**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ**

***Институт Принтмедиа и информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА №14**

**Дисциплина:** Введение в программирование

**Тема:** Инструменты и технологии создания FrontEnd в web-приложениях

**Выполнила: студентка группы 201-723**

Круглова Анастасия Михайловна

**Дата:** 02.12.2020

**Проверил:** Колодочкин Александр Алексеевич

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2020**

<https://github.com/nastyakrul/VVP_.git>

Оглавление

[Задача №1. 3](#_Toc57807420)

[Задача №2. 3](#_Toc57807421)

[Задача №3. 4](#_Toc57807422)

[Задача №4. 5](#_Toc57807423)

[Задача №5. 6](#_Toc57807424)

[Задача №6. 7](#_Toc57807425)

Задача №1.

Даны целые положительные числа A и B (A < B). Вывести все целые числа от A до B включительно; при этом каждое число должно выводиться столько раз, каково его значение (например, число 3 выводится 3 раза).

#include <iostream>

#include <locale.h>

#include <stdio.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int a,b,i,j;//ввод переменных

printf("Введите первое целое положительное число: "); //ввод значения с клавиатуры

scanf\_s("%d", &a);

printf("Введите второе целое положительное число: "); //ввод значения с клавиатуры

scanf\_s("%d", &b);

if ((a < b) && (a>0) && (b>0)) //ввод условия

{

for (i = a; i < b + 1; i++) //внешний цикл

{

for (j = 0; j < i; j++) //внутренний цикл

{

printf("%d\n", i); //вывод результата на экран

}

}

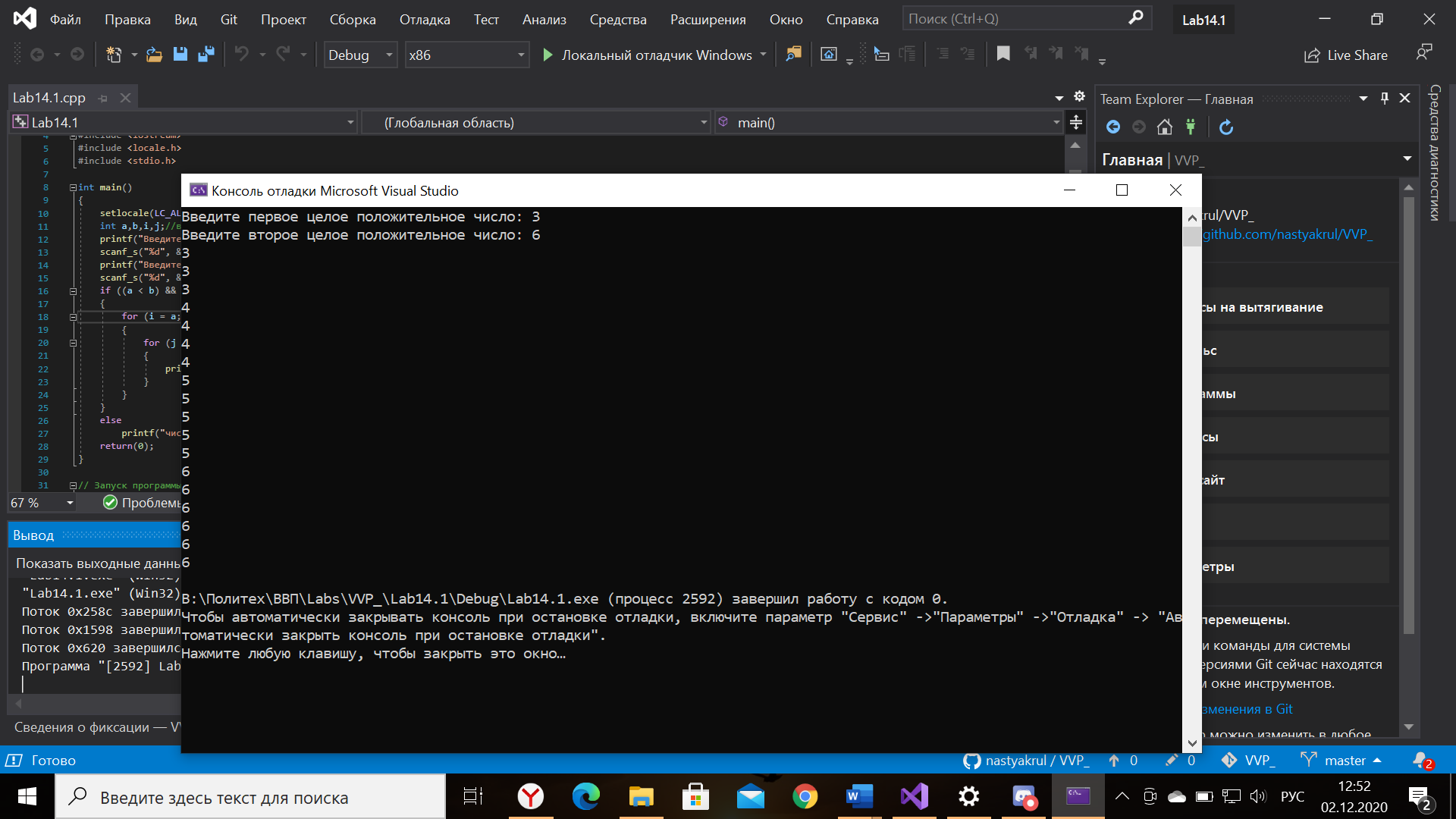
}

else

printf("числа должны быть положительными и первое должно быть меньше второго\n"); //подсказка

return(0);

}



Задача №2.

Даны положительные числа A и B (A > B). На отрезке длины A размещено максимально возможное количество отрезков длины B (без наложений). Не используя операции умножения и деления, найти длину незанятой части отрезка A.

#include <iostream>

#include <locale.h>

#include <stdio.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int a, b, i, j;//ввод переменных

printf("Введите A: "); //ввод значения с клавиатуры

scanf\_s("%d", &a);

printf("Введите B: "); //ввод значения с клавиатуры

scanf\_s("%d", &b);

if ((a > b) && (a > 0) && (b > 0)) //ввод условия

{

while (a >= b) //ввод цикла

{

a = a - b; //уменьшаем число A на B

}

printf("Длина незанятой части отрезка A: %d\n", a); //вывод результата на экран

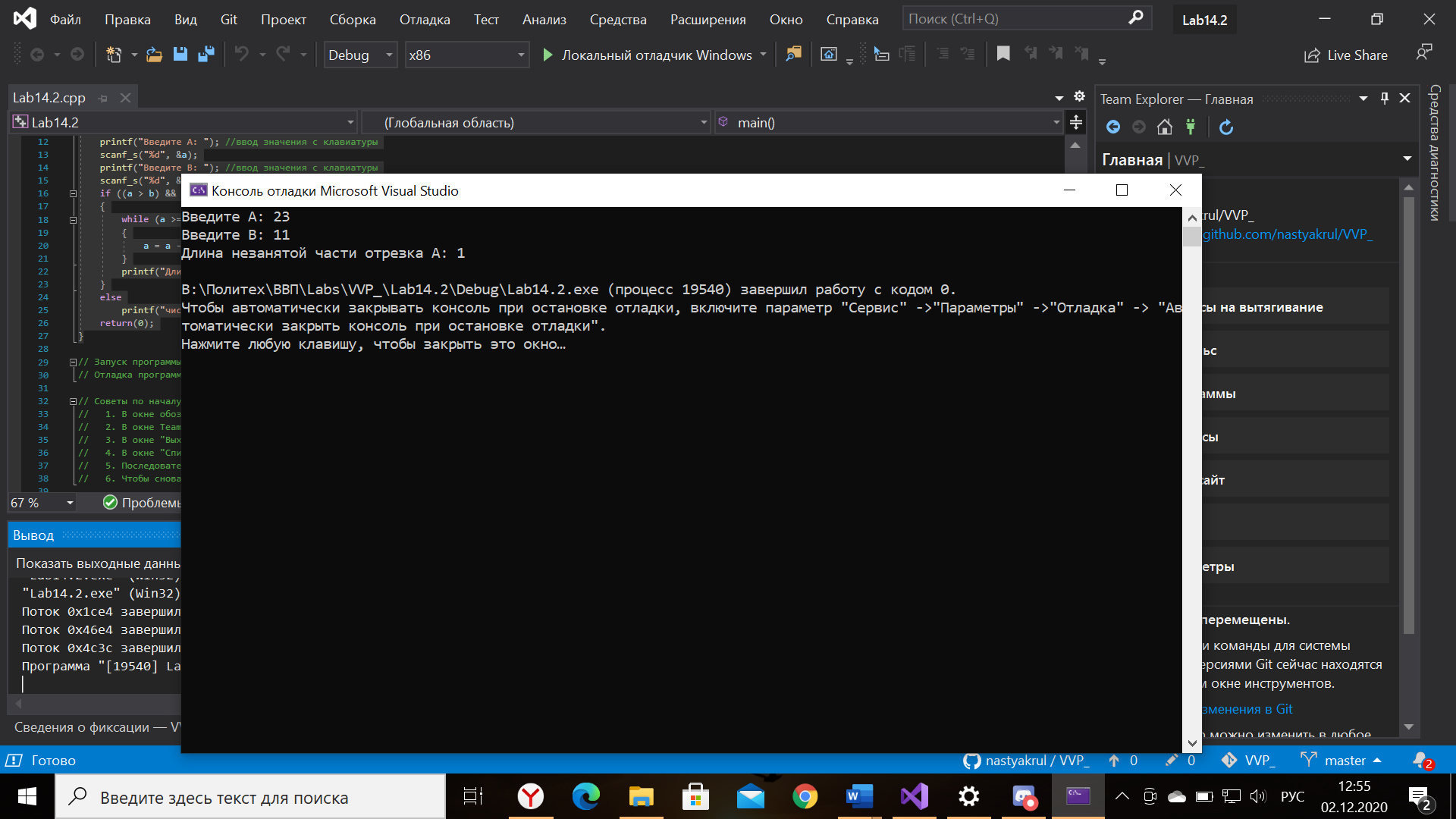
}

else

printf("числа должны быть положительными и первое должно быть больше второго\n"); //подсказка

return(0);

}



Задача №3.

Дано целое число N (> 1). Вывести наименьшее из целых чисел K, для которых сумма 1 + 2 + . . . + K будет больше или равна N, и саму эту сумму.

#include <iostream>

#include <locale.h>

#include <stdio.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int n,k,i;//ввод переменных

printf("Введите целое число N: "); //ввод значения с клавиатуры

scanf\_s("%d", &n);

if (n > 1) //ввод условия

{

k = 0; //начальные значения

i = 0;

while (i < n) //ввод цикла

{

k = k + 1; //увеличение числа

i = i + k; //сумма

}

printf("Наименьшее k = %d\nСумма будет равна %d\n", k, i); //вывод на экран значений

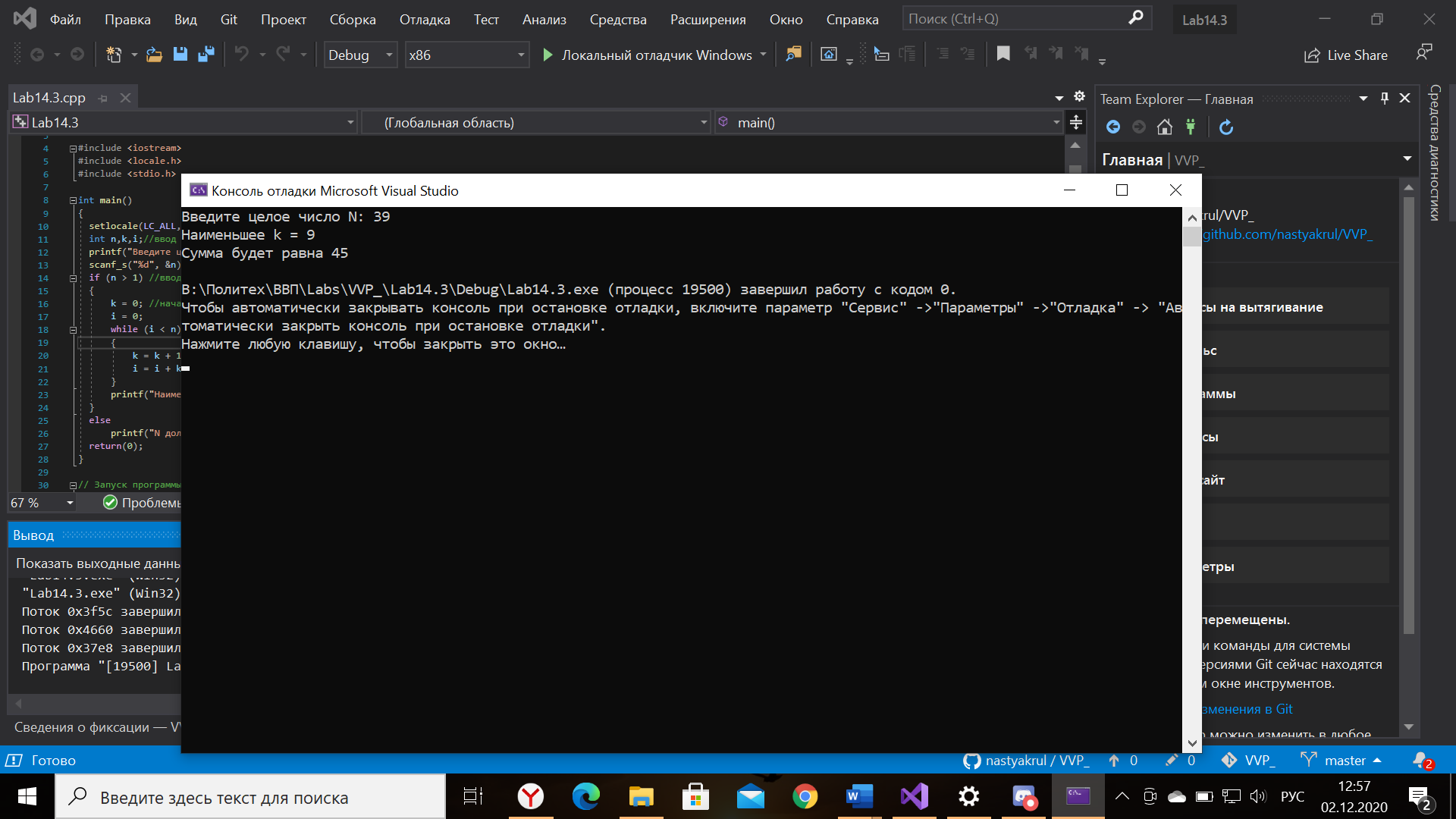
}

else

printf("N должно быть больше 1\n"); //подсказка

return(0);

}



Задача №4.

Начальный вклад в банке равен 1000 руб. Через каждый месяц размер вклада увеличивается на P процентов от имеющейся суммы (P — вещественное число, 0 < P < 25). По данному P определить, через сколько месяцев размер вклада превысит 1100 руб., и вывести найденное количество месяцев K (целое число) и итоговый размер вклада S (вещественное число).

#include <iostream>

#include <locale.h>

#include <stdio.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int s,k;

double p;//ввод переменных

printf("Введите P: "); //ввод значения с клавиатуры

scanf\_s("%lf", &p);

if ((0 < p) && (p < 25)) //ввод условия

{

s = 1000; //начальные значения

k = 0;

while (s <= 1100) //ввод цикла

{

s = s \* (p / 100 + 1); //находим размер вклада за месяц

k++; //увеличиваем число месяца на 1

}

printf("Количество месяцев: %d\nИтоговый размер вклада: %d\n", k, s); //вывод результата на экран

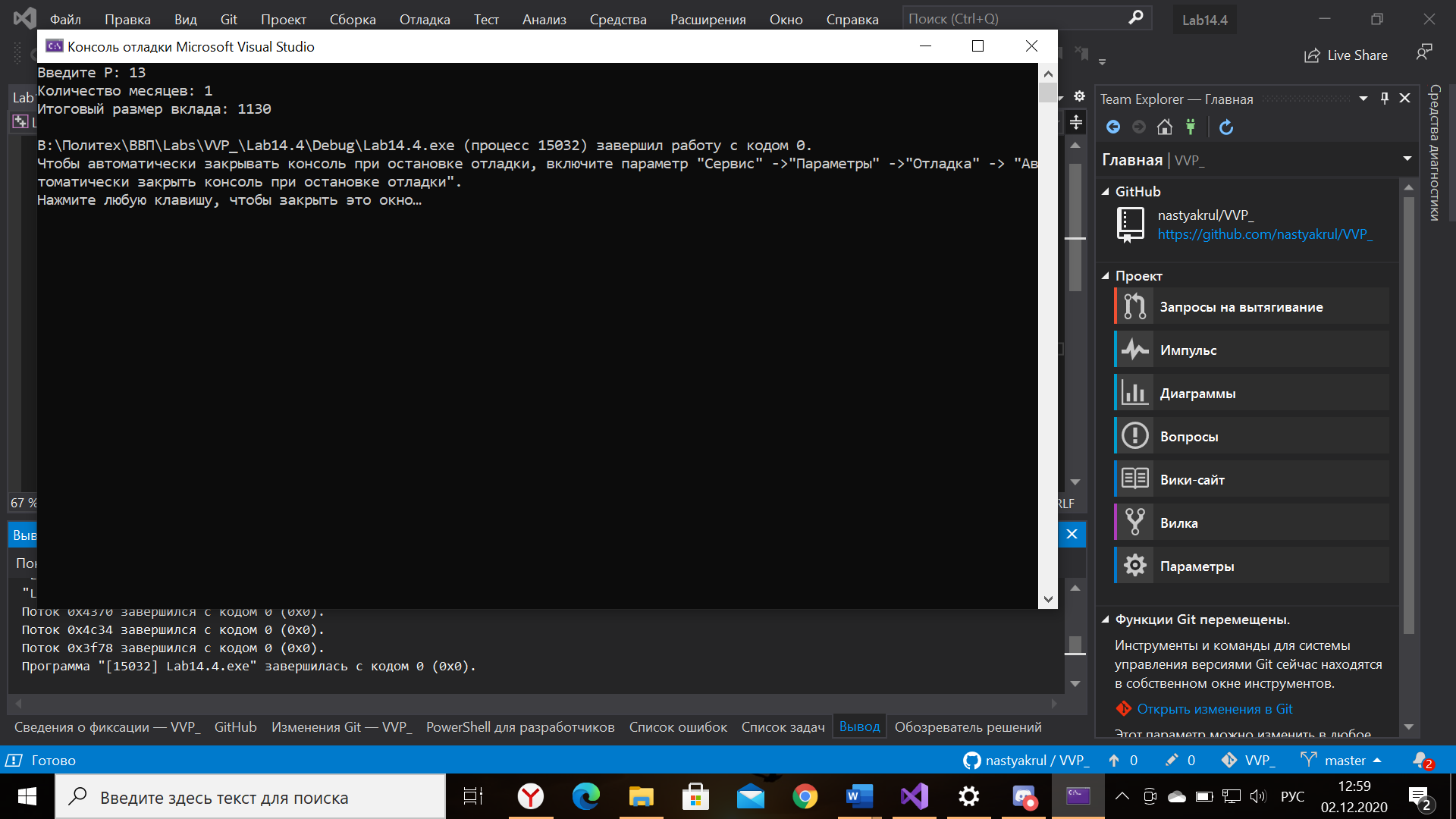
}

else

printf("Процентное число долно быть больше 0, но меньше 25\n"); //подсказка

return(0);

}



Задача №5.

Даны целые положительные числа A и B. Найти их наибольший общий делитель (НОД), используя алгоритм Евклида

#include <iostream>

#include <locale.h>

#include <stdio.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int a,b, s;//ввод переменных

printf("Введите A: "); //ввод значения с клавиатуры

scanf\_s("%d", &a);

printf("Введите B: "); //ввод значения с клавиатуры

scanf\_s("%d", &b);

if ((a > 0) && (b > 0)) //числа должны быть положительными

{

while ((a != 0) && (b != 0)) //ввод цикла

{

if (a > b) //ввод условия

{

a = a % b; //остаток от деления a на b

}

else

b = b % a; //остаток от деления b на a

}

s = a + b; //результат

printf("НОД = %d\n", s); //вывод результата на экран

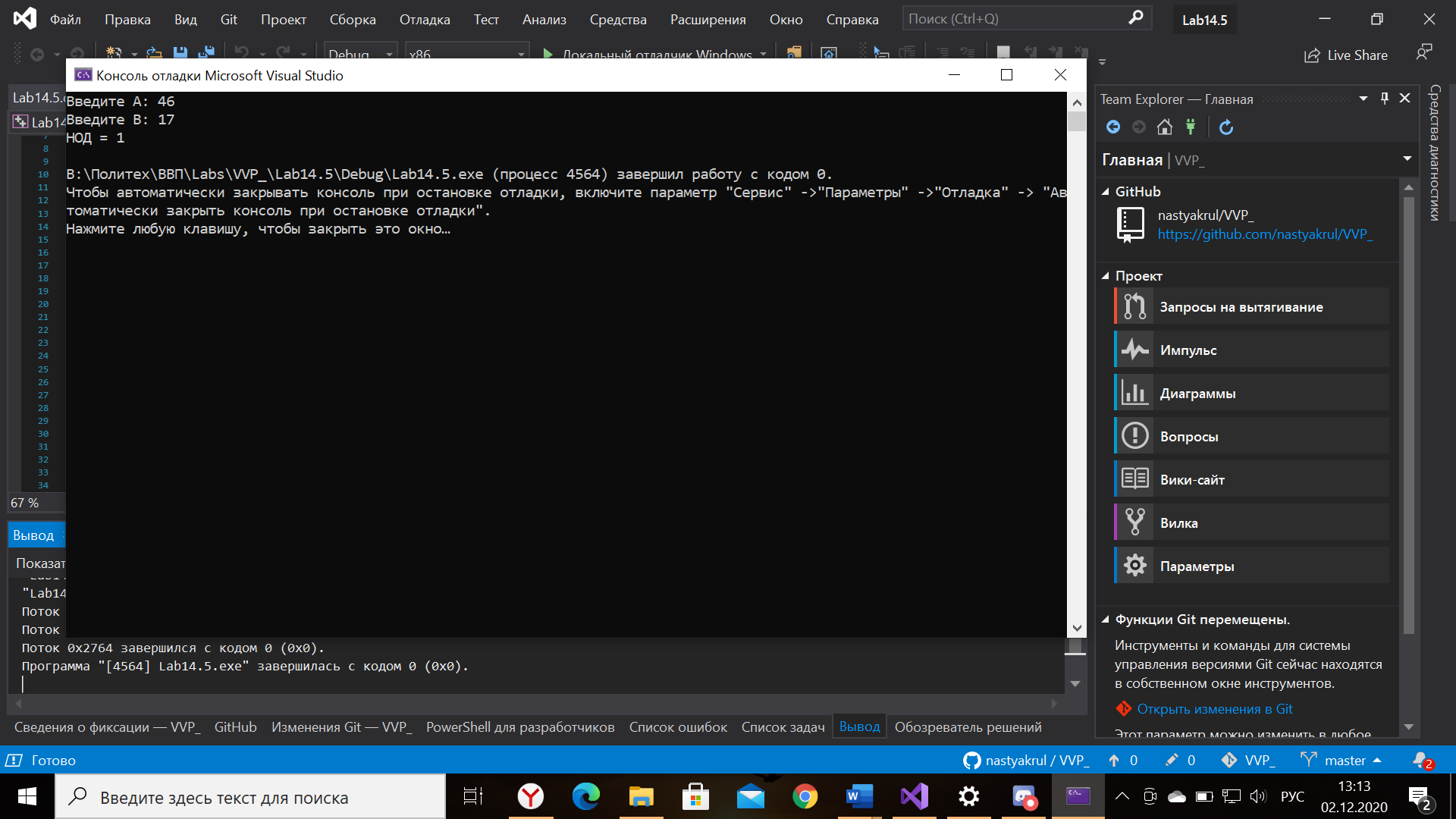
}

else

printf("Числа должны быть положительные\n"); //подсказка

return (0);

}



Задача №6.

#include <iostream>

#include <locale.h>

#include <stdio.h>

int main()

{

setlocale(LC\_ALL, "Russian"); //установка русского языка

int n,f1,f2,f,k;//ввод переменных

printf("Введите N: "); //ввод значения с клавиатуры

scanf\_s("%d", &n);

if (n > 1) //число должно быть больше 1

{

f1 = 1; //ввод начальных значений

f2 = 1;

f = 0;

k = 2;

while (f < n) //цикл пока введенное число больше очередного члена ряда Фибоначчи

{

k++; //увеличение переменной

f = f2 + f1; //нахождение нового значения

f2 = f1; //перестановка

f1 = f;

}

printf("K = %d\n", k); //вывод результата на экран

}

else

printf("N должно быть больше 1\n"); //подсказка

return(0);

}

