

КОНЦЕПЦІЯ СТВОРЕННЯ АВТОМАТИЗОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ „НАСЕЛЕННЯ”

Ходаков В.Є., Шеховцов А.В., Бараненко Р.В.

Херсонський національний технічний університет, м. Херсон
кафедра інформаційних технологій

E-mail: hod@ist.com.ua

Abstract

Hodakov V.E., Shechovtsov A.V., Baranenko R.V. The concept of creation of automated information system „Population”. In the given article the aspects of designing of the automated information system "Population", which basic component is the database "The Population of Ukraine", the features of data structure in this system, criterions which the projected automated system should satisfy are considered, the algorithms of information search in system are analyzed and the using of integrational and multicriterion approaches for creation of system and signature hash for organization of text search are offered. The new approach, which consists in use for creation of system of the working automated systems of bodies of the Ministry of Justice of Ukraine, State tax administration of Ukraine, Pension fund of Ukraine, Ministry of work and social policy of Ukraine, Fund of social insurance from accidents on manufacture and occupational diseases, temporary loss of work capacity and their regional divisions for formation of the "Population" system is proposed.

Вступ. Реформування національної економіки призвело до необхідності модернізації системи та принципів державного управління, що полягає, насамперед, в зміні підходів до державного менеджменту, необхідності прийняття оптимальних й науковообґрунтованих управлінських рішень, що повинні базуватися на використанні сучасних інформаційних технологій управління, серед яких значне місце займають автоматизовані системи управління, автоматизовані інформаційні системи і системи електронної обробки даних.

Постановка проблеми. Сформована в Україні система територіального управління за рядом параметрів вже не відповідає сучасним вимогам [1].

Виконання державою однієї із своїх найголовніших функцій — соціального захисту населення потребує негайної автоматизації наступних управлінських процесів:

- підготовки й виготовлення списків населення;
- аналізу і прогнозування демографічної ситуації;
- розробки принципів і напрямків соціальної політики;
- створення ефективного механізму соціальних гарантій і соціального захисту населення;
- оцінки якісних характеристик населення;
- формування гендерних відносин у суспільстві.

Першими кроками для вирішення соціальних проблем населення з використанням сучасних підходів до державного управління є розробка відповідної нормативної бази, що вже відбувається, й створення автоматизованої інформаційної системи (АІС), яка інтегрувала би у собі різноманітні відомості про населення й функції аналізу й прогнозування поточної демографічної ситуації та можливість розробки відповідних рекомендацій для прийняття виважених рішень. На жаль, на даний момент такої системи в країні не існує.

Іншою важливою задачею, що потребує негайного вирішення, є утворення територіальних виборчих округів за абсолютно новою концепцією, що також потребує наявності відомостей про населення. Якщо раніше виборчі округи формувалися з приблизно рівною кількістю виборців, що пояснювалося мажоритарною системою, то нині вони

матимуть приблизно однакову кількість виборчих дільниць. Тоді рівна кількість виборців давала різну кількість виборчих дільниць. Наприклад, у великих містах 80 виборчих дільниць «охоплювали» 200 тис. виборців, але, щоб «набрати» ті ж 200 тисяч виборців у Тернопільській, Вінницькій та деяких інших областях, потрібно було 300 і більше дільниць. Зрозуміла річ, місцевим окружкам важче організувати роботу сотень дільниць, ніж десятків. А головне — прийняти в ніч голосування стільки протоколів!

Сьогодні закон підходить до цього по-іншому. У Києві, де колись було 12, а потім 10 округів, буде всього сім, але в кожному з них — майже по півмільйона виборців. У Донецькій області було 23 округи, тепер лише 17, на Дніпропетровщині відповідно 17 — тепер 12. У той же час на Тернопільщині замість 5 округів буде 8, на Вінниччині замість 8-ми — 10, на Волині — майже вдвічі більше [50]. Звісно, що цей процес також потребуватиме відповідного інформаційного й програмного забезпечення.

Аналіз останніх досліджень. Проблемам створення автоматизованих систем управління, інформаційних систем та баз даних присвячені праці багатьох вчених [1–15]. Головним аспектом є застосування системного підходу для проектування АСУ [3]. Автоматизовані інформаційні системи повинні реалізовувати швидкодіючі алгоритми інформаційного пошуку.

Основна маса праць з дослідження обчислювальної складності алгоритмів пошуку інформації зв'язана з розробкою нових ефективних алгоритмів пошуку, що знаходять численні застосування в різних областях, таких, як машинне проектування, машинна графіка, бібліотечно-інформаційні системи, робототехніка, системи штучного інтелекту і багатьох інших [16–23]. У цих роботах оцінюється складність пропонуємих алгоритмів (найчастіше порядок складності) й порівнюється зі складністю раніше розроблених алгоритмів. У ряді праць пропонується інший підхід, зв'язаний із введенням математичних моделей обчислень, що використовуються головним чином для одержання нижніх оцінок складності обчислень [2, 16, 24–28]. Серед цих моделей найбільш відомою є так зване алгебраїчне дерево обчислень Бен-Ора [26]. Як різновид алгебраїчного дерева обчислень можна розглядати алгебраїчне дерево рішень порядку d [28]. У випадку, коли d дорівнює 1, виходить лінійне дерево рішень, з використанням якого отримані докази ряду нижніх оцінок складності [27, 29–31]. Проте, питання розробки ефективних алгоритмів інформаційного пошуку залишаються й досі актуальними.

Ціль статті. Метою роботи є визначення основних проблем створення АІС «Населення», особливостей організації даних у системі, критеріїв, яким має задовольняти автоматизована система, аналіз існуючих та розробка нових алгоритмів інформаційного пошуку в системі та розробка моделей системи, що забезпечать оптимальне функціонування автоматизованої системи.

Основний матеріал. Автоматизовані системи застосовуються на рівні оперативного управління яким-небудь об'єктом з метою автоматизації управлінської діяльності. АІС служить для рішення задач, що мають високий зміст операцій з обробки даних. До таких операцій відносяться: збирання даних, маніпулювання ними, збереження даних і підготовка документів [2].

Пропонується створити автоматизовану інформаційну систему „Населення”, до **основних задач функціонування** якої відносяться:

- розробка методології оцінки якості населення на макро- і мікрорівні;
- розробка методології аналізу якості робочої сили;
- аналіз процесів реформування освіти;
- оцінка якісних характеристик окремих груп населення: дітей, пенсіонерів, інвалідів;
- розробка методології аналізу соціально-економічних проблем населення;
- вивчення, аналіз і прогнозування соціального розвитку села, зовнішньої трудової міграції;
- моніторинг бідності, адресна соціальна допомога.

В *основу концепції* створення АІС „Населення” покладені наступні принципи:

– використання наявних баз даних, що вже функціонують в різноманітних державних установах;

– інтеграція відомостей про населення в єдину документальну базу даних (БД);

– відкритість архітектури проектуємої системи;

– мінімізація витрат на створення й впровадження БД та АІС;

– забезпечення максимальної достовірності даних, що зберігаються в БД;

– забезпечення прийняттого часу пошуку необхідних відомостей в БД;

– простий та зручний користувальницький інтерфейс.

Основним компонентом АІС „Населення” має бути БД „Населення України”.

Найбільш зрозумілим способом створення відповідної БД є організація перепису населення. Однак це дуже трудомістка й дорога процедура і її неможливо провести без попередньої підготовки, тому що вона потребує значних фінансових, інформаційних й людських ресурсів. Економічно більш вигідно на основі інтеграційного підходу створити БД, що інтегрувала б відомості про населення з уже існуючих автоматизованих систем та БД, що функціонують у різних державних установах та відомствах.

Враховуючи перелічені функції АІС „Населення”, можна визначити її структуру, проаналізувавши структуру інформаційних потоків, що функціонують на регіональному та державному рівнях з інформацією про населення країни.

АІС „Населення” має складатися з трьох рівнів: державного, регіонального та місцевого.

Перший — *державний* рівень містить інформацію про все населення країни. Основними джерелами для формування БД „Населення України” є відомості із Державної податкової адміністрації (ДПА), Пенсійного фонду України (ПФУ), Центральної виборчої комісії (ЦВК), Головного обчислювального центру Єдиної державної автоматизованої паспортної системи (ГОЦ ЄДАПС).

На другому — *регіональному* рівні відомості, відсортовані на першому рівні для будь-якої адміністративно-територіальної одиниці України, надходять для подальшого уточнення до обчислювального центру обласних державних адміністрацій (ОЦА). Тут вони поповнюються додатковими відомостями з обласних управлінь: МНС, соціального захисту населення, державної служби зайнятості населення, міськ- та райвиконкомів. Після чого формується регіональна БД з відомостями про населення регіону. Далі проводиться сортування для формування основи БД для місцевого рівня й пересилання необхідної інформації на третій рівень.

На третьому — *місцевому* рівні проводиться доповнення БД відомостями про населення територіальної громади з уточненням відомостей, що надходять із: РАГСів (акти народження), пенсійних відділів УМВС та УСБУ, державного департаменту України з виконання покарань, районних управлінь соціального захисту населення (інваліди, отримувачі субсидій та різноманітних пільг), районних управлінь державної служби зайнятості населення, військових з’єднань, розташованих на території відповідного населеного пункту.

В БД „Населення України” повинні знайти відображення *відомості про наступні шари населення країни*:

1) осіб, які зареєстровані згідно з Законом України „Про Державний реєстр фізичних осіб — платників податків та інших обов’язкових платежів”;

2) осіб, які зареєстровані згідно з Законом України "Про загальнообов’язкове державне пенсійне страхування”;

3) осіб, зареєстрованих згідно з Постановою Кабінету Міністрів України від 02.04.2002 №450 «Про затвердження Порядку централізованого оформлення, виготовлення та видачі паспорта громадянина України для виїзду за кордон»;

4) осіб, які зареєстровані згідно з Законом України „Про державну реєстрацію юридичних осіб та фізичних осіб — підприємців”;

5) осіб, які тимчасово не працюють, та зареєстровані згідно з Законом України „Про зайнятість населення”;

6) осіб, які знаходяться на пенсійному забезпеченні згідно з Законами України „Про пенсійне забезпечення” та „Про пенсійне забезпечення військовослужбовців, осіб начальницького і рядового складу органів внутрішніх справ та деяких інших осіб”;

7) осіб, які зареєстровані згідно з Законом України „Про свободу пересування та вільного вибору проживання”;

8) осіб, яким виповниться на день призначення найближчих виборів 18 років, а також дітей до 18 років;

9) осіб, які проходять строкову службу в Збройних силах України;

10) осіб, які знаходяться на утриманні в Державному департаменті України з питань виконання покарань;

11) осіб, зареєстрованих в службі видачі субсидій відповідно до Постанови Кабінету Міністрів України від 22.10.97 № 1050 „Про заходи щодо подальшого вдосконалення надання населенню субсидій для відшкодування витрат на оплату житлово-комунальних послуг, придбання скрапленого газу, твердого та рідкого пічного побутового палива”;

12) осіб, зареєстрованих Міністерством з надзвичайних ситуацій України як ліквідатори або потерпілі від наслідків катастрофи на Чорнобильській АЕС;

13) осіб без постійного місця проживання;

14) осіб, які померли в період за останні 10 років до дати впровадження функціонування АІС „Населення”;

15) осіб, зареєстрованих згідно з Законом України „Про державну соціальну допомогу особам, які не мають права на пенсію, та інвалідам”;

16) осіб, зареєстрованих згідно з Постановою Кабінету міністрів України №117 „Про Єдиний державний автоматизований реєстр осіб, які мають право на пільги” від 29.01. 2003 р.

Дані про всіх цих осіб мають бути занесені до бази даних (БД) АІС „Населення”, над якими потім будуть виконуватися операції обробки даних, вибірки й сортування.

Таким чином, АІС „Населення” містить підсистеми: „Пенсіонери”, „Військовослужбовці”, „Отримувачі пільг”, „Виборці”, „Померлі громадяни”, „Інваліди”, „Тимчасово непрацюючі”, „Ув'язнені”, „Неповнолітні громадяни”.

Для визначення критеріїв, яким має задовольняти БД „Населення України” можна використати багатокритеріальний підхід, оскільки істотною особливістю задач проектування баз даних є наявність не одного, а декількох критеріїв (показників якості), що визначають ефективність функціонування БД. До них відносяться мінімальні витрати на створення БД, максимальна складність БД, максимальна швидкодія БД і т.д. Усі зазначені критерії взаємозалежні і суперечливі. Це визначає багатокритеріальний характер задач проектування БД [13, 32, 33]. Облік сукупності критеріїв ефективності БД значно ускладнює розробку критеріальних методів їхнього рішення. Тому в багатьох випадках, для спрощення задачі проектування і практичного рішення, визначають головний показник ефективності, і інші показники переводять до розряду обмежень [14, 15, 34]. Однак такий підхід не завжди ефективний, тому що отриманий варіант БД не враховує всієї множини вимог, що висуваються до проекту. Крім того, не завжди можливо виділити один критерій в якості головного [35].

Разом з тим необхідно враховувати наступні особливості процесу проектування БД: високу вірогідність вихідних даних задач проектування; наявність факторів і обмежень, що важко формалізуються; велику розмірність розв'язуваних задач; значну складність розв'язуваних задач, великі витрати трудових, фінансових і обчислювальних ресурсів. Зазначені фактори дозволяють віднести задачу проектування БД до слабо структуризованих [35].

Найбільш загальним і універсальним підходом до рішення задачі багатокритеріального проектування БД є формування узагальненого скалярного критерію, що враховує всі різноманітні часткові критерії. У загальному випадку часткові

критерії $k_i(x)$ мають різний фізичний зміст, розмірність, діапазон вимірів, різні екстремальні значення. Тому спочатку їх необхідно привести до деякого загального (ізоморфного) виду. З цією метою використовується функція корисності часткових критеріїв наступного виду [36, 37]:

$$P_i[k_i(x)] = \frac{k_i(x) - k_{in\lambda}(x)}{k_{in\lambda}(x) - k_{in\chi}(x)} \alpha_i \quad (1)$$

де $k_i(x)$ — поточне значення i -го критерію;
 $k_{in\chi}(x)$ — найгірше значення i -го критерію;
 $k_{in\lambda}(x)$ — найкраще значення i -го критерію;
 α_i — показник нелінійності.

У залежності від виду екстремуму (напрямку домінування) маємо:

$$k_{in\lambda}(x) = \begin{cases} \max_{x \in X} k_i(x), & \text{якщо } k_i(x) \rightarrow \max, \\ \min_{x \in X} k_i(x), & \text{якщо } k_i(x) \rightarrow \min; \end{cases} \quad (2)$$

$$k_{in\chi}(x) = \begin{cases} \min_{x \in X} k_i(x), & \text{якщо } k_i(x) \rightarrow \max, \\ \max_{x \in X} k_i(x), & \text{якщо } k_i(x) \rightarrow \min; \end{cases}$$

Інтервал $[k_{in\lambda}(x), k_{in\chi}(x)]$ містить у собі крапки екстремумів усіх часткових критеріїв. Таким чином, значення $k_{in\chi}(x)$ і $k_{in\lambda}(x)$, $i = \overline{1, n}$ є границями наближеної області компромісів [37]. Тому в цьому випадку компромісні рішення, обрані з наближеної області компромісів, необхідно перевірити на належність області Парето.

В подальшому, будемо виходити з того, що відомі точні кількісні значення вагових коефіцієнтів a_i часткових критеріїв $k_i(x)$ або функцій корисності $p_i[k_i(x)]$. У цьому випадку узагальнений критерій синтезу БД буде мати такий вигляд:

$$W(x) = \sum_{i=1}^n a_i p_i[k_i(x)], i = \overline{1, n}, \sum_{i=1}^n a_i = 1, \quad (3)$$

де n — число часткових критеріїв;
 а принцип оптимальності — у вигляді:

$$x^0 = \arg \max_{x \in X} \sum_{i=1}^n a_i p_i[k_i(x)], i = \overline{1, n}, \sum_{i=1}^n a_i = 1. \quad (4)$$

Іноді зручніше вирішувати задачу мінімізації. І в цьому випадку оптимізаційна функція має вигляд:

$$x^0 = \arg \min_{x \in X} \sum_{i=1}^n a_i \overline{p}_i[k_i(x)], i = \overline{1, n}, \sum_{i=1}^n a_i = 1, \quad (5)$$

де $\overline{p}_i[k_i(x)] = 1 - p_i[k_i(x)]$ — функція втрати корисності.

Для визначення компромісного рішення, що належить області Парето, варто використовувати універсальну схему [36, 37]:

$$W(x^*) = \arg \max_{x^* \in \Omega_x} \frac{1}{n} \left\{ \sum_{i=1}^n p_i[k_i(x)]^\beta \right\}^{1/\beta}, \quad (6)$$

де x^* — компромісне рішення;
 Ω_x — область Парето;
 β — адаптаційний параметр.

або

$$W(x^*) = \arg \min_{x^* \in \Omega_x} \frac{1}{n} \left\{ \sum_{i=1}^n \overline{p}_i [k_i(x)] \right\}^{1/\beta}. \quad (7)$$

Конкретне значення коефіцієнта β можна визначити за формулою:

$$\beta = \frac{\log n}{\log(1 + \varepsilon)}, \quad (8)$$

де ε — відносна точність завдання або зміни критеріїв.

Тоді модель багатокритеріального проектування БД зводиться до пошуку екстремальних значень (6) або (7) при обмеженнях на фінансові, трудові й обчислювальні ресурси, що виділяються для проектування.

В разі ж застосування інтеграційного підходу для створення БД „Населення України” виникає проблема надмірності даних, визвана дублюванням відомостей про населення в різноманітних документах, що зберігаються в багатьох відомчих БД. Для вирішення цієї проблеми необхідно розробити оптимальні критерії вибірки документів із допоміжних БД до БД „Населення України”, а також розробити ефективні алгоритми інформаційного пошуку, вибірки й сортування вмісту БД.

Задача пошуку ідентичних об’єктів складається з пошуку в інформаційному масиві об’єкта, ідентичного об’єктові-запитові. Опишемо цю задачу формально.

Нехай дана множина X , на якій задане відношення лінійного порядку \preceq , тобто таке бінарне відношення на $X \times X$, що для будь-яких $x, y, z \in X$ задовольняє умовам:

- рефлексивності $x \preceq x$;
- транзитивності $(x \preceq y) \& (y \preceq z) \rightarrow (x \preceq z)$;
- антисиметричності $(x \preceq y) \& (y \preceq x) \rightarrow (x = y)$;
- зв’язності $(x \preceq y) \vee (y \preceq x)$.

Розглянемо наступний тип задач пошуку $S_{id} = \langle X, X, \rho_{id} \rangle$, де відношення пошуку ρ_{id} є відношення ідентичності, тобто $x \rho_{id} y \Leftrightarrow x = y$. Тип S_{id} будемо називати типом пошуку ідентичних об’єктів.

Нехай $N_0 = N \cup \{0\}$ — множина цілих невід’ємних чисел.

Нехай

$$L_l(l) = \begin{cases} 0, & \text{якщо } l = 0 \\ \log l + 1, & \text{якщо } l = 1, 2, 3 \\ \log l + 2, & \text{якщо } l \geq 4 \end{cases} \quad (9)$$

— функція, визначена на множині N_0 .

Доведемо наступне допоміжне твердження.

Лема 1 Нехай $L_l(l)$ — функція, визначена вище, нехай $k, m \in N$, нехай

$$r_l(k, m) \stackrel{def}{=} \max \left\{ \sum_{i=1}^m L_l(l_i) : l_i \in N_0, \dots, l_m \in N_0, \sum_{i=1}^m l_i = k \right\}.$$

$$\text{Тоді } r_l(k, m) = \left(k - \left[\frac{k}{m} \right] \cdot m \right) \cdot L_l \left(\left[\frac{k}{m} \right] + 1 \right) + \left(m - k + \left[\frac{k}{m} \right] \cdot m \right) \cdot L_l \left(\left[\frac{k}{m} \right] \right).$$

Доказ: Якщо до визначити функцію $L_l(l)$ на позитивну піввісь числової прямої, наприклад, у такий спосіб: $L_l(x) = \begin{cases} x, & \text{якщо } 0 \leq x \leq 4 \\ \log x + 2, & \text{якщо } x \geq 4 \end{cases}$, то отримана функція буде безперервною й ввігнутою. Ввігнутість цієї функції пояснює результат леми.

Більш докладний доказ будемо вести від протилежного.

Припустимо, що лема невірна, тобто

$$r_1(k, m) = \sum_{i=1}^m L_1(l_i') > \left(k - \left\lfloor \frac{k}{m} \right\rfloor \cdot m\right) \cdot L_1\left(\left\lfloor \frac{k}{m} \right\rfloor + 1\right) + \left\lfloor \frac{k}{m} \right\rfloor \cdot m \cdot L_1\left(\left\lfloor \frac{k}{m} \right\rfloor\right), \quad (10)$$

і серед чисел $l_i' (i = \overline{1, m})$ існує 2 числа, різниця яких не менше двох. Без обмеження спільності можемо вважати, що $l_1' - l_2' \geq 2$.

$$\text{Нехай } l_1'' = \left\lfloor \frac{l_1' + l_2'}{2} \right\rfloor, l_2'' = \left\lceil \frac{l_1' + l_2'}{2} \right\rceil. l_1'' + l_2'' = l_1' + l_2' \text{ та } l_1'' - l_2'' \leq 1.$$

Тому що функція $L_1(x)$ ввігнута, то за властивостями ввігнутих функцій

$$L_1(l_1'') + L_1(l_2'') \geq L_1(l_1') + L_1(l_2'). \quad (11)$$

Якщо в нерівності (11) нерівність строга, то одержали протиріччя, тому що $\sum_{i=1}^m L_1(l_i')$

— не максимальна; якщо ж ні, то позначимо $l_i'' = l_i' (i = \overline{3, m})$ і перейдемо до розгляду суми

$$\sum_{i=1}^m L_1(l_i'') = \sum_{i=1}^m L_1(l_i') = r_1(k, m).$$

Якщо серед чисел $l_i'' (i = \overline{1, m})$ немає пари чисел, різниця яких перевищує 1, то одержимо протиріччя з нерівністю (10), якщо ж є, то знову проробимо операцію, описану вище, і позбудемося від такої пари, і так будемо робити доти, поки або не одержимо строгу нерівність у нерівності (11), або не прийдемо до розбиття $l_1^{(n)}, \dots, l_m^{(n)}$, такого, що $\sum_{i=1}^m L_1(l_i^{(n)}) = r_1(k, m)$, і всі вони відрізняються не більше ніж на 1, і, тим самим, одержимо протиріччя з нерівністю (10), що доводить лему.

Нехай на X заданий ймовірностний простір $\langle X, \sigma, P \rangle$. Нехай $I_m^{-1}(x)$ — перемикач такий, що

$$I_m^{-1}(x) = i, \text{ якщо } x \in X_i (i = \overline{1, m}), \quad (12)$$

де X_1, X_2, \dots, X_m — розбиття множини X

(тобто $X = X_1 \cup X_2 \cup \dots \cup X_m$ та $X_i \cap X_j = \emptyset$, якщо $i \neq j$) таке, що

$$P(x_i) \leq c/m (i = \overline{1, m}), \quad (13)$$

де $c = const$, що не залежить від m . Тому що $P(x) = \sum_{i=1}^m P(X_i) \leq c$, то звідси відразу

випливає, що $c \geq 1$.

Нехай

$$I_a^{-2}(x) = \begin{cases} 1, & \text{якщо } x \leq a \\ 2, & \text{в іншому випадку} \end{cases}, a \in X, \quad (14)$$

$$f_a(x) = \begin{cases} 0, & \text{якщо } x \neq a \\ 1, & \text{якщо } x = a \end{cases}, a \in X. \quad (15)$$

Нехай

$$G_1 = \{I_m^{-1}(x) : m \in N\}, \quad (16)$$

$$G_2 = \{I_a^{-2}(x) : a \in X\}. \quad (17)$$

$$J = \{f_a(x) : a \in X\} \quad (18)$$

$$F = \langle J, G_1 \cup G_2 \rangle \quad (19)$$

Справедлива наступна теорема.

Теорема 1 Нехай $I = \{X, V, \rho_{id}\}$ — задача пошуку ідентичних об'єктів, тобто задача типу S_{id} , де $|V| = k$, F — базова множина, обумовлена співвідношеннями (12)–(19), c — константа, обумовлена співвідношенням (13), $L_l(l)$ — функція, обумовлена співвідношенням (9). Тоді

$$I < T(I, F, 2 \cdot k + m - 1) \leq \frac{c}{m} \left(\left(k - \left\lfloor \frac{k}{m} \right\rfloor \cdot m \right) \cdot L_l \left(\left\lfloor \frac{k}{m} \right\rfloor + 1 \right) + \left(m - k + \left\lfloor \frac{k}{m} \right\rfloor \cdot m \right) \cdot L_l \left(\left\lfloor \frac{k}{m} \right\rfloor \right) \right) + I.$$

Зокрема, $I < T(I, F, (2 + c) \cdot k) < 2$ і $T(I, F) \sim I$ при $k \rightarrow \infty$.

Доказ:

Побудуємо мережу U_m^0 , що має вид дерева, таким чином.

Візьмемо вершину β_0 й оголосимо її коренем мережі U_m^0 . Випустимо з β_0 m ребер, припишемо їм числа від 1 до m , оголосимо β_0 крапкою переключення і припишемо їй перемикач $I_m^1(x)$.

Нехай $V_i = X_i \cap V, l_i = |V_i|, i = \overline{1, m}$.

Кінець ребра з номером i позначимо β_i .

Для всіх таких i , що $V_i \neq \emptyset$, проробимо наступну процедуру. Випустимо з вершини β_i бінарне збалансоване дерево з l_i кінцевими вершинами, висоти $\lceil \log l_i \rceil$, усі ребра якого орієнтовані від кореня до кінцевих вершин. Якщо в цьому дереві є внутрішня вершина, з якої виходять ребра, що ведуть одне до кінцевої вершину, а інше до внутрішньої (якщо така вершина є, то вона тільки одна), то з кінцевої вершини, у яку веде одне з цих ребер, випустимо одне ребро. Отримане дерево позначимо D_i . Оголосимо кінцеві вершини цього дерева листами і зіставимо їм зліва направо у порядку зростання елементи з V_i .

Нехай β — довільна внутрішня вершина дерева D_i . Позначимо через V_β безліч записів, що відповідають листам гілки, що росте з β . Нехай β' — вершина, до якої веде ліве (якщо воно одне, то єдине) ребро з β . Нехай $y_\beta = \max_{y \in V_{\beta'}} y$.

Оголосимо всі внутрішні вершини дерева D_i , з яких виходять ребра, що ведуть до внутрішньої вершини, вершинами переключення і для кожної такої вершини β лівому ребру, що з неї виходить, припишемо 1, а правому — 2, а самій вершині припишемо перемикач $I_{y_\beta}^2(x)$.

Усі ребра дерева D_i , що входять до листів, оголосимо предикатними і припишемо кожному такому ребру, що веде до листа із записом y , предикат $f_y(x)$.

Отриману мережу з $|V| = k$ листами позначимо U_m^0 . Це мережа над базовою множиною F .

Покажемо, що U_m^0 вирішує задачу $I = \langle X, V, \rho_{id} \rangle$.

Нехай $y \in V_i (i \in \overline{1, m})$, тобто лист α належить дереву D_i . До нього веде єдиний ланцюг, що складається з не більш ніж $\lceil \log l_i \rceil + 1$ ребер, причому всі ці ребра, крім останнього, перемикальні. Тому що $I_m^1(y) = i$ і перше ребро в цьому ланцюзі є i -е ребро, що виходить з кореня, то провідність першого ребра в ланцюзі дорівнює 1. Останнє ребро в ланцюзі — предикатне з предикатом $f_y(x)$, і його провідність дорівнює $f_y(y) = 1$. Покажемо, що провідність інших ребер ланцюга дорівнює 1. Візьмемо будь-яке з цих ребер

(β, β') . Якщо (β, β') — ліве, що виходить з β , то $y \in V_{\beta'}$ і предикат, приписаний вершині β , $I^2_{y\beta}(y) = 1$, тому що $y \leq y_{\beta} = \max_{y \in V_{\beta'}} y'$.

Якщо (β, β') — праве ребро, що виходить з β , то $y > y_{\beta}$ і $I^2_{y\beta}(y) = 2$, тим самим провідність ребра (β, β') в обох випадках дорівнює 1.

Таким чином, ми показали, що $\varphi_{\alpha}(x) = f_y(x)$. З огляду на довільність листа α , одержимо, що мережа U^0_m вирішує задачу $I = \langle X, V, \rho_{id} \rangle$.

Обсяг мережі U^0_m дорівнює $Q(U^0_m) \leq m + \sum_{i=1}^m \max(0, 2 \cdot l_i - 1) \leq 2 \cdot k + m - 1$.

Складність отриманої мережі U^0_m дорівнює

$$M_x T(U^0_m, x) = \int_X T(U^0_m, x) P(dx) = \sum_{i=1}^m \int_{X_i} T(U^0_m, x) P(dx) \leq \sum_{i=1}^m (1 + L_1(l_i)) \cdot P(X_i) =$$

$$T(U^0_m) = 1 + \sum_{i=1}^m L_1(l_i) \cdot P(X_i) \leq 1 + \frac{c}{m} \sum_{i=1}^m L_1(l_i) \leq \frac{c}{m} \sum_{i=1}^m L_1(l_i) \leq 1 + r_1(k, m) \frac{c}{m} =$$

$$\frac{c}{m} \left(\left(k - \left\lfloor \frac{k}{m} \right\rfloor \cdot m \right) \cdot L_1 \left(\left\lfloor \frac{k}{m} \right\rfloor + 1 \right) + \left(m - k + \left\lfloor \frac{k}{m} \right\rfloor \cdot m \right) \cdot L_1 \left(\left\lfloor \frac{k}{m} \right\rfloor \right) \right) + 1$$

При $m =]c \cdot k[$ $Q(U^0_m) \leq 2k +]c \cdot k[-1 \leq (2+c) \cdot k$.

Тому що $c \geq 1$, те $m \geq k$. Якщо $m = k$, то $k/m = 1$ і $r_1(m, k) = 0 \cdot L_1(2) + k \cdot L_1(1) = k \cdot L_1(1)$.

Якщо $m > k$, то $k/m = 0$ і $r_1(m, k) = k \cdot L_1(1) + (m - k) \cdot L_1(0) = k \cdot L_1(1)$.

Отже, $T(I, F, (2+c) \cdot k) \leq T(U^0_m) \leq 1 + \frac{c}{]ck[} k \cdot L_1(1) < 2$.

Якщо взяти $m = k \cdot \alpha(k)$, де $\alpha(k) \rightarrow \infty$ при $k \rightarrow \infty$ і $\alpha(k) > 1$ при кожному k , то $T(I, F) \leq 1 + \frac{c \cdot k}{k \cdot \alpha(k)} < 1$.

З іншого боку, $T(I, F) \geq 1$, тому що з кореня повинне виходити хоча б одне ребро, оскільки $k > 0$.

Тим самим теорема доведена.

Оскільки БД „Населення України” містить відомості про населення в формі електронних документів, то особливу значимість одержує також задача повнотекстового пошуку, що потребує розробки ефективних й швидкодіючих алгоритмів її вирішення.

Найбільш розповсюджені в даний час методи пошуку, що використовуються інформаційно-пошуковими системами (ІПС), можна розділити на дві групи:

- 1) пошук за ключовими словами (термінами);
- 2) кластерні методи.

Кластерні методи засновані на наступному спостереженні: близькі за значенням функції відповідності документи звичайно вибираються однаковими пошуковими запитами. Таким чином, розбивши всі документи на групи або кластери і побудувавши в кожній групі характерного представника: центроїд кластера, можна порівнювати запит не з кожним документом окремо, а спочатку тільки з центроїдами. Якщо центроїд релевантний запиту, то потрібно продовжувати пошук всередині кластера, ні — перейти до розгляду іншого кластера.

Пошук за ключовими словами (термінами) є досить ефективним і може бути легко реалізованим, тому й одержав найбільше поширення. Для пошуку за ключовими словами використовують, в основному, індекси наступних двох типів:

- 1) інвертовані файли (ІФ) [38–41]
- 2) сигнатурні файли (СФ) [39–41].

ІФ — це множина пар: ключове слово, адреса входження ключового слова в документ. Основним недоліком ІФ є його великий обсяг.

На відміну від ІФ розмір СФ складає близько 50% від розміру вихідних даних.

Пошукові запити можуть бути контекстно-залежні й контекстно-незалежні. У випадку контекстно-незалежних запитів передбачається, що документ складається з деяких неподільних компонентів: у більшості випадків, слів. Контекстно-залежні запити можуть бути дуже різноманітні, починаючи з пошуку підстроки в документі, і закінчуючи умовою на наявність визначених ключових слів в одному реченні, абзаці й т.д.

Пошук на точну відповідність не дозволяє знайти слово, якщо в документі воно зустрічається в іншій граматичній формі, тому більшість інформаційно-пошукових систем здійснює пошук з урахуванням змінюваності слова.

В електронних документах бувають як природні орфографічні помилки, так і помилки введення, коли користувач набирає терміни запиту неправильно, тому ППС повинна «вміти» знаходити досить «схожі» слова. Ключовим моментом пошуку за подібністю є вибір міри ступеня «подібності». У цій якості рекомендується використовувати метрику (функцію) Левенштайна, що також часто називається відстанню редагування. Відстань Левенштайна між словами u і v дорівнює мінімальній кількості операцій редагування, необхідних для перетворення u до v .

Вибір як міру близькості метрики Левенштайна обумовлений двома факторами. По-перше, відстань Левенштайна формалізує інтуїтивне поняття про «помилку», а по-друге, існує безліч алгоритмів ефективного його обчислення [42–46]. Фактично, при пошуку потрібно не стільки значення відстані між рядками u і v , скільки знання чи перевищує $L(u, v)$ якість наперед задане граничне значення.

В даний час для пошуку терміна в документі застосовують різні методи. *Метод повного перебору* здійснює послідовне зчитування даних з диска і порівняння з термінами запиту. Основними перевагами цього методу є простота реалізації і досить висока швидкість роботи. При послідовному зчитуванні великих файлів (за умови невисокої фрагментації на диску) досягається пікова швидкість читання. Крім того, цей метод дозволяє реалізувати багатофункціональний пошук, наприклад, за підстрокою або регулярним виразом, без яких-небудь істотних обмежень.

Суть *методу розширення вибірки* полягає в побудові множини всіляких «помилкових» слів, наприклад, що виходять з вихідного в результаті однієї операції редагування, після чого побудовані терміни шукаються в документі (на точну відповідність).

Ідея *методу n-грам* полягає в наступному: якщо рядок v «схожий» на рядок u , то в них повинні бути які-небудь спільні підстроки. Тому буває доцільно будувати інвертований файл, у якому роль документів грають самі терміни, а роль термінів — підстроки довжини n , названі також n -грамами. Основний недолік методу n -грам — великий розмір файлу.

У методі *trie-дерев* усі рядки, що мають спільний початок, розташовуються в одному піддереві. Кожне ребро позначене деяким рядком. Термінальним вершинам («листам») відповідають слова списку.

Звичайно *trie-дерев* використовуються для пошуку за підстроками, але їх можна використовувати, і досить ефективно, для пошуку за подібністю [47]. Щоб забезпечити пошук за підстрокою, необхідно зберігати в *trie-дереві* всі суфікси (або префікси термінів). При цьому, як і у випадку методу n -грам розмір індексу в 2–10 разів перевищує розмір документа.

Триангуляційні дерева [46] дозволяють індексувати множини довільної структури, за умови, що на них задана метрика. Існує досить багато різних модифікацій цього методу, але

всі вони не занадто ефективні у випадку текстового пошуку і частіше використовуються для організації пошуку в базі даних зображень або інших складних об'єктів.

Однак найбільш оптимальним методом пошуку є хеширування за сигнатурою (ХС) [48]. Ідея методу полягає в наступному.

Нехай заданий непорожній алфавіт A і відомі ймовірності появи різних символів алфавіту. Нехай також на множині символів A задана функція $f(\alpha)$, що відображає букви в числа від 1 до m . Ця функція задає розбиття алфавіту на m підмножин.

Визначення: Сигнатурою $sign(w)$ слова w називається вектор розмірності m , k -ий елемент якого дорівнює одиниці, якщо в слові w є символ α такий, що $f(\alpha) = k$, і нулю в протилежному випадку. Номером сигнатури слова називається число $H(w) = \sum_{i=0}^{m-1} 2^i sign(w)_{i+1}$.

$H(w)$ є хеш-функцією, що відображає множину слів у відрізок цілих чисел від 0 до $2^m - 1$, що дозволяє організувати документ у вигляді хеш-таблиці.

Нехай слово \tilde{u} отримане з u в результаті однієї операції редагування: заміни, додавання (видалення) букви або перестановки символів. У силу визначення сигнатури, бітові вектори $sign(\tilde{u})$ і $sign(u)$ відрізняються не більш ніж в одному розряді у випадку операції додавання (видалення), і не більш ніж в двох розрядах — у випадку заміни. Якщо при заміні символів змінюються два елементи сигнатури, то загальна кількість одиниць (модуль сигнатури) залишається незмінним, а довільні перестановки, взагалі, не впливають на сигнатуру. Ця властивість дозволяє використовувати сигнатурний хеш для пошуку за подібністю — на відповідність цілому рядкові, а також за підстрокою і найпростішим регулярним виразом.

Метод ХС можна розглядати, як метод, зворотній методу розширення вибірки: спочатку визначаються хеш-списки документа, що містять «потенційні» терміни, а потім відбувається читання цих списків і безпосередня перевірка на предмет відповідності запиту. За рахунок того, що значна кількість «схожих» слів (з однаковими сигнатурами) розташовуються фізично (і логічно) у суміжних сторінках пам'яті, вдається уникнути багаторазових і неефективних читань з великою кількістю позиціонування.

Метод хеширування за сигнатурою має наступні переваги:

- 1) дозволяє здійснювати з високою швидкістю пошук на точну рівність й пошук, що допускає одну – дві «помилки» у завданні пошукового запиту;
- 2) ХС ефективно, як у випадку «прямих» читань з диска, так і з кеш-пам'яті;
- 3) ХС використовує компактний індекс;
- 4) відрізняється простотою реалізації.

ХС властивий і один досить істотний недолік: воно повільно працює, якщо індекс фрагментований: тобто в тому випадку, якщо списки слів з однаковими сигнатурами розкидані по несуміжних секторах на диску. В даний час це не є великою проблемою, тому що, з одного боку, розміри пам'яті ПК часто дозволяють завантажити документ повністю, а з іншого боку, дефрагментація документа, як правило, здійснюється протягом декількох хвилин.

Таким чином, враховуючи функції й критерії, яким має задовольняти АІС „Населення”, можна процес створення автоматизованої інформаційної системи умовно розділити на наступні **етапи**:

I. Розробка програмного забезпечення (ПЗ).

1. Розробка ПЗ для першого рівня:

- програми розпізнавання й пошуку ідентичних записів, внесених на російській і українській мовах;
- програми групування записів у базі даних;
- програми пошуку необхідних записів за заданим запитом;
- програми вибірки необхідних записів за заданим запитом;
- програми формування загальної бази даних;

– програми формування баз даних з необхідною інформацією із загальної бази даних для обчислювальних центрів обласних державних адміністрацій.

2. Розробка ПЗ для другого рівня:

– програми формування баз даних підсистем з необхідними відомостями із загальної бази даних „Населення України”;

– програми доповнення бази даних «Населення» інформацією з міськ-, райвиконкомів, РАГС.

3. Розробка ПЗ для третього рівня:

– програми доповнення бази даних «Населення» інформацією з військових частин;

– програми доповнення бази даних «Населення» інформацією з пенсійних відділів УМВС, УСБУ, міськ-, райвійськкоматів;

4. Розробка програмного забезпечення для створення сайту.

5. Налагодження ПЗ для захисту інформації, побудованого на основі криптоалгоритму „Wolkey” [49], розробленого співробітниками кафедри ІТ ХНТУ, для використання в АІС „Населення”.

На всіх рівнях необхідно керуватися критерієм цінності програм:

– можливість роботи програм під управлінням операційних систем фірми Microsoft (Windows 98, NT, 2000 Professional, 2000 Server, 2003 Server);

– прийнятна трудомісткість налаштування;

– зручний інтерфейс;

– можливість друкування різноманітних звітів, довідок, документів.

II. Роботи на організаційному рівні:

1. Створення технологічної документації й інструктивних матеріалів для супроводу ПЗ і технічного забезпечення.

2. Навчання персоналу — користувачів АІС „Населення” всіх рівнів.

Висновки. Авторами визначені основні проблеми й етапи створення АІС «Населення», особливості організації даних у системі, критерії, яким має задовольняти автоматизована система; пропонується використання інтеграційного підходу для створення автоматизованої системи та багатокритеріального підходу для проектування бази даних, проведений аналіз інформаційного пошуку в системі, що забезпечать оптимальне функціонування автоматизованої системи та запропоноване хеширування за сигнатурою в якості метода повнотекстного пошуку.

Література

1. С.А. Карпенко Региональная система управления социально-экономическим развитием как объект изучения // Ученые записки Таврического национального университета им. В.И. Вернадского Серия „География” — Симферополь: ТНУ, 2001. — Том 14. — №1. — С. 63–69.
2. Ю.М. Бардачов, В.Є. Ходаков, А.В. Шеховцов, Р.В. Бараненко Аналіз створення та підходи реалізації автоматизованої системи „Реєстр Виборців України” // Вестник Херсонского национального технического университета. — 2005. — №1(21). — С. 153–165.
3. Ходаков В.Е., Бардачев Ю.Н., Мартынов А.Н. Автоматизация и компьютеризация информационных процессов учреждений: учебное пособие. — Киев, Херсон, 1997. — 264 с.
4. П.М. Таланчук, Ю.О. Скрипник, В.О. Дубровний. Засоби вимірювання в автоматичних інформаційних та керуючих системах: Підруч. для студентів вузів. — К.: Райдуга, 1994 — 672 с.: іл.
5. Рогольский Ф.Б. Методологические аспекты создания систем управления открытыми технологическими комплексами в регионе // Математические модели в образовании,

- науке и промышленности: Сб. науч. трудов. — С.-Пб.: Санкт-Петербургское отделение МАН ВШ, 2003. — С. 183–188.
6. В.В. Сальдо, А.В. Шеховцов Стратегия управления городом — создание и функционирование СУБД города // Вестник Херсонского государственного технического университета. — 2003. — №2 (18). — С. 179–183.
 7. Тимчук Н.Ф. Автоматизация планирования комплексного развития регионов. — К.: Техніка, 1986.
 8. Дружинин В.В., Конторов Д.С. Системотехника. — М.: Радио и связь, 1985. — 200 с.
 9. Николаев В.И., Брук В.М. Системотехника: методы и приложения. — Л.: Машиностроение, 1985. — 199 с.
 10. Месарович М., Мако Д., Такахара И. Теория иерархических многоуровневых систем. — М.: МИР, 1973. — 344 с.
 11. Берталанфи Л. Общая теория систем — обзор проблем и результатов. — В кн.: Системные исследования; Ежегодник. — М.: Наука, 1969.
 12. Молчанов А.А. Моделирование и проектирование сложных систем. — К.: Вища школа. 1988. — 359 с.
 13. Евсеев В.В., Барский В.И., Калита Н.И. Многокритериальная оптимизация в задачах проектирования распределенных систем обработки информации // Прикладная информатика автоматизированных систем проектирования, управления, программированной эксплуатации. Тезисы докл. XII объединенного семинара. — Калининград: КТИРПХ, 1987. — С. 87–88.
 14. Цегелик Г.Г. Системы распределенных баз данных. — Львов: Свит, 1990. — 168 с.
 15. Лаздынь С.В., Телятников А.О. Оптимизация распределенных баз данных использованием генетических алгоритмов // Вестник Херсонского государственного технического университета. — 2004. — №1 (19). — С. 236–239.
 16. Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979.
 17. Кнут Д. Искусство программирования для ЭВМ: Т.3: Сортировка и поиск. — М.: Мир, 1978.
 18. Ли Д., Препарата Ф. Вычислительная геометрия. Обзор // Кибернетический сб. — 1987. — Вып. 24. — С. 5–96.
 19. Ньюмен У. М., Спруэлл Р. Ф. Основы интерактивной машинной графики. — М.: Мир, 1976.
 20. Солтон Дж. Динамические библиотечно-информационные системы. — М.: Мир, 1979.
 21. Chazelle B. M. Filtering search: a new approach to query-answering // Proc. 24th IEEE Annu. Symp. Found. Comput. Sci. — Nov. 1983. — P. 122–132.
 22. Edelsbrunner H., Overmars M. H., Siedel R. Some methods of computational geometry applied to computer graphics // IIG, Technische Univ. Graz, Austria, Tech. Rep. F117. — June 1983.
 23. Lee D. T., Wong C. K. Quintary trees: A file structures for multidimensional database system // ACM Trans. Database Syst. — Sept. 1980. — V. 1, №1. — P. 339–353.
 24. Гасанов Э.Э. Некоторые оценки сложности поиска информации // Физическое и математическое моделирование дискретных систем. Межвузовский сборник трудов №56. — М.: Изд-во Моск. энерг. ин-та, 1985. — С. 43–47.
 25. Гасанов Э.Э. О виде оптимальных информационных сетей для отношений линейного квазиупорядка. Препринт Р-5-303 ИЯФ АН УзССР. — Ташкент, 1987.
 26. Ben-Or M. Lower bounds for algebraic computation trees // Proc. 15th ACM Annu. Symp. Theory Comput. — Apr. 1983. — P. 80–86.
 27. Dobkin D.P., Lipton R.J. On the complexity of computations under varying sets of primitives // J. Comput. Syst. Sci. — 1979. — V. 18. — P. 86–91.

28. Steele J.M., Yao A.C. Lower bounds for algebraic decision trees // *J. Algorith.* — 1982.
29. Dobkin D.P. A nonlinear lower bound on search tree programs for solving knapsack problems // *J. Comput. Syst. Sci.* — 1976. — V. 13. — P. 69–73.
30. Dobkin D.P., Lipton R.J. A lower bound of $1/2n^2$ on linear search programs for the knapsack problem // *J. Comput. Sci.* — 1978. — V. 16. — P. 413–417.
31. Yao A.C., Rivest R.L. On the polyhedral decision problem // *SIAM J. Comput.* — 1980.
32. Петров Э.Г., Аннамухамедов О.Б., Евсеев В.В. и др. Синтез информационно-вычислительного обеспечения распределенных АСПИ. Ч.1/ Под ред. Э.Г. Петрова. — Ашхабад: Ылым, 1988. — 200 с.
33. Кулагин О.А. Многокритериальная модель проектирования схемы размещения данных в информационной сети // *Приборостроение*, 1997. — № 9. — С. 12–15.
34. Жадан А.П., Евсеев В.В., Барский В.И. Размещение информационных фондов в сетях ЭВМ. — *Техника средств связи. Серия СС. Вып. 1.* — 1986. — С. 38–40.
35. В.В. Евсеев, О.С. Богуш Многокритериальный подход к проектированию распределенной базы данных // *Вестник Херсонского национального технического университета.* — 2005. — №1 (21). — С. 317–320.
36. Овезгельдиев О.А., Петров Э.Г., Петров К.Э. Синтез и идентификация моделей многофакторного оценивания и оптимизации. — К.: Наукова думка, 2002. — 161 с.
37. Евсеев В.В. Методи автоматизованого проектування комплексу технічних засобів автоматизованих систем. Навч. посібник. — К.: ІЗМН, 1996. — 116 с.
38. Э. Озкарахан Машины баз данных и управление базами данных. — М.: Мир, 1989.
39. Christos Faloutsos and Douglas Oard *Survey of Information Retrieval and Filtering Methods* // University of Maryland.
40. J. Zobel and A. Moffat and K. Ramamohanarao Inverted files versus signature files for text indexing // Collaborative Information Technology Research Institute. Departments of Computer Science. RMIT and The University of Melbourne. — Australia. — feb 1995. — Technical report No TR-95-5.
41. C.J. van Rijsbergen *Information Retrieval* // London: Butterworths, 1979.
42. A. Ehrenfeucht, D. Haussler. A New Distance Metric on Strings Computable in Linear Time. // *Discrete Applied Mathematics.* — 1988. — №20. — pp. 191–203.
43. U. Masek, M.S. Peterson A faster algorithm for computing string-edit distances // *Journal of Computer and System Sciences.* — 1980. — №20(1). — pp. 785–807.
44. P.H. Sellers, *The Theory of Computation of Evolutionary Distances: Pattern recognition.* // *Journal of Algorithms.* — 1980. — №1. — pp. 359–373.
45. E. Ukkonen, *Approximate String Matching over Suffix-Trees.* // In *Proceedings of the Fourth Annual Symposium on Combinatorial Pattern Matching.* — Padova, Italy. — June, 1993. — pp. 229–242.
46. R.A. Wagner and M.J. Fisher *The String to String Correction Problem.* // *Journal of the ACM.* — 1974. — №21(1). — pp. 168–173.
47. H. Shang & T. H. Merrett *Tries for Approximate String Matching* // *IEEE Trans. on Knowledge and Data Engineering*, special issue on Digital Libraries, Nabil R. Adam, ed. — August, 1996. — p. 540.
48. Бойцов Л.М., Использование хеширования по сигнатуре для поиска по сходству. // *Прикладная математика и информатика.* — М.: Изд-во факультета ВМиК, МГУ. — 2000. — № 7.
49. Свідоцтво про реєстрацію авторського права на твір №12097, “Комп’ютерна програма криптографічного перетворення інформації “Wonderful key “Wolkey”. Автори: Граб М.В., Пилипенко М.В., Бараненко Р.В., Цивільський Ф.М., Шаганян С.М., Lunegov Maksim. Опубл. 24.01.2005.
50. На порозі нових виборів. // *Урядовий кур’єр.* — 26.11.2005. — №226. — С. 6.