МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ

Федеральное государственное бюджетное

образовательное учреждение

высшего образования

«ВЯТСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ФАКУЛЬТЕТ КОМПЬЮТЕРНЫХ И ФИЗИКО-МАТЕМАТИЧЕСКИХ НАУК

КАФЕДРА ПРИКЛАДНОЙ МАТЕМАТИКИ И ИНФОРМАТИКИ

Отчёт по лабораторной работе №9.2

по дисциплине «Теория алгоритмов»

Анализ рекурсивных алгоритмов. Сортировка слиянием, сортировка Хоара, бинарный поиск

Выполнил: студент группы ФИб-4302-51-00 \_\_\_\_\_\_\_\_\_\_\_ / Д.А.Савин/

Проверил: доцент каф. ПМиИ \_\_\_\_\_\_\_\_\_\_\_ /Е.В.Разова/

Киров 2021

Задание 1

Сортировка слиянием:

|  |
| --- |
| **static** **int**[] **MergeSort**(**int**[] arr, **int** l, **int** r)  {  **if** (l < r)  {  **int** m = (l + r) / **2**;  MergeSort(arr, l, m);  MergeSort(arr, m + **1**, r);  MergeMass(arr, l, m, r);  }  **return** arr;  } |

Начало рекурсии – получение середины за

Разбиение на 2 равные части:

Выход из рекурсии: слияние 2 массивов за

Сортировка Хоара:

|  |
| --- |
| **static** **int**[] **QuickSort**(**int**[] arr, **int** minInd, **int** maxInd)  {  **if** (minInd >= maxInd)  {  **return** arr;  }  **int** Index = Partition(arr, minInd, maxInd);  QuickSort(arr, minInd, Index - **1**);  QuickSort(arr, Index + **1**, maxInd);  **return** arr;  } |

Начало рекурсии: получение опорного индекса и разделение массива за ;

Разбиение на 2 части:

В лучшем случае:

В худшем случае:

В лучшем случае: ;

Временная оценка: ;

В худшем случае: ;

По времени работы сортировка слиянием эффективнее, так как в худшем случае будет быстрее, чем быстрая сортировка в худшем случае и одинаково эффективна в лучшем случае.

Задание 2

Бинарный поиск:

|  |
| --- |
| **static** **int** **BinarySearch**(**int**[] arr, **int** **value**, **int** first, **int** last)  {  **if** (first > last)  **return** -**1**;  **int** m = (first + last) / **2**;  **int** midValue = arr[m];  **if** (midValue == **value**)  **return** m;  **else**  {  **if** (midValue > **value**)  {  **return** **BinarySearch**(arr, **value**, first, m - **1**);  }  **else**  {  **return** **BinarySearch**(arr, **value**, m + **1**, last);  }  }  } |

Начало рекурсии: получение среднего значения массива ;

Разбиение на 2 части и продолжение в нужной ;

Пока не найден элемент, если элемент найден продолжения не происходит.

В лучшем случае – не происходит рекурсии, выходит сразу

В худшем случае

Задание 3

Бинарный поиск:

|  |
| --- |
| Procedure **Soch** (i : Integer);  Var k : Integer;  Begin  If i>n Then Print(a)  Else For k:=**1** To n Do  Begin  a[i]:=k;  Soch(i+**1**);  End;  End; |

Начало рекурсии: получение условие и вывод элементов массива ;

Вызов n рекурсий: ;

1.

;

;

предположим, при

– верно при

Предположим, что для всех неравенство выполняется (индуктивное предположение).

;

2.

;