Содержание

[1 Постановка задачи 3](#_Toc121151303)

[1.1 Исходное условие задачи 3](#_Toc121151304)

[1.2 Модифицированное условие задачи 3](#_Toc121151305)

[1.3 Исходные данные 3](#_Toc121151306)

[2 Структура данных 4](#_Toc121151307)

[3 Детали и методика решения задачи 5](#_Toc121151308)

[3.1 Детали задачи 5](#_Toc121151309)

[3.2 Методика решения задачи 5](#_Toc121151310)

[3.3 Проверка исходных данных с помощью цикла с постусловием 6](#_Toc121151311)

[4 Результаты расчетов 8](#_Toc121151312)

[Приложение А 9](#_Toc121151313)

[Приложение Б 14](#_Toc121151314)

# Постановка задачи

## Исходное условие задачи

Есть два целых натуральных числа, состоящих до 50 цифр. Вывести на экран их произведение.

## Модифицированное условие задачи

Есть два целых числа (Num\_1 и Num\_2), состоящих до 127 цифр, в определенной системе счисления. Вывести на экран разность чисел ArrayOfMul.

## Исходные данные

Программа получает от пользователя :

* систему счисления;
* уменьшаемое;
* вычитаемое.

Вывод данных происходит в следующем виде:

Произведение чисел Num\_1 и Num\_2 равно ArrayOfMul

# Структура данных

Таблица 1 – Данные

|  |  |  |
| --- | --- | --- |
| Элементы данных | Рекомендуемый тип | Назначение |
| Num\_1 | ShortString | Первый множитель |
| Num\_2 | ShortString | Второй множитель |
| N | SmallInt | Вспомогательная переменная для подсчетов |
| I | SmallInt | Вспомогательная переменная для подсчетов |
| J | SmallInt | Вспомогательная переменная для подсчетов |
| К | SmallInt | Отвечает за сдвиг на разряд |
| RunkOfNum | SmallInt | Промежуточное значение умножения n-х разрядов чисел |
| Z1 | SmallInt | Знак первого числа |
| Z2 | SmallInt | Знак второго числа |
| S | SmallInt | Система счисления |
| LengthA | Integer | Длина ArrayOfMul |
| ArrayOfSub | Array[1..257] of SmallInt | Массив для произведения Num\_1 и Num\_2 |
| L | Boolean | Переменная для проверки на ошибки |

# Детали и методика решения задачи

## Детали задачи

Детали задачи таковы:

* числа должны быть введены без знака, если они положительные;
* числа должны быть целыми;
* числа могут быть отрицательными;
* система счисления должна быть от 2-ой до 62-ой, так как дальше в таблице ASCII используются другие символы;
* числа должны быть введены со знаком, если они отрицательные;
* количество цифр в числе должно быть не больше 127;
* второе число может быть больше, чем первое;
* буквы в системах счисления, предполагающих их, вводятся на латинице.

## Методика решения задачи

Проанализировав условие и детали задачи, можно сделать вывод, что для решения необходимы следующие пункты:

* ввод числа должен быть в тип Shortstring, т. к. тип Integer может содержать числа в промежутке от -231 до 231 - 1 (7 символов), a Shortstring – 256 символов;
* использования функции ord для получения значения кода цифры в таблице ASCII;
* «перевод в десятичные числа» (например, дана 16-тиричная система счисления, где для записи чисел используются как цифры, так и буквы; во время «перевода в десятичные числа» от кода, в зависимости буква это или цифра, будет отниматься разница кода и значения в десятичной системе счисления, а именно для букв – 55, а для цифр – 48):

Таблица 2 – Пример к «переводу в десятичные числа»

|  |  |  |
| --- | --- | --- |
| Значение цифры числа | Код в ASCII | Значение в десятичной системе |
| А | 65 | 10 |
| 2 | 50 | 2 |

* использование массива типа smallint для подсчётов;
* наличие переменных z1 и z2, которые будут отвечать за знаки (если равна 1 – число положительное; -1 – отрицательное) при выводе начальных значений чисел и подсчётах.

Сами расчёты будут проходить по следующим пунктам:

1. Проверка чисел на их длину, чтобы выявить, по какому числу делать внешний цикл, так как удобнее умножать на меньшее.
2. «Перевод в десятичное число» цифры первого числа, умножение её на Z1 и запись в RankOfNum.
3. «Перевод в десятичное число» цифры второго числа, умножение её на Z2 и RankOfNum, запись результата которой будет записано в RankOfNum.
4. Если RunkOfNum больше, чем S-1, то к следующему элементу массива прибавляется (RankOfNum div S) (как переход к новому разряду); а от RunkOfNum отнимается (RankOfNum div S) \* S.
5. Если ArrayofMul[I] больше, чем S-1, то к следующему элементу массива прибавляется 1 (как переход к новому разряду); а от ArrayOfMul[I] отнимается (ArrayOfMul[I] div S) \* S.
6. Увеличение К для смещения начала сложения новых произведений цифр Num\_1 и Num\_2.

## Проверка исходных данных с помощью цикла с постусловием

При проверке исходных данных использовался цикл с постусловием, так как в таком случае цикл будет запущен хотя бы один раз:

Repeat

Write('Введите первое число: ');

readln(Num\_1);

j := length(Num\_1);

for I := (LengthA-1) downto (LengthA-length(Num\_1)) do

Begin

if (ord(Num\_1[J]) < 48) or

((ord(Num\_1[J]) >= 58) and (ord(Num\_1[J]) <= 64))

then

Begin

Writeln('Вы ввели неверный символ! Попробуйте ещё

раз');

L := False;

End

Else

L := True;

if ord(Num\_1[J]) <= 57 then

ArrayOfSum[I] := ord(Num\_1[J]) - 48

Else

ArrayOfSum[I] := ord(Num\_1[J]) - 55;

j := j - 1;

End;

Until L;

С помощью этого цикла мы повторяем вывод фразы «Введите первое число: », получение данных от пользователя, если было правильно введено значение, иначе фразу «Вы ввели неверный символ! Попробуйте еще раз!» до тех пор, пока полученные значения не будет выполнять всем требованиям. Так же происходит и с вводом второго числа, и системы счисления.

# Результаты расчетов

Вследствие выполнения программы на экран выводятся следующие результаты расчетов:

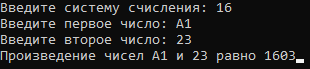


Рисунок 1– Результаты расчётов

Приложение А

(обязательное)

Исходный код программы

program Произведение\_СС;

{

Multiplying two numbers (around 126 digits in each)

}

uses

System.SysUtils;

var

Num\_1, Num\_2: shortstring;

N, K, I, j, Z1, Z2, RankOfNum, S: smallint;

L: boolean;

ArrayOfMul: array [1 .. 256] of shortint;

// Num\_1, Num\_2 - numbers, which must be multiplied

// I, J, K, N - auxiliary variables for counting

// RunkOfNum - intermediate value of n-digit

// multiplying of numbers

// Z1, Z2 - variables of signs of numbers

// ArrayOfMul - array to multiply Num\_1 and Num\_2

// L - variable to check for errors

// S - the numbering system

const

LengthA = length(ArrayOfMul) + 1;

// LengthA - Length ArrayOfSum

begin

// Requiring and checking the Numbering system

Repeat

Write('Введите систему счисления: ');

readln(S);

if (S < 2) or (S > 62) then

Begin

Writeln('');

L := False;

End

Else

L := True;

Until L;

Repeat

Write('Введите первое число: ');

readln(Num\_1);

// Checking for sign of the first number

if Num\_1[1] = '-' then

Begin

Delete(Num\_1, 1, 1);

Z1 := -1;

End

else

Z1 := 1;

// Requiring the second number

Write('Введите второе число: ');

readln(Num\_2);

// Checking for sign of the second number

if Num\_2[1] = '-' then

Begin

Delete(Num\_2, 1, 1);

Z2 := -1;

End

else

Z2 := 1;

I := LengthA - 1;

// Determine the number by which the work will occur

if length(Num\_1) >= length(Num\_2) then

Begin

For j := length(Num\_2) downto 1 do

if (ord(Num\_2[j])<48) or

((ord(Num\_2[j])>=58) and (ord(Num\_2[j])<=64)) then

Begin

Writeln('Вы ввели неверный символ! Попробуйте ещё

раз');

L := False;

End

Else

Begin

for N := length(Num\_1) downto 1 do

if (ord(Num\_1[N])<48) or ((ord(Num\_1[N])>=58)

and (ord(Num\_1[N])<=64)) then

Begin

Writeln('Вы ввели неверный символ! Попробуйте

ещё раз');

L := False;

End

Else

Begin

// Adding digits of the first number

// as like they are in 10th numbering system

if ord(Num\_1[N]) <= 57 then

RankOfNum := Z1 \* (ord(Num\_1[N]) - 48)

Else

RankOfNum := Z1 \* (ord(Num\_1[N]) - 55);

if ord(Num\_2[j]) <= 57 then

RankOfNum:= RankOfNum\*Z2\*(ord(Num\_2[j])-48)

Else

RankOfNum:= RankOfNum\*Z2\*(ord(Num\_2[j])-55);

if RankOfNum > (S-1) then

Begin

// Changing the rank

ArrayOfMul[I-1] := ArrayOfMul[I-1]+

(RankOfNum div S);

RankOfNum:=RankOfNum-((RankOfNum div S)\*S);

End;

ArrayOfMul[I] := ArrayOfMul[I] + RankOfNum;

// A new rank of the sum

If ArrayOfMul[I] > (S-1) then

Begin

ArrayOfMul[I]:=ArrayOfMul[I]-(ArrayOfMul[I]

div S) \* S;

ArrayOfMul[I-1] := ArrayOfMul[I-1] + 1;

End;

I := I - 1;

L := True;

End;

K := K + 1;

I := LengthA - 1 - K;

End;

End

Else

Begin

For N := length(Num\_1) downto 1 do

if (ord(Num\_1[N]) < 48) or

((ord(Num\_1[N])>=58) and (ord(Num\_1[N])<=64)) then

Begin

Writeln('Вы ввели неверный символ! Попробуйте ещё

раз');

L := False;

End

Else

Begin

for j := length(Num\_2) downto 1 do

if (ord(Num\_2[j]) < 48) or ((ord(Num\_2[j])>=58)

and (ord(Num\_2[j])<=64)) then

Begin

Writeln('Вы ввели неверный символ! Попробуйте

ещё раз');

L := False;

End

Else

Begin

// Adding digits of the first number

// as like they are in 10th numbering system

if ord(Num\_1[N]) <= 57 then

RankOfNum := Z1 \* (ord(Num\_1[N]) - 48)

Else

RankOfNum := Z1 \* (ord(Num\_1[N]) - 55);

if ord(Num\_2[j]) <= 57 then

RankOfNum:=RankOfNum\*Z2\*(ord(Num\_2[j])-48)

Else

RankOfNum:=RankOfNum\*Z2\*(ord(Num\_2[j])-55);

if RankOfNum > (S-1) then

Begin

// Changing the rank

ArrayOfMul[I-1] := ArrayOfMul[I-1] +

(RankOfNum div S);

RankOfNum:=RankOfNum-((RankOfNum div S)\*S);

End;

ArrayOfMul[I] := ArrayOfMul[I] + RankOfNum;

// A new rank of the sum

If ArrayOfMul[I] > (S-1) then

Begin

ArrayOfMul[I]:=ArrayOfMul[I]-(ArrayOfMul[I]

div S) \* S;

ArrayOfMul[I-1] := ArrayOfMul[I-1] + 1;

End;

I := I - 1;

L := True;

End;

K := K + 1;

I := LengthA - 1 - K;

End;

End;

Until L;

// Recreating the sign of the numbers

if Z1 = -1 then

Insert('-', Num\_1, 1);

if Z2 = -1 then

Insert('-', Num\_2, 1);

Write('Произведение чисел ', Num\_1, ' и ', Num\_2,

' равно ');

// Finding the start of the mul

if length(Num\_1) >= length(Num\_2) then

j := lengthA - 2\*length(Num\_1)

else

J := LengthA - 2\*Length(Num\_2);

if ArrayOfMul[j] = 0 then

While ArrayOfMul[j] = 0 do

j := j + 1;

If ArrayOfMul[J] < 0 then

Write('-');

// Outputting the mul in the required numbering system

for I := j to (LengthA - 1) do

if abs(ArrayOfMul[I]) <= 9 then

Write(char((abs(ArrayOfMul[I]) + 48)))

Else

Write(char((abs(ArrayOfMul[I]) + 55)));

readln;

end.

Приложение Б

(обязательное)

Тестовые наборы

Тест 1

Исходные данные:

* система счисления – 10;
* первое число – 1;
* второе число – 999999999999999999999999999999999.

Ожидаемый результат:

999999999999999999999999999999999

Полученный результат:



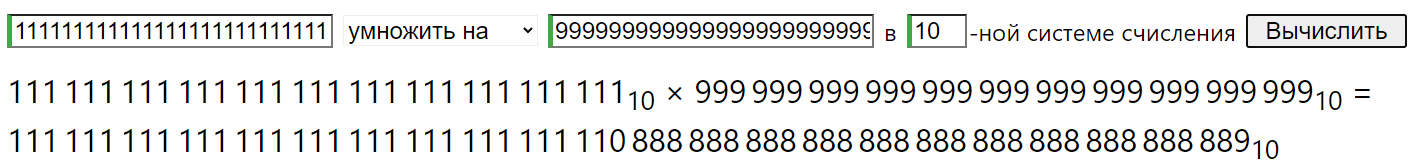
Рисунок 2 – Тест 1

Тест 2

Исходные данные:

* система счисления – 10;
* первое число – 111111111111111111111111111111111;
* второе число – 999999999999999999999999999999999.

Ожидаемый результат:



Полученный результат:

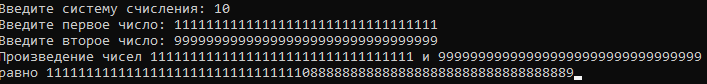


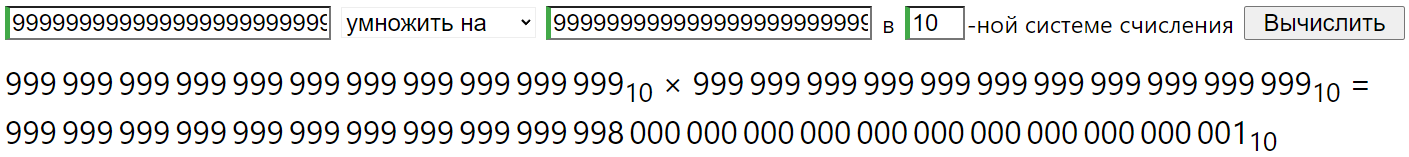
Рисунок 3 – Тест 2

Тест 3

Исходные данные:

* система счисления – 10;
* первое число – 999999999999999999999999999999999;
* второе число – 999999999999999999999999999999999.

Ожидаемый результат:



Полученный результат:

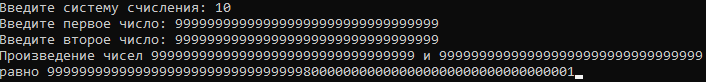


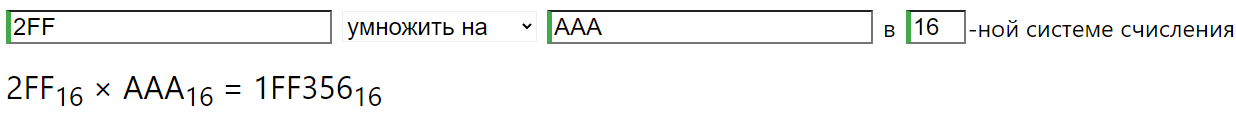
Рисунок 4 – Тест 3

Тест 4

Исходные данные:

* система счисления – 16;
* первое число – 2FF;
* второе число – AAA.

Ожидаемый результат:



Полученный результат:



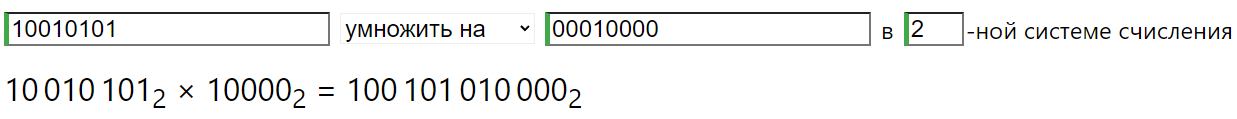
Рисунок 5 – Тест 4

Тест 5

Исходные данные:

* система счисления – 2;
* первое число – 1001 0101;
* второе число – 0001 0000.

Ожидаемый результат:



Полученный результат:

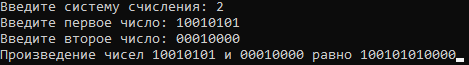


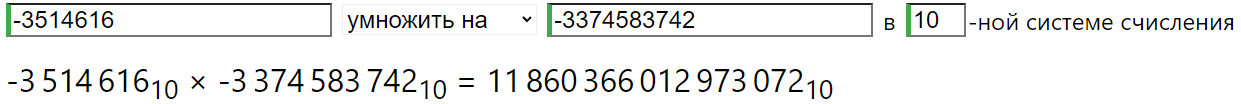
Рисунок 6 – Тест 5

Тест 6

Исходные данные:

* система счисления – 10;
* первое число – -3514616;
* второе число – -3374583742.

Ожидаемый результат:



Полученный результат:

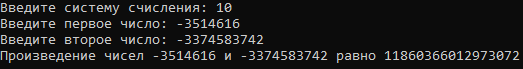


Рисунок 7 – Тест 6

Тест 7

Таблица 3 – Тестовые ситуации

|  |  |  |
| --- | --- | --- |
| Тест | Тестовые ситуации (вместе с расчётами) | Проверка полученного результата |
| 1. |  | [См. Детали задачи](#_Детали_задачи) |
| 2. |  | [См. Детали задачи](#_Детали_задачи) |
| 3. |  | [См. Детали задачи](#_Детали_задачи) |
| 4. |  | [См. Детали задачи](#_Детали_задачи) |
| 5. |  | [См. Детали задачи](#_Детали_задачи) |
| 6. |  | [См. Детали задачи](#_Детали_задачи) |