СКБ223\_2 Печенежский Арсений ЛР5 Вариант 1

Сначала разберёмся с алгоритмом. Я решил представленный массив отрезков отсортировать по первому элементу, то есть по началу. Таким образом я получил будто бы числовую ось, где каждый следующий отрезок находится правее предыдущего или содержится в нём. Функция sort() как раз эту сортировку реализует. Эта функция находится в funs.cpp. Был реализован просто алгоритм сортировки пузырьком.

void sort(float begin[], float end[], int n) {

for (int i = 0; i < n - 1; i++) {

for (int j = 0; j < n - 1; j++) {

if (begin[j] > begin[j + 1]) {

float temp = begin[j];

begin[j] = begin[j + 1];

begin[j + 1] = temp;

temp = end[j];

end[j] = end[j + 1];

end[j + 1] = temp;

}

}

}

}

Begin и end два массива. В первом находятся начала отрезков, во втором- концы. N просто кол-во элементов, то есть кол-во отрезков.

После сортировки просто пробегаем с начала по этим отрезкам и смотрим: если начало текущего находится слева (меньше) конца предыдущего, то присоединяем (меняем конец на максимальный из концов). Если нет – закидываем получившийся большой отрезок в новые массивы begin1 и end1, которые работают аналогично старым. Реализация в ConsoleApplication3.cpp, то есть в главном файле в main.

float help\_end = end[0], help\_begin = begin[0];

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

if (begin[i] < help\_end) {

if (help\_end < end[i]) help\_end = end[i];

}

else {

begin1[j] = help\_begin;

end1[j] = help\_end;

help\_end = end[i];

help\_begin = begin[i];

j++;

}

}begin1[j] = help\_begin;

end1[j] = help\_end;

help\_end -конец текущего отрезка, что получаем склейкой, help\_begin – начало.

После пробега мы получаем наши отрезки в begin1 и end1, то есть это наш ответ.

Теперь про считывание и вывод. Из-за запреты на string пришлось извращаться, и поэтому я обратился к обычному сравниванию указателей на константные char переменные, а именно к strcmp().

if ((strcmp(argv[1], "--fromfile") == 0))

Была осуществлена проверка на некорректный ввод, выводит "Incorrect input (0)". А также проверка на неверный ввод отрезков (если начало больше конца), и выводит "Incorrect input (1)".

Для вывода примерно та же проверка на флаг. Если его нет, запускается обычный вывод через консоль. Если есть, запускается функция вывода, которая была создана чтобы на писать один и тот же код дважды, то есть для красоты. Функция output(), которая находится в funs.cpp.

void output(char\* f, float a[], float b[], int n) {

std::ofstream fou(f);

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

fou << a[i] << " " << b[i] << "\n";

}

}

Теперь к Makefile. Там я просто запускаю всё, и делаю объектный файл hello. Так же были реализованы 2 очистки: clean (удаление объектного файла) и distclean (удаление всего, то есть объектного и всех текстовых).

В итоге, STL и string не использовались, алгоритм работает, запуск через Makefile реализован. Google tests нет, как и Cmake.