МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
Київський національний університет імені Тараса Шевченка
Кафедра програмних систем і технологій

Звіт з лабораторної роботи 2.4

Тема: «Сортування послідовностей»

Виконав: студент групи ІПЗ - 12
Богатько Олександр
Геннадійович
Перевірила: викладач
Юрчук Ірина Аркадіївна

Київ 2020

1.Алгоритм природного злиття

Код програми:

```cpp
#include<iostream>
#include <vector>
using namespace std;

void Print(vector<int>& arr) {
    for(int i : arr)
        cout<<i<<" ";
}
void Merge(vector<int>& b, vector<int>& c, vector<int>& arr, int& check)
{
    int i = 0, j = 0, k = 0;
    check++;
    while (i != b.size() && j != c.size()) {
        check++;
        if (b[i] < c[j]) arr[k] = b[i++]; else arr[k] = c[j++];
        k++;
    }
    check++;
    if (i != b.size())
        while (i < b.size()) {
            check++;
            arr[k++] = b[i++]; }
    else
        while (j < c.size()) {
            check++;
            arr[k++] = c[j++]; }
    cout << "After merging: "; Print(arr);
    cout << endl;
}
void MergeSort(vector<int>& arr, int& check)
{
    vector<int> b, c;
    int tmp = -100;
    do
    {
        int count = 0;
        for (int & i : arr) {
            if (tmp > i) count++;
            if (count % 2 == 0) b.push_back(i);
            else if (count % 2 == 1) c.push_back(i);
            tmp = i; }
        if (c.empty() || b.empty())
            return;
            cout <<"First stripe: ";
        Print(b);
        cout << endl;
        cout << "Second stripe: ";
        Print(c);
        cout << endl;
        Merge(b, c, arr, check);
        b.clear();
        c.clear();
    } while (true);
}
int main() {
    int size, r, mode, check = 0;
    cout << "Enter the number of elements: ";
    cin >> size;
    vector<int> arr;
    cout << "1. Random fill\n 2. Manual fill\nChoose mode: ";
    cin >> mode; switch(mode) {
        case 1: for(int i = 0; i < size; ++i)
                arr.push_back(rand()%100); cout << "Generated array: ";
            Print(arr);
            cout << endl; break;
        case 2: cout << "Enter array: ";
            for(int i = 0; i < size; ++i)
            {
                cin >> r;
                arr.push_back(r); }

            break; default: break;
    }
    cout << "Array before Sorting: "; Print(arr);
    cout << endl; MergeSort(arr,check);
    cout << "Array after Sorting: "; Print(arr);
    cout << endl;
    cout << "Comparisons: " << check;
    return 0; }
```

```
Enter the number of elements: 10
1. Random fill
 2. Manual fill
Choose mode: 1
Generatedarray: 7 49 73 58 30 72 44 78 23 9
Array before Sorting: 7 49 73 58 30 72 44 78 23 9
First stripe: 7 49 73 30 72 23
Second stripe: 58 44 78 9
After merging: 7 49 58 44 73 30 72 23 78 9
First stripe: 44 73 23 78
Second stripe: 7 49 58 30 72 9
After merging: 7 44 49 58 30 72 9 73 23 78
First stripe: 30 72 23 78
Second stripe: 7 44 49 58 9 73
After merging: 7 30 44 49 58 9 72 23 73 78
First stripe: 9 72
Second stripe: 7 30 44 49 58 23 73 78
After merging: 7 9 30 44 49 58 23 72 73 78
First stripe: 23 72 73 78
Second stripe: 7 9 30 44 49 58
After merging: 7 9 23 30 44 49 58 72 73 78
Array after Sorting: 7 9 23 30 44 49 58 72 73 78
Comparisons: 60
Process finished with exit code 0
```

```
Enter the number of elements: 10
1. Random fill
 2. Manual fill
Choose mode: 2
Enter array: 91
32
21
44
53
62
77
89
5
11
Array before Sorting: 91 32 21 44 53 62 77 89 5 11
First stripe: 91 21 44 53 62 77 89
Second stripe: 32 5 11
After merging: 32 5 11 91 21 44 53 62 77 89
First stripe: 32 21 44 53 62 77 89
Second stripe: 5 11 91
After merging: 5 11 32 21 44 53 62 77 89 91
First stripe: 21 44 53 62 77 89 91
Second stripe: 5 11 32
After merging: 5 11 21 32 44 53 62 77 89 91
Array after Sorting: 5 11 21 32 44 53 62 77 89 91
Comparisons: 36
Process finished with exit code 0
```

2.Алгоритм багатошляхового злиття.

Код программи:

```cpp
#include<iostream>
#include <vector>
using namespace std;

void Print(vector<int>& arr) {
    for(int i : arr)
        cout << i <<" ";
}
void Merge(vector<int>& b, vector<int>& c, vector<int>& d, vector<int>& arr, int& check)
{
    vector<int> buf;
    int i = 0, j = 0, k = 0;
    check++;
    while (i != b.size() && j != c.size()) {
        check++;
        if (b[i] < c[j])
            buf.push_back(b[i++]);
        else buf.push_back(c[j++]);
        k++;
    }
    if (i != b.size())
        while (i < b.size())
        {
            check++;
            buf.push_back(b[i++]);
        }
    else
        while (j < c.size()) {
            check++;
            buf.push_back(c[j++]); }
    cout << "Merging first two stripes: ";
    Print(buf);
    cout << endl;
    i = 0, j = 0, k = 0;
    check++;
    while (i != buf.size() && j != d.size()) {
        check++;
        if (buf[i] < d[j])
            arr[k] = buf[i++];
        else arr[k] = d[j++];
        k++;
    }
    if (i != buf.size())
        while (i < buf.size()) {
            check++;
            arr[k++] = buf[i++]; }
    else
        while (j < d.size()) {
            check++;
            arr[k++] = d[j++]; }
    cout << "After merging: ";
        Print(arr);
    cout << endl;
}
void MergeSort(vector<int>& arr, int& check) {
    vector<int> b, c, d;
    int tmp = -100;
    do
    {
        int count = 0;
        for (int & i : arr) {
            if (tmp > i) count++;
            if (count % 3 == 0) b.push_back(i);
            else if (count % 3 == 1) c.push_back(i);
            else if (count % 3 == 2) d.push_back(i);
            tmp = i;
        }
        if (c.empty() && b.empty() || c.empty() && b.empty() ||
b.empty() && d.empty())
            return;
        cout <<"First stripe: ";
        Print(b);
        cout << endl;
        cout << "Second stripe: ";
        Print(c);
        cout << endl;
        cout << "Third stripe: "; Print(d);
        cout << endl;
        Merge(b, c, d, arr, check);
        b.clear();
        c.clear();
        d.clear();
    } while (true); }
int main() {
    int size, r, mode, check = 0; cout << "Enter the number of elements: ";
    cin >> size;
    vector<int> arr;
    cout << "1. Random fill\n"
            "2. Manual fill\n"
            "Choose mode: ";
```

```
Enter the number of elements: 10
1. Random fill
2. Manual fill
Choose mode: 1
Generated array: 7 49 73 58 30 72 44 78 23 9
Array before Sorting: 7 49 73 58 30 72 44 78 23 9
First stripe: 7 49 73 44 78
Second stripe: 58 23
Third stripe: 30 72 9
Merging first two stripes: 7 49 58 23 73 44 78
After merging: 7 30 49 58 23 72 9 73 44 78
First stripe: 9 73
Second stripe: 7 30 49 58 44 78
Third stripe: 23 72
Merging first two stripes: 7 9 30 49 58 44 73 78
After merging: 7 9 23 30 49 58 44 72 73 78
First stripe:
Second stripe: 7 9 23 30 49 58
Third stripe: 44 72 73 78
Merging first two stripes: 7 9 23 30 49 58
After merging: 7 9 23 30 44 49 58 72 73 78
Array after Sorting: 7 9 23 30 44 49 58 72 73 78
Comparisons: 57
Process finished with exit code 0
```

```
Enter the number of elements: 10
1. Random fill
2. Manual fill
Choose mode: 2
Enter array: 21
13
52
46
85
97
59
74
33
66
Array before Sorting: 21 13 52 46 85 97 59 74 33 66
First stripe: 21 59 74
Second stripe: 13 52 33 66
Third stripe: 46 85 97
Merging first two stripes: 13 21 52 33 59 66 74
After merging: 13 21 46 52 33 59 66 74 85 97
First stripe:
Second stripe: 13 21 46 52
Third stripe: 33 59 66 74 85 97
Merging first two stripes: 13 21 46 52
After merging: 13 21 33 46 52 59 66 74 85 97
Array after Sorting: 13 21 33 46 52 59 66 74 85 97
Comparisons: 35
Process finished with exit code 0
```

```
    cin >> mode;
    switch(mode) {
        case 1: for(int i = 0;i< size; ++i) arr.push_back(rand()%100);
            cout << "Generated array: "; Print(arr);
            cout << endl;
            break;
        case 2: cout << "Enter array: ";
        for(int i = 0; i < size; ++i) {
        cin >> r;
        arr.push_back(r); }
            break; default: break;
    }
    cout << "Array before Sorting: ";
    Print(arr);
    cout << endl;
    MergeSort(arr, check);
    cout << "Array after Sorting: ";
    Print(arr);
    cout << endl;
    cout << "Comparisons: " << check;
    return 0; }
```