. Ви-

ходить, що один раз розв'язана задача
вже не буде розв'язуватись.

Процес розробки алгоритмів динамічного
програмування складається з 4 етапів:

1. опис структури оптимального розв'язку

2. рекурсивне визначення значення, що відпо-
відає оптимальному розв'язку,

3. обчислення значення, що відповідає
оптимальному розв'язку

4. утворення оптимального розв'язку,
на основі виконання пунктів 1-3).

3. $A = (k, l, m, n)$

I) 1) $m \rightarrow _$

2) $n \rightarrow _$

3) $ll \rightarrow .nnn$ (заключна)

[- - порожній символ]
[. - заключно]

k l l m n l l

k l l _ n l l

k l l _ _ l l

k n n n _ _ l l

II) якщо n n n use замінити пороками
символами.

1) m → _

2) n n n! → _ _ _ (закінчення)

3) n → _

4) l l → n n n!

k l l m n l l

k l l _ n l l

k l l _ _ l l

k n n n! _ _ l l

k _ _ _ _ l l