

Отчет по лабораторной 13

Задания

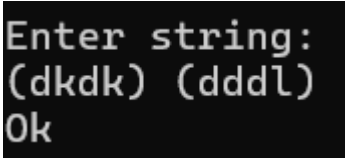
1	<p>1. Изучить способы преобразования символов, выполнив программы в правой части.</p> <p>Выполнить прокрутки программ и записать условия.</p> <p>Опробовать программы с различным текстом.</p>	<pre>#include <iostream> void main() { int num = 5; char symb, new_symbol = ' '; symb = num + '0'; std::cout<<symb<<' '; if (symb >= '0' && symb <= '9') num = symb - '0'; std::cout<<num<<' '; symb = 'b'; if (symb >= 'a' && symb <= 'z') new_symbol = symb - 'a' + 'A'; std::cout<<new_symbol<<' '; }</pre> <p>Заменить четвертую и пятую строчки на операторы:</p> <pre>char *pc; pc = &symb; *pc = num + '0'; std::cout<<*pc<<' ';</pre> <p>Объяснить результат.</p>	<pre>#include <iostream> void main() { char str[] = "Text"; int count = 0; char *pstr; pstr = &str[0]; if (str) while (*pstr++) ++count; std::cout<<count; }</pre> <p>В данной программе указатели может содержать нулевое значение поэтому перед операцией разыменования его следует проверить.</p>																		
2	<p>2. Выполнить программу, приведенную в правой части.</p> <p>Объяснить принцип использования функций стандартной библиотеки.</p>	<p><u>Пример.</u> Имеется адрес файла в сети, например, http://belstu.by/p1/p2/file1.htm. Определить имя последней папки (каталога).</p> <pre>#include <cstring> #include <iostream> using namespace std; void main() { char str[50], s[20]; int i, j, i1, i2; cout<<"Input string:"; cin>> str; for(i = strlen(str); i >= 0; i--) if (str[i] == '\\') break; i2 = i; for (j = (i2 - 1); j >= 0; j--) if (str[j] == '\\') break; i1 = j; if (i2 == i1) printf(" '\\/'); else { strncpy_s(s, &str[i1+1], i2-i1-1); s[i2 - i1 - 1] = 0; cout<<"katalog:"<< s<<"\n"; } }</pre>	<table><tr><th>Функция</th><th>Краткое описание функции</th></tr><tr><td>strcmp</td><td>int strcmp(const char *str1, const char *str2); Сравнивает строки str1 и str2. Если str1 < str2, то результат <0, если str1 = str2, то результат = 0, если str1 > str2, то результат >0.</td></tr><tr><td>strcpy</td><td>char* strcpy(char*s1, const char *s2); Копирует байты из строки s1 в строку s2</td></tr><tr><td>strlen</td><td>int strlen (const char *str); Вычисляет длину строки str</td></tr><tr><td>strncat</td><td>char *strncat(char *s1, const char *s2, int kol); Приписывает kol символов строки s1 к строке s2</td></tr><tr><td>strncpy</td><td>char *strncpy(char *s1, const char *s2, int kol); Копирует kol символов строки s2 в строку s1</td></tr><tr><td>strnset</td><td>char *strnset(char *str, int c, int kol); Заменяет первые kol символов строки s1 символом c</td></tr><tr><td>atoi</td><td>int atoi(char *str); Преобразует числовые символы в целое число</td></tr><tr><td>atof</td><td>float atof(char *str); Преобразует числовые символы в вещественное число</td></tr></table>	Функция	Краткое описание функции	strcmp	int strcmp(const char *str1, const char *str2); Сравнивает строки str1 и str2. Если str1 < str2, то результат <0, если str1 = str2, то результат = 0, если str1 > str2, то результат >0.	strcpy	char* strcpy(char*s1, const char *s2); Копирует байты из строки s1 в строку s2	strlen	int strlen (const char *str); Вычисляет длину строки str	strncat	char *strncat(char *s1, const char *s2, int kol); Приписывает kol символов строки s1 к строке s2	strncpy	char *strncpy(char *s1, const char *s2, int kol); Копирует kol символов строки s2 в строку s1	strnset	char *strnset(char *str, int c, int kol); Заменяет первые kol символов строки s1 символом c	atoi	int atoi(char *str); Преобразует числовые символы в целое число	atof	float atof(char *str); Преобразует числовые символы в вещественное число
Функция	Краткое описание функции																				
strcmp	int strcmp(const char *str1, const char *str2); Сравнивает строки str1 и str2. Если str1 < str2, то результат <0, если str1 = str2, то результат = 0, если str1 > str2, то результат >0.																				
strcpy	char* strcpy(char*s1, const char *s2); Копирует байты из строки s1 в строку s2																				
strlen	int strlen (const char *str); Вычисляет длину строки str																				
strncat	char *strncat(char *s1, const char *s2, int kol); Приписывает kol символов строки s1 к строке s2																				
strncpy	char *strncpy(char *s1, const char *s2, int kol); Копирует kol символов строки s2 в строку s1																				
strnset	char *strnset(char *str, int c, int kol); Заменяет первые kol символов строки s1 символом c																				
atoi	int atoi(char *str); Преобразует числовые символы в целое число																				
atof	float atof(char *str); Преобразует числовые символы в вещественное число																				
3	<p>3. Изучить способы работы с символьными массивами, выполнив программу в правой части.</p>	<pre>void main() { char s, t[20]; int i, j, n; cout<<"Input string: "; gets_s(t); cout<<"Input symbol: "; cin>>s; n = strlen(t); for(i = 0; i <= n; i++) if (t[i] == s) for(j = i; j <= n; j++) t[j] = t[j + 1]; for(i = 0; i < n; i++) cout << t[i]; }</pre> <p><u>Пример.</u> Из предложения удалить все символы, совпадающие с символом, введенным с клавиатуры.</p>																			

1	<div data-bbox="261 85 544 548"> <p>1. Изучить способы преобразования символов, выполнив программы в правой части.</p> <p>Выполнить прокрутки программ и записать условия.</p> <p>Опробовать программы с различным текстом.</p> </div> <div data-bbox="544 85 1046 548"> <pre>#include <iostream> void main() { int num = 5; char symb, new_symbol = ' '; symb = num + '0'; std::cout<<symb<<' '; if (symb >= '0' && symb <= '9') num = symb - '0'; std::cout<<num<<' '; symb = 'b'; if (symb >= 'a' && symb <= 'z') new_symbol = symb - 'a' + 'A'; std::cout<<new_symbol<<' '; }</pre> <p>Заменить четвертую и пятую строчки на операторы:</p> <pre>char *pc; pc = &symb; *pc = num + '0'; std::cout<<*pc<<' ';</pre> <p>Объяснить результат.</p> </div> <div data-bbox="1046 85 1439 548"> <pre>#include <iostream> void main() { char str[] = "Text"; int count = 0; char *pstr; pstr = &str[0]; if (str) while (*pstr++) ++count; std::cout<<count; }</pre> <p>В данной программе указатели может содержать нулевое значение поэтому перед операцией разыменования его следует проверять.</p> </div>
4	<div data-bbox="261 562 544 869"> <p>4. Выполнить программу в правой части.</p> <p>Закомментировать строку <code>cin.ignore...</code></p> <p>Объяснить возникшие проблемы при выполнении программы.</p> </div> <div data-bbox="544 562 1046 869"> <pre>void main() { char t[3][20]; int i, n; cout<<"Input size (<=3): "; cin>>n; cin.ignore(cin.rdbuf()->in_avail()); for(i = 0; i < n; i++) { cout<<"Input string: "; gets_s(t[i]); } for(i = 0; i < n; i++) puts(t[i]); }</pre> </div> <div data-bbox="1046 562 1439 869"> <p>У потока ввода есть буфер чтения, в котором он хранит символы. При неоднократном чтении символов в буфере могут остаться ненужные символы.</p> <p>Буфер очищается с помощью:</p> <pre>cin.ignore(cin.rdbuf()->in_avail());</pre> <p>Функция <code>rdbuf()</code> обращается к буферу, функция <code>in_avail()</code> определяет количество символов, которое нужно</p> </div>
5	<div data-bbox="261 882 544 1451"> <p>5. В правой части записаны два варианта решения задачи. Выполнить программы и объяснить различия между ними.</p> <p>Внести изменения в программу с тем, чтобы проверялось не только количество скобок, но и правильность их расстановки (первой в тексте должна быть открывающая скобка).</p> </div> <div data-bbox="544 882 1439 1451"> <p>проигнорировать с использованием <code>ignore()</code>.</p> <p>Пример проверки соответствия количества открывающих и закрывающих круглых скобок в строке.</p> <div data-bbox="555 1003 963 1444"> <pre>#include <stdio.h> void main() { char s[256]; int i, count; puts("Enter string: "); gets_s(s); for(count=i=0; s[i] != 0; i++) { if (s[i] == '(') count++; if (s[i] == ')') count--; } if (!count) puts("Ok\n"); else puts("Not Ok\n"); }</pre> </div> <div data-bbox="963 1003 1439 1444"> <pre>#include <stdio.h> void main() { char s[256]; int count; char *ps; puts("Enter string: "); gets_s(s); for(count = 0, ps = s; *ps != 0; ps++) { if (*ps == '(') count++; if (*ps == ')') count--; } if (count == 0) puts("Ok\n"); else puts("Not Ok\n"); }</pre> </div> </div>
6 7	<p>1. Преобразовать строку: после каждой буквы a добавить символ «!».</p> <p>2. Дана строка символов, состоящая из букв, цифр, запятых, точек, знаков «+» и «-». Найти количество цифр.</p>
Доп	<p>5. Дополнительные задания.</p> <ol style="list-style-type: none"> В заданной последовательности слов найти все слова, имеющие заданное окончание. В имеющемся словаре найти группы слов, записанных одними и теми же буквами и отличающиеся только порядком расположения. Из заданного предложения удалить те слова, которые уже встречались в предложении раньше. Отредактировать заданное предложение, удаляя из него все слова с нечетными номерами и переворачивая слова с четными номерами. Пример: HOW DO YOU DO преобразовать в OD OD. Даны два предложения. Найти самое короткое из слов первого предложения, которого нет во втором предложении.


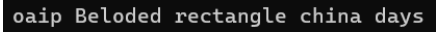
Номер задания	Скриншот кода	Скриншот результата
1 1	<pre> #include <iostream> void main() { int num = 5; char symb, new_symbol = ' '; char* pc; pc = &symb; *pc = num + '0'; std::cout << *pc << ' '; if (symb >= '0' && symb <= '9') num = symb - '0'; std::cout << num << ' '; symb = 'b'; if (symb >= 'a' && symb <= 'z') new_symbol = symb - 'a' + 'A'; std::cout << new_symbol << ' '; } </pre>	5 5 B
	<pre> #include <iostream> void main() { int num = 9; char symb, new_symbol = ' '; char* pc; pc = &symb; *pc = num + '0'; std::cout << *pc << ' '; if (symb >= '0' && symb <= '9') num = symb - '0'; std::cout << num << ' '; symb = 'g'; if (symb >= 'a' && symb <= 'z') new_symbol = symb - 'a' + 'A'; std::cout << new_symbol << ' '; } </pre>	9 9 G
	<p>num = 5</p> <p>symb = num + '0' → symb = '5'</p> <p>Вывести symb → 5</p> <p>Если symb >= '0' и symb <= '9', то</p> <p>num = symb - '0' → num = 5</p> <p>symb = 'b'</p> <p>Если symb >= 'a' и symb <= 'z', то</p> <p>new_symbol = symb - 'a' + 'A' →</p> <p>new_symbol = 'B'</p> <p>Вывести new_symbol → B</p>	
1 2	<pre> #include <iostream> void main() { char str[] = "Text"; int count = 0; char* pstr; pstr = &str[0]; if (str) while (*pstr++) ++count; std::cout << count; } </pre>	4

Номер задания	Скриншот кода	Скриншот результата
	<pre>#include <iostream> void main() { char str[] = "OAIP BP GOOD LUCK!"; int count = 0; char* pstr; pstr = &str[0]; if (str) while (*pstr++) ++count; std::cout << count; }</pre>	<div>18</div>
	<pre>str = "Text" count = 0</pre> <p>Установить указатель <code>pstr</code> на первый символ строки</p> <p>Если строка не пустая</p> <p>Пока текущий символ, на который указывает <code>pstr</code>, не равен <code>'\0'</code></p> <p>Увеличить <code>count</code> на 1</p> <p>Переместить указатель на следующий символ</p> <p>Вывести <code>count</code> → 4</p>	
2	<pre>#include <cstring> #include <iostream> using namespace std; void main() { char str[50], s[20]; int i, j, i1, i2; cout << "Input string:"; cin >> str; for (i = strlen(str); i >= 0; i--) if (str[i] == '\\') break; i2 = i; for (j = (i2 - 1); j >= 0; j--) if (str[j] == '\\') break; i1 = j; if (i2 == i1) printf(" '\\ ' "); else { strncpy_s(s, &str[i1 + 1], i2 - i1 - 1); s[i2 - i1 - 1] = 0; cout << "katalog:" << s << "\n"; } }</pre>	<div>Input string:http://belstu.by/p1/p2/file1.htm</div> <div>katalog:p2</div>

Номер задания	Скриншот кода	Скриншот результата
3	<pre> #include <iostream> using namespace std; void main() { char s, t[20]; int i, j, n; cout << "Input string: "; gets_s(t); cout << "Input symbol: "; cin >> s; n = strlen(t); for (i = 0; i <= n; i++) if (t[i] == s) for (j = i; j <= n; j++) t[i] = t[i + 1]; for (i = 0; i < n; i++) cout << t[i]; } </pre>	<pre> Input string: sdsdsd Input symbol: s ddddd </pre>
4	<pre> #include <iostream> using namespace std; void main() { char t[3][20]; int i, n; cout << "Input size (<=3): "; cin >> n; cin.ignore(cin.rdbuf()->in_avail()); for (i = 0; i < n; i++) { cout << "Input string: "; gets_s(t[i]); } for (i = 0; i < n; i++) puts(t[i]); } </pre>	<pre> Input size (<=3): 2 Input string: 233 Input string: 22 233 22 </pre>
	<p>cin.ignore() обеспечивает удаление остатка из буфера (в данном случае символа <code>\n</code>), чтобы следующие операции ввода работали корректно.</p> <p>Если эту строку закомментировать, программа не будет работать как задумано, так как первая строка ввода будет потеряна.</p>	

Номер задания	Скриншот кода	Скриншот результата
5 1	<pre> #include <stdio.h> using namespace std; void main() { char s[256]; int i, count, isTrue = 1; puts("Enter string: "); gets_s(s); for (count = i = 0; s[i] != 0; i++) { if (s[i] == '(') { count++; } if (s[i] == ')') { count--; if (count < 0) { isTrue = 0; break; } } } if (count == 0 && isTrue) puts("Ok\n"); else puts("Not Ok\n"); } </pre>	 <pre> Enter string: (dkdk) (ddd\l) Ok </pre>
5 2	<pre> #include <stdio.h> using namespace std; void main() { char s[256]; int count = 0; int valid = 1; char* ps; puts("Enter string: "); gets_s(s); for (ps = s; *ps != 0; ps++) { if (*ps == '(') { count++; } if (*ps == ')') { count--; if (count < 0) { valid = 0; break; } } } if (count == 0 && valid) puts("Ok\n"); else puts("Not Ok\n"); } </pre>	 <pre> Enter string: (hello) (bye) Not Ok </pre>

Номер задания	Скриншот кода	Скриншот результата
6 1	<pre> #include <iostream> #define _CRT_SECURE_NO_WARNINGS using namespace std; int main() { setlocale(LC_ALL, "ru"); char arr[] = "variant 7 task 1 for 13 lab"; int length = strlen(arr); int newLength = length; for (int i = 0; i < length; ++i) { if (arr[i] == 'a') { ++newLength; } } char* result = new char[newLength + 1]; int j = 0; for (int i = 0; i < length; ++i) { result[j++] = arr[i]; if (arr[i] == 'a') { result[j++] = '!'; } } result[newLength] = '\0'; strcpy(arr, result); cout << "Преобразованный массив: " << arr << endl; delete[] result; return 0; } </pre>	<pre> Преобразованный массив: va!ria!nt 7 ta!sk 1 for 13 la!b </pre>
6 2	<pre> #include <iostream> using namespace std; int main() { setlocale(LC_ALL, "rus"); char arr[] = "a1b2,c3+d4-e5.---2"; int count = 0; char* ptr = arr; while (*ptr != '\0') { if (*ptr >= '0' && *ptr <= '9') { count++; } ptr++; } cout << "Количество цифр: " << count << endl; return 0; } </pre>	<pre> Количество цифр: 6 </pre>

Номер задания	Скриншот кода	Скриншот результата
Доп 1	<pre> #include <iostream> using namespace std; void main() { char arr[] = "oaip opi kyar math day programm"; char end[] = "aip"; bool isHave = false; char* word = strtok(arr, " "); while (word != nullptr) { int wordLength = strlen(word); int endLength = strlen(end); if (endLength > wordLength) { word = strtok(nullptr, " "); continue; } bool haveEnd = true; for (int i = 0; i < endLength; ++i) { if (word[wordLength - endLength + i] != end[i]) { haveEnd = false; break; } } if (haveEnd) { if (isHave) { cout << word; } else { cout << word; isHave = true; } } word = strtok(nullptr, " "); } if (!isHave) { cout << "Нет слов с таким окончанием"; } cout << endl; } </pre>	
Доп 3	<pre> #include <iostream> using namespace std; int main() { char arr[] = "oaip Beloded rectangle china days rectangle china days"; char words[50][50]; int wordCount = 0; char* word = strtok(arr, " "); while (word != nullptr) { bool isRepeat = false; for (int i = 0; i < wordCount; i++) { if (strcmp(words[i], word) == 0) { isRepeat = true; break; } } if (!isRepeat) { strcpy(words[wordCount], word); wordCount++; } word = strtok(nullptr, " "); } for (int i = 0; i < wordCount; i++) { cout << words[i]; if (i != wordCount - 1) { cout << " "; } } cout << endl; return 0; } </pre>	

Номер задания	Скриншот кода	Скриншот результата
Доп 5	<pre>#include <iostream> using namespace std; int main() { setlocale(LC_ALL, "rus"); char arr1[] = "tasks laboratory angles oaip anime tables"; char arr2[] = "tasks laboratoru anime angles"; char words1[50][50]; char words2[50][50]; int wordCount1 = 0, wordCount2 = 0; char* word = strtok(arr1, " "); while (word != nullptr) { strcpy(words1[wordCount1], word); wordCount1++; word = strtok(nullptr, " "); } word = strtok(arr2, " "); while (word != nullptr) { strcpy(words2[wordCount2], word); wordCount2++; word = strtok(nullptr, " "); } int minLength = 1000; char shortestWord[50]; for (int i = 0; i < wordCount1; ++i) { bool isInSecond = false; for (int j = 0; j < wordCount2; ++j) { if (strcmp(words1[i], words2[j]) == 0) { isInSecond = true; break; } } if (!isInSecond) { int wordLength = strlen(words1[i]); if (wordLength < minLength) { minLength = wordLength; strcpy(shortestWord, words1[i]); } } } if (minLength != 1000) { cout << "Самое короткое слово первого предложения, которого нет во втором: " << shortestWord << endl; } else { cout << "Нет слов первого предложения, которых нет во втором." << endl; } return 0; }</pre>	<p>Самое короткое слово первого предложения, которого нет во втором: oaip</p>