МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»**

**(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

**Лабораторная работа №6**

по дисциплине: Основы программирования тема: «Обработка символьных строк»

Выполнил: ст. группы ПВ-202 Буйвало Анастасия Андреевна

Проверил:

Притчин Иван Сергеевич Брусенцева Валентина

Станиславовна

Белгород 2020 г.

# Лабораторная работа №6

**«Обработка символьных строк»**

**Цель работы**: получение навыков работы со строками. Закрепление навыков работы с подпрограммами.

# Задания для подготовки к работе:

1. Изучите строковый тип, организацию работы со строками в Паскале; стандартные строковые функции.
2. Разбейте задачу соответствующего варианта на подзадачи, таким образом, чтобы решение каждой̆ подзадачи описывалось подпрограммой̆, а основная программа состояла бы в основном из вызовов подпрограмм.
3. Опишите блок-схему алгоритма решения задачи в укрупненных блоках.
4. Для каждой̆ подзадачи опишите используемые структуры данных, спецификацию и блок-схему алгоритма
5. Опишите блок-схему алгоритма решения задачи с использованием блоков «предопределённый̆ процесс».
6. закодируйте алгоритм, не используя подпрограммы стандартной библиотеки для обработки строк. Если в библиотеке есть подпрограмма для решения выделенной подзадачи, то следует описать ее самостоятельно, сохранив название, назначение и список параметров стандартной.
7. Подберите наборы тестовых данных с обоснованием их выбора.

# Задания к работе:

* 1. Наберите программу, отладьте ее и протестируйте.
  2. Выполните анализ ошибок, выявленных при отладке программы.

# Задание варианта №7:

Преобразовать строку, заменяя каждую цифру соответствующим ей числом пробелов.

# Выполнение работы:

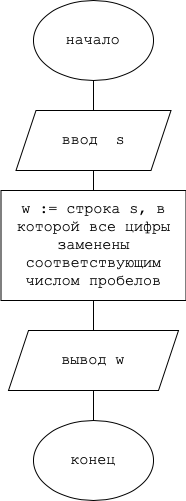
1. **Выделение подзадач**

Выделим следующие подзадачи:

* 1. Ввод строки;
  2. Замена каждой цифры соответствующим числом пробелов:
* поиск цифры
* замена цифры пробелами
  1. Вывод строки.

Опишем алгоритм в укрупненных блоках в терминах выделенных подзадач.

# Блок-схема алгоритма в укрупненных блоках:

****

# Описание подпрограмм:

Функция search\_and\_swap:

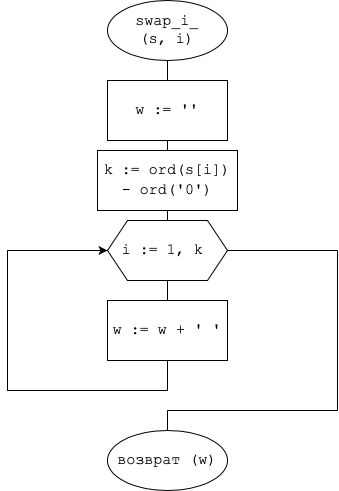
Выделим подзадачу

Функция swap\_i\_:

Спецификация

1. Заголовок: function swap\_i\_ (s : string, i: integer):string.
2. Назначение : возвращает строку, в которой цифра стоящая на i позиции в строке s заменена на соответствующее количество пробелов.
3. Входные параметры: s, i.
4. Выходные параметры: нет.

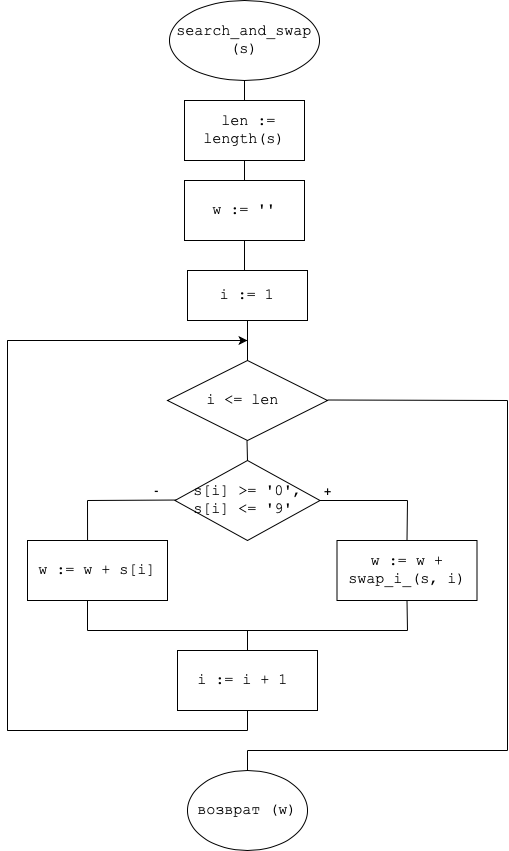
Блок-схема:



Спецификация

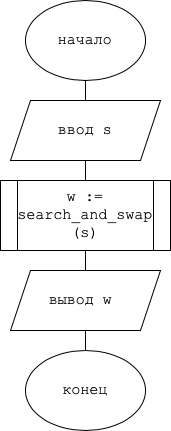
1. Заголовок: function search\_and\_swap (s : string): string;
2. Назначение: возвращает строку, в которой все цифры из строки s заменены соответствующим числом пробелов
3. Входные параметры: s
4. Выходные параметры: нет

Блок-схема:



# Блок-схема алгоритма решения задачи c блоками

**«предопределенный процесс»**

****

1. **Тестовые данные:**

|  |  |  |
| --- | --- | --- |
| № | Исходные данные | Результаты |
| *s* | *w* |
| 1 | 33 коровы | коровы |
| 2 | 3 коровы | коровы |
| 3 | коровы | коровы |

1. **Текст программы:**

program lab6;

{возвращает строку, в которой цифра стоящая на i позиции в строке s заменена на соответствующее количество пробелов }

function swap\_i\_ (s : string; i : integer) : string;

var k : integer;

w : string;

begin

w := ‘’;

k := ord(s[i]) - ord(‘0’);

for i:= 1 to k do

w :=w + ‘ ’;

swap\_i\_ := w;

end;

{возвращает строку, в которой все цифры из строки s заменены соответствующим числом пробелов}

function search\_and\_swap (s : string) : string;

var i : integer;

w : string;

begin

w := ‘’;

len := length(s);

i := 1;

while (i <= len) do

begin

if (s[i] >= ‘0’) and (s[i] <= ‘9’) then

w := w + swap\_i\_(s, i)

else

w := w + s[i];

i := i + 1;

end;

search\_and\_swap := w;

end;

var s, w : string;

len : integer;

begin

writeln(‘ введите строку ’);

read (s);

w := search\_and\_swap(s);

writeln (w);

end.

# Анализ допущенных ошибок:

* При наборе программы ошибок допущено не было;

# Результаты работы программы:

