Министерство науки и образования Украины

Харьковский национальный университет радиоэлектроники

Кафедра Бикс

Отчет

По лабораторной работе № 6:

Тема: «Использование контейнерных классов и алгоритмов

стандартной библиотеки»

Выполнила: Принял:

Ст. гр БИКС-14-1 Олешко О.И

Д.Т.Ф. Тхао

Харьков 2016

#include <iostream>

#include <vector>

#include <time.h>

#include <Windows.h>

#include <algorithm>

#include <iomanip>

using namespace std;

#define len 100000

typedef struct {

LARGE\_INTEGER start;

LARGE\_INTEGER stop;

} stopWatch;

void startTimer(stopWatch \*timer) {

QueryPerformanceCounter(&timer->start);

}

void stopTimer(stopWatch \*timer) {

QueryPerformanceCounter(&timer->stop);

}

double LIToSecs(LARGE\_INTEGER \* L) {

LARGE\_INTEGER frequency;

QueryPerformanceFrequency(&frequency);

return ((double)L->QuadPart / (double)frequency.QuadPart);

}

double getElapsedTime(stopWatch \*timer) {

LARGE\_INTEGER time;

time.QuadPart = timer->stop.QuadPart - timer->start.QuadPart;

return LIToSecs(&time);

}

void Swap(int &a, int &b)

{

int t = a;

a = b;

b = t;

}

void SortdecA(int \*a, int size)

{

int min = 0;

for (int i = 1; i < size - 1; min = 0, i++) {

for (int j = 0; j < size - i; j++){

if (a[j] < a[min]) {

min = j;

}

}

if (min != (size - i)) Swap(a[min], a[size - i]);

}

}

void SortdecV(vector<int> v, int size)

{

int min = 0;

for (int i = 1; i < size - 1; min = 0, i++) {

for (int j = 0; j < size - i; j++){

if (v[j] < v[min]) {

min = j;

}

}

if (min != (size-i)) swap(v[min], v[size - i]);

}

}

bool mod2(int value){

return value % 2;

}

void GenArray(int a[len + 1]) {

for (int i = 0; i < len; i++) {

a[i] = rand();

} a[len] = len;

}

void GenVector(vector<int> &v) {

v.reserve(len);

for (int i = 0; i < len;i++) v.push\_back(rand());

}

double TimeArray(int a[len], int x) {

stopWatch timer;

startTimer(&timer);

switch (x) {

case 1: {

for (int i = 4; i < len; a[i] = 0, i += 5); break;

}

case 2: {

for (int i = len - 1; i >= 0;){

if (a[i] % 2) {

a[len]--;

for (int j = i; j < a[len]; j++) a[j] = a[j + 1];

}

else i--;

} break;

}

case 3: {

//SortdecA(a, 0, a[len], 0); break;

SortdecA(a, a[len]); break;

}

}

stopTimer(&timer);

return getElapsedTime(&timer)\*1000;

}

double TimeVector(vector<int> v, int x)

{

stopWatch timer;

startTimer(&timer);

switch (x) {

case 1: {

for (int i = 4; i < len; v.at(i) = 0, i += 5); break;

}

case 2: {

for (int i = 0; i < v.size();) {

if (v[i] % 2) v.erase(v.begin() + i);

else i++;

} break;

}

case 3: {

//SortdecV(v.begin(), 0, v.size(), 0);

SortdecV(v, v.size()); break;

}

}

stopTimer(&timer);

return getElapsedTime(&timer)\*1000;

}

double TimeVector\_alg(vector<int> v\_alg, int x)

{

stopWatch timer;

startTimer(&timer);

switch (x) {

case 1: {

for (int i = 4; i < len; v\_alg.at(i) = 0, i += 5); break;

}

case 2: {

v\_alg.erase(remove\_if(v\_alg.begin(), v\_alg.end(), mod2), v\_alg.end());

break;

}

case 3: {

sort(v\_alg.begin(), v\_alg.end());

for (int i = 0; i < (v\_alg.size() / 2); i++)

swap(v\_alg[i], v\_alg[v\_alg.size() - i - 1]);

break;

}

}

stopTimer(&timer);

return getElapsedTime(&timer)\*1000;

}

int main()

{

srand(time(0));

int a[len+1]; vector<int> v, v\_alg;

GenArray(a);

GenVector(v);

GenVector(v\_alg);

double ta, tv, tv\_alg;

cout << setw(15) << "Array" << setw(15) << "Vector" << setw(15) << "Vector\_alg" << endl;

for (int i = 1; i < 4; i++) {

ta = TimeArray(a, i);

tv = TimeVector(v, i);

tv\_alg = TimeVector\_alg(v\_alg, i);

cout << i << setw(15) << ta << setw(15) << tv << setw(15) << tv\_alg << endl;

}

system("pause");

return 0;

}

