Министерство образования Республики Беларусь

Учреждение образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра программного обеспечения информационных технологий

Дисциплина: Основы алгоритмизации и программирования (ОАиП)

ОТЧЕТ

по разминочной работе №4

Тема работы: Длинная арифметика (сложение)

Выполнил

студент: гр. 551003 Дементей В.С

Проверила: Фадеева Е.П.

Минск 2015

Содержание

[1 Постановка задачи 4](#_Toc434759563)

[2 Структура данных 5](#_Toc434759564)

[3 Результаты расчетов 6](#_Toc434759565)

[Приложение А 7](#_Toc434759566)

[Приложение Б 9](#_Toc434759567)

# Постановка задачи

Произвести ввод двух целых, положительных чисел с количеством символов больше 50. Произвести сложение этих чисел. Расчет ответа меньше чем за 15 секунд.

# Структура данных

Таблица 1 - Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| str0 | string | Конечный результат |
| str1 | string | Первое число |
| str2 | string | Второе число |
| sim0 | integer | Количество символов результирующей строки и количество повторений |
| sim1 | integer | Количество символов первой строки |
| sim2 | integer | Количество символов второй строки |
| a | integer | Промежуточные значения, которые принимают вид одного символа строки |
| b | integer |
| c | integer |
| k | integer | Переходящий десяток при сложении |
| g | integer | Разность строк |
| sim | char | Используем при перевороте числа, как сохранение символа |

# Результаты расчетов

Для эффективной работы программы переворачиваем начальные числа. Выравниваем количество символов по максимальному значению. Производим сложение чисел. Проверяем возможность создания нового разряда.

Переворачиваем число в исходный вид. В результате получаем сумму первой и второй строки.

Приложение А

(обязательное)

Исходный код программы

**Program** Slojenie;

{$APPTYPE CONSOLE}

**Uses**

SysUtils,

windows;

**Const** chr=45;

null=48;

// Константа для проверки на отрицательность и на нуль

**Var** str0,str1,str2:string;

sim0,sim1,sim2,a,b,c,i,k,g:integer;

sim:char;

{str0-конечный результат

str1-первое число

str2-второе число

sim0-количество символов 3 строки

sim1-количество символов первой строки

sim2-количество символов второй строки

а,b,c- промежуточные значения, которые принимают вид одного символа

k-переходящий десяток при сложении

g-разность строк

sim-используем при перевороте числа, как сохранение символа }

**Begin**

//Подключаем русский язык

setconsoleoutputcp(1251);

setconsolecp(1251);

//вводим 2 числа с проверкой на отрицательность

**repeat**

writeln('Введите первое число');

readln(str1);

**if** ord(str1[1])=chr **then**

writeln('Вы ввели отрицательное число, введите положительное');

**until** ord(str1[1])<>chr;

**repeat**

writeln('Введите второе число');

readln(str2);

**if** ord(str2[1])=chr **then**

writeln('Вы ввели отрицательное число, введите положительное');

**until** ord(str2[1])<>chr;

// расчет количества символов

sim1:=length(str1);

sim2:=length(str2);

sim0:=0;

(\*присвоем максимальное количество символов

конечной строке и выравниваем количество символов

во второй строке\*)

**if** sim1>sim2 **then**

**begin**

sim0:=sim1;

g:=sim1-sim2;

**for** i:=1 **to** g **do**

str2:='0'+str2;

**end**;

**if** sim1<sim2 **then**

**begin**

sim0:=sim2;

g:=sim2-sim1;

**for** i:=1 **to** g **do**

str1:='0'+str1;

**end**;

**if** sim1=sim2 **then**

sim0:=sim1;

str0:=str2;

//переворачиваем первое число

**for** i:=1 **to** (length(str1) **div** 2) **do**

**begin**

sim:=str1[i];

str1[i]:=str1[length(str1)-i+1];

str1[length(str1)-i+1]:=sim;

**end**;

//переворачиваем второе число

**for** i:=1 **to** (length(str2) **div** 2) **do**

**begin**

sim:=str2[i];

str2[i]:=str2[length(str2)-i+1];

str2[length(str2)-i+1]:=sim;

**end**;

k:=0;

// начинаем складывать числа

**for** i:=1 **to** sim0 **do**

**begin**

//преобразуем один символ строки в число

a:=strtoint(str1[i]);

b:=strtoint(str2[i]);

//складываем числа

c:=b+a+k;

k:=0;

// проверка на прибавление нового разряда

**if** (i=sim0) **and** (c>=10) **then**

str0:=str0+'1';

//проверка на переход десятка

**if** c>=10 **then**

**begin**

k:=1;

c:=c-10;

**end**;

//преобразуем число в один символ строки

str0[i]:=inttostr(c)[1];

**end**;

//переворачиваем число в исходную форму

**for** i:=1 **to** (length(str0) **div** 2) **do**

**begin**

sim:=str0[i];

str0[i]:=str0[length(str0)-i+1];

str0[length(str0)-i+1]:=sim;

**end**;

// проверка на нули в начале

**while** ord(str0[1])=null **do**

**begin**

**for** i:=1 **to** length(str0) **do**

str0[i]:=str0[i+1];

**end**;

writeln('Результат сложения');

writeln(str0);

readln;

end.

Приложение Б

(обязательное)

Тестовые наборы

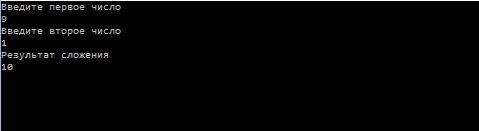
Тест 1

Тестовая ситуация: Проверка создания нового разряда.

Исходные данные: 9+1

Ожидаемый результат: 10

Результат:



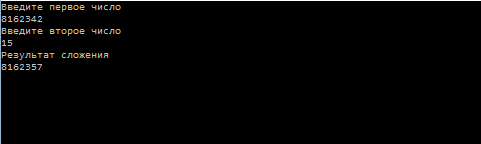
Тест 2

Тестовая ситуация: Проверка ситуации в которой, количество символов первой строки превосходит количество символов второй строки

Исходные данные: 8162342+15

Ожидаемый результат: Вывод корректных данных

Результат:

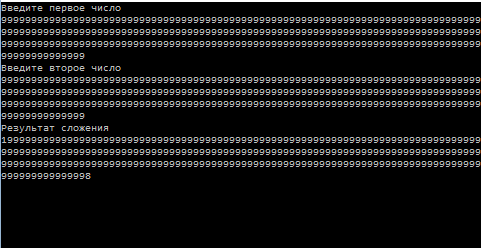


Тест 3

Тестовая ситуация: Ввод максимально допустимого значения

Ожидаемый результат: Вывод данных со сдвигом одного числа

Результат:



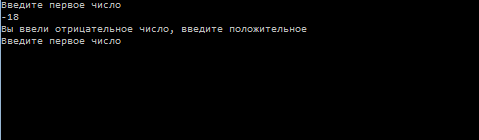
Тест 4

Тестовая ситуация: Проверка на отрицательность.

Исходные данные: -18

Ожидаемый результат: Предупреждение с повтором ввода

Результат:



Тест 5

Тестовая ситуация: Введем множество нулей и в конце число и сложим с обычным числом

Исходные данные: 0000000000000000000000000008+2

Ожидаемый результат: 10

Результат:

