Лабораторная работа №4:

«Библиотеки».

Работу выполнила Кубрак Екатерина Алексеевна

Студент группы A-03-19

14.05.2020г.

1) Ввод из произвольного потока.

vector<double>input\_numbers(istream& in, size\_t count)

{

vector<double>result(count);

for(size\_t i=0; i<count; i++)

{

in >> result[i];

}

return result;

}

Вызов функции:

const auto numbers = input\_numbers(cin, number\_count);

2) Закоммитила изменения и отправила на GitHub.

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:переписала функцию input\_numbers()"

[lab04 a7bb9a6] code:переписала функцию input\_numbers()

6 files changed, 17 insertions(+), 3 deletions(-)

create mode 100644 bin/Debug/lab03-histogram.exe

rewrite obj/Debug/main.o (84%)

create mode 100644 "~$\320\261\320\276\321\200\320\260\321\202\320\276\321\200\320\275\320\260\321\217 \321\200\320\260\320\261\320\276\321\202\320\260 \342\204\2264.docx"

create mode 100644 "\320\233\320\260\320\261\320\276\321\200\320\260\321\202\320\276\321\200\320\275\320\260\321\217 \321\200\320\260\320\261\320\276\321\202\320\260 \342\204\2264.docx"

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 20, done.

Counting objects: 100% (20/20), done.

Delta compression using up to 8 threads

Compressing objects: 100% (10/10), done.

Writing objects: 100% (12/12), 649.68 KiB | 6.84 MiB/s, done.

Total 12 (delta 3), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (3/3), completed with 3 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

9974d18..a7bb9a6 lab04 -> lab04

3) Упорядочение передачи данных по программе.

1.

struct Input

{

vector<double> numbers;

size\_t bin\_count;

};

Input read\_input(istream& in)

{

Input data;

cerr << "Enter number count: ";

size\_t number\_count;

in >> number\_count;

cerr << "Enter numbers: ";

data.numbers = input\_numbers(in, number\_count);

size\_t bin\_count;

cerr << "Enter column count: ";

in >> data.bin\_count;

return data;

}

int main()

{

// Ввод данных

Input data = read\_input(cin);

// Обработка данных

double min, max;

find\_minmax(data.numbers, min, max);

const auto bins = make\_histogram(data.numbers, data.bin\_count, min, max);

// Вывод данных

show\_histogram\_svg(bins);

return 0;

}

Закоммитила:

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:добавила функцию read\_input()"

[lab04 5de91e7] code:добавила функцию read\_input()

3 files changed, 28 insertions(+), 13 deletions(-)

rewrite obj/Debug/main.o (85%)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 17, done.

Counting objects: 100% (17/17), done.

Delta compression using up to 8 threads

Compressing objects: 100% (7/7), done.

Writing objects: 100% (9/9), 303.03 KiB | 3.12 MiB/s, done.

Total 9 (delta 5), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (5/5), completed with 4 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

a7bb9a6..5de91e7 lab04 -> lab04

2.

vector<size\_t> make\_histogram(struct Input data, double min, double max)

{

vector<size\_t> bins(data.bin\_count);

for (double number : data.numbers)

{

size\_t bin = (size\_t)((number - min) / (max - min) \* data.bin\_count);

if (bin == data.bin\_count)

{

bin--;

}

bins[bin]++;

}

return (bins);

}

int main()

{

// Ввод данных

const auto input = read\_input(cin);

// Обработка данных

double min, max;

find\_minmax(input, min, max);

const auto bins = make\_histogram(input, min, max);

// Вывод данных

show\_histogram\_svg(bins);

return 0;

}

Закоммитила:

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:переписала функцию make\_histogram(), чтобы она принимала Input"

[lab04 51b4ba4] code:переписала функцию make\_histogram(), чтобы она принимала Input

7 files changed, 29 insertions(+), 22 deletions(-)

rewrite bin/Debug/lab03-histogram.exe (61%)

rewrite obj/Debug/histogram.o (94%)

rewrite obj/Debug/main.o (95%)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 25, done.

Counting objects: 100% (25/25), done.

Delta compression using up to 8 threads

Compressing objects: 100% (11/11), done.

Writing objects: 100% (13/13), 340.41 KiB | 1.70 MiB/s, done.

Total 13 (delta 8), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (8/8), completed with 6 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

5de91e7..51b4ba4 lab04 -> lab04

3.

Input read\_input(istream& in, bool prompt)

{

Input data;

if(prompt == true)

{

cerr << "Enter number count: ";

}

size\_t number\_count;

in >> number\_count;

if(prompt == true)

{

cerr << "Enter numbers: ";

}

data.numbers = input\_numbers(in, number\_count);

size\_t bin\_count;

if(prompt == true)

{

cerr << "Enter column count: ";

}

in >> data.bin\_count;

return data;

}

int main()

{

// Ввод данных

const auto input = read\_input(cin, true);

// Обработка данных

double min, max;

find\_minmax(input, min, max);

const auto bins = make\_histogram(input, min, max);

// Вывод данных

show\_histogram\_svg(bins);

return 0;

}

Закоммитила:

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:добавила переменную типа bool"

[lab04 bb0fa41] code:добавила переменную типа bool

3 files changed, 11 insertions(+), 3 deletions(-)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 17, done.

Counting objects: 100% (17/17), done.

Delta compression using up to 8 threads

Compressing objects: 100% (7/7), done.

Writing objects: 100% (9/9), 185.54 KiB | 3.04 MiB/s, done.

Total 9 (delta 5), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (5/5), completed with 5 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

51b4ba4..bb0fa41 lab04 -> lab04

4) Установка cURL.

Скачала архив, извлекла curl-7.70.0-win64-mingw, переместила в папку с проектом, переименовала и добавила в игнорируемые.

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git status

On branch lab04

Your branch is up to date with 'origin/lab04'.

Untracked files:

(use "git add <file>..." to include in what will be committed)

.gitignore

nothing added to commit but untracked files present (use "git add" to track)

5) Подключение cURL к проекту.

1. Вызвала функцию curl\_global\_init(CURL\_GLOBAL\_ALL), появились ошибки.

2. Добавила #include <curl/curl.h>, больше нет этих ошибок, но появилась другая.

3. В открывшемся диалоге Сompile указала curl/include и нажала OK, больше нет этой ошибки, но появилась другая.

4. Вернулась к Search directories, теперь на дочернюю вкладку Linker, и добавила каталог curl/lib. На вкладке Linker settings (соседняя с Search directories) нажала Add и ввела libcurl.dll.a, после чего сохранила настройки и покинула диалог нажатиями OK.

Теперь сборка программы прошла успешно.

6) Установка динамических библиотек.

При запуске lab04.exe ,будет ошибка (главное слово в ней libcurl-x64.dll). Чтобы её решить скопировала libcurl-x64.dll в bin/Debug. Программа запускается.

Теперь не найдены будут libcrypto-1\_1.dll и libssl-1\_1.dll. Скачала.

Программа работает.

Закоммитила:

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:подключение сURL к проекту и установка динамических библиотек"

[lab04 2e3f1a3] code:подключение сURL к проекту и установка динамических библиотек

8 files changed, 45 insertions(+), 1 deletion(-)

create mode 100644 .gitignore

delete mode 100644 bin/Debug/lab04.exe

create mode 100644 bin/Debug/libcurl-x64.dll

rewrite obj/Debug/main.o (91%)

rewrite obj/Debug/svg.o (61%)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 23, done.

Counting objects: 100% (23/23), done.

Delta compression using up to 8 threads

Compressing objects: 100% (10/10), done.

Writing objects: 100% (13/13), 840.55 KiB | 3.18 MiB/s, done.

Total 13 (delta 6), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (6/6), completed with 5 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

bb0fa41..2e3f1a3 lab04 -> lab04

7) Статическая сборка программы.

1. Вернулась в Linker settings, выбрала в списке libcurl.dll.a, нажала Edit, изменила имя на libcurl.a и нажала OK.
2. Перешла на вкладку Compiler settings (рядом с Linker settings), выбрала дочернюю вкладку #defines. Там в текстовое ввела CURL\_STATICLIB и вышла из диалога кнопкой OK.

Запустила программу, появилось множество ошибок.

Удалила все изменения, сделанные в попытках собрать программу статически, при помощи git.

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git reset --hard HEAD

HEAD is now at 2e3f1a3 code:подключение сURL к проекту и установка динамических библиотек

8) Получение аргументов программы.

int main(int argc, char\* argv[])

{

curl\_global\_init(CURL\_GLOBAL\_ALL);

if(argc > 1)

{

cout << argc << endl;

for(size\_t i=0; i < argc; i++)

{

cout << "argv[" << i << "]=" << argv[i] << endl;

}

return 0;

}

// Ввод данных

const auto input = read\_input(cin, true);

// Обработка данных

double min, max;

find\_minmax(input, min, max);

const auto bins = make\_histogram(input, min, max);

// Вывод данных

show\_histogram\_svg(bins);

return 0;

}

Закоммитила:

$ git add -A

warning: LF will be replaced by CRLF in lab03-histogram.cbp.

The file will have its original line endings in your working directory

warning: LF will be replaced by CRLF in lab03-histogram.layout.

The file will have its original line endings in your working directory

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:получение аргументов программы"

[lab04 43b0c82] code:получение аргументов программы

6 files changed, 31 insertions(+), 5 deletions(-)

create mode 100644 lab03-histogram.layout

rewrite obj/Debug/main.o (71%)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 22, done.

Counting objects: 100% (22/22), done.

Delta compression using up to 8 threads

Compressing objects: 100% (10/10), done.

Writing objects: 100% (12/12), 265.15 KiB | 2.19 MiB/s, done.

Total 12 (delta 7), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (7/7), completed with 7 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

2e3f1a3..43b0c82 lab04 -> lab04

* lab03-histogram.exe -x --y /z w (снаяала выводится количество аргументов, первый элемент массива присваивает имя программы, последующие элементы другие аргументы)

Выводится:

5

argv[0]=lab03-histogram.exe

argv[1]=-x

argv[2]=--y

argv[3]=/z

argv[4]=w

* lab03-histogram.exe param "with spaces" (снаяала выводится количество аргументов, первый элемент массива присваивает имя программы, последующие элементы другие аргументы)

Выводится:

3

argv[0]=lab03-histogram.exe

argv[1]=param

argv[2]=with spaces

* lab03-histogram.exe param <marks.txt >NUL (скачайте marks.txt) (выполняется программа, так как количество аргументов равно 1, результат программы обнуляется)

Enter number count: Enter numbers: Enter column count:

9) Загрузка файла по сети.

int main(int argc, char\* argv[])

{

curl\_global\_init(CURL\_GLOBAL\_ALL);

if(argc > 1)

{

CURL \*curl = curl\_easy\_init();

if(curl)

{

CURLcode res;

curl\_easy\_setopt(curl, CURLOPT\_URL, argv[1]);

res = curl\_easy\_perform(curl);

curl\_easy\_cleanup(curl);

}

return 0;

}

// Ввод данных

const auto input = read\_input(cin, true);

// Обработка данных

double min, max;

find\_minmax(input, min, max);

const auto bins = make\_histogram(input, min, max);

// Вывод данных

show\_histogram\_svg(bins);

return 0;

}

Закоммитила:

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:загрузка файла по сети"

[lab04 08e06b6] code:загрузка файла по сети

3 files changed, 9 insertions(+), 6 deletions(-)

rewrite obj/Debug/main.o (83%)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 17, done.

Counting objects: 100% (17/17), done.

Delta compression using up to 8 threads

Compressing objects: 100% (7/7), done.

Writing objects: 100% (9/9), 251.36 KiB | 1.76 MiB/s, done.

Total 9 (delta 5), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (5/5), completed with 4 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

43b0c82..08e06b6 lab04 -> lab04

10) Обработка ошибок.

int main(int argc, char\* argv[])

{

curl\_global\_init(CURL\_GLOBAL\_ALL);

if(argc > 1)

{

CURL \*curl = curl\_easy\_init();

if(curl)

{

CURLcode res;

curl\_easy\_setopt(curl, CURLOPT\_URL, argv[1]);

res = curl\_easy\_perform(curl);

if(res != CURLE\_OK)

{

cout << curl\_easy\_strerror(res);

exit(1);

}

curl\_easy\_cleanup(curl);

}

return 0;

}

// Ввод данных

const auto input = read\_input(cin, true);

// Обработка данных

double min, max;

find\_minmax(input, min, max);

const auto bins = make\_histogram(input, min, max);

// Вывод данных

show\_histogram\_svg(bins);

return 0;

}

Закоммитила:

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git add -A

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git commit -m "code:обработка ошибок"

[lab04 d3b450c] code:обработка ошибок

4 files changed, 6 insertions(+), 1 deletion(-)

kubra@DESKTOP-2QN1GL1 MINGW64 ~/OneDrive/Рабочий стол/lab04/cs-lab03 (lab04)

$ git push

Enumerating objects: 19, done.

Counting objects: 100% (19/19), done.

Delta compression using up to 8 threads

Compressing objects: 100% (8/8), done.

Writing objects: 100% (10/10), 127.49 KiB | 2.41 MiB/s, done.

Total 10 (delta 6), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (6/6), completed with 6 local objects.

To https://github.com/Ekaterina-Kubrak/cs-lab03.git

08e06b6..d3b450c lab04 -> lab04

11) Работа с буфером, загруженным по сети.

Input download(const string& address)

{

stringstream buffer;

curl\_global\_init(CURL\_GLOBAL\_ALL);

CURL \*curl = curl\_easy\_init();

if(curl)

{

CURLcode res;

curl\_easy\_setopt(curl, CURLOPT\_URL, address.c\_str());

res = curl\_easy\_perform(curl);

if(res != CURLE\_OK)

{

cout << curl\_easy\_strerror(res);

exit(1);

}

curl\_easy\_cleanup(curl);

}

return read\_input(buffer, false);

}

int main(int argc, char\* argv[])

{

Input input;

if(argc > 1)

{

input = download(argv[1]);

}

else

{

input = read\_input(cin, true);

}

double min, max;

find\_minmax(input, min, max);

const auto bins = make\_histogram(input, min, max);

show\_histogram\_svg(bins);

return 0;

}

12) Сохранение данных из сети в буфер.