

Лекция 5. Ввод и вывод информации в Си

В языке Си ввод и вывод информации возможно осуществлять с использованием консоли. Ввод и вывод информации осуществляется через функции стандартной библиотеки *stdio.h*, посредством следующих функций:

printf() - для вывода информации
scanf_s(), *getchar()* - для ввода информации
gets_s() – ввод строки символов

5.1. Функция *getchar()*

Самый простой механизм ввода заключается в чтении по одному символу за раз, обычно с терминала пользователя, с помощью функции *getchar()*. Функция *getchar()* при каждом к ней обращении возвращает следующий вводимый символ. При этом принимаемый символ будет записан в ASCII коде. ASCII коды основных символов представлены в таблице 3.1.

Функция *getchar()* применяется не так часто, так как может считывать только по одному символу и только в ASCII коде. Т.е. если мы используем данную функцию для ввода с клавиатуры, например, цифры 5, переменная в которую мы считаем вводимый символ будет содержать десятичное число 53.

Пример 5.1. Код программы считывания символа с клавиатуры.

```
#include <stdio.h>           //подключение заголовочного файла
#include <stdlib.h>          //для перехода на русский язык

char a;                      //объявление переменных типа char
int b;                       //объявление переменных типа integer

void main()                  //основной цикл
{                             //начало основного цикла
    a = getchar();           //считывание введенного символа в a
    b = getchar();           //считывание введенного символа в b
}
```

В ходе выполнения кода данной программы будут считаны два символа введенные с клавиатуры и записаны в соответствующие переменные «*a*» и «*b*». Причем первая переменная является типа *char*, а вторая типа *integer*.

5.2. Функция *printf()*

Для вывода в консоль информации используется функция *printf()*. Она переводит данные в символьное представление и выводит полученные изображения символов на экран. При этом у пользователя имеется возможность форматировать данные, то есть влиять на их представление на экране.

Функция **printf()** имеет следующую структуру:

printf("Строка Форматов", объект₁, объект₂, ..., объект_n)

При этом на экран будет выводиться информация, заключенная в кавычках. Строка форматов состоит из следующих элементов:

1. Управляющие символы – это символы, которые не выводятся на экран, а определяют расположение выводимых символов (расположение курсора). Отличительной чертой управляющего символа является наличие обратного слэша «\» перед ним.

2. Текста, т.е. информации представленной для непосредственного вывода на экран.

3. форматов, предназначенных для вывода значений переменных различных типов. Отличительной чертой формата является наличие символа процент «%» перед ним.

В таблице 5.1 представлены основные управляющие символы, применяемые в языке Си при выводе в консоль:

Таблица 5.1.

Управляющие символы в языке Си

Символ	Значение символа
\n	перевод строки
\t	горизонтальная табуляция
\v	вертикальная табуляция
\b	возврат на символ
\r	возврат на начало строки
\a	звуковой сигнал

В таблице 5.2 представлены основные форматы, применяемые при выводе информации в языке Си.

Таблица 5.2.

Форматы вывода информации в языке Си

Формат	Значение формата
%d	целое число типа int со знаком в десятичной системе счисления
%u	целое число типа unsigned int
%x	целое число типа int со знаком в шестнадцатеричной системе счисления
%o	целое число типа int со знаком в восьмеричной системе счисления
%hd	целое число типа short со знаком в десятичной системе счисления;
%hu	целое число типа unsigned short
%hx	целое число типа short со знаком в шестнадцатеричной системе счисления

%ld	целое число типа long int со знаком в десятичной системе счисления
%lu	целое число типа unsigned long int
%lx	целое число типа long int со знаком в шестнадцатеричной системе счисления
%f	вещественный формат (числа с плавающей точкой типа float)
%lf	вещественный формат двойной точности (числа с плавающей точкой типа double)
%e	вещественный формат в экспоненциальной форме (числа с плавающей точкой типа float в экспоненциальной форме)
%c	символьный формат
%s	строковый формат

Отметим, что форма данной функции позволяет выводить на экран текст и значения переменных различного типа. Причем вывод нескольких переменных может быть выполнен в одной функции. При этом их форматы объектов просто разделяются запятыми.

После использования функции **printf** окно консоли автоматически закрывается. Для того, чтобы это не происходило, можно после данной функции использовать функцию **getchar**. Причем если в программе применяется ни одна функция вывода, то необходимо в конце программы прописывать столько же функций **getchar**, т.е. команда закрытия консоли генерируется в конце каждой функции вывода в консоль. При этом если используется одна функция **printf** для вывода нескольких переменных, то количество функций **getchar** должно соответствовать количеству выводимых переменных.

Изначально в библиотеки **stdio.h** имеются только латинские символы. Для вывода информации в кириллице необходимо подключить еще одну библиотеку: **stdlib.h**. В данной библиотеке функция перехода в консоли на русский язык записывается как **system("chcp 1251")**. Также часто используемой является функция отчистки экрана консоли - **system("cls")**.

Приведем ряд примеров по вводу информации с помощью функции **getchar()** и вывода ее на экран.

Пример 5.2. Пример программы вывода на экран фразы «Информационные технологии».

```
#include <stdio.h>           //подключение заголовочного файла
#include <stdlib.h>          //для перехода на русский язык

void main()                  //основной цикл
{                             //начало основного цикла
    system("chcp 1251");      //переходим на русский язык
    system("cls");           //очищаем окно консоли
    printf("Информационные технологии\n"); //выводим в консоль фразу
    getchar();
```

```
}
```

```
//конец основного цикла
```

В результате выполнения программы в консоль будет выведена соответствующая фраза, причем за счет использования управляющего символа `\n` курсор будет находиться на следующей строке:

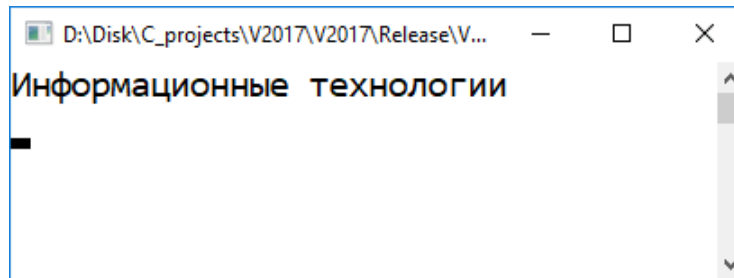


Рис. 5.1. Окно вывода в консоль строки «Информационные технологии»

Пример 5.3. Пример программы ввода трех символов и вывода на экран их ASCII кодов в десятичном виде и самих символов.

```
#include <stdio.h>                                //подключение заголовочного файла
#include <stdlib.h>                                //для перехода на русский язык

int a, b, c;                                     //объявление переменных типа integer

void main()                                     //основной цикл
{
    system("chcp 1251");                         //начало основного цикла
    system("cls");                               //переходим в консоли на русский язык
    printf("Нажмите 3 клавиши и Enter: ");       //очищаем окно консоли
    a = getchar();                               //выводим в консоль фразу
    b = getchar();                               //ожидание ввода символа с клавиатуры
    c = getchar();                               //ожидание ввода символа с клавиатуры
    printf("ASCII код клавиши 1 - %d\n",a);       //ожидание ввода символа с клавиатуры
    printf("ASCII код клавиши 2 - %d\n",b);       //ASCII кода клавиши 1 в дес-ой форме
    printf("ASCII код клавиши 3 - %d\n",c);       //ASCII кода клавиши 2 в дес-ой форме
    printf("Клавиша 1 - %c\n", a);               //ASCII кода клавиши 3 в дес-ой форме
    printf("Клавиша 2 - %c\n", b);               //вывод символа клавиши 1 в формате char
    printf("Клавиша 3 - %c\n", c);               //вывод символа клавиши 2 в формате char
    getchar(); getchar(); getchar();             //вывод символа клавиши 3 в формате char
    getchar(); getchar(); getchar();
}                                                 //конец основного цикла
```

Выполнение данной программы будет осуществляться по следующему алгоритму:

1. Будет открыто окно консоли (рис. 5.2а).
2. Далее необходимо ввести с клавиатуры три любых символа, например, «1», «R» и «%» (рис. 5.2б).
3. После нажатия клавиши Enter на экран будут выведены десятичные значения ASCII кодов данных символов и сами символы (рис. 5.2в).

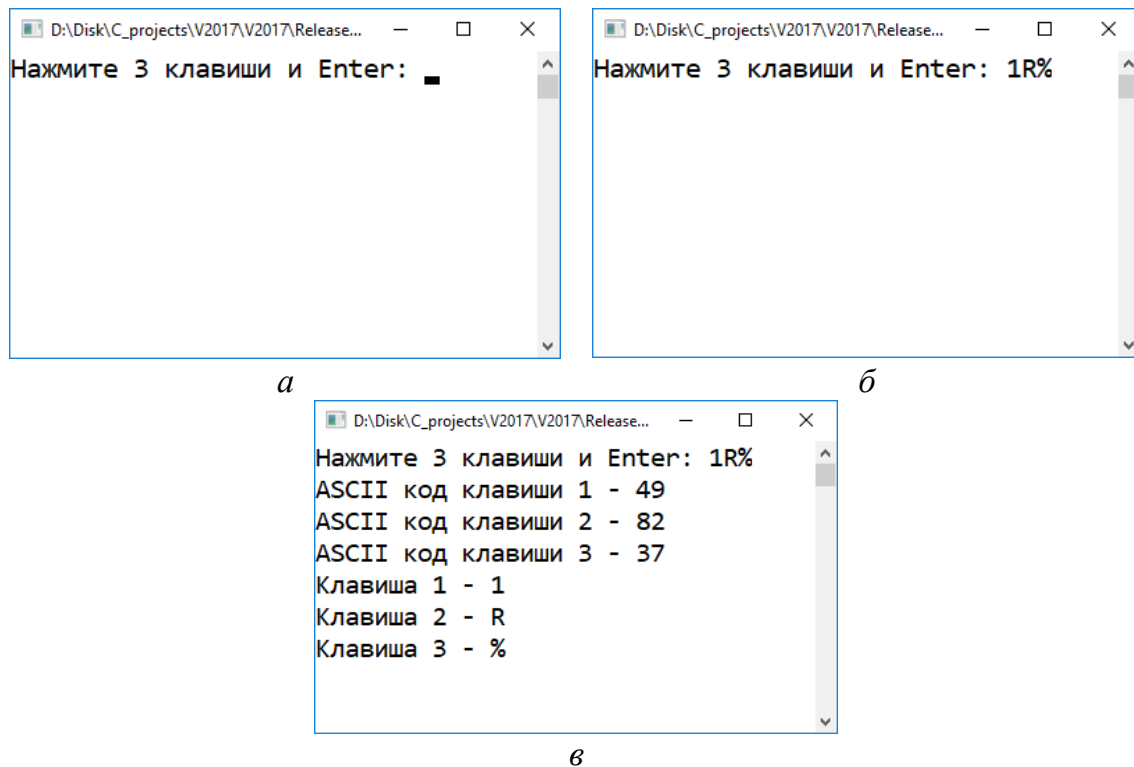


Рис. 5.2. Программа ввода трех символов и вывода на экран их ASCII кодов в десятичном виде и самих символов

Пример 5.4. Пример программы ввода трех символов и вывода на экран их ASCII кодов в десятичном виде и самих символов. В отличие от примера 5.3, в данном случае вывод будет осуществляться с использованием двух функций *printf*. Таким образом будет показана возможность вывода нескольких объектов в рамках одной функции вывода.

```
#include <stdio.h>                                     //подключение заголовочного файла
#include <stdlib.h>                                     //для перехода на русский язык

int a, b, c;                                           //объявление переменных integer

void main()                                           //основной цикл
{
    system("chcp 1251");                             //начало основного цикла
    system("cls");                                    //переходим на русский язык
    printf("Нажмите 3 клавиши и Enter: ");           //очищаем окно консоли
    a = getchar();                                    //выводим в консоль фразу
    b = getchar();                                    //ожидание ввода символа
    c = getchar();                                    //ожидание ввода символа
    printf("ASCII коды - %d, %d, %d\n", a,b,c);       //ожидание ввода символа
    printf("Клавиши - %c, %c, %c\n", a,b,c);         //ASCII коды клавиш в дес-ой форме
    getchar(); getchar();                            //вывод символов клавиш в char
}                                                     //конец основного цикла
```

Выполнение данной программы будет осуществляться по следующему алгоритму:

1. Будет открыто окно консоли (рис. 5.3а).

2. Далее необходимо ввести с клавиатуры три любых символа, например, «1», «R» и «%» (рис. 5.3б).

3. После нажатия клавиши Enter на экран будут выведены десятичные значения ASCII кодов данных символов и сами символы (рис. 5.3в).

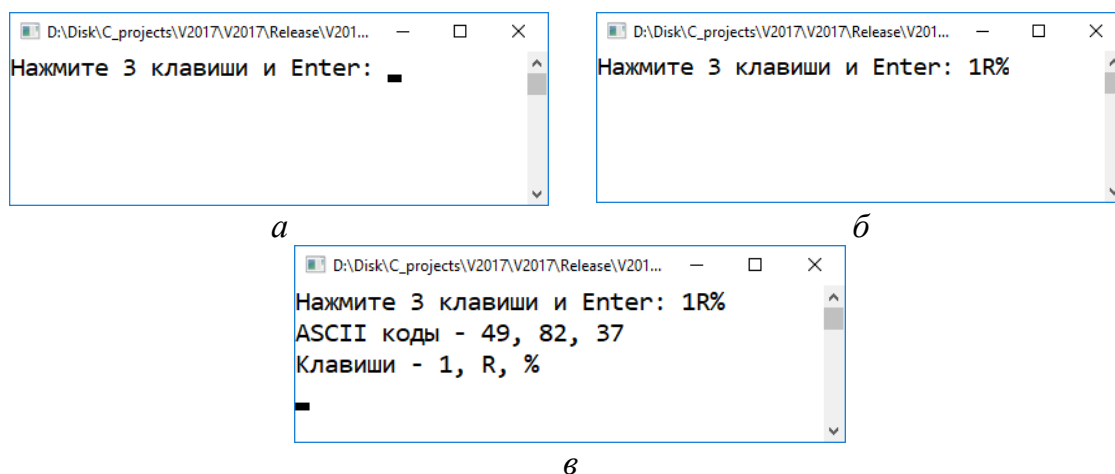


Рис. 5.3. Программа ввода трех символов и вывода на экран их ASCII кодов в десятичном виде и самих символов

Пример 5.5. Пример программы ввода двух символов и вывода на экран первого символа и ASCII кода в десятичном виде второго символа.

```
#include <stdio.h> //подключение заголовочного файла
#include <stdlib.h> //для перехода на русский язык

char a; //объявление переменных типа char
int b; //объявление переменных типа integer

void main() //основной цикл
{ //начало основного цикла
    system("chcp 1251"); //переходим на русский язык
    system("cls"); //очищаем окно консоли
    a = getchar(); //считывание введенного символа в a
    b = getchar(); //считывание введенного символа в b
    printf("Вы ввели символы: %c,%d", a,b); //выводим фразу и символы
    getchar(); getchar();
} //конец основного цикла
```

После запуска данной программы необходимо будет ввести два символа. Символы могут быть как одинаковыми, так и различными. Для примера введем одинаковые символы – цифру «1» (рис. 5.4а). Далее нажав клавишу Enter на экран будет выведена фраза «Вы ввели символы: » и первым будет выведен сам символ, а через запятую его ASCII код в десятичном виде (рис. 5.4б).

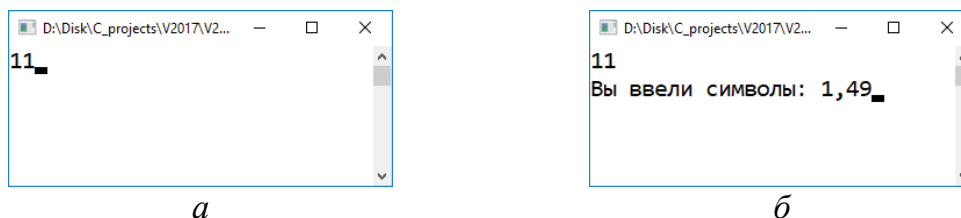


Рис. 5.4. Программа ввода трех символов и вывода на экран их ASCII кодов в десятичном виде и самих символов

5.3. Функция `scanf_s()`

Наиболее универсальной функцией для ввода информации является функция *scanf_s()*. Функция форматированного ввода данных с клавиатуры *scanf_s()* выполняет чтение данных, вводимых с клавиатуры, преобразует их во внутренний формат и передает вызывающей функции. Общая форма записи функции *scanf_s()*:

scanf_s ("СтрокаФорматов", адрес1, адрес2,...);

Строка форматов аналогична функции ***printf()***. Для формирования адреса переменной используется символ амперсанд '&': **адрес = &объект**. Строка форматов и список аргументов для функции обязательны. Окончанием считывания с консоли будет символ нажатия клавиши ***Enter***.

Функция `scanf_s()` может работать сразу с несколькими переменными. Предположим, что необходимо ввести два целых числа с клавиатуры. Формально для этого можно дважды вызвать функцию `scanf_s()`, однако лучше воспользоваться такой конструкцией:

```
scanf_s(" %d, %d ", &n, &m);
```

Функция `scanf_s()` интерпретирует это так, как будто ожидает, что пользователь введет число, затем – запятую, а затем – второе число. Все происходит так, как будто требуется ввести два целых числа следующим образом:

88,221 или 88, 221

Функция `scanf_s()` возвращает число успешно считанных элементов. Если операции считывания не происходило, что бывает в том случае, когда вместо ожидаемого цифрового значения вводится какая-либо буква, то возвращаемое значение равно 0.

Пример 5.6. Пример программы ввода трех десятичных чисел и вывода их на экран. Для ввода чисел, а не символов в ASCII коде необходимо использовать функцию *scanf_s()*.

```
#include <stdio.h> //подключение заголовочного файла
#include <stdlib.h> //для перехода на русский язык

int a, b, c; //объявление переменных типа integer

void main() //основной цикл
{ //начало основного цикла
    system("chcp 1251"); //переходим в консоли на русский язык
```

```
//подключение заголовочного файла
//для перехода на русский язык
```

```
//объявление переменных типа integer
```

```
//основной цикл
//начало основного цикла
```

```
//переходим в консоли на русский язык
```

```

system("cls");
printf("Введите число 1: ");
scanf_s("%d", &a);
printf("Введите число 2: ");
scanf_s("%d", &b);
printf("Введите число 3: ");
scanf_s("%d", &c);
printf("Число 1 - %d\n", a);
printf("Число 2 - %d\n", b);
printf("Число 3 - %d\n", c);
getchar(); getchar(); getchar();
}

```

//очищаем окно консоли
//выводим в консоль фразу
*//считывание числа в переменную **a***
//выводим в консоль фразу
*//считывание числа в переменную **b***
//выводим в консоль фразу
*//считывание числа в переменную **c***
//вывод числа 1 в десятичной форме
//вывод числа 2 в десятичной форме
//вывод числа 3 в десятичной форме
//конец основного цикла

В ходе выполнения данной программы будет выведено стартовое окно для ввода первого десятичного числа, далее вводим первое число (например, 125) и нажимаем **Enter**. Далее, вводим второе число (например, 1038) и вновь нажимаем **Enter** (рис. 5.5б). Вводим третье число (например, -25) и вновь нажимаем **Enter** и все три введенных числа будут выведены на экран (рис. 5.5в).

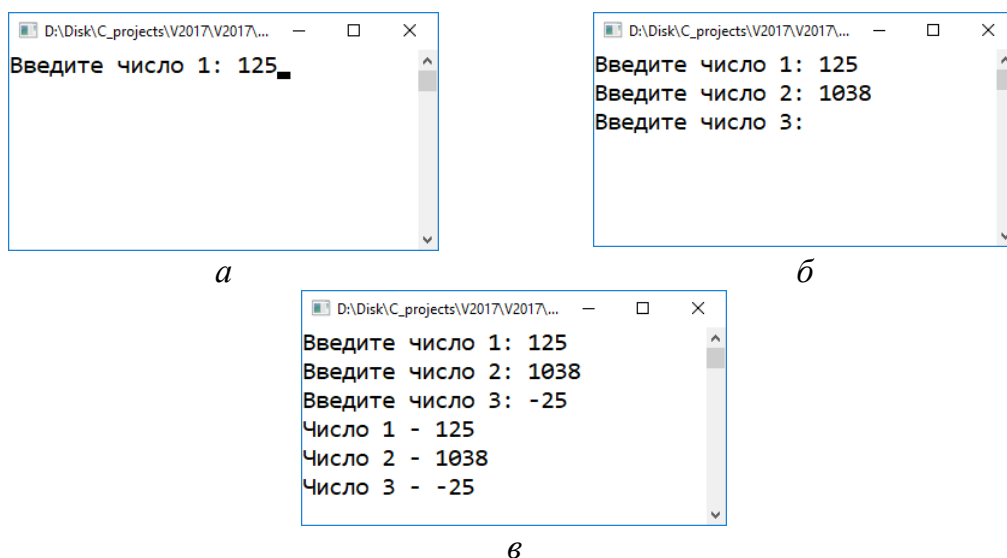


Рис. 5.5. Программа ввода трех десятичных чисел и вывода их на экран

При этом все три числа будут сохранены в соответствующих переменных «*a*», «*b*» и «*c*», соответственно. В данной программе все три переменные определены как *integer*, и в функциях считывания и вывода использовались десятичные форматы. Поэтому при попытке ввода или вывода числа с точкой программа либо будет закрыта, либо будет выполнен некорректный вывод.

Пример 5.7. Пример программы ввода трех чисел с точкой и вывода их на экран. Например, осуществим ввод и вывод следующих чисел: 1.25, -30.54 и 250.1.

Для ввода и вывода чисел с плавающей точкой необходимо в функциях использовать формат *%f*, а не *%d*. А также определить переменные, в которые будет осуществляться ввод, типа *float*.


```

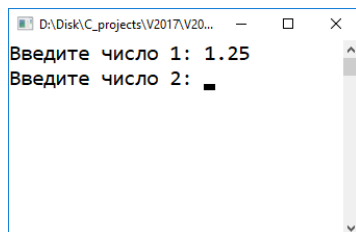
#include <stdio.h>                                //подключение заголовочного файла
#include <stdlib.h>                                //для перехода на русский язык

float a, b, c;                                    //объявление переменных типа float

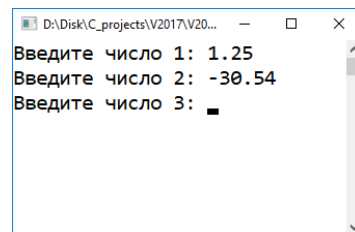
void main()                                       //основной цикл
{                                                 //начало основного цикла
    system("chcp 1251");                          //переходим в консоли на русский язык
    system("cls");                                //очищаем окно консоли
    printf("Введите число 1: ");                 //выводим в консоль фразу
    scanf_s("%f", &a);                           //считывание числа в переменную a
    printf("Введите число 2: ");                 //выводим в консоль фразу
    scanf_s("%f", &b);                           //считывание числа в переменную b
    printf("Введите число 3: ");                 //выводим в консоль фразу
    scanf_s("%f", &c);                           //считывание числа в переменную c
    printf("Число 1 - %f\n", a);                 //вывод числа 1 с точкой
    printf("Число 2 - %f\n", b);                 //вывод числа 2 с точкой
    printf("Число 3 - %f\n", c);                 //вывод числа 3 с точкой
    getchar(); getchar(); getchar();
}                                                 //конец основного цикла

```

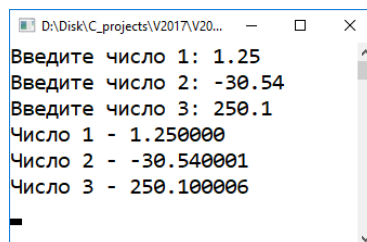
В ходе выполнения данной программы будет выведено стартовое окно для ввода первого числа с точкой, далее вводим первое число (1.25) и нажимаем **Enter** (рис. 5.6а). Далее, вводим второе число (-30.54) и вновь нажимаем **Enter** (рис. 5.6б). Далее, вводим третье число (-250.1) и вновь нажимаем **Enter** и все три введенных числа будут выведены на экран (рис. 5.6в).



а



б



в

Рис. 5.6. Программа ввода трех чисел с точкой и вывода их на экран

При этом все три числа будут сохранены в соответствующих переменных «*a*», «*b*» и «*c*», соответственно.

Пример 5.8. Пример программы ввода трех чисел с точкой и вывода их на экран. Например, осуществим ввод и вывод следующих чисел: 1.25, -30.54 и 250.1. при этом для ввода будем использовать одну функцию *scanf_s()*.

#include <stdio.h>	//подключение заголовочного файла
#include <stdlib.h>	//для перехода на русский язык
float a, b, c;	//объявление переменных типа float
void main()	//основной цикл
{	//основного цикла
system("chcp 1251");	//переходим в консоли на русский язык
system("cls");	//очищаем окно консоли
printf("Введите три числа: ");	//выводим в консоль фразу
scanf_s("%f,%f,%f", &a,&b,&c);	//считывание трех числа в a, b, c
printf("Число 1 - %f\n", a);	//вывод числа 1 с точкой
printf("Число 2 - %f\n", b);	//вывод числа 2 с точкой
printf("Число 3 - %f\n", c);	//вывод числа 3 с точкой
getchar(); getchar(); getchar();	
}	//конец основного цикла

После запуска программы на экран будет выведена фраза «Введите три числа:». Далее необходимо ввести три числа разделяя их запятыми. При этом введенные числа будут сохранены в формате плавающей точки в соответствующие переменные (рис. 5.7а). При нажатии клавиши **Enter** на экран будут выведены все три числа (рис. 5.7б).

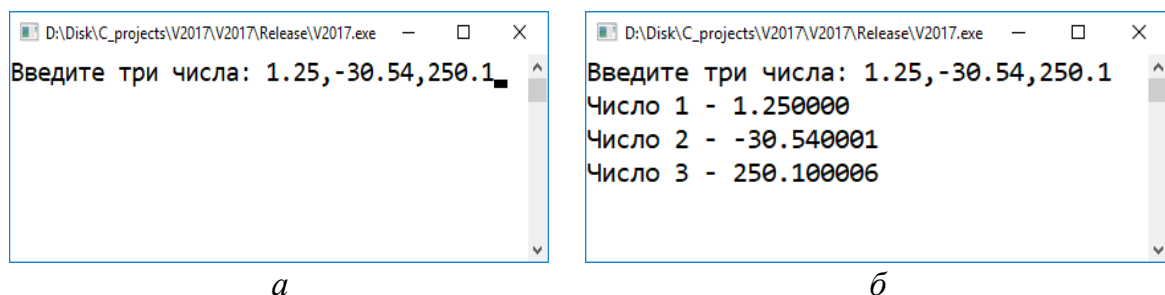


Рис. 5.7. Программа ввода трех чисел с точкой и вывода их на экран

5.4. Функция `get s()`

Для считывания строки символов, например, какого-либо предложения или слова, необходимо задать следующую переменную:

char имя переменной[N];

где N – количество символов в строке. При этом количество символов можно задавать больше, в этом случае просто будет зарезервировано больше ячеек памяти. В ином случае можно потерять часть вводимой информации. Считывание строки будет осуществляться с помощью функции:

```
get_s (имя переменной);
```

Пример 5.9. Пример программы ввода фамилии и дальнейшего ее вывода на экран. Ввод осуществляется в символьную переменную «a». Для вывода строки в функции *printf* используется формат %s.

```
#include <stdio.h>           //подключение заголовочного файла
#include <stdlib.h>           //для перехода на русский язык
```

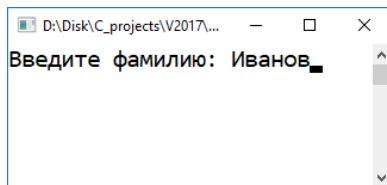
```

char a[20]; //объявление переменных типа char

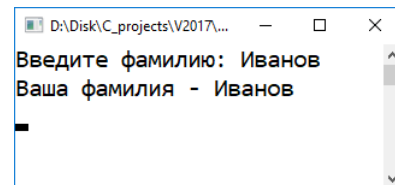
void main() //основной цикл
{
    system("chcp 1251"); //начало основного цикла
    system("cls"); //переходим в консоли на русский язык
    printf("Введите фамилию: "); //выводим в консоль фразу
    gets_s(a); //функция ввода строки символов в a
    printf("Ваша фамилия - %s\n", a); //вывод введенной строки на экран
    getchar();
} //конец основного цикла

```

В ходе выполнения кода программы будет выведена фраза «Введите фамилию:», вводим фамилию (рис. 5.8а). После ввода фамилии нажимаем клавишу **Enter**. Далее будет выведена фраза «Ваша фамилия - » и фамилия (рис. 5.8б).



а



б

Рис. 5.8. Программа ввода фамилии и дальнейшего ее вывода на экран

Пример 5.10. Пример программы ввода фамилии, имени и отчества, и вывода их в одной строке. Для вывода строки ФИО без запятых в функции **printf_s** используется форматы **%s** также не разделенные запятыми. В функции автоматически происходит привязка формата к соответствующей переменной (по порядку).

```

#include <stdio.h> //подключение заголовочного файла
#include <stdlib.h> //для перехода на русский язык

char fam[20]; //объявление переменных типа char
char name[20]; //объявление переменных типа char
char otch[20]; //объявление переменных типа char

void main() //основной цикл
{
    system("chcp 1251"); //начало основного цикла
    system("cls"); //переходим на русский язык
    printf("Введите фамилию: "); //выводим в консоль фразу
    gets_s(fam); //функция ввода строки в fam
    printf("Введите имя: "); //выводим в консоль фразу
    gets_s(name); //функция ввода строки в name
    printf("Введите отчество: "); //выводим в консоль фразу
    gets_s(otch); //функция ввода строки в otch
    printf("Ваше ФИО - %s %s %s\n", fam, name, otch); //вывод на экран
    getchar();
} //конец основного цикла

```

При выполнении кода программы сначала необходимо будет ввести фамилию и нажать **Enter** (рис. 5.9а). Далее вводим имя и отчество (рис. 5.9б-в). После нажатия клавиши **Enter** на экран будет выведена фраза «Ваше ФИО – Фамилия Имя Отчество» (рис. 5.9г).

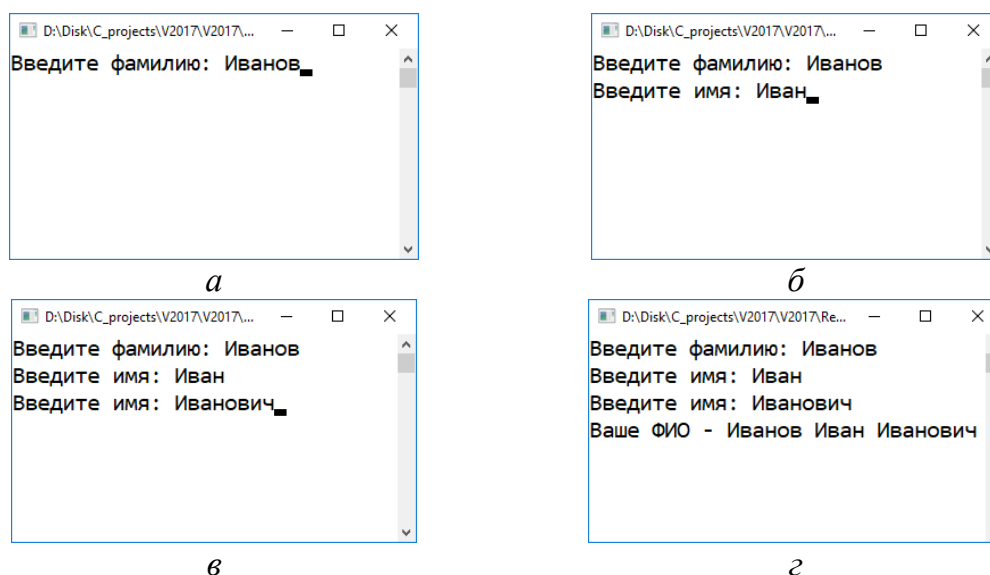


Рис. 5.9. Программа ввода фамилии, имени и отчества, и вывода их в одной строке