Министерство Образования Республики Беларусь

Учреждение образования

«Гомельский государственный технический университет

имени П. О. Сухого»

Кафедра «Информатика»

**Лабораторная работа № 5**

по дисциплине: **«Знакомство со стандартной утилитой**

**Gnumаке для построения проектов в ОСUnix/Linux»**

**Инструментальные средства разработки Linux**

Выполнил студент

группы ИП-31

Васюков И. А.

Проверил преподаватель

Самовендюк Н.В.

**Цель:** Ознакомиться с техникой компиляции программ на языке программирования С (С++) в среде ОС семейства Unix/Linux, а также получить практические навыки использования утилиты GNU make для сборки проекта.

**Ход работы:**

1. Ознакомиться с теоретическим материалом.

2. Воспользоваться утилитой makeдля автоматизации сборки проектов из лабораторной работы №4 «Инструментальные средства разработки Linux»

**Листинг программы**

**MakeFile**

СС=g++

CFLAGS=-c

FinalName=lab4

IshKat=src

BinKat=bin

compile:

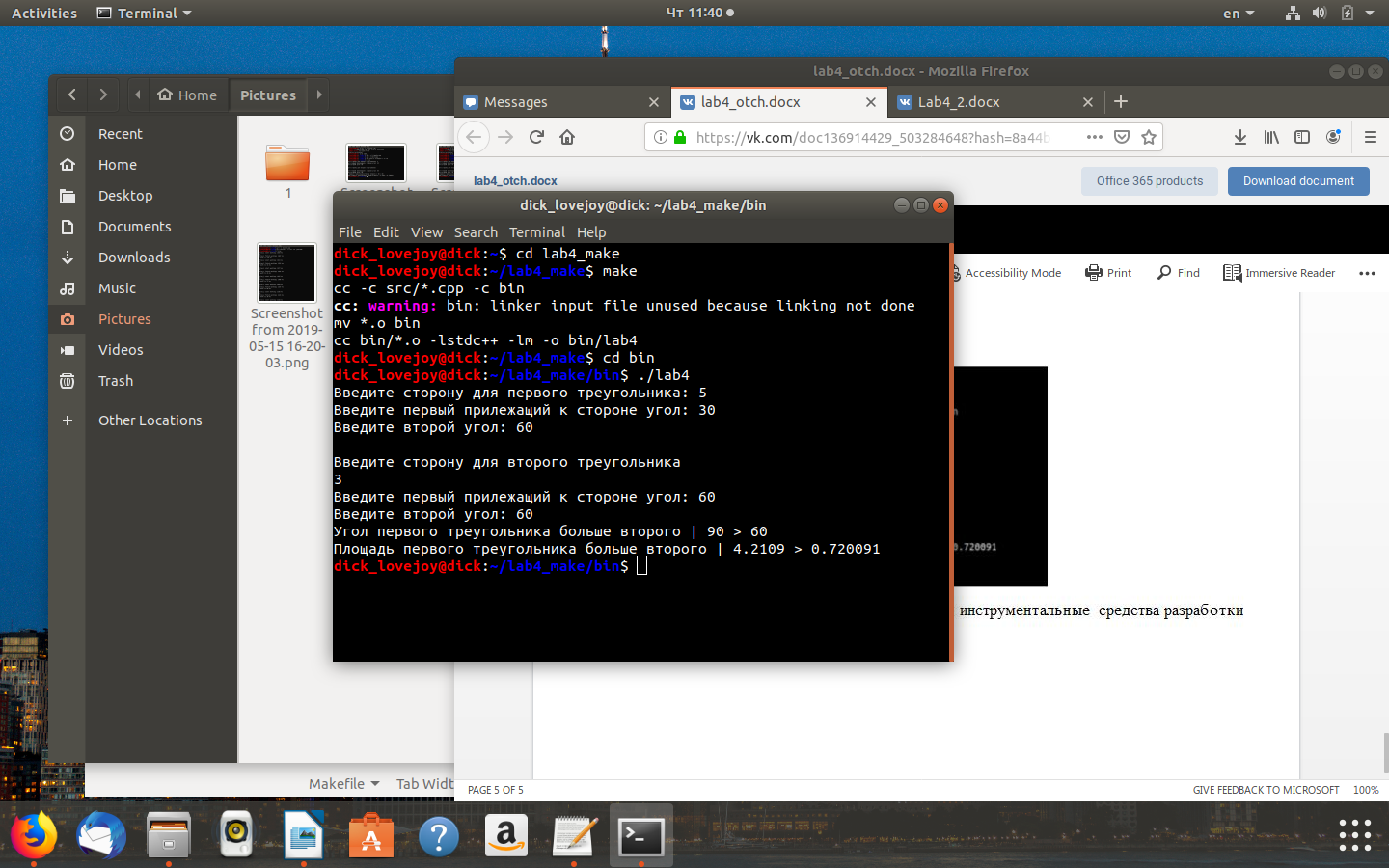
$(CC) $(IshKat)/\*.cpp -c $(BinKat)

mv \*.o $(BinKat)

$(CC) $(BinKat)/\*.o -lstdc++ -lm -o $(BinKat)/$(FinalName)

clean:

rm -rf $(BinKat)/\*.o

**Результат выполнения**

3. Создать make-файл с высоким уровнем автоматизированной обработки исходных файлов программы согласно следующим условиям:

• имя скомпилированной программы (выполняемый или бинарный файл), флаги компиляции и имена каталогов с исходными файлами и бинарными файлами (каталоги,src, bin и т. п.) задаются с помощью переменных в makefile;

• зависимости исходных файлов на языке С (С++) и цели в make-файле должны формироваться динамически;

• наличие цели clean, удаляющей временные файлы;

• каталог проекта должен быть структурирован следующим образом:

src — каталог с исходными файлами;

bin — каталог с бинарными файлами (скомпилированными);

makejile.

|  |  |  |  |
| --- | --- | --- | --- |
| Варианта | Структура данных (STL) | Функции обработки | Тип сортировки |
| 4 | List | Определение минимального элемента. | Вставками/ Выбором минимального |

Все функции размещаются в отдельных файлах.

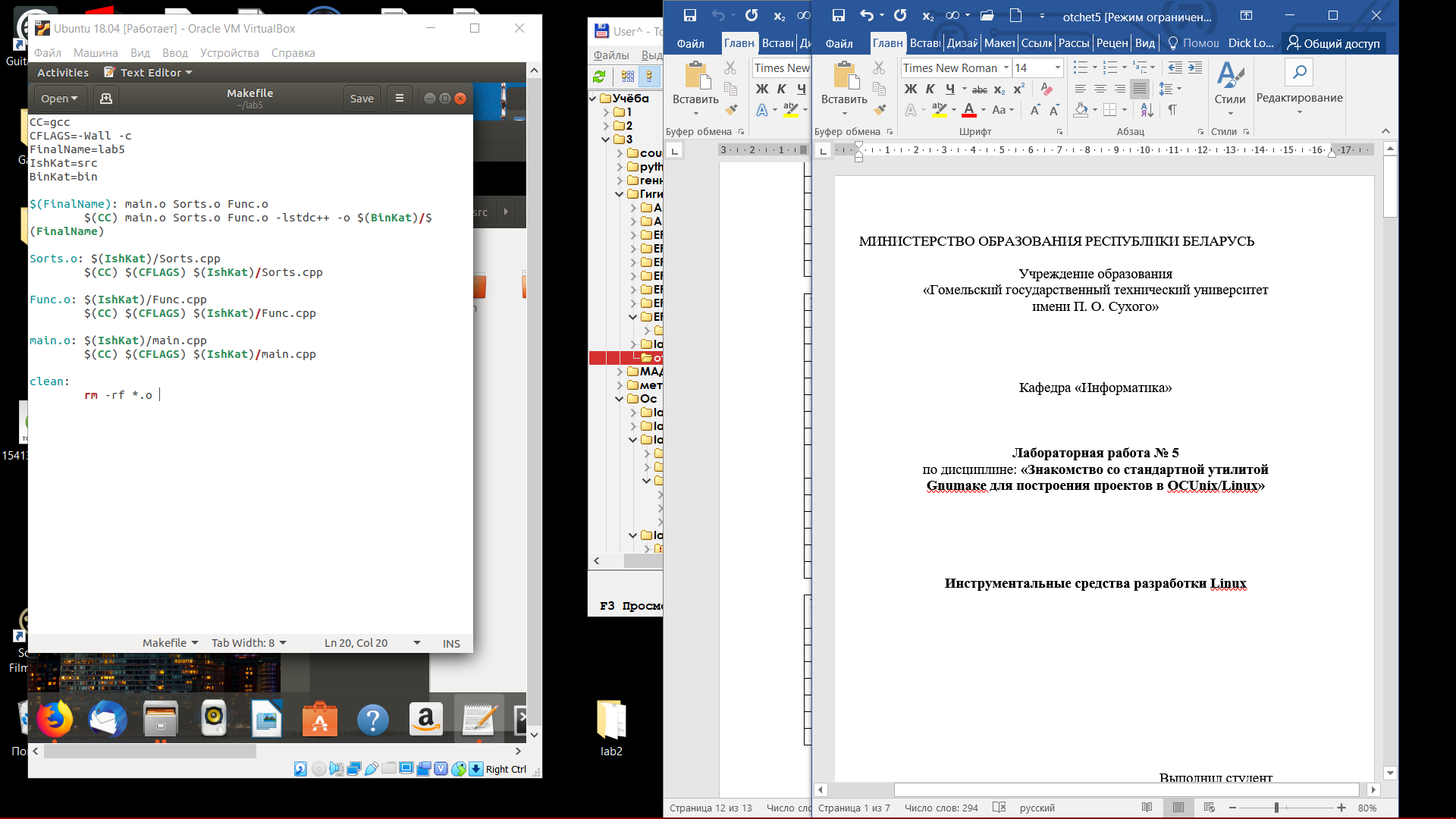
4. Выполнить программу (скомпилировать, при необходимости отладить) для первого варианта сортировки данных.

5. Изменить тип сортировки и выполнить программу.

5. Показать, что при изменении одного исходного файла и последующем вызове make будут исполнены только необходимые команды компиляции (неизмененные файлы перекомпилированы не будут) и изменены атрибуты и/или размер объектных файлов (файлы с расширением .о).

**Листинг программы**

**Makefile**



**Func.h**

#include <iostream>

#include <list>

using namespace std;

void Task(list<int> &list);

**Func.cpp**

#include "Func.h"

void Task(list<int> &list)

{

auto min = list.begin();

for ( auto i = list.begin(); i != list.end(); i++)

{

if(\*min > \*i)

min = i;

}

cout << "Minimal number:" << \*min;

}

**Sorts.h**

#pragma once

#include <list>

using namespace std;

void Sort1(list<int> &list);

void Sort2(list<int> &list);

**Sorts.cpp**

#include "Sorts.h"

void Sort1(list<int> &list)

{

for ( auto i = list.begin(); i != list.end();)

{

int value = \*i;

i = list.erase(i);

auto place = list.begin();

for( auto j = list.begin(); j != i; j++)

{

if(value > \*j)

{

place = j;

place++;

}

}

list.insert(place, value);

}

}

void Sort2(list<int> &list)

{

for (auto i = list.begin(); i != list.end(); i++)

{

auto min = i;

for(auto j = i; j != list.end(); j++)

{

if(\*min > \*j)

min = j;

}

int value = \*i;

\*i = \*min;

\*min = value;

}

}

**main.cpp**

#include "Func.h"

#include "Sorts.h"

int main()

{

int n;

setlocale(0,"");

cout << "Enter List size: ";

cin >> n;

list<int> list;

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

{

cout << "Enter" << i << " element: " << endl;

int element;

cin >> element;

list.push\_back(element);

}

cout << "List: ";

for ( auto i = list.begin(); i != list.end(); i++)

cout << \*i << " ";

cout << endl;

Task(list);

Sort1(list);

cout << endl << "Sorted list: ";

for ( auto i = list.begin(); i != list.end(); i++)

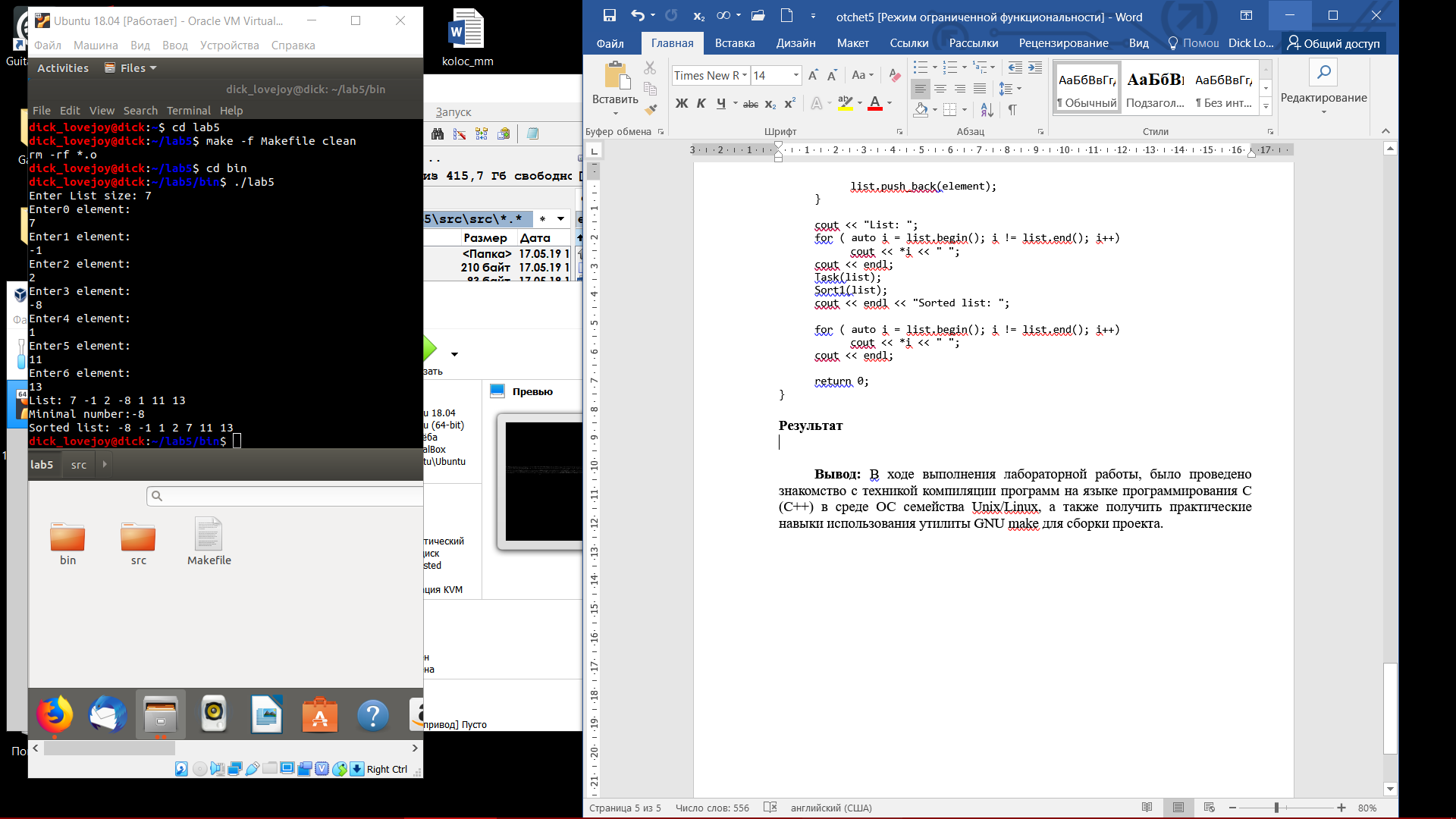
cout << \*i << " ";

cout << endl;

return 0;

}

**Результат**



**Вывод:** В ходе выполнения лабораторной работы, было проведено знакомство с техникой компиляции программ на языке программирования С (С++) в среде ОС семейства Unix/Linux, а также получить практические навыки исполssьзования утилиты GNU make для сборки проекта.