

第一部分

实验

实验 1

C 语言程序的运行环境和运行方法

1.1 实验目的

- (1) 了解 C 语言集成开发环境 CodeBlocks 的使用方法。
- (2) 了解在该开发环境中如何编辑、编译、连接和运行一个 C 语言程序。
- (3) 通过运行简单的 C 语言程序，初步了解 C 语言源程序的特点。

1.2 实验内容及步骤

1.2.1 任务一：C 语言开发环境 CodeBlocks 的下载、安装及配置

C 语言开发环境 CodeBlocks 可到其官方网站 www.codeblocks.org 下载，下载后的 CodeBlocks 为英文版，下载后双击安装文件，按照安装提示步骤即可实现安装。

安装完成后，在【开始】菜单中选择【所有程序】→【CodeBlocks】，可启动 CodeBlocks，如图 1-1 所示。

可对 CodeBlocks 进行汉化，步骤如下：

(1) 下载 CodeBlocks 的汉化包，将汉化包中的“zh_CN”文件夹复制到 CodeBlocks 的安装根目录\CodeBlocks\share\CodeBlocks\locale 下。如果没有该目录，则新建 locale 目录。

(2) 在 CodeBlocks 启动界面中选择【Setting】→【Environment】（见图 1-2），在打开的“Environment settings”对话框中选择【View】，选中【Internationalization】并单击【Chinese(Simplified)】按钮（见图 1-3），再单击【OK】按钮。

(3) 关闭 CodeBlocks 后，双击 CodeBlocks 图标重新启动即可，见图 1-4。

(4) CodeBlocks 编译环境的配置。在图 1-4 中，选择【设置】→【编译器】，打开“编译器设置”窗口（见图 1-5），在左侧选择【全局编译器设置】，在右侧单击【可执行工具链】，然后单击【自动检测】按钮，会在前面的地址栏中添加编译器的安装目录，之后单击【确定】按钮，即可完成配置。

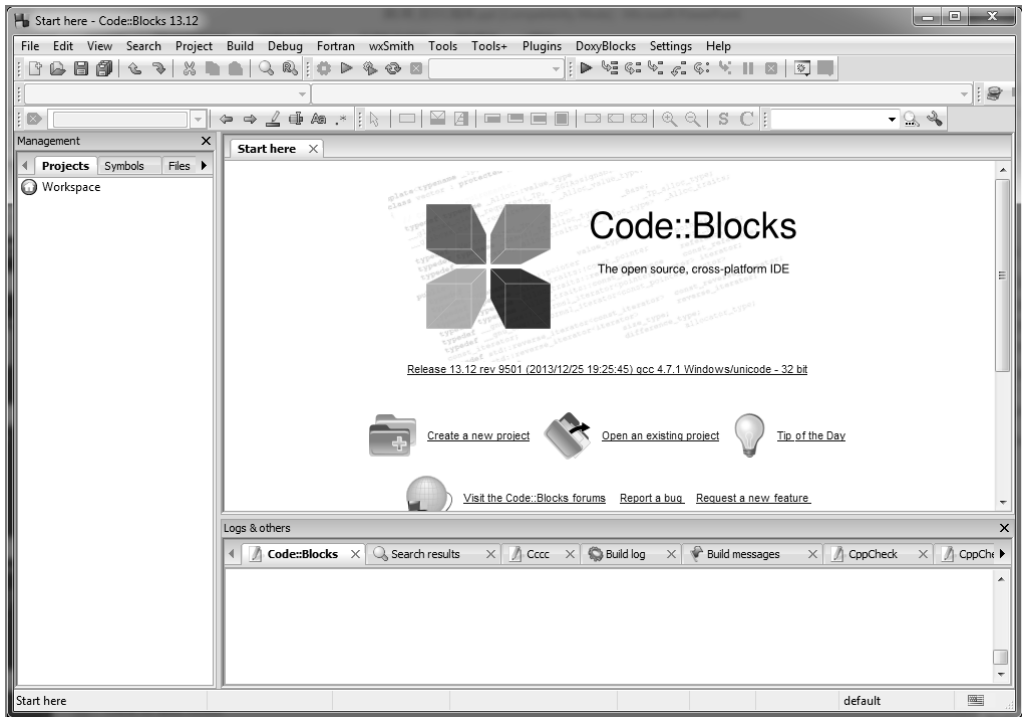


图 1-1 CodeBlocks 启动界面

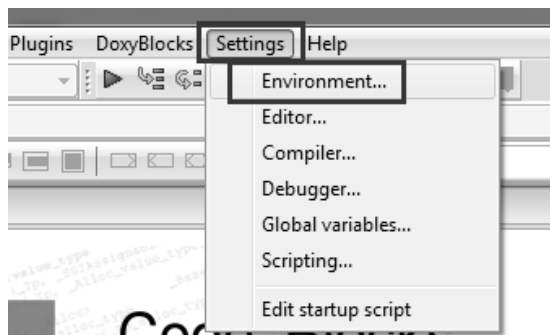


图 1-2 Settings → Environment

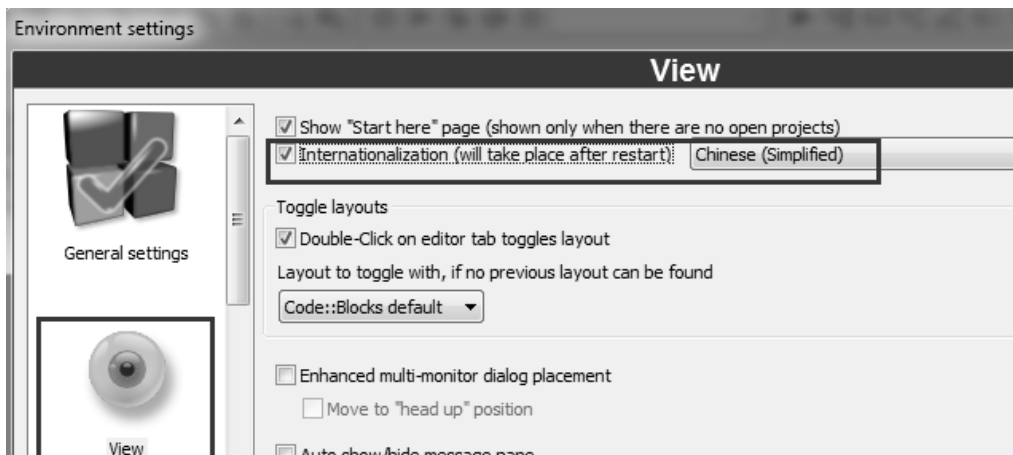


图 1-3 “Environment settings” 对话框

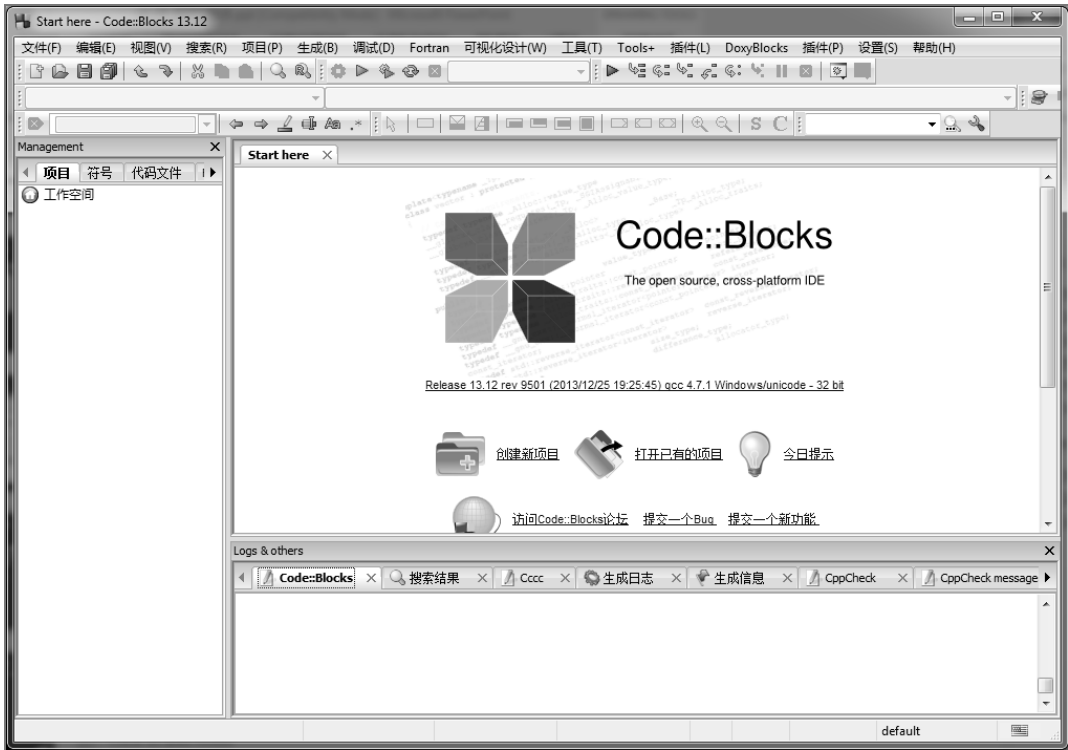


图 1-4 CodeBlocks 汉化界面



图 1-5 编译器设置

1.2.2 任务二：通过 C 语言示例，熟悉 CodeBlocks 的使用方法

在 CodeBlocks 中，按照以下步骤编写一个简单的 C 语言程序的源文件，然后运行程序、查看程序运行结果。

1. 创建工程

(1) 在 CodeBlocks 启动界面中单击【创建新项目】(见图 1-6)，或者单击工具栏中的【文件】→【新建】→【项目】，打开“新建项目”对话框，见图 1-7。

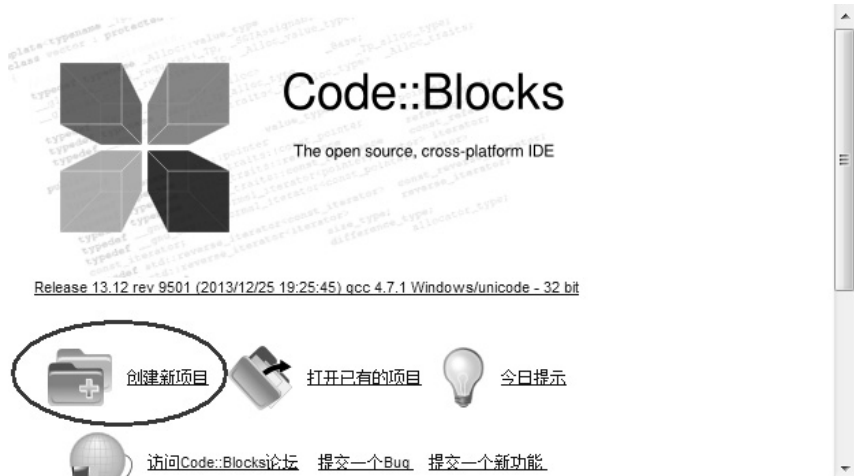


图 1-6 创建新项目



图 1-7 “新建项目”对话框

(2) 在图 1-7 中, 单击【Console application】, 再单击【前进】按钮, 打开“Console application”对话框, 见图 1-8。



图 1-8 “Console application”对话框

(3) 单击【下一步】按钮, 出现图 1-9, 选择编程语言“C”后单击【下一步】按钮。



图 1-9 选择编程语言

(4) 在弹出的对话框中填写项目名称和保存路径, 见图 1-10。

(5) 填写完成后, 单击【下一步】按钮, 出现如图 1-11 所示的对话框, 从中选择编译器, 并勾选【创建“调试/Debug”编译配置】和【创建“发行/Release”编译配置】。

(6) 单击【完成】按钮, 即可创建工程 CTest, 见图 1-12。



图 1-10 填写项目名称和保存路径



图 1-11 选择编译器



图 1-12 创建新工程 CTest

2. 向工程添加源文件（新创建的工程默认有一个 main.c 源文件）

(1) 在【文件】菜单中选择【新建】→【文件】，出现如图 1-13 所示的对话框，从中选择【C/C++ source】，然后单击【前进】按钮。

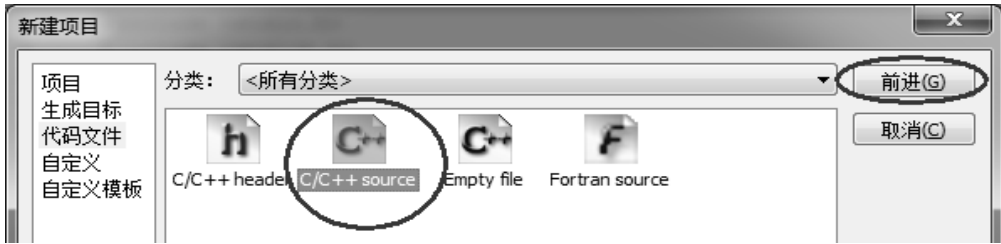


图 1-13 新建源文件

(2) 在出现的对话框中选择编程语言 C（见图 1-14），然后出现如图 1-15 所示的对话框。



图 1-14 选择编程语言



图 1-15 文件名

(3) 选择文件名“file1.c”，并勾选【将文件添加到当前活动项目包含以下生成目标】，

选中【Debug】和【Release】，单击【完成】按钮，即可将源文件 file1.c 添加到项目 CTest 中。

3. 编写源文件

双击 file1.c 源文件，在 CodeBlocks 中打开 file1.c 的程序编辑区，输入如下代码并保存。

```
#include <stdio.h>

int main() {
    printf("Hello, C 语言程序设计!\n");    //输出 "Hello, C 语言程序设计!", 并换行
    printf("I'm glad to meet you!\n");    //输出 "I'm glad to meet you!", 并换行
    getchar();
    return 0;
}
```

4. 编译和链接

单击【构建】按钮（见图 1-16）进行编译和连接。如果源文件没有错误，将产生目标文件；如果有错误，编译器将提示有关错误，见图 1-17。



图 1-16 编译和链接

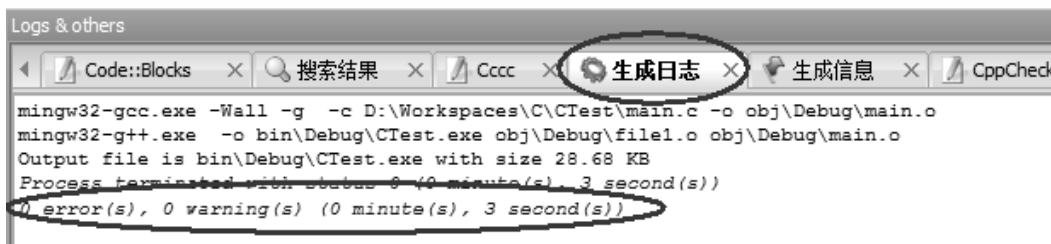


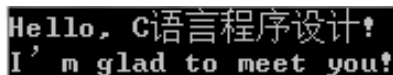
图 1-17 生成日志

5. 运行

在工具栏中单击【运行】按钮（见图 1-18），或在可执行文件所在目录中双击此可执行文件，即可运行该文件。运行结果见图 1-19。



图 1-18 运行



```
Hello, C语言程序设计!  
I'm glad to meet you!
```

图 1-19 运行结果

1.2.3 任务三：编写并运行程序 1_3

根据实验任务一和任务二中 CodeBlocks 的使用以及运行 C 程序的步骤和方法，按照如下步骤依次编写下面的源程序：（1）创建工程；（2）向工程添加源文件；（3）编写源文件；（4）编译；（5）链接；（6）运行。

运行并查看结果，如果程序有错，请根据提示信息修改。

源程序 1_3 如下：

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
    int num;                // 定义一个名为 num 的变量
    num = 1;                // 为变量 num 赋值
    printf("I am a student!\n");    // 使用输出函数
    printf("My favorite number is %d, because it is the first.\n",num);
    return 0;
}
```

1.2.4 任务四：编写并运行程序 1_4

根据实验任务一和任务二中 CodeBlocks 的使用以及运行 C 程序的步骤和方法，按照如下步骤依次编写下面的源程序：（1）创建工程；（2）向工程添加源文件；（3）编写源文件；（4）编译；（5）链接；（6）运行。

运行并查看结果，如果程序有错，请根据提示信息修改。

源程序 1_4 如下：

```
#include <stdio.h>
#include <stdlib.h>
void main( )                // 定义主函数
{
    // 主函数体开始
    int max(int x,int y);    // 对被调用函数 max()的声明
    int a,b,c;                // 定义变量 a, b, c
    printf("input a & b:");
    scanf("%d,%d",&a,&b);    // 输入变量 a 和 b 的值
    c = max(a,b);            // 调用 max()函数, 将得到的值赋给 c
    printf("max=%d\n",c);    // 输出 c 的值
}
int max(int x,int y)        // 定义 max()函数, 函数值为整型, 形式参数 x 和 y 为整型
{
```

```
// 函数体开始
int z;
if (x>y)
    z = x;
else
    z = y;
return(z);
}

// 定义本函数中用到的变量 z 为整型

// 将 z 的值返回, 通过 max() 带回到调用函数的位置
```