Лабораторная работа № 1:

Многофайловые проекты

Оглавление

Цель:

Создать программу, разделенную на два модуля.

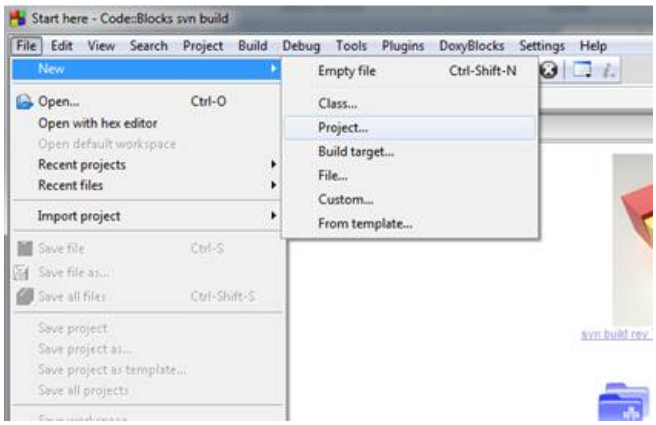
Инструкция:

Для того чтобы наиболее просто разбить программу на модули в Code::Blocks необходимо:

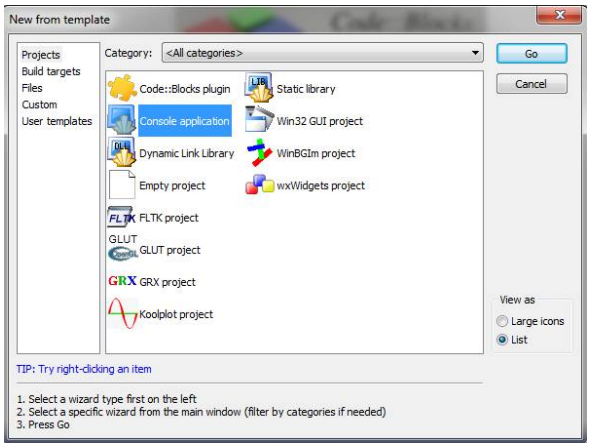
1. *Создатьконсольноеприложение.* ***File->New->Project->Console Application.****Назвать его* ***main****. Создастся проект, в котором подключен один cpp-файл, который называется main. Это будет главный модуль.*

a. *Открыть среду программирования CodeBlocks.*

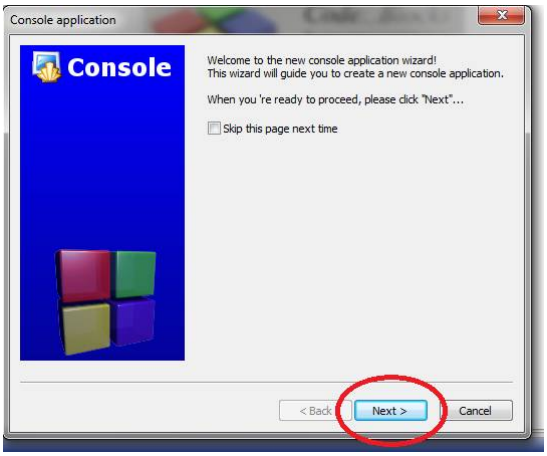
b.



c.

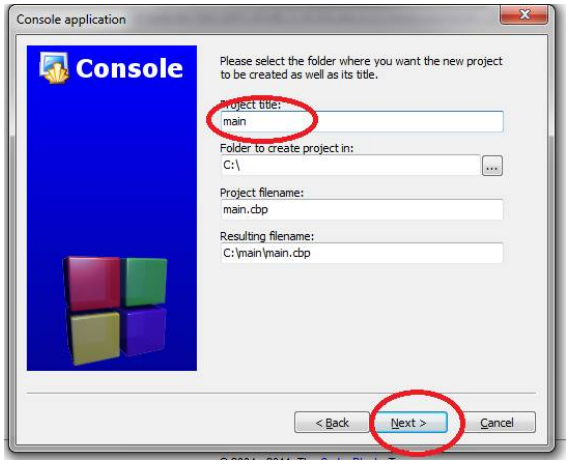


d.

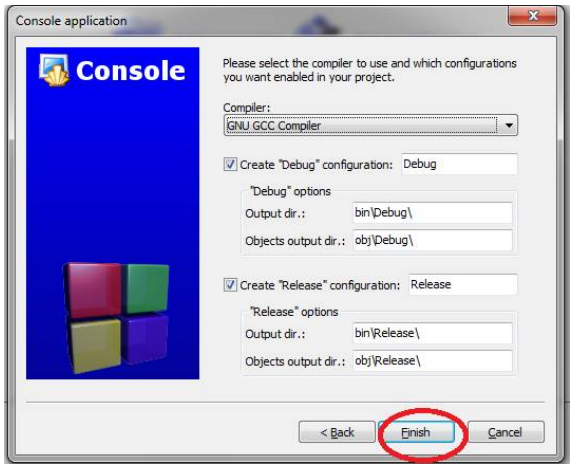


e.

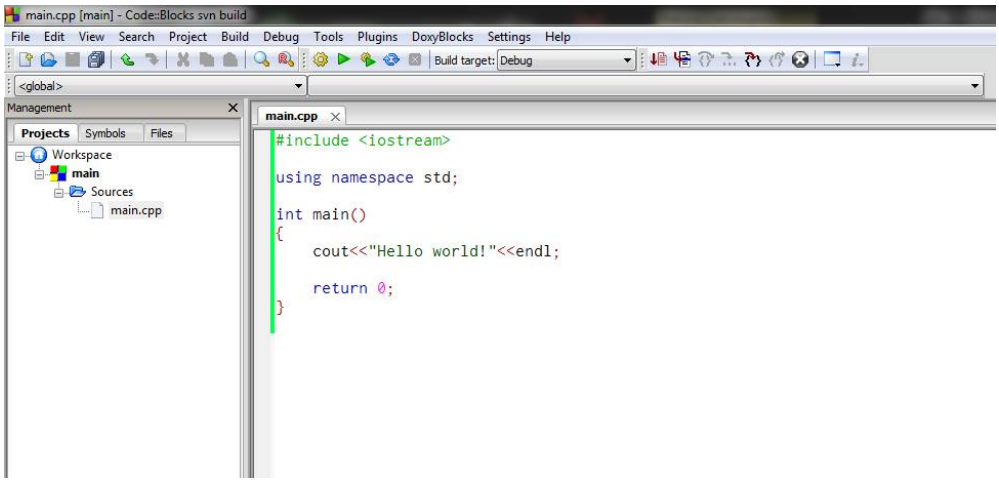
f.



g.

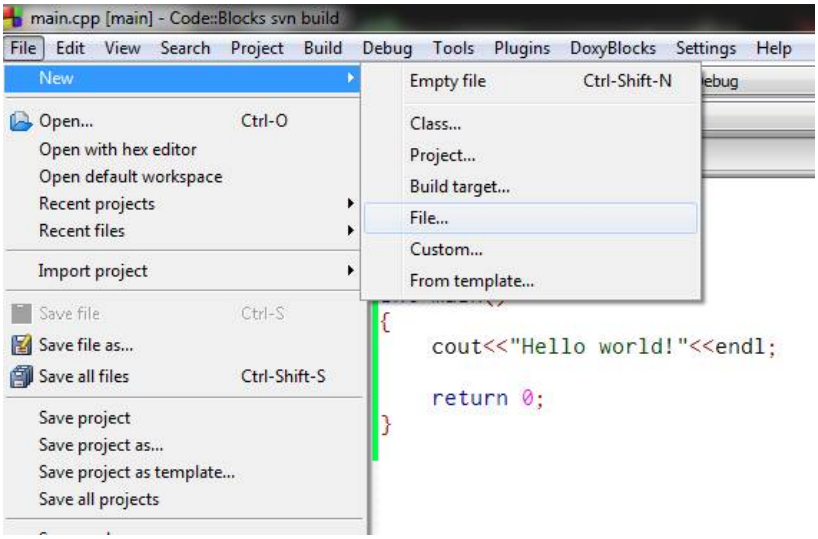


h.

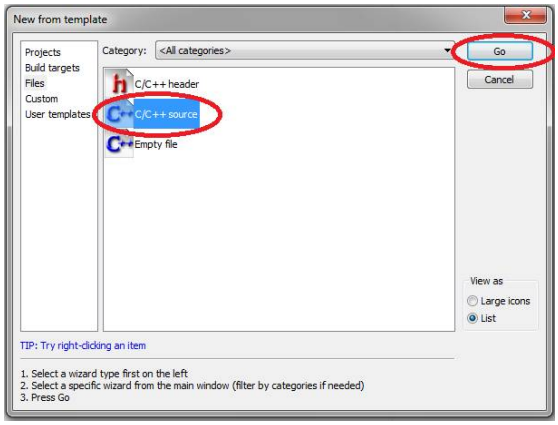


2. *Не выходя из проекта создать файлы для подключаемых модулей. Модулем будем называть пару файлов с одним названием и расширениями cpp и h. Для создания cpp-файла необходимо выполнить:* ***File->New->File->C/C++ Source****, а для создания h-файла:****File->New->File->Heder****. При создании обязательно ставить галочки* ***Debug****и* ***Realize****. После создания этих файлов, они должны появиться в дереве проекта. Это будет значить, что они будут совместно компилироваться со всем проектом. Если возникают проблемы с компоновкой, то необходимо еще раз подключить нужные cpp и h файлы к проекту:* ***Project->Addfiles.***

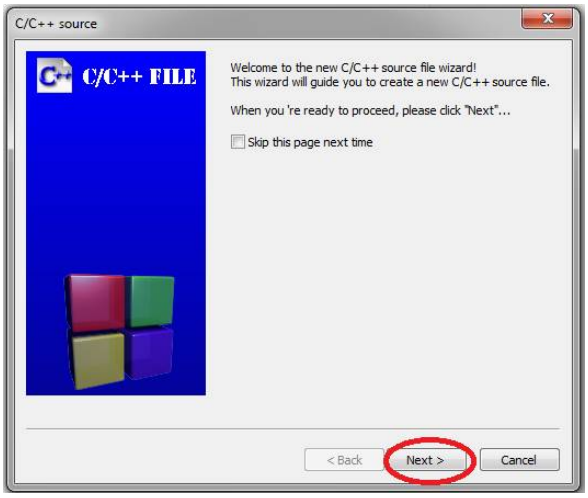
   a.


main.cpp [main] - Code::Blocks svn build
File Edit View Search Project Build Debug Tools Plugins DoxyBlocks Settings Help
New
Open... Ctrl-O
Open with hex editor
Open default workspace
Recent projects
Recent files
Import project
Save file Ctrl-S
Save file as...
Save all files Ctrl-Shift-S
Save project
Save project as...
Save project as template...
Save all projects
Empty file Ctrl-Shift-N
Class...
Project...
Build target...
File...
Custom...
From template...
cout<<"Hello world!"<<endl;
return 0;

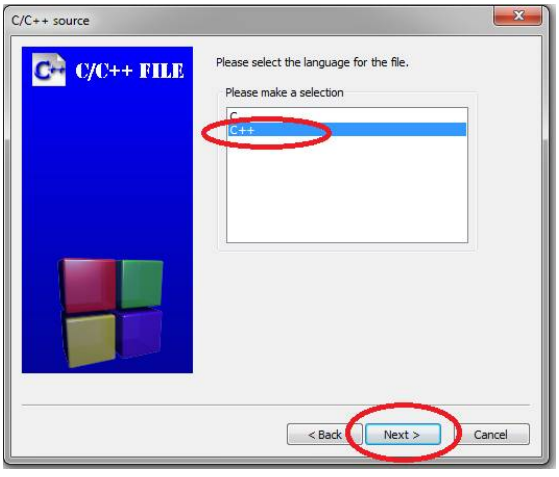
b.


New from template
Projects
Build targets
Files
Custom
User templates
Category: <All categories>
C/C++ header
C/C++ source
Empty file
Go
Cancel
View as
Large icons
List
TIP: Try right-clicking an item
1. Select a wizard type first on the left
2. Select a specific wizard from the main window (filter by categories if needed)
3. Press Go

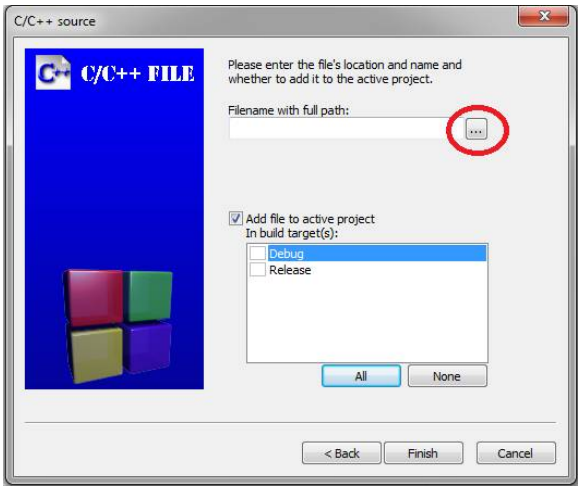
c.


C/C++ source
C/C++ FILE
Welcome to the new C/C++ source file wizard!
This wizard will guide you to create a new C/C++ source file.
When you 're ready to proceed, please click "Next"...
Skip this page next time
< Back
Next >
Cancel

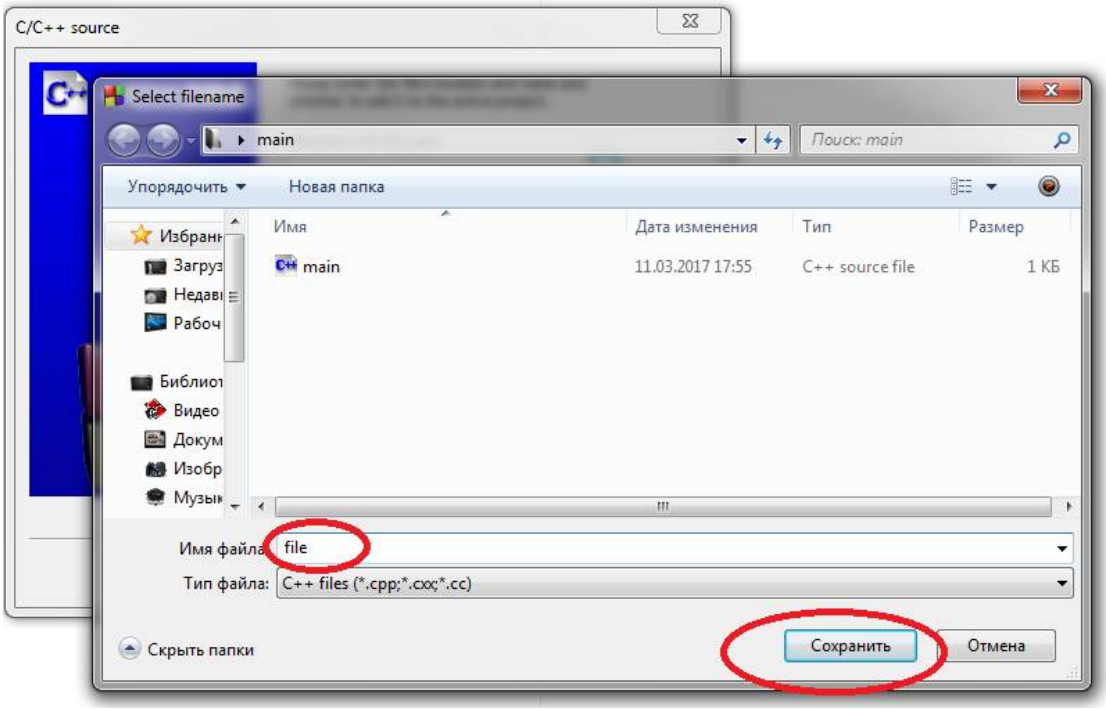
d.


C/C++ source
C/C++ FILE
Please select the language for the file.
Please make a selection
C
C++
< Back
Next >
Cancel

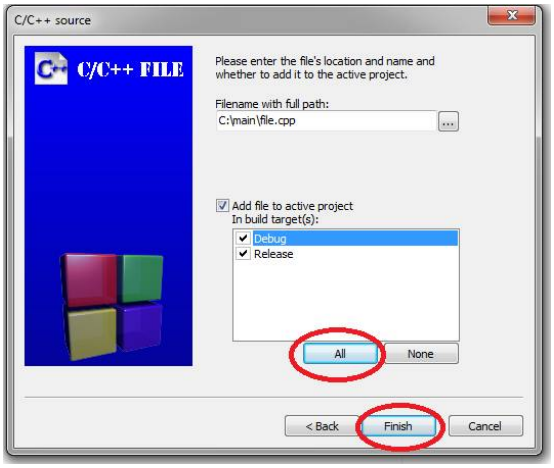
e.


C/C++ source
C/C++ FILE
Please enter the file's location and name and whether to add it to the active project.
Filename with full path:
...
Add file to active project
In build target(s):
Debug
Release
All
None
< Back
Finish
Cancel

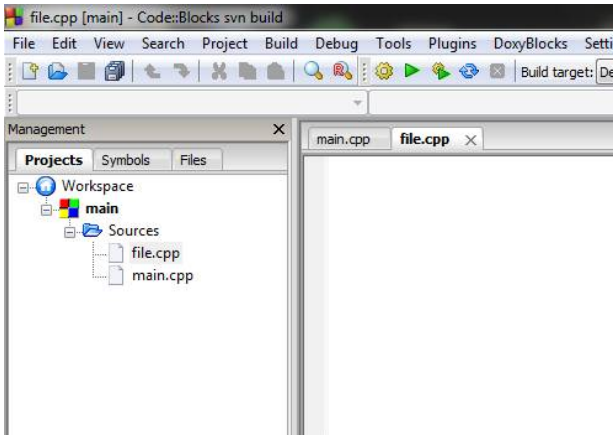
f.


C/C++ source
Select filename
main
Поиск: main
Упорядочить
Новая папка
Имя
Дата изменения
Тип
Размер
main
11.03.2017 17:55
C++ source file
1 КБ
Имя файла:
file
Тип файла:
C++ files (*.cpp;*.cxx;*.cc)
Скрыть папки
Сохранить
Отмена

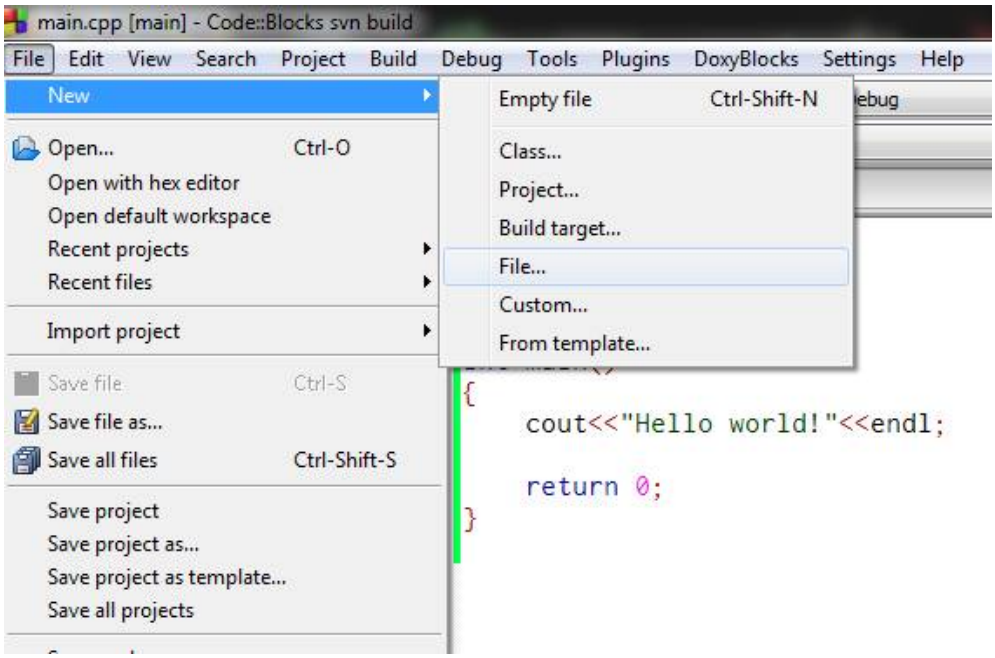
g.


C/C++ source
C/C++ FILE
Please enter the file's location and name and whether to add it to the active project.
Filename with full path:
C:\main\file.cpp
Add file to active project
In build target(s):
Debug
Release
All
None
< Back
Finish
Cancel

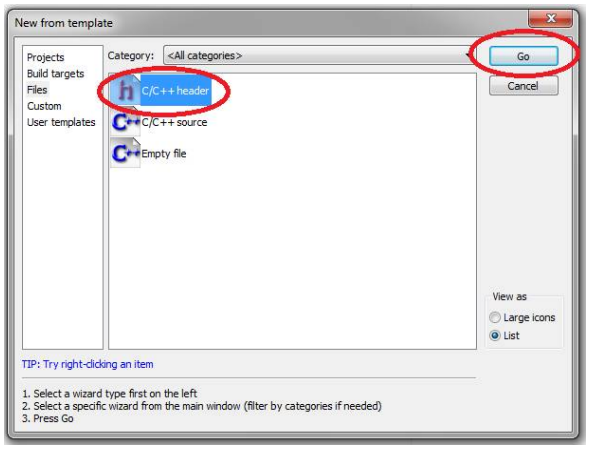
h.


file.cpp [main] - Code::Blocks svn build
File Edit View Search Project Build Debug Tools Plugins DoxyBlocks
Management
Projects
Symbols
Files
Workspace
main
Sources
file.cpp
main.cpp
main.cpp
file.cpp

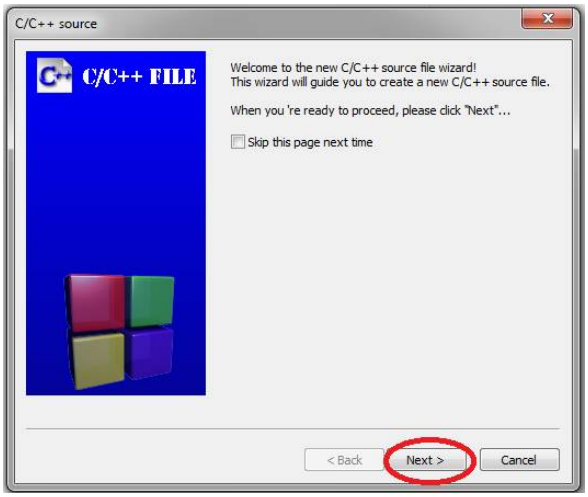
i.


main.cpp [main] - Code::Blocks svn build
File Edit View Search Project Build Debug Tools Plugins DoxyBlocks Settings Help
New
Open... Ctrl-O
Open with hex editor
Open default workspace
Recent projects
Recent files
Import project
Save file Ctrl-S
Save file as...
Save all files Ctrl-Shift-S
Save project
Save project as...
Save project as template...
Save all projects
Empty file Ctrl-Shift-N
Class...
Project...
Build target...
File...
Custom...
From template...
cout<<"Hello world!"<<endl;
return 0;

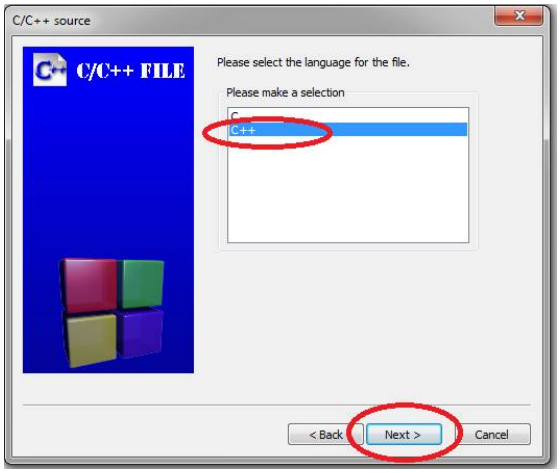
j.


New from template
Projects
Build targets
Files
Custom
User templates
Category:
<All categories>
Go
Cancel
C/C++ header
C/C++ source
Empty file
View as
Large icons
List
TIP: Try right-clicking an item
1. Select a wizard type first on the left
2. Select a specific wizard from the main window (filter by categories if needed)
3. Press Go

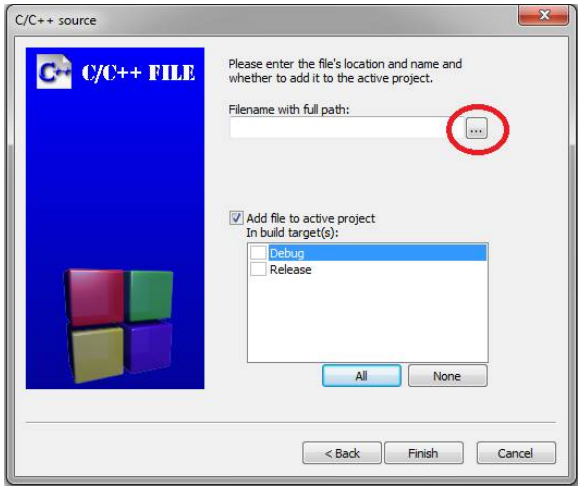
k.


C/C++ source
C/C++ FILE
Welcome to the new C/C++ source file wizard!
This wizard will guide you to create a new C/C++ source file.
When you 're ready to proceed, please click "Next"...
Skip this page next time
< Back
Next >
Cancel

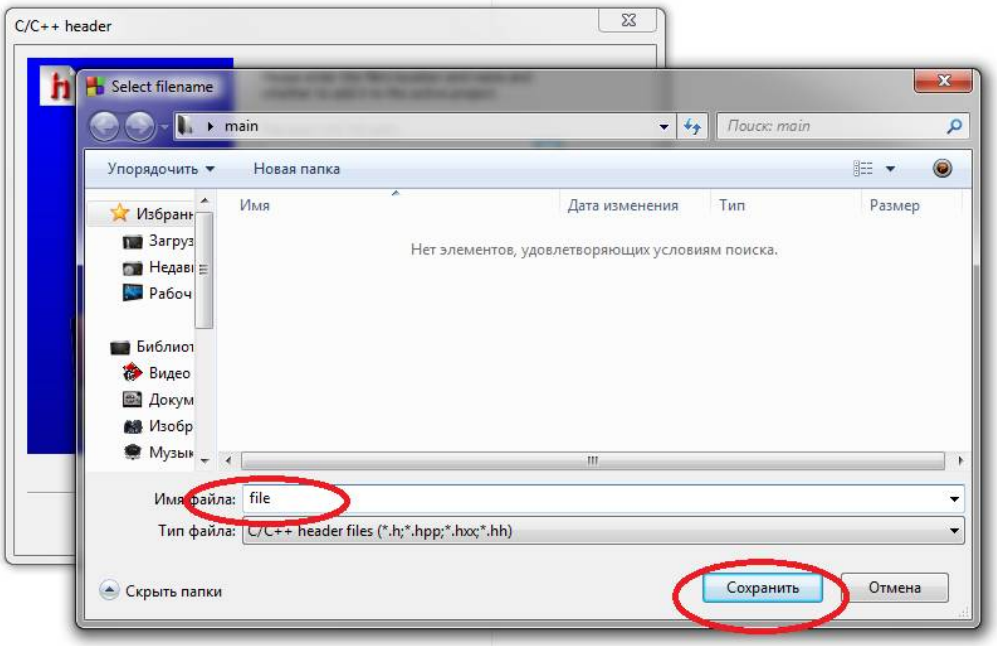
l.


C/C++ source
C/C++ FILE
Please select the language for the file.
Please make a selection
C++
< Back
Next >
Cancel

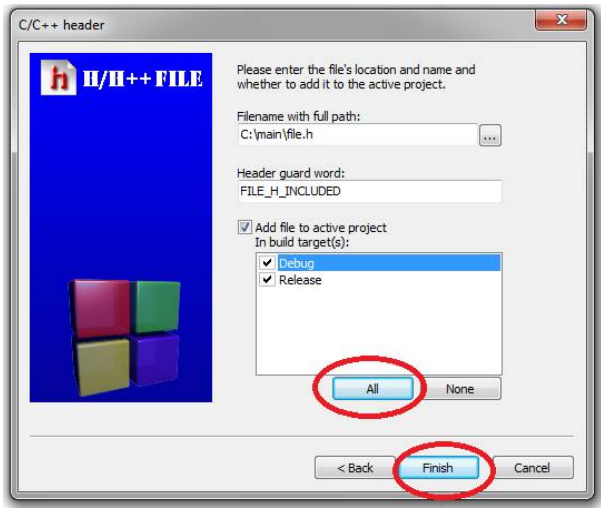
m


C/C++ source
C/C++ FILE
Please enter the file's location and name and whether to add it to the active project.
Filename with full path:
...
Add file to active project
In build target(s):
Debug
Release
All
None
< Back
Finish
Cancel

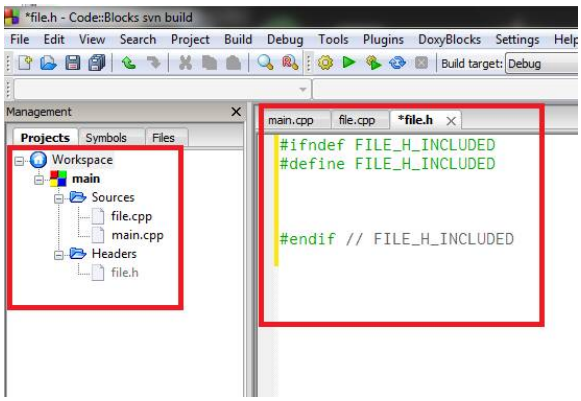
n.

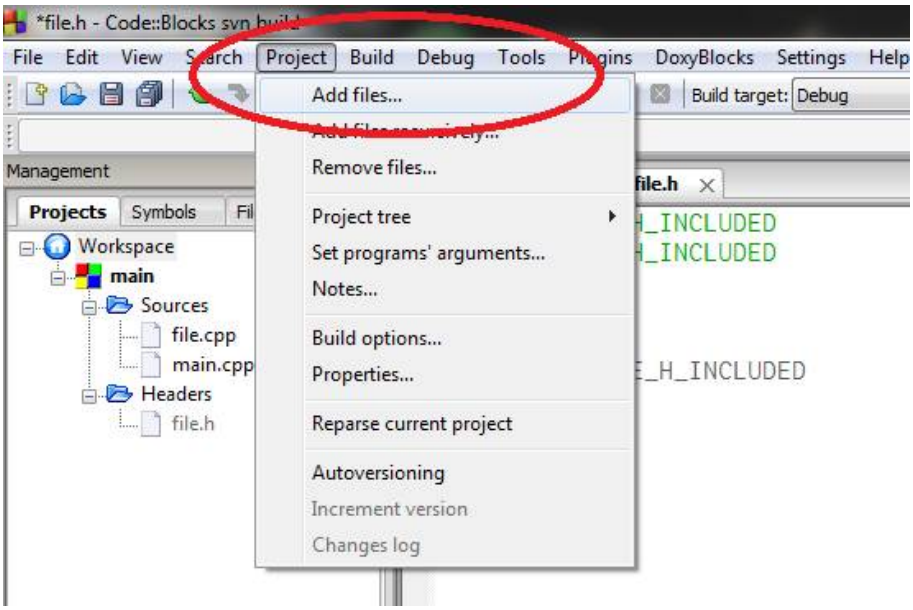

C/C++ header
Select filename
main
Поиск: main
Упорядочить
Новая папка
Имя
Дата изменения
Тип
Размер
Нет элементов, удовлетворяющих условиям поиска.
Имя файла:
file
Тип файла:
C/C++ header files (*.h;*.hpp;*.hxx;*.hh)
Скрыть папки
Сохранить
Отмена

o.

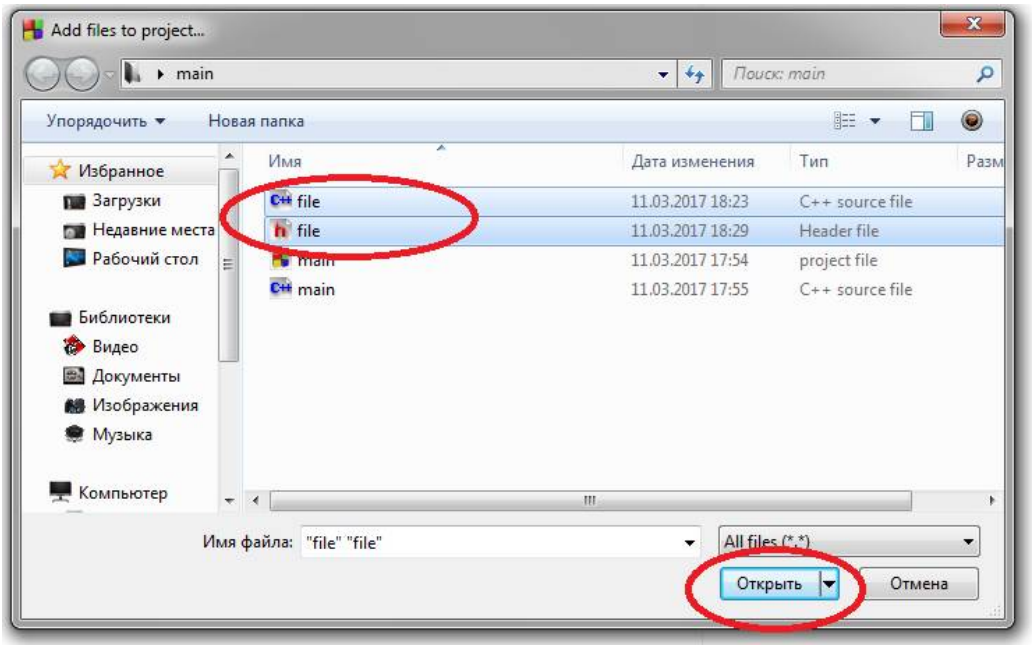

C/C++ header
H/H++ FILE
Please enter the file's location and name and whether to add it to the active project.
Filename with full path:
C:\main\file.h
Header guard word:
FILE_H_INCLUDED
Add file to active project
In build target(s):
Debug
Release
All
None
< Back
Finish
Cancel

p.

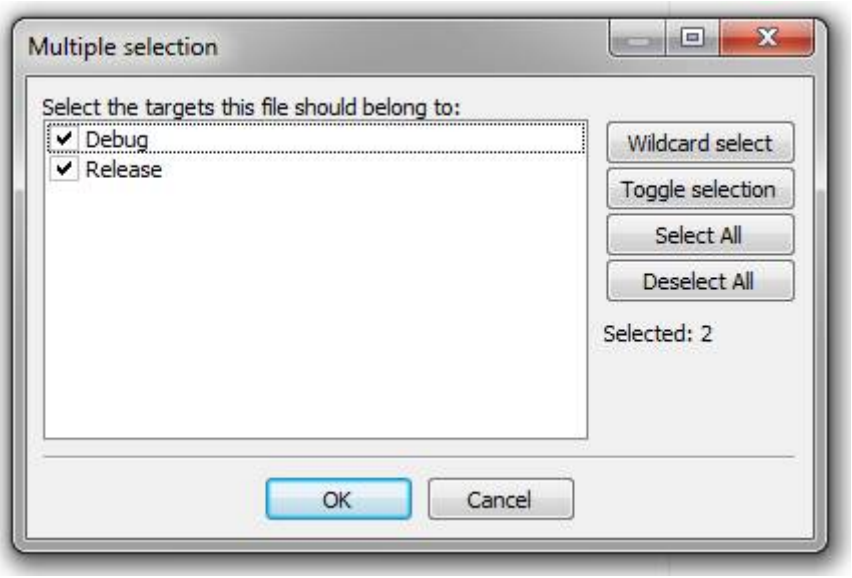

*file.h - Code::Blocks svn build
File Edit View Search Project Build Debug Tools Plugins DoxyBlocks Settings Help
Build target: Debug
Management
Projects Symbols Files
Workspace
main
Sources
file.cpp
main.cpp
Headers
file.h
main.cpp file.cpp *file.h
#ifndef FILE_H_INCLUDED
#define FILE_H_INCLUDED
#endif // FILE_H_INCLUDED


File Edit View Search Project Build Debug Tools Plugins DoxyBlocks Settings Help
Add files...
Remove files...
Project tree
Set programs' arguments...
Notes...
Build options...
Properties...
Reparse current project
Autoversioning
Increment version
Changes log
Build target: Debug
Management
Projects Symbols
Workspace
main
Sources
file.cpp
main.cpp
Headers
file.h

q.

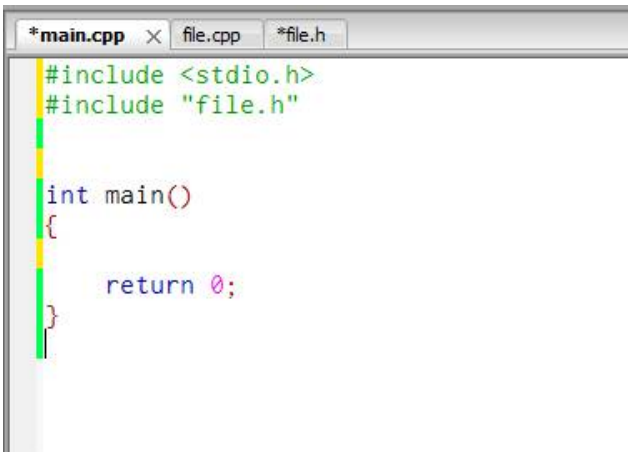

Add files to project...
main
Поиск: main
Упорядочить
Новая папка
Избранное
Загрузки
Недавние места
Рабочий стол
Библиотеки
Видео
Документы
Изображения
Музыка
Компьютер
Имя
Дата изменения
Тип
file 11.03.2017 18:23 C++ source file
file 11.03.2017 18:29 Header file
main 11.03.2017 17:54 project file
main 11.03.2017 17:55 C++ source file
Имя файла: "file" "file"
All files (*.*)
Открыть
Отмена

r.

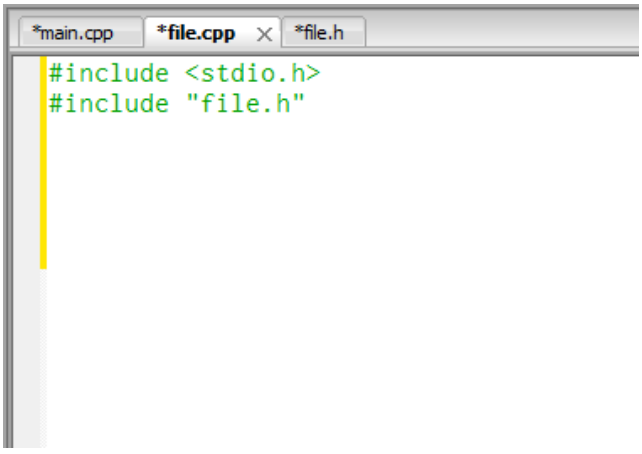


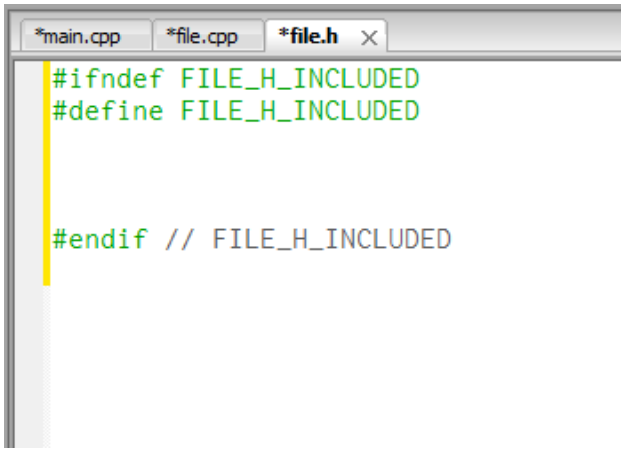
3. *Заполняем файлы подключаемого модуля: в cpp-файл переносим описания нужных функций, а в h-файл прототипы этих функций. h-файл необходимо подключать во всех cpp-файлах, в которых будут вызываться функции, прототипы которых в нем содержатся с помощью директивы вида* ***#include "file.h"****. Если название файла (в данном примере это file.h)пишется в двойных кавычках – то он будет искаться в той же папке, в которой находится проект. Вместо одного названия можно писать полный путь к файлу.*

a.



b.





4. *Аналогично можно создавать нужное количество модулей и подключать к проекту.*

5. *Далее заполняем необходимыми блоками кода подготовленные файлы.*

Задания для самостоятельного выполнения:

1. Создать проект согласно инструкции. Набрать представленный ниже код в соответствующие файлы проекта. Транслировать программу.

Вводить числа: 10 и 20.

Проанализировать результат.

main.cpp

```
#include <stdio.h>
#include "file.h"


int main()
{
    int A,B,res;

    printf("Enter first number, A = ");
    scanf("%d",&A);
    printf ("\n\n");

    printf("Enter second number, B = ");
    scanf("%d",&B);
    printf ("\n\n");

    res = Func(A,B);
    printf("A + B = %d + %d = %d", A,B,res);
    printf ("\n\n");
    return 0;
}
```

file.cpp

```
#include <stdio.h>
#include "file.h"


int Func(int a, int b)
{
    return a+b;
}
```

file.h

```
#ifndef FILE_H_INCLUDED
#define FILE_H_INCLUDED

int Func(int a, int b);

#endif // FILE_H_INCLUDED
```

2. Создать проект согласно инструкции. Набрать представленный ниже код в соответствующие файлы проекта. Транслировать программу.

Вводить числа: 1, 5, 5, 10, 5, 6, 7, -1

Проанализировать результат. Что выполняет программа?

*main.cpp | file.cpp | file.h

```c
#include <stdio.h>
#include "file.h"


int main()
{

    printf("\t Res = %d", Func());

    printf ("\n");

    return 0;
}
```

*main.cpp | file.cpp | *file.h

```c
#ifndef FILE_H_INCLUDED
#define FILE_H_INCLUDED

int Func();

#endif // FILE_H_INCLUDED
```

*main.cpp | file.cpp | *file.h

```c
#include <stdio.h>
#include "file.h"


int Func()
{
    int num=0, sum=0;

    printf("Enter number's, for stop  - press number < 0 (negative number) \n\n");

    scanf("%d",&num);
    printf ("\n");

    while(num>0)
        {
            if(num%5==0)
                {
                    sum+=num;
                }

            scanf("%d",&num);
            printf ("\n");

        }

    return sum;
}
```

Домашнее задание:

БЛОК А

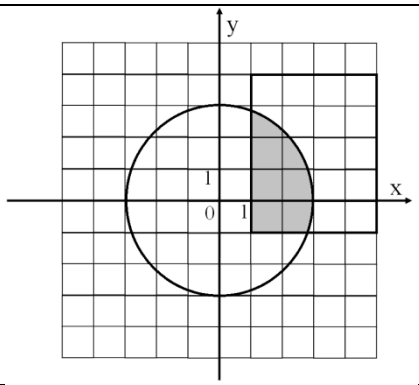
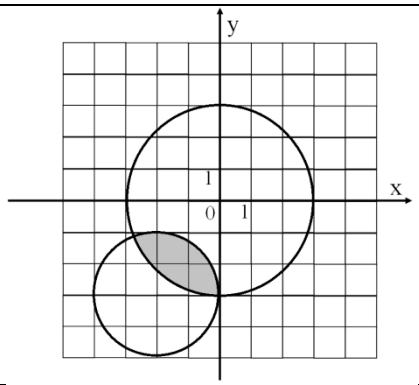
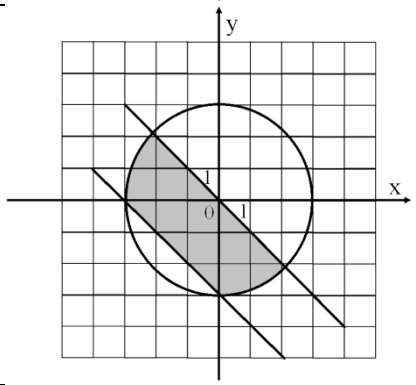
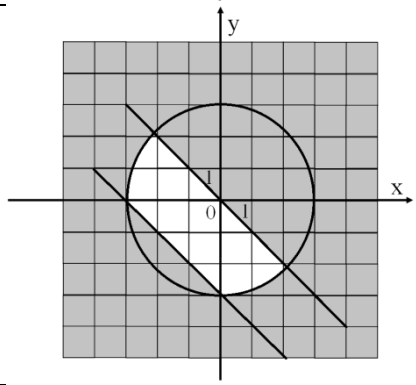
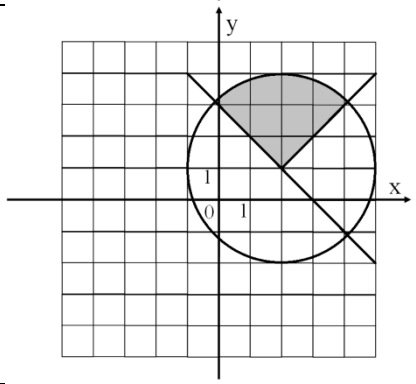
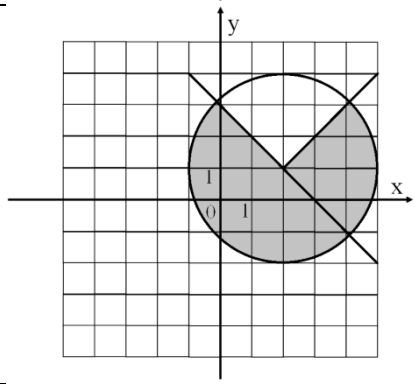
Задание на отметку «удовлетворительно»

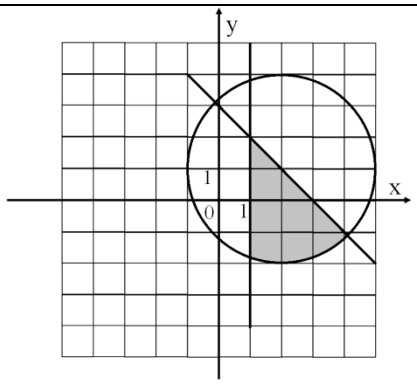
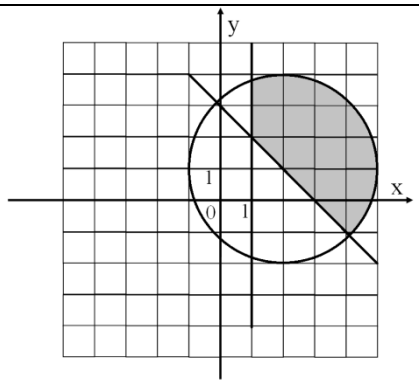
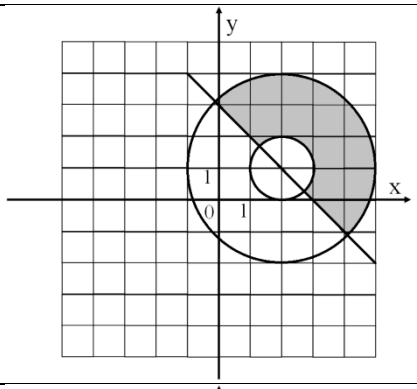
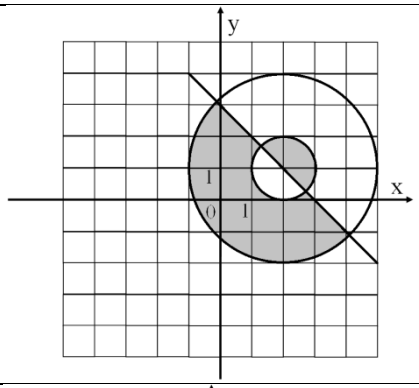
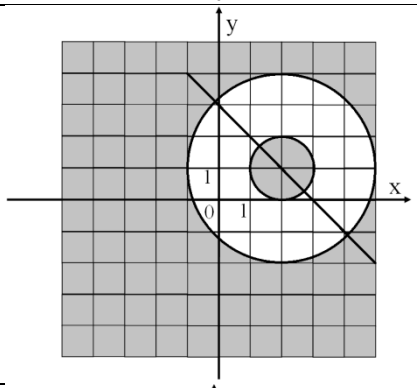
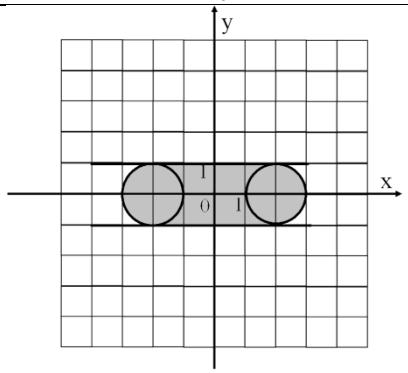
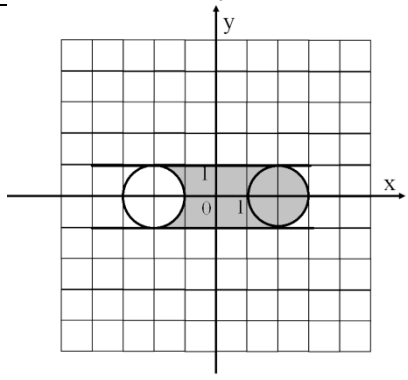
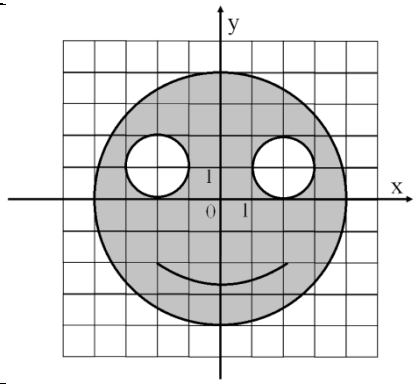
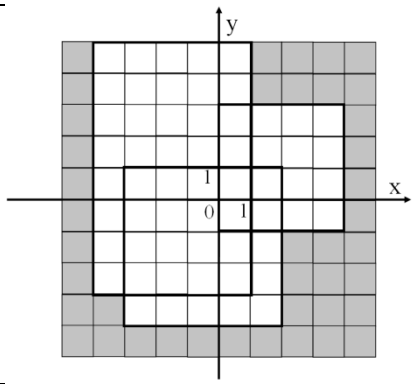
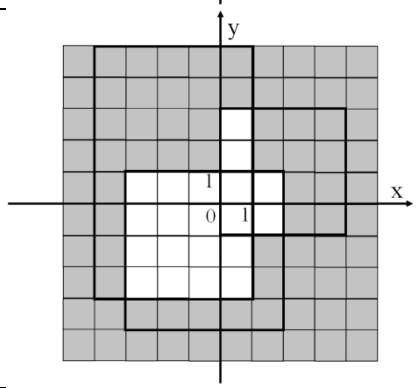
С экрана считываются целые числа, пока не будет введено положительное. Посчитать, сколько было введено кратных семи, отрицательных чисел. Результат выводить на экран.

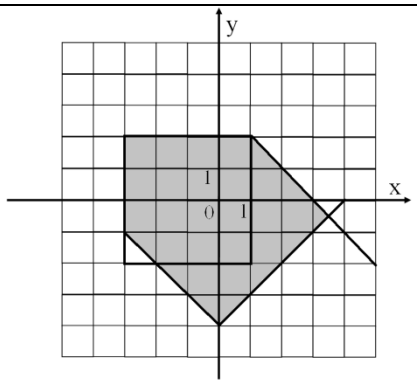
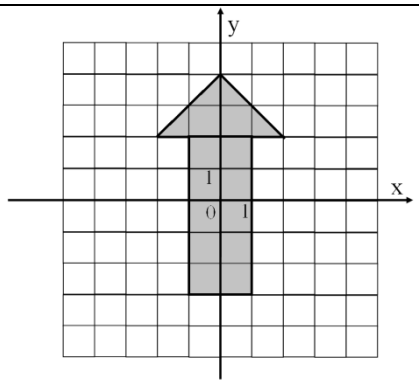
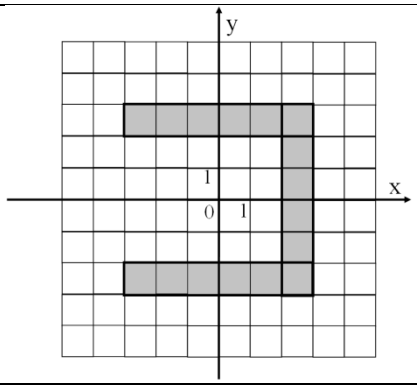
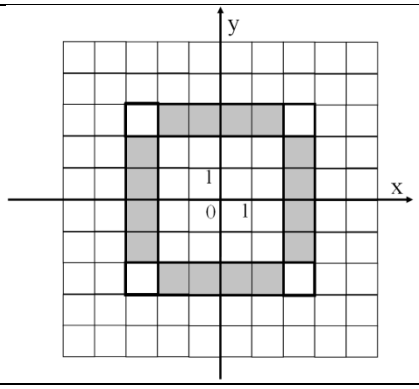
БЛОК В

Задания на отметку «хорошо»

Дана заштрихованная область и точка с координатами (x,y). Написать программу, определяющую лежит ли заданная точка в заштрихованной области:

1.		2.	
3.		4.	
5.		6.	

7.		8.	
9.		10.	
11.		12.	
13.		14.	
15.		16.	

17.		18.	
19.		20.	

БЛОК С

Задания на отметку «отлично»

1. Дана точка A(x,y). Определить, лежит ли она в треугольнике с вершинами в точках (x1,y1), (x2,y2), (x3,y3).
2. Найдите угол между двумя прямыми: a1 x + b1 y + c1=0 и a2 x + b2 y + c2=0.
3. Найти координаты точек пересечения прямой ax + b = y и окружности радиуса R с центром в начале координат.
4. Найти уравнение прямой, проходящей через точку M(x,y), перпендикулярно вектору $\bar{n}$=(A,B).
5. Дано уравнение плоскости ax+by+cz+d=0; определить, лежат ли точки M1(x1,y1,z1) и M2(x2,y2,z2) по одну сторону плоскости, если по разные, то определить координаты точки пересечения плоскости отрезком M1M2.
6. Даны две точки A(x1,y1) и B(x2,y2), найти на прямой AB точку M такую, чтобы она была расположена по ту же сторону от точки A, что и точка B, и чтобы отрезок AM был втрое больше отрезка AB.
7. Дана точка A(x1,y1). Найти точку B такую, что точка C пересечения прямой AB с осью ординат делит отрезок AB в отношении 2/3, а точка D пересечения прямой AB с осью абсцисс делит отрезок AB в отношении 3/4.
8. Даны три положительных числа. Определить, можно ли построить треугольник со сторонами, длины которых равны этим числам.
9. Заданы координаты вершин прямоугольника: (x1,y1), (x2,y2), (x3,y3), (x4,y4).Определить площадь части прямоугольника, расположенной в I координатной четверти.
10. Заданы координаты вершин прямоугольника: (x1,y1), (x2,y2), (x3,y3), (x4,y4).Определить площадь части прямоугольника, расположенной в II координатной четверти.
11. Заданы координаты вершин прямоугольника: (x1,y1), (x2,y2), (x3,y3), (x4,y4).Определить площадь части прямоугольника, расположенной в III координатной четверти.

12. Заданы координаты вершин прямоугольника: (x1,y1), (x2,y2), (x3,y3), (x4,y4).Определить площадь части прямоугольника, расположенной в IV координатной четверти.
13. Составить программу перевода натурального числа из десятичной системы счисления в двоичную.
14. Составить программу перевода натурального числа из двоичной системы счисления в десятичную.
15. Составить программу перевода натурального числа из десятичной системы счисления в восьмеричную.
16. Составить программу перевода натурального числа из восьмеричной системы счисления в десятичную.
17. Дано натуральное число n. Переставьте его цифры так, чтобы получилось максимальное число, записанное теми же цифрами.
18. Дано натуральное число n. Переставьте его цифры так, чтобы получилось минимальное число, записанное теми же цифрами.
19. Дано натуральное число k. Написать программу, которая выведет k-ую цифру последовательности 24610121416…, в которой выписаны подряд все натуральные четные числа.
20. Дано натуральное число k. Написать программу, которая выведет k-ую цифру последовательности 13579111315…, в которой выписаны подряд все натуральные нечетные числа.

Требования к оформлению программ:

1. **Содержание**. Программа должна делать то, что предусмотрено заданием. Не надо выполнять лишних действий, заданием не предусмотренных.
2. **Спецификация.** В преамбуле программы в комментариях указывать сведения:
    - Кто выполнил.
    - Что делает программа (кратко).
    - Что на входе (имена входных файлов и т.д.).
    - Что на выходе (что является результатов работы программы?).
3. **Ввод и вывод**
    - Приглашения к вводу (например, сколько чисел, какого типа и через какой разделитель нужно вводить).
    - Контрольный вывод (все введенные данные выводить на экран, и только после этого выполнять необходимые вычисления.)
    - «Защита от дурака». Проверять вводимые данные на корректность. Например, если необходимо считать количество чего – то, то эта величина не может быть отрицательной и т.д.
4. **Структура кода.** Набираемый код должен быть хорошо структурированным. Использовать:
    - Отступы.
    - Комментарии – поясняют решение программы.
    - Осмысленные названия переменных.
    - Пояснения о назначении переменных в комментариях (кроме счетчиков).
5. **Декомпозиция кода**
    - Функциональная. Программу оформлять с помощью функций.
6. **Многофайловые проекты**

- Все проекты должны состоять минимум из двух модулей: главного и подключаемого.

Главный модуль **main.cpp**. В нем оставить только функцию main, в которой вызывать функции, описанные в других модулях.

Модуль с описанием пользовательских функций **file.cpp** и **file.h.**В этот модуль перенести определение всех функций,необходимых для выполнения задания.

Контрольные вопросы:

1. Перечислите основные файлы консольного приложения.
2. Для чего нужен файл с расширением .cpp?
3. Для чего нужен файл с расширением .h?
4. Что такое компиляция?
5. Что такое компановка?
6. Что такое раздельная компиляция?