МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

Учреждения образования «БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ

ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ»

Факультет Информационных технологии

Кафедра Информационных систем и технологий

Специальность 1-40 05 01 Информационные системы и технологии

**РЕФЕРАТ**

на тему:

«Bogosort»

Выполнил студент Iкурса 3 группы специальности ИСиТ Задорожный Н.П

(Ф.И.О)

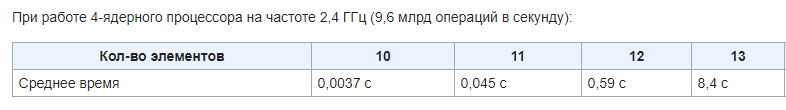
Руководитель

(учен. степень, звание, должность, подпись, Ф.И.О.)

Минск 2019

Bogosort является очень неэффективным алгоритмом сортировки. Её используют только в образовательных целях, противопоставляя другим, более реалистичным алгоритмам. Eё суть состоит в том, что в случае если массив не отсортирован перемещать его и проверить является ли он отсортированным, и так до тех пор, пока массив не будет отсортирован.

Нотация Big O для Bogosort’а O(n\*n!), что говорит о том, что данная сортировка крайне неэффективна.



Массив из 32 элементов будет сортироваться больше чем лет

Функция, которая проверяет является ли массив отсортированным:

int correct(int \*arr, int size) { // Функция, проверяющая в правильном ли порядке расположены элементы в массиве

while (size-- > 0) // Пока не пройдет все элементы массива

if (arr[size - 1] > arr[size])

return 0;

return 1;

}

Функция, реализующая перемещевание массива в случае, когда массив не отсортирован:

void shuffle(int \*arr, int size) { // Функция перемещки массива

for (int i = 0; i < size; ++i)

swap(arr[i], arr[(rand() % size)]); // Перемещка в случайном порядке

}

Функция, bogoSort, которая реализует сам этот алгоритм.

void bogoSort(int \*arr, int size) { // Главная функция, реализующая саму сортировку

while (!correct(arr, size)) // в случае некорректности перемешать

shuffle(arr, size); // Вызов функции перемещивания

}

Функция main, отвечающая за заполнение массива и для передачи его на сортировку.

void main()

{

int arr[10]; // Создадим массив из 10 целочисленных элементов

for (int i = 0; i < 10; i++) { // Заполнение массива случайными числами

arr[i] = rand() % 20;

cout << arr[i] << ' '; // Вывод элементов до сортировки

}

bogoSort(arr, 10); // Вызов функции сортировки

cout << endl;

for (int i = 0; i < 10; i++) { // Вывод отсортированного массива

cout << arr[i] << ' ';

}

system("pause");

}

Результат работы этой сортировки:

