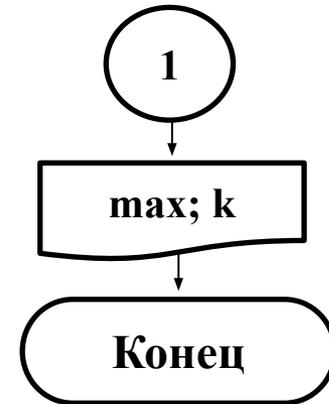
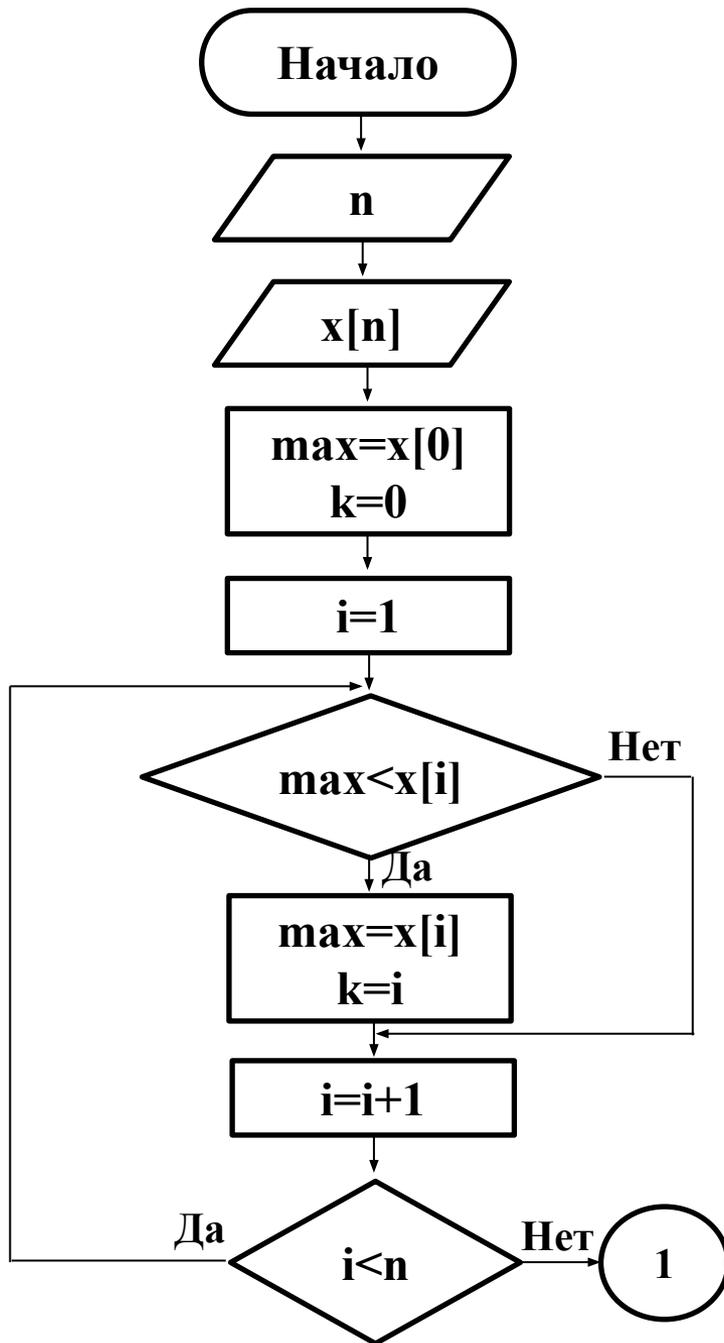


Нахождение наибольшего и наименьшего значения

Найти максимальный элемент и его порядковый номер в заданном массиве целых чисел $x[n]$.

Обозначения:

- $x[n]$ - массив целых чисел;
- n - количество элементов;
- i - номер элемента;
- \max - максимальный элемент;
- k - номер максимального элемента.



Алгоритмический язык Си++

Краткая история развития C

- Автор – Деннис Ритчи (Dennis Ritchie).
- Компьютер – DEC PDP-11 под управлением ОС Unix (1970-е гг.)
- Проектировался как рабочий инструмент для системного программиста - разработчика операционной системы UNIX и нацеливался автором на создание хорошо структурированных программ.

Краткая история развития С

- Основа – язык **BCPL** (1960-е гг.) Мартина Ричардса (Martin Richards), предназначенный для написания компиляторов для других языков.
- Оказал влияние на язык **B** Кена Томпсона (Ken Thompson). Является потомком BCPL и непосредственным предшественником С.
- Развитие языка B привело к созданию непосредственно языка **C** (1970 г.)

Краткая история развития C

- К 1973 году язык Си стал достаточно силён, и большая часть ядра UNIX, первоначально написанная на ассемблере PDP-11/20, была переписана на Си.
- Одно из самых первых ядер операционных систем, написанное на языке, отличном от ассемблера.

Краткая история развития C

- В 1983 году Американский национальный институт стандартов сформировал комитет X3J11 для создания спецификации стандарта C.
- В 1989 году, после длительного и напряжённого процесса, стандарт был завершён и утверждён как ANSI X3.159-1989 «Язык программирования C».
- Эту версию часто называют «ANSI C», или «C89».

C++

- Создатель – Бьёрн Страуструп (начало 1980-х г.)
- В C были добавлены классы, наследование классов, строгая проверка типов, inline-функции и аргументы по умолчанию.
- Первый компилятор Cfront от Bell Laboratory.
- Ранние версии языка, первоначально именовавшегося «C with classes» («Си с классами»), стали доступны с 1980 года.

C++

- 1983 г. – название «Си с классами» заменено на C++: добавлены новые возможности, такие как виртуальные функции, перегрузка функций и операторов, ссылки, константы, пользовательский контроль над управлением свободной памятью, улучшенная проверка типов и новый стиль комментариев.
- В 1998 году - опубликован стандарт языка ISO/IEC 14882:1998 (известный как C++98).

Краткая характеристика языка C++

Место С++ в мире языков

Языки высокого уровня	Языки среднего уровня	Языки низкого уровня
Ада	С	Ассемблер
Модула-2	С++	
Паскаль	FORTH	
COBOL	Макро Ассемблер	
Фортран		
Бейсик		

Место C++ в мире языков

- Определение C как языка среднего уровня означает, что он объединяет элементы языков высокого уровня с функциональностью ассемблера.
- Как язык среднего уровня, C++ позволяет манипулировать битами, байтами и адресами — основными элементами, с которыми работает компьютер.
- Код C++ имеет высокую степень переносимости.

Примеры программ

```
/* Пример 1 */  
# include <iostream>  
# include <locale>  
# include <conio.h>  
using namespace std;  
int main()  
{  
    setlocale(LC_ALL, "rus");  
    //clrscr();  
    system("cls");  
  
    cout <<"Hallo!" << endl;  
    cout << "Мы начинаем изучать язык Си++ " << endl;  
  
    while (!kbhit());  
    return 0;  
}
```

Пример 2

Напечатать таблицу температур по Фаренгейту в диапазоне от 0 до 300 с шагом 20 и их эквивалентов по стоградусной шкале Цельсия, используя для перевода формулу

$$c = 5/9 * (f - 32)$$

Таблица имен

- f1 – начальное значение температуры по Фаренгейту;
- f2 - конечное значение температуры по Фаренгейту;
- step – шаг изменения температуры по Фаренгейту;
- fahr – текущее значение температуры по Фаренгейту;
- cel - текущее значение температуры по Цельсию.

// Пример 2. Печать таблицы температур Фаренгейт - Цельсий

```
# include <iostream>
# include <locale>
# include <conio.h>
using namespace std;
int main()
{
    int f1, f2, step;
    float fahr, cel;
    //clrscr();
    system("cls");
    f1=0;
    f2=300;
    step=20;
    fahr=f1;
    while (fahr<=f2)
    {
        cel=(5.0/9.0)*(fahr-32.0);
        cout << "    " << fahr << "    " << cel << endl;
        fahr=fahr+step;
    }
    while (!kbhit());
    return 0;
}
```

Алфавит языка

Алфавит языка

- Алфавит языка — это тот набор символов (знаков), который допустим в данном языке.

Алфавит языка C++ включает в себя:

- прописные и строчные латинские буквы;
- арабские цифры 0 – 9;
- специальные знаки “ { } , | [] () + - / % * . \ < > = & # и др.

Типы, операции, выражения

Имя или идентификатор

- **Имена**, используемые для переменных, функций, меток и других определяемых пользователем объектов, называются **идентификаторами**.
- Первым символом должна быть буква или знак подчеркивания, а за ним могут стоять буквы, числа или знак подчеркивания.

Имя или идентификатор

Корректные	Некорректные
count	{count
test123	+test123
high_balance	High.,balance

Имя или идентификатор

- C++ - регистрозависимый язык - прописные и строчные буквы трактуются по-разному.
- kwadro, KWADRO, kwAdro – это разные имена.

Ключевые слова C++

- Ключевые слова — это предварительно определенные зарезервированные идентификаторы, имеющие специальные значения.
- Их использование в программе в качестве идентификаторов не допускается.

Ключевые слова C++

asm; auto; bool; break; case; catch; char; class;
const; const_cast; continue; default; delete; do;
double; dynamic_cast; else; enum; explicit;
export; extern; false; float; for; friend; goto;
if; inline; int; long; mutable; namespace; new;
operator; private; protected; public; register;
reinterpret_cast; return; short; signed; sizeof;
static; static_cast; struct; switch; template; this;
throw; true; try; typedef; typeid; typename;
union; unsigned; using; virtual; void; volatile;
wchar_t; while.

Типы данных

Типы данных

- Тип данных – множество значений, набор операций, которые можно применять к таким значениям и, возможно, способ реализации хранения значений и выполнения операций.
- Любые данные, которыми оперируют программы, относятся к определенным типам.

Типы данных

- Две группы:
 - основные;
 - составные.

Основные (или стандартные) ТИПЫ ДАННЫХ

- `int` - целый;
- `char` - символьный;
- `wchar_t` – расширенный символьный;
- `bool` – логический;
- `float` - вещественный;
- `double` – вещественный с двойной
точностью;
- `void` - пустой, не имеющий значения.

Спецификаторы типа

- - short –короткий;
- - long - длинный;
- - signed - знаковый;
- - unsigned – беззнаковый.

Допустимые комбинации типов и спецификаторов

Тип переменной	Длина в байтах
Char	1
Unsigned char	1
Signed char	1
Int	2
Unsigned int	2
Signed int	2
Short int	2
Unsigned short int	2
Signed short int	2
Long int	4

Допустимые комбинации типов и спецификаторов

Тип переменной	Длина в байтах
Signed long int	4
Unsigned long int	4
Float	4
Double	8
Long double	10

Составные типы данных

- массивы;
- перечисления;
- функции;
- структуры;
- ссылки;
- указатели;
- объединения;
- классы.

Целый тип (int)

Целый тип (int)

`signed` - старший разряд знаковый
0 – число положительное,
1 – число отрицательное

`unsigned` - беззнаковый

Примечание

Типы:

short int, long int, signed int, unsigned int

можно сокращать до:

short, long, signed, unsigned соотвественно.

СИМВОЛЬНЫЙ ТИП (char)

Кодировка символов

Однобайтная кодировка:

- код символа храниться в одном байте;
- возможно использование до 256 различных СИМВОЛОВ;
- используется код ASCII
(American Standard Code for Information Interchange – американский стандартный код для обмена информацией).

ОСНОВНЫЕ КОДЫ

32 Пробел

33 !

34 “

35 #

36 \$

37 %

38 &

39 *

40 (

41)

42 *

43 +

44 ,

45 -

46 .

47 /

Двухбайтная кодировка

- Unicode
- Символы могут принимать значения в диапазоне от 0 до 65 535.

Логический тип (bool)

Логический тип (bool)

- Величины логического типа могут принимать значения только:

true false

Вещественные типы (или типы с плавающей точкой)

float

double

long double

Диапазоны значений простых ТИПОВ для IBM PC

Тип	Диапазон значений	Размер (байт)
Bool	True , false	1
Signed char	-128 ... 127	1
Unsigned char	0 ... 255	1
Signed short int	-32 768 ... 32767	2
Unsigned short int	0 ... 65 535	2
Signed long int	-2 147 483 648 ... 2 147 483 647	4
Unsigned long int	0 ... 4 294 967 295	4
float	3.4e-38 ... 3.4e+38	4
Double	1.7e-308 ... 1.7e+308	8
Long double	3.4e-4932 ... 3.4e+4932	10

Переменные

Переменные

- Переменная - это именованное место хранения какого-то типа данных.
- Конкретное *значение* переменной хранится в ячейках оперативной памяти.

Формат объявления переменных

[класс памяти] [const] тип <список имён>
[инициализатор]

Значения класса памяти

- `auto` – создаются при входе в функцию и уничтожаются при выходе из нее;
- `extern` – глобальная переменная, доступна везде, где описана;
- `static` – если переменная объявлена внутри функции, то она сохраняет свое значение между вызовами функции;
- `register` – регистровая, рекомендует размещать переменную в регистрах процессора

Примеры

- `int k,l;`
- `short pi;`
- `double a,b,c;`
- `const char txt ='c';`
- `unsigned int p;`
- `int x=7;`
- `float q=0.5, f(2.0);`

/* Пример 4. Объявление переменных в разных местах программы.
Использование глобальных и локальных переменных и
формальных параметров. */

```
#include<conio.h>
#include<iostream>
#include <stdio.h>
using namespace std;
char ch; // глобальная переменная ch
int main()
{
    int n; // локальная переменная n
    cout << "Введите символ:";
    ch=getche(); // использование глобальной переменной
    ch
    cout (<< "Введите количество символов:";
    scanf("%d",&n);
    print_str(n);
    print_str(int m) // формальный параметр m
    {
        int j; // локальная переменная j
        for (j=0; j<m; j++)
            cout << ch; // глобальная переменная ch
    }
    return 0;
}
```

Константы

Константы

- Константа — способ адресации данных, изменение которых рассматриваемой программой не предполагается или запрещается.
- Виды констант:
 - целые;
 - вещественные;
 - символные;
 - строковые.

Виды констант

Константа	Примеры
Целая	
Формат десятичный	456; -12
Формат восьмеричный	06327
Формат шестнадцатеричный	0x4AF8
Вещественная	
Формат десятичный	123.45
Формат экспоненциальный	6.7e-3
Символьная	'a'; '\r'; '7'; 'ф'
Строковая	"\t аргумент функция"

Суффиксы констант

- Суффиксы целочисленных констант:

u, l, h, U, L, H

- Суффиксы чисел с плавающей точкой:

l, L, f, F

Примеры

Вид константы	Тип данного
12h; 34H	short int
23L; -123l	long int
37lu; 37Lu; 37ul; 9UL	unsigned long
77uh	unsigned short
12.34f; 67.7E-24F	float
1.39l; 12.0L2e+10	float double

Константы

- Шестнадцатеричная константа начинается с символов: 0x;
- Восьмеричная константа начинается с нуля.

```
int hex=0xff;  
int oct=023;
```

Константы

- “abba” - строковая константа;
- “a” - строковая константа;
- ‘a’ - СИМВОЛЬНАЯ КОНСТАНТА;

Управляющие константы (символьные константы с обратным слешем)

- `\n` перевод строки;
- `\r` возврат каретки;
- `\t` горизонтальная табуляция;
- и др.