Темнохуд Михаил, ПЗПИ-16-2

Лабораторная работа №1

# Задание:

15) Для кожного елемента заданого масиву цілих чисел визначити мінімальне число бітів, достатню для запису цих чисел. Сформувати масив, кожен елемент якого буде дорівнює кількості бітів, необхідного для зберігання відповідного елемента.

# Код программы:

// lab1.cpp : Defines the entry point for the console application.

#include "stdafx.h"

#define PRINT\_HEADER \_tprintf(\_T("15. MYKHAILO TEMNOKHUD\r\n\r\n"));

#define PRINT\_TASK\_TITLE \_tprintf(\_T("=== Minimum amount of bits for numbers ===\r\n"));

void GetBitCountArray(int \* array, int length, int \* result);

void GetBinCount(int n, int \* result);

int main()

{

//DECALRATION

int length = 10;

int \* numbers = new int[length];

int \* result = new int[length];

//INITIALIZING

for (int i = 0; i < length; i++)

{

\*(numbers + i) = (i + 1) \* 2;

}

//CALCULATING

GetBitCountArray(numbers, length, result);

//OUTPUT

PRINT\_HEADER;

PRINT\_TASK\_TITLE;

for (int i = 0; i < length; i++)

{

\_tprintf(\_T("%4d: %4d\r\n"), \*(numbers + i), \*(result + i));

}

//FREE MEMORY

delete numbers;

delete result;

\_fgetchar();

return 0;

}

void GetBitCountArray(int \* array, int length, int \* result)

{

for (int i = 0; i < length; i++)

{

GetBinCount(array[i], result + i);

}

}

void GetBinCount(int n, int \* result)

{

\*result = 0;

while (n > 0)

{

\*result += 1;

n /= 2;

}

}

# Результат выполнение:

