**Глоссарий на тему «Строки»**

Строки - это основной способ работы с текстом в C++, а std::string упрощает работу с ними.

Конкатенация строк - процесс сложения строк.

std::string — класс стандартной библиотеки языка C++, реализующий понятие "строка". Описание этого класса содержится в заголовочном файле iostream, внутри которого заголовочный файл string. Но при использовании строк желательно подключать явно #include <string>.

std::cout - это поток для вывода информации (cOUT = вывод).

std::cin - это поток для ввода информации (cIN = ввод).

std::getline() - функция, которая предназначена для ввода данных из потока, например, для ввода данных из консольного окна.

substr - метод для выделения подстроки (например, text.substr(7, 5); // Выделить подстроку, начиная с позиции 7, длиной 5 символов).

replace - метод для замены части строки на другую (например, text.replace(4, 3, "замена"); // Заменить часть строки с позиции 4 на "замена").

clear - метод для очистки строки (например, text.clear(); // Очистить содержимое строки).

empty - метод для проверки строки на пустоту (например, if (text.empty()) { /\* строка пуста \*/ } else { /\* строка не пуста \*/ }).

erase - метод для удаления части строки (например, text.erase(2, 4); // Удалить 4 символа, начиная с позиции 2).

insert - метод для вставки символов внутри строки (например, text.insert(5, "вставка "); // Вставить "вставка " на позицию 5).

append - метод для добавления символов в конец строки (например, text.append("добавление"); // Добавить "добавление" в конец строки).

find - метод для поиска подстроки в строке (например, size\_t pos = text.find("поиск"); // Найти позицию подстроки "поиск").

rfind - метод для поиска подстроки в строке справа налево (например, size\_t pos = text.rfind("поиск"); // Найти позицию подстроки "поиск" справа налево).