

# Функции работы со строками

Автор: Третьяков Андрей

Действие	Pascal	C
<b>Символы</b>		
Определения	c: Char; i: Integer;	char c; int i;
Получить символ по ASCII-коду	c := Chr(i);	c = i;
Получить ASCII-код символа	i := Ord(c);	i = c;
<b>Строки</b>		
Определения	s, s1, s2, t: String;	char s[1024], s1[1024], s2[1024], t[1024], *p;
i-й символ	s[i] <i>//отсчёт от 1</i>	s[i] <i>//отсчёт от 0</i>
Первый символ	s[1]	s[0]
Окончание строки		'\0'
Заголовочный файл		#include <string.h>
Длина строки	i := Length(s);	i = strlen(s);
	<i>Использует значение s[0]</i>	<i>Вычисляет длину, пробежав всю строку до '\0'. Символ '\0' не считается</i>
Присваивание	t := s;	strcpy(t, s); <i>//опасная</i>
		<i>В буфере t должно быть достаточно места для s и '\0'</i>
		strncpy(t, s, sizeof t);
Конкатенация		<i>t может остаться без '\0', если он не поместится</i>
	s := s1 + s2;	strcat(s1, s2); <i>//опасная</i>
		strncat(s1, s2, n);
Сравнение		<i>//n первых символов из s2</i>
		<i>В буфере s1 должно быть достаточно места для s1, s2 и '\0'</i>
	if s1 = s2	if (strcmp(s1, s2) == 0)
	if s1 <> s2	if (strcmp(s1, s2) != 0)
Сравнение без учёта регистра	if s1 > s2 <i>//&lt;, &gt;=, &lt;=</i>	if (strcmp(s1, s2) > 0) <i>//&lt;, &gt;=, &lt;=</i>
		strncmp(s1, s2, n)
Подстрока	t := Copy(s, idx, len);	<i>//сравнивает n первых символов</i>
Вставить подстроку	Insert(t, s, idx); <i>//t в s</i>	stricmp / strcmpi / strcasemp
Удалить подстроку	Delete(s, idx, len);	strnicmp / strncmpi / strncasemp
Найти подстроку (первое вхождение)	i := Pos(t, s); <i>//возвр. 0, если t в s //не найдена</i>	p = strstr(s, t); <i>//возвр. NULL, если t в s не найдена, //иначе - указатель на подстроку t //получить индекс: i = strstr(s, t) - s;</i>
Найти первый символ с	i := Pos(c, s);	p = strchr(s, c);
Найти последний символ с		p = strrchr(s, c);
<b>Преобразования</b>		
		#include <stdlib.h>
		#include <stdio.h>
Число в строку	Str(i, s);	itoa(i, s, 10); <i>//нет в стандарте и в Linux</i>
		sprintf(s, "%d", i);
		<i>В буфере s должно быть достаточно места для числа и '\0'</i>
Строку в число	Val(s, i, j); <i>//результат в i</i>	i = atoi(s);
		sscanf(s, "%d", &i);

Действие	Pascal	C
<b>Ввод-вывод</b>		
		#include <stdio.h>
Ввод символа	Read(c);	c = getchar();
		c = getc(stdin);
		scanf("%c", &c);
Вывод символа	Write(c);	putchar(c);
		putc(c, stdout);
		printf("%c", c);
Ввод строки без перехода на новую строку	Read(s);	scanf("%s", s); fscanf(stdin, "%s", s); <i>//до первого пробельного символа</i> <i>//(пробел, TAB, '\n')</i>
Ввод строки с переходом на новую строку	ReadLn(s);	gets(s); <i>//очень опасная!</i> <i>//В буфере s должно быть достаточно</i> <i>//места для строки и '\0';</i> <i>//удаляет '\n' из s</i>
		fgets(s, sizeof s, stdin); <i>//может оставить '\n' в s,</i> <i>//но '\0' гарантирован</i>
		scanf("%s\n", s);
		<i>//в конце s не будет '\n', но будет '\0'</i>
Вывод строки без перехода на новую строку	Write(s);	printf("%s", s);
		fprintf(stdout, "%s", s);
		fputs(s, stdout); <i>//не добавляет '\n' и '\0'</i>
Вывод строки с переходом на н.с.	WriteLn(s);	printf("%s\n", s);
		puts(s); <i>//добавляет '\n'</i>