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 №7

Тема работы: Длинная арифметика (деление)

Выполнил

студент: гр. 551003 Дементей В.С

Проверила: Фадеева Е.П.

Минск 2015

Содержание

[1 Постановка задачи 4](#_Toc437300541)

[2 Структура данных 5](#_Toc437300542)

[Приложение А 7](#_Toc437300543)

[Приложение Б 10](#_Toc437300544)

# Постановка задачи

Произвести ввод двух целых, положительных чисел с количеством символов равным 50 и меньше. Произвести деление этих чисел. Вывод ответа меньше чем за 15 секунд, вывести на экран целую часть и остаток от деления.

Убрать символы, которые не имеют смысла.

# Структура данных

Таблица 1 - Данные

|  |  |  |  |
| --- | --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение | Примечание |
| str1 | string | Первое число |  |
| str2 | string | Второе число |  |
| str3 | string | Результат |  |
| str\_ost | string | Остаток |  |
| Arr\_dividend | array[1.. fifty] of  integer | Массив делимого |  |
| Arr\_divider | array[1.. fifty] of  integer | Массив делителя |  |
| Arr\_temp\_dividend | array[1.. fifty] of integer | Часть делимого |  |
| Arr\_temp\_last\_dividend | array[1.. fifty] of  integer | Последний неотрицательный остаток |  |
| Arr\_temp\_divider | array[1.. fifty] of  integer | Массив делителя с учетом разрядности |  |
| Arr\_temp | array[1.. fifty] of  integer | Массив с остатком после разности |  |
| Arr\_result | array[1.. fifty] of  integer | Результирующий массив |  |
| number\_is\_negative | boolean | Тест на отрицательность |  |
| i | integer | Счетчик цикла |  |
| j | integer | Счетчик цикла |  |
| length\_dividend | integer | Длина делимого |  |
| length\_divider | integer | Длина делителя |  |
| itr\_result | integer | Счетчик текущего разряда частного |  |
| copy\_sim | integer | Количество символов для копирования |  |
| socer | integer | Счетчик символов |  |
| soccer\_0 | integer | Счетчик нулей |  |
| temp | integer | Временный результат |  |
| Ten | Integer | Число 10 | Константа |
| input | string | Допустимые символы | Константа |
| Fifty | Integer | Ограничение количества символов при вводе (по условию 50) | Константа |

Приложение А

(обязательное)

Исходный код программы

**Program** Div\_2;

{$APPTYPE CONSOLE}

**uses**

SysUtils;

**Const**

input : string ='0123456789';

ten=10;

fifty=50;

**Var**

str1,str2 : string;

str3,str\_ost : string;

i,j,length\_dividend, length\_divider : integer;

itr\_result,copy\_sim,temp : integer;

Arr\_dividend : **array**[1..fifty] **of** integer;

Arr\_divider : **array**[1..fifty] **of** integer;

Arr\_temp\_dividend : **array**[1..fifty] **of** integer;

Arr\_temp\_last\_dividend : **array**[1..fifty] **of** integer;

Arr\_temp\_divider : **array**[1..fifty] **of** integer;

socer,socer\_0 : integer;

Arr\_temp : **array**[1..fifty] **of** integer;

Arr\_result : **array**[1..fifty] **of** integer;

number\_is\_negative : boolean;

**Begin**

**repeat**

writeln('Enter dividend ');

readln(str1);

socer:=0;

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

**for** j:=1 **to** (length(input)) **do**

**if** (chr(ord(input[j])))=(str1[i]) **then**

socer:=socer+1;

**if** socer<>(length(str1)) **then**

writeln('Error retype');

**until** (socer=(length(str1)));

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

Arr\_dividend[i]:=strtoint(str1[i]);

**repeat**

writeln('Enter divider ');

readln(str2);

socer:=0;

socer\_0:=0;

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

**begin**

**for** j:=1 **to** (length(input)) **do**

**if** (chr(ord(input[j])))=(str2[i]) **then**

socer:=socer+1;

**if** (input[1])=(str2[i]) **then**

socer\_0:=socer\_0+1;

**end**;

**if** (socer<>(length(str2))) **or** (socer\_0 = length(str2)) **then**

writeln('Error retype');

**until** (socer=(length(str2)))**and** (socer\_0 <> length(str2) );

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

Arr\_divider[i]:=strtoint(str2[i]);

length\_dividend :=length(str1);

length\_divider := length(str2);

//Копируем количество символов, которые будут делиться

copy\_sim:=length\_divider;

**while** copy\_sim <= length\_dividend **do**

**begin**

itr\_result:=0;

//Копируем делимое во временный массив прибаляя новый разряд

**for** i:=1 **to** copy\_sim **do**

**begin**

Arr\_temp\_dividend[i]:= Arr\_dividend[i];

//Проверка на дописывание нулей для того чтобы поставить нули на места

**if** (copy\_sim - i >= length\_divider) **then**

Arr\_temp\_divider[i]:=0

**else**

Arr\_temp\_divider[i]:= Arr\_divider[i - copy\_sim + length\_divider]

**end**;

**if** copy\_sim < length\_divider **then**

number\_is\_negative:=true

**else**

number\_is\_negative:=false;

//Начинаем деление текущей итерации через разность

**while** number\_is\_negative=false **do**

**begin**

i:=copy\_sim;

**while** (i>= 1) **and** (number\_is\_negative=false) **do**

**begin**

temp:= Arr\_temp\_dividend[i] - Arr\_temp\_divider[i];

**if** (temp < 0) **then**

**begin**

j:=i-1;

//Заимствование десятка у старшего разряда

**while** (Arr\_temp\_dividend[j] = 0) **and** (j >= 1) **do**

**begin**

Arr\_temp\_dividend[j]:=9;

j:= j-1;

**end**;

**if**( j < 1) **then**

number\_is\_negative:= true

**else**

**begin**

Arr\_temp\_dividend[j]:=Arr\_temp\_dividend[j] - 1;

temp:= temp + 10;

**end**;

**end**;

Arr\_temp[i]:= temp;

i:=i-1;

**end**;

**if** (number\_is\_negative=false) **then**

**begin**

//Сохраняем результаты разности в два массива

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

**begin**

Arr\_temp\_dividend[i]:=Arr\_temp[i];

Arr\_temp\_last\_dividend[i]:=Arr\_temp[i];

**end**;

itr\_result:=itr\_result+1;

**end**

**else**

//Перезаписываев часть результат деления в основной массив

**begin**

**if**(itr\_result>0) **then**

**begin**

**for** i:=1 **to** copy\_sim **do**

**begin**

Arr\_dividend[i]:=Arr\_temp\_last\_dividend[i];

**end**;

**end**;

**end**;

**end**;

//Запись результата

Arr\_result[copy\_sim]:=itr\_result;

copy\_sim:=copy\_sim+1;

**end**;

**for** i:=1 **to** copy\_sim-1 **do**

str3:=str3+inttostr(Arr\_result[i]);

**while** (ord(str3[1])=48) **and**(length(str3)<>1) **do**

delete(str3,1,1);

writeln('The integer part = ', str3);

**for** i:=1 **to** copy\_sim-1 **do**

str\_ost:=str\_ost+inttostr(Arr\_dividend[i]);

**while** (ord(str\_ost[1])=48) **and**(length(str\_ost)<>1) **do**

delete(str\_ost,1,1);

writeln('The residue = ',str\_ost);

readln;

**end**.

Приложение Б

(обязательное)

Тестовые наборы

Тест 1.1

Тестовая ситуация: Проверка функциональности

Исходные данные:

Первое число: 888

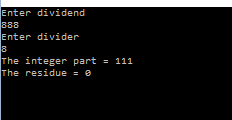
Второе число: 8

Ожидаемый результат:

The integer part = 111

The residue = 0

Результат:



Тест 1.2

Тестовая ситуация: Проверка функциональности (большие цифры)

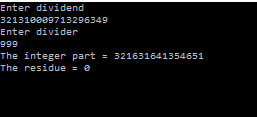
Исходные данные:

Первое число: 321310009713296349

Второе число: 999

Ожидаемый результат:

Результат:



Тест 2.1

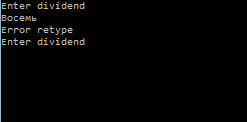
Тестовая ситуация: Проверка ввода данных (буквы и знаки)

Исходные данные:

Первое число: Восемь

Ожидаемый результат: Error retype

Результат:



Тест 2.2

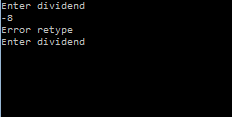
Тестовая ситуация: Проверка ввода данных (отрицательное число)

Исходные данные:

Первое число: -8

Ожидаемый результат: Error retype

Результат:



Тест 2.3

Тестовая ситуация: Деление на нуль

Исходные данные:

Первое число: 321310009713296349

Второе число: 0

Ожидаемый результат: Error retype

Результат:

