

第一、二章 引言

本章学习目标：

- ✓ 初步理解“程序”的含义。
- ✓ 初步理解 C 程序的基本结构。
- ✓ 初步认识 C 语言的单词集
- ✓ 理解标准与实现的关系。
- ✓ 初步理解编写 C 程序的四个阶段。
- ✓ 学会 Visual Studio 2012 的安装和使用。
- ✓ 初步掌握使用 Visual Studio 2012 调试 C 程序的方法。
- ✓ 能模仿样例在屏幕上打印信息。

1.1 上机实践题

一、安装 VS2012

实验目的

1. 掌握 VS2012 的安装方法及注意事项。

实验步骤

准备工作：安装前准备好 VS2012 安装包，检查系统配置是否符合以下要求。

- ✓ 支持的操作系统

Windows 7 (x86 和 x64)；

Windows 8 Release Preview 或更高版本 (x86 和 x64)；

Windows Server 2008 R2 (x64)；

Windows Server 2012 候选发布版本 (x64)。

- ✓ 硬件要求

1.6 GHz 或更快的处理器；


1 GB RAM (如果在虚拟机上运行，则为 1.5 GB)；

10 GB 的可用硬盘空间；

5400 RPM 硬盘；

以 1024 x 768 或更高显示分辨率运行的支持 DirectX 9 的视频卡。

步骤 1: 解压  VS2012_ULT_chs.iso

找到  vs_ultimate.exe

双击 vs2012 安装程序，启动 Visual Studio 2012 安装界面，如图 1-1 所示。

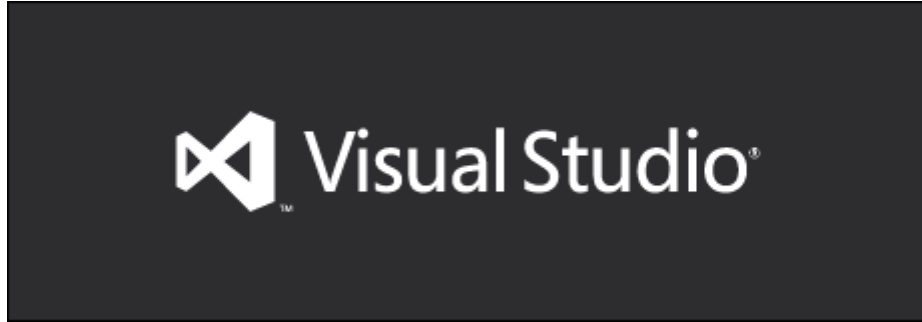


图 1-1 安装 Visual Studio 2012 启动界面

步骤 2: 稍等片刻，进入 Visual Studio 2012 安装界面，如图 1-2 所示。选择安装位置
和同意条款，点击【下一步】按钮。

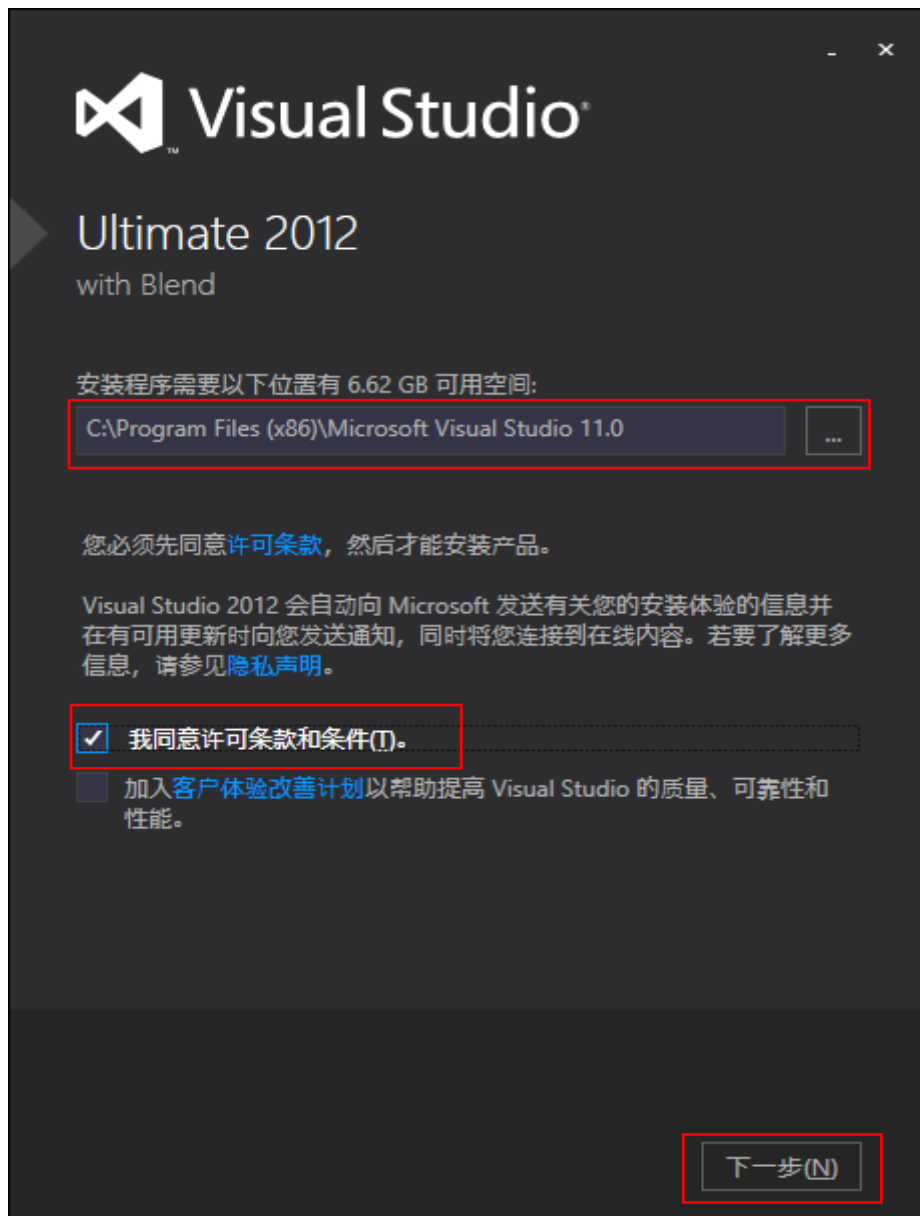


图 1-2 Visual Studio 2012 安装界面

步骤 3: 进入功能和组件选择界面, 如图 1-3 所示。选择要安装的功能组件, 默认全选, 点击【安装】按钮。



图 1-3 功能和组件选择界面

步骤 4：进入安装过程界面，如图 1-4 所示。该步骤时间较长，请耐心等待。

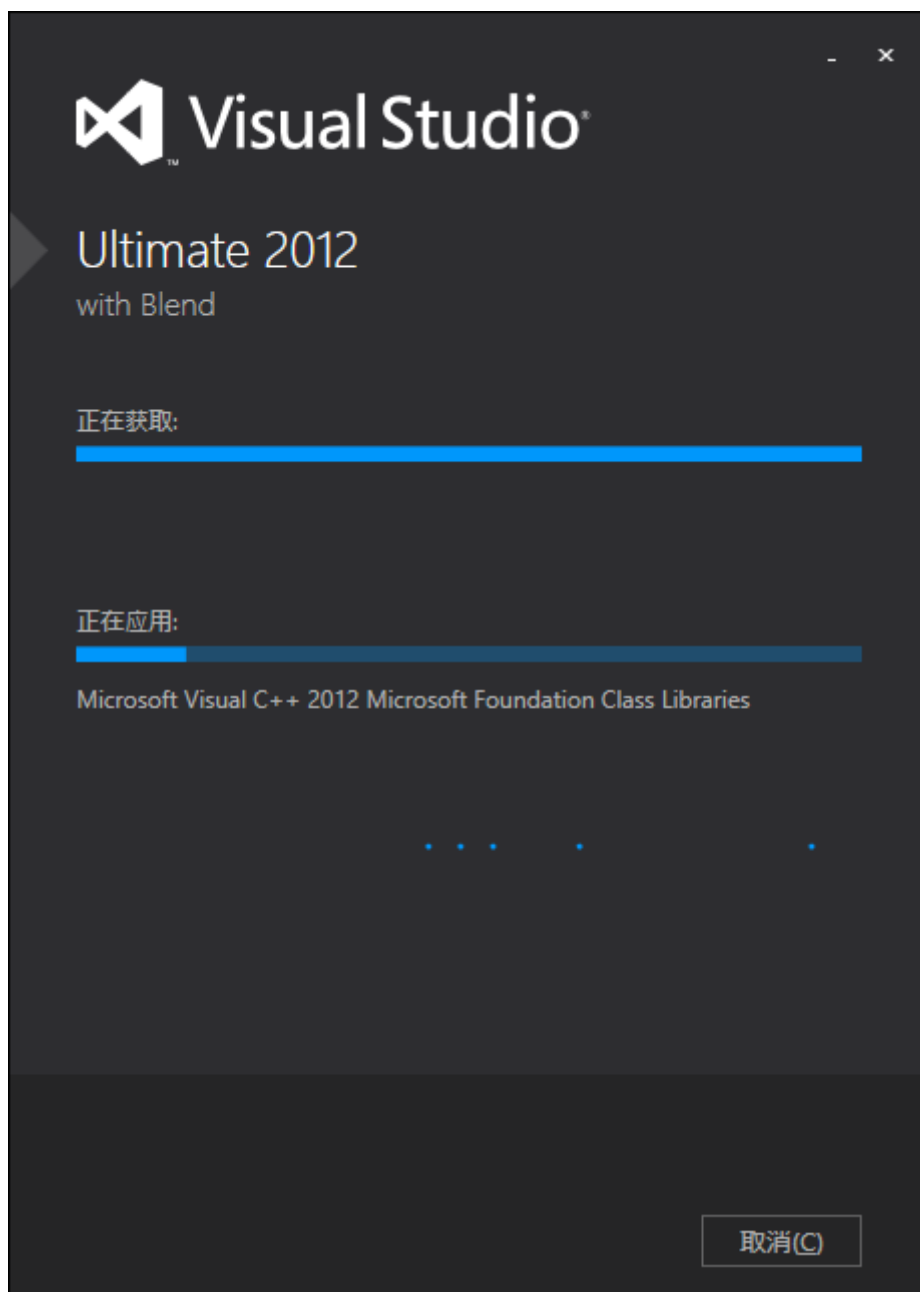


图 1-4 Visual Studio 2012 安装过程界面

步骤 5: 进入安装完成界面，如图 1-5 所示。点击【启动】按钮，进入 Visual Studio 2012 启动界面。



图 1-5 Visual Studio 2012 安装完成界面

步骤 6: 进入 Visual Studio 2012 启动界面, 如图 1-6 所示, 稍等片刻。

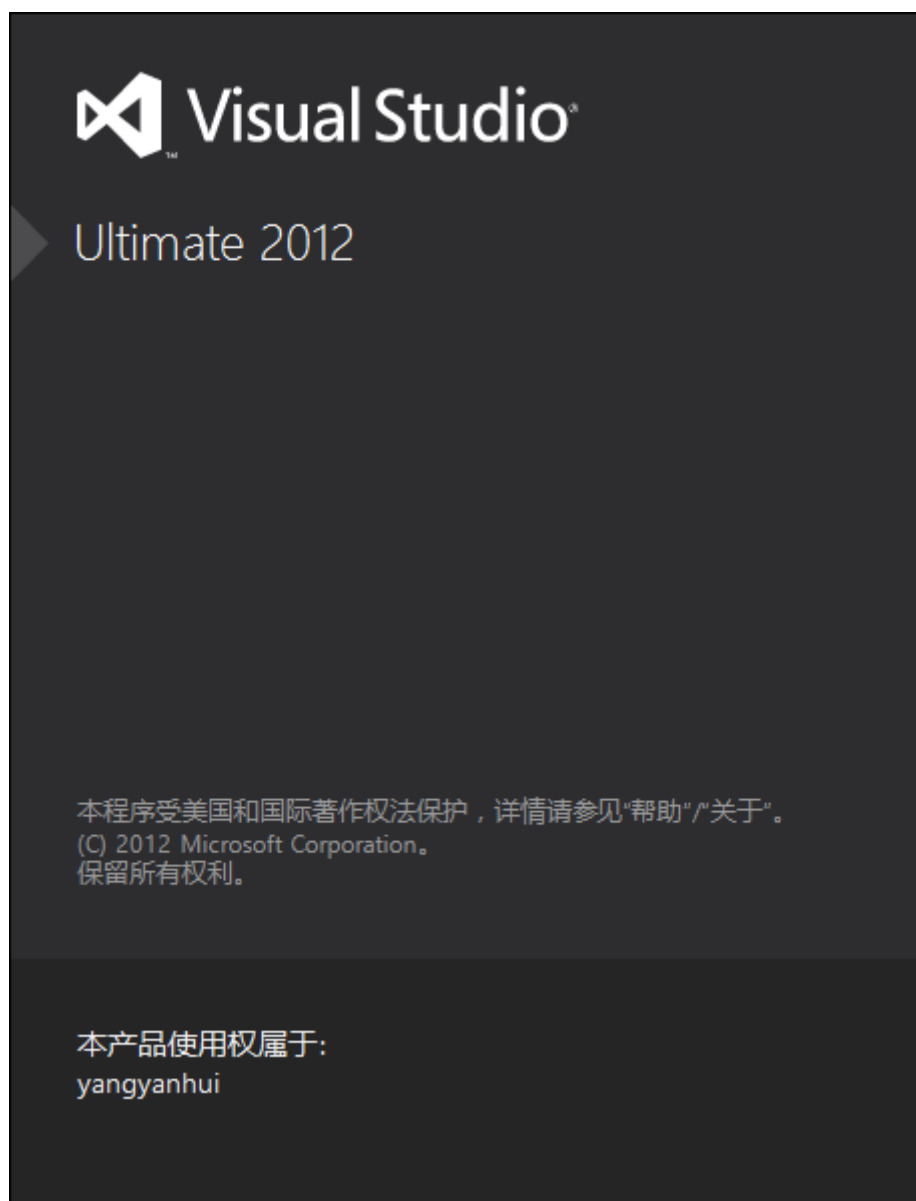


图 1-6 Visual Studio 2012 启动界面

步骤 7：第一次运行 Visual Studio 2012，会弹出运行环境自动配置界面，如图 1-7 所示。

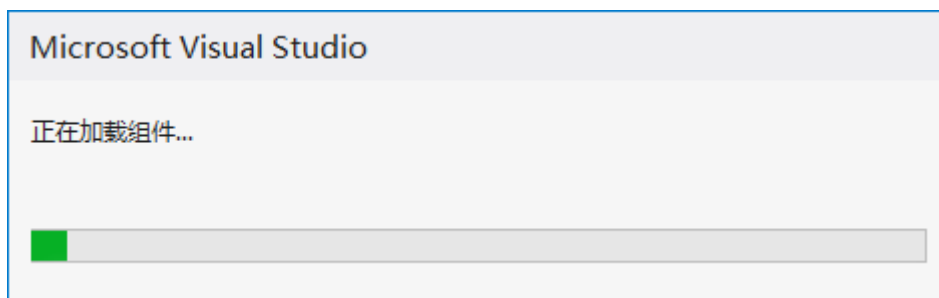


图 1-7 Visual Studio 2012 自动配置界面

步骤 8：Visual Studio 2012 程序运行界面，如图 1-8 所示。

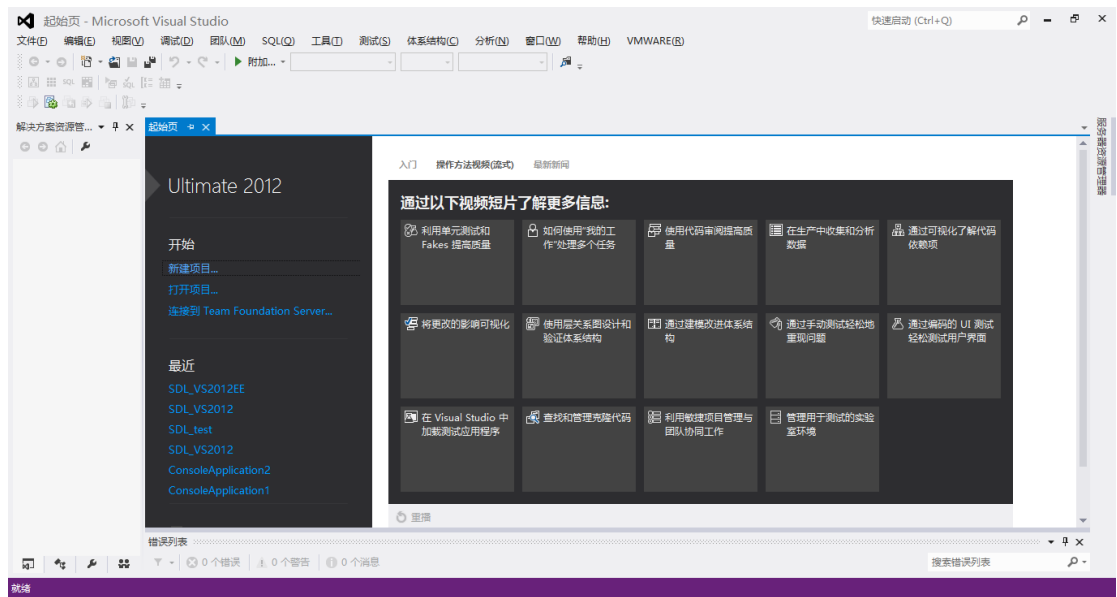


图 1-8 Visual Studio 2012 程序界面

实验结果/结论

1. 实验结果

- ✓ 具备了 c 程序 IDE 开发环境。

二、配置 VS2012

实验目的

1. 掌握 vs2012 基本配置。

实验步骤

步骤 1：认识界面，vs2012 界面如图 2-1 所示。最上面红色框内是菜单栏；左侧为解决方案/工程管理区，可以根据需要展示不同视图，如：解决方案资源管理器（现阶段最常用）、工具箱等；右侧为工作区，编码工作主要在此区完成。

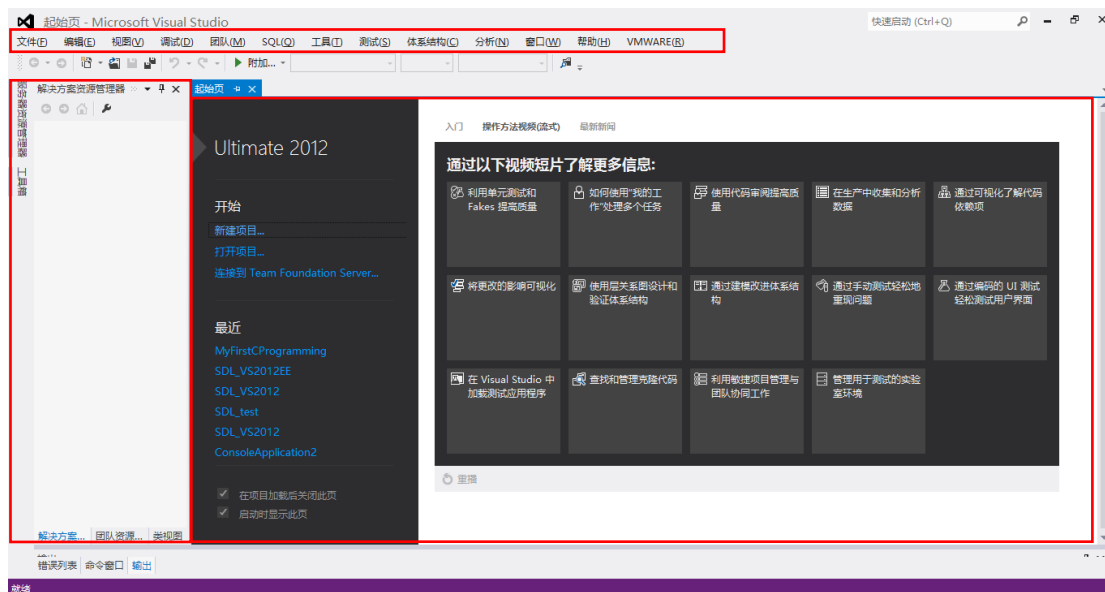


图 2-1 Visual Studio 2012 程序界面

步骤 2: 配置工具界面, vs2012 默认启动界面有解决方案资源管理器、团队资源管理器、输出等窗口, 这些窗口点击右侧关闭按钮后, 退出 vs2012 界面, 通过视图菜单可以将相关窗口调出, 如图 2-2 所示。

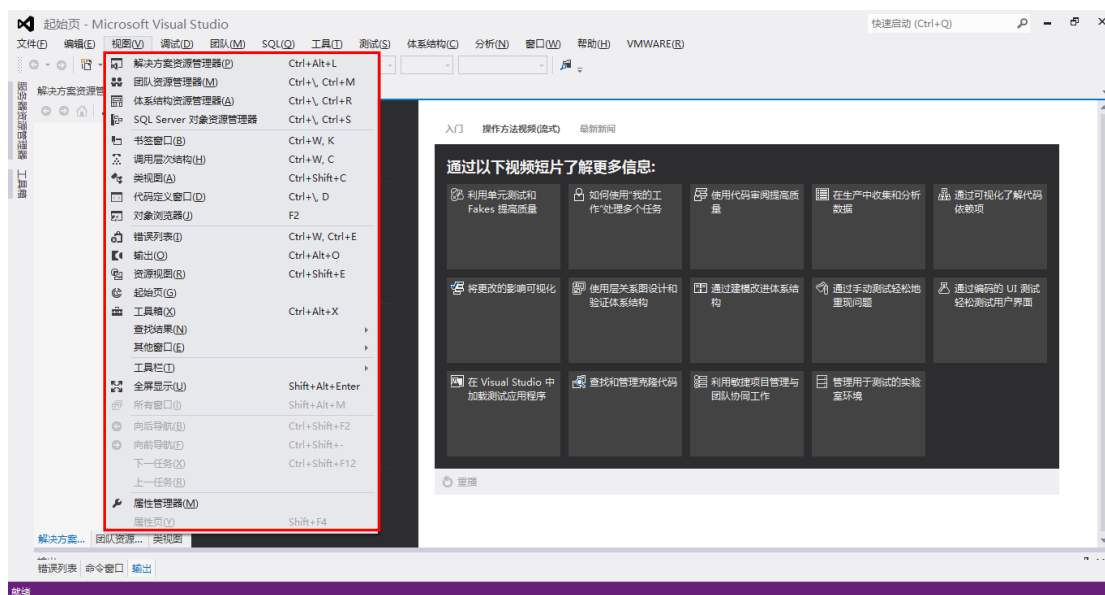


图 2-2 Visual Studio 2012 视图菜单界面

步骤 3: 工具配置, 通过【工具】→【选项】, 配置 vs2012 的界面特效、代码字体颜色、项目默认的存放目录等, 如图 2-3 所示。

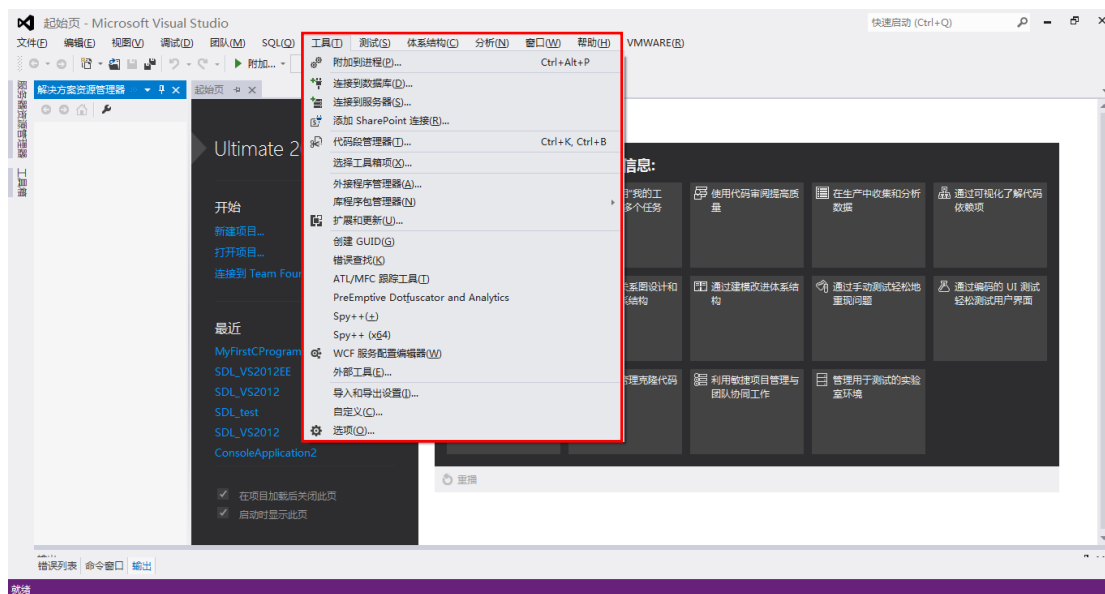


图 2-3 Visual Studio 2012 工具菜单界面

步骤 4: 设置 vs2012 视觉体验、显示项, 如图 2-4 所示。

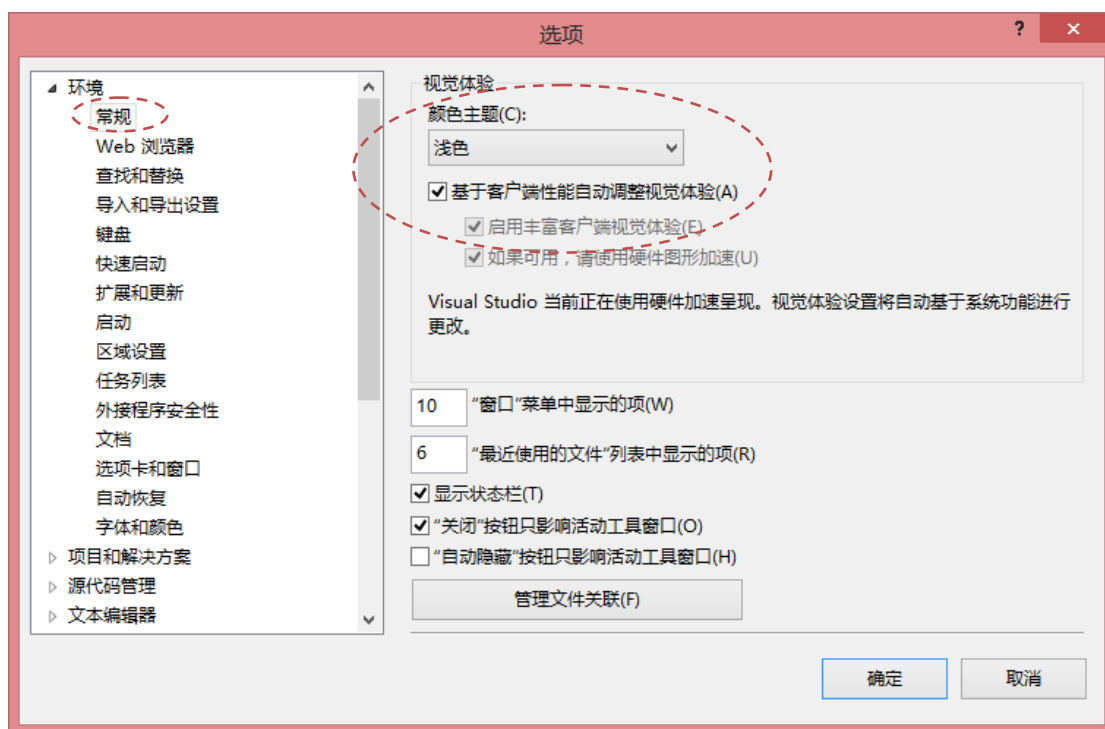


图 2-4 Visual Studio 2012 环境-常规选项卡

步骤 5: 设置代码编辑界面背景色、代码显示格式。可以为显示项分别配置各自的格式, 如图 2-5 所示。

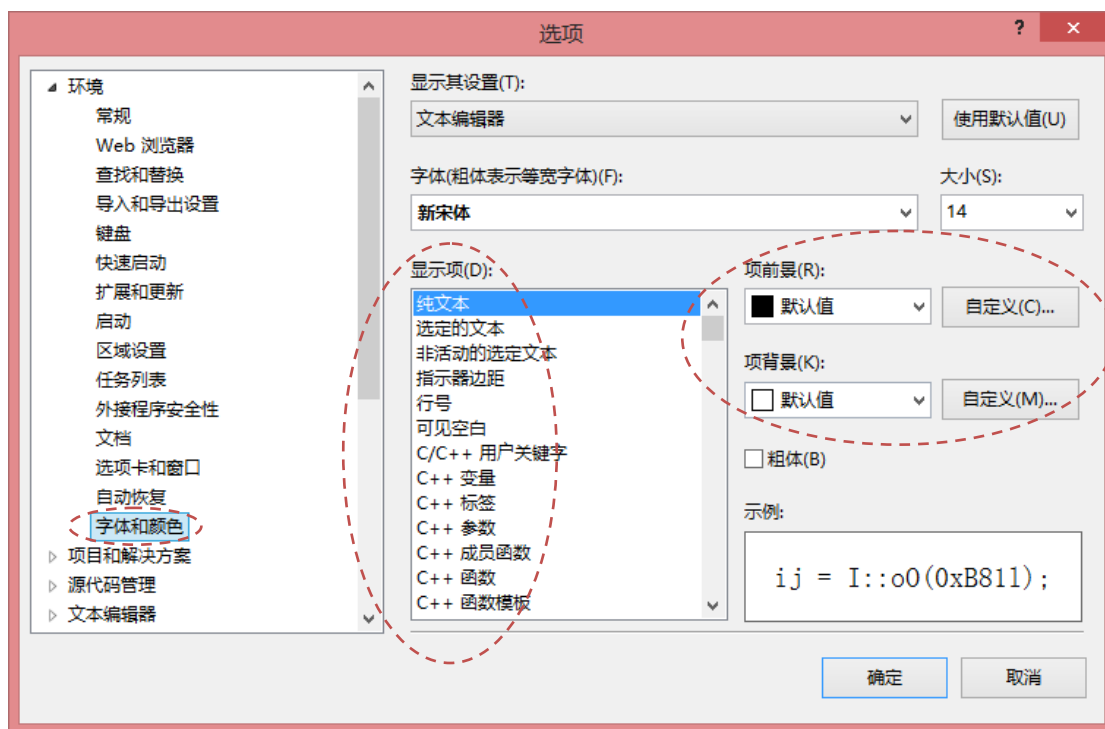


图 2-5 Visual Studio 2012 环境-字体和颜色选项卡

步骤 6: 设置解决方案/项目默认存放位置, 通过项目和解决方案-常规选项卡设置, 如图 2-6 所示。

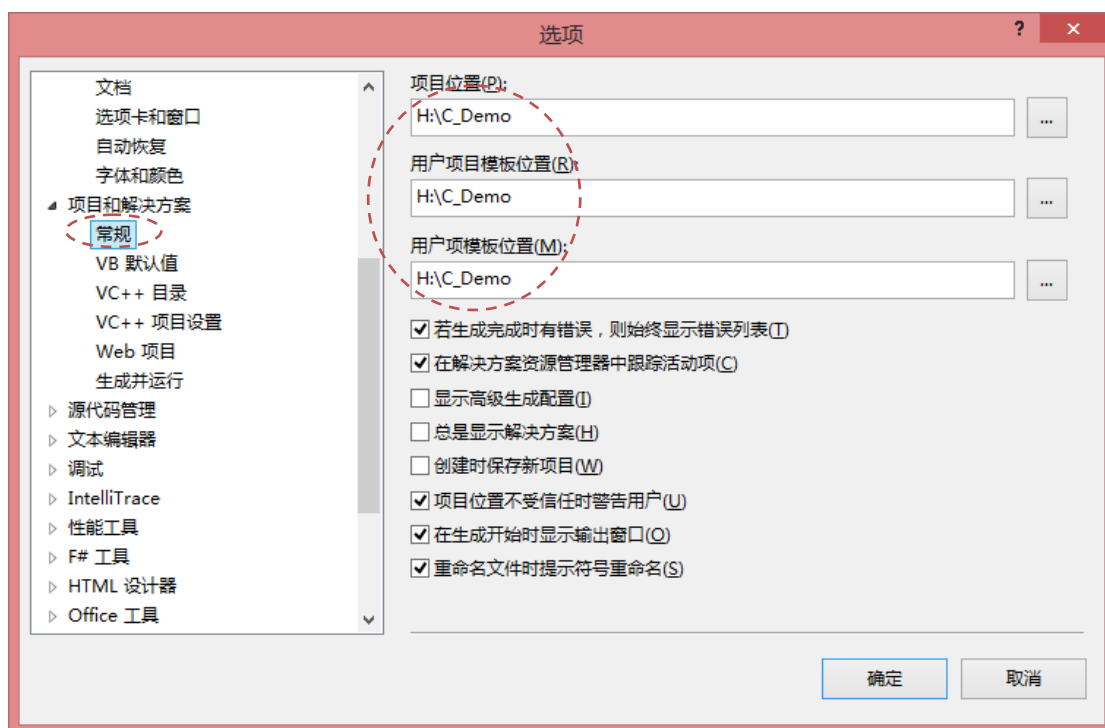


图 2-6 Visual Studio 2012 项目和解决方案-常规选项卡

步骤 7: 上面的设置是最常用的一些设置, 除此之外, 可以设置扩展和更新、导入导出、区域等等, 通过选择左侧的选项卡, 右侧会列出相应的选项供用户配置

VS2012 环境。

实验结果/结论

1. 实验结果

- ✓ 掌握 VS2012 基本配置是初学者的基本能力，可以让 VS2012 更符合用户的使用习惯，同时可以避免因环境不熟悉代码的各种问题。

三、VS2012 编辑、编译、链接、运行 C 程序

实验目的

1. 掌握 VS2012 开发 C 程序的基本方法
2. 了解 C 程序开发的基本步骤。
3. 了解集成开发环境“集成”了什么。

实验步骤

步骤 1：启动 VS2012。

步骤 2：新建项目，选择【文件】→【新建】→【项目】，单击，如图 3-1 所示。

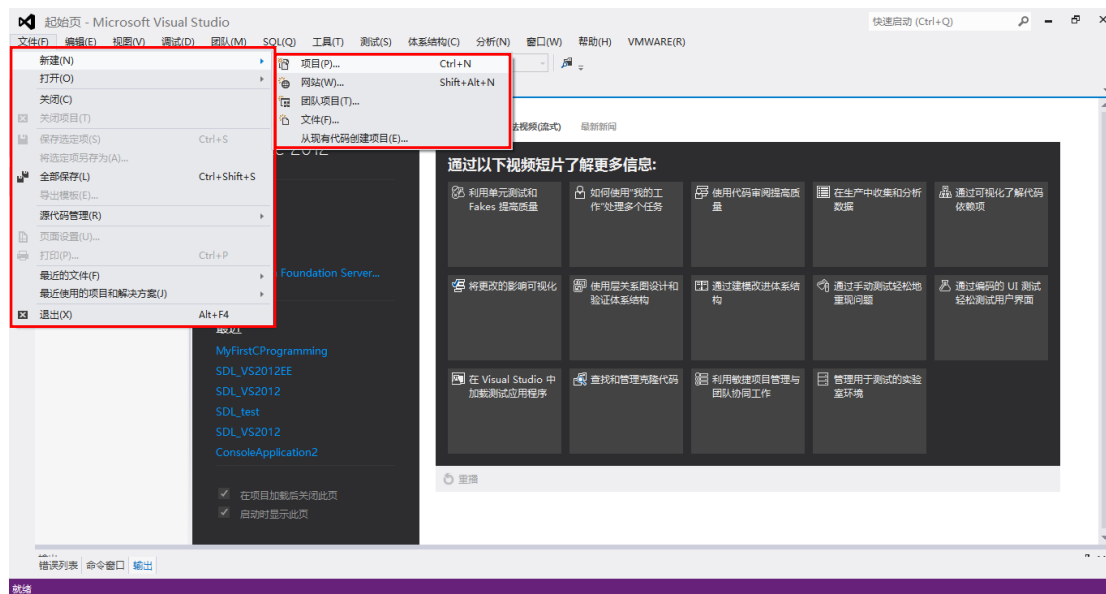


图 3-1 新建项目

步骤 3：进行相应的输入和配置之后，单击确定按钮，如图 3-2 所示。

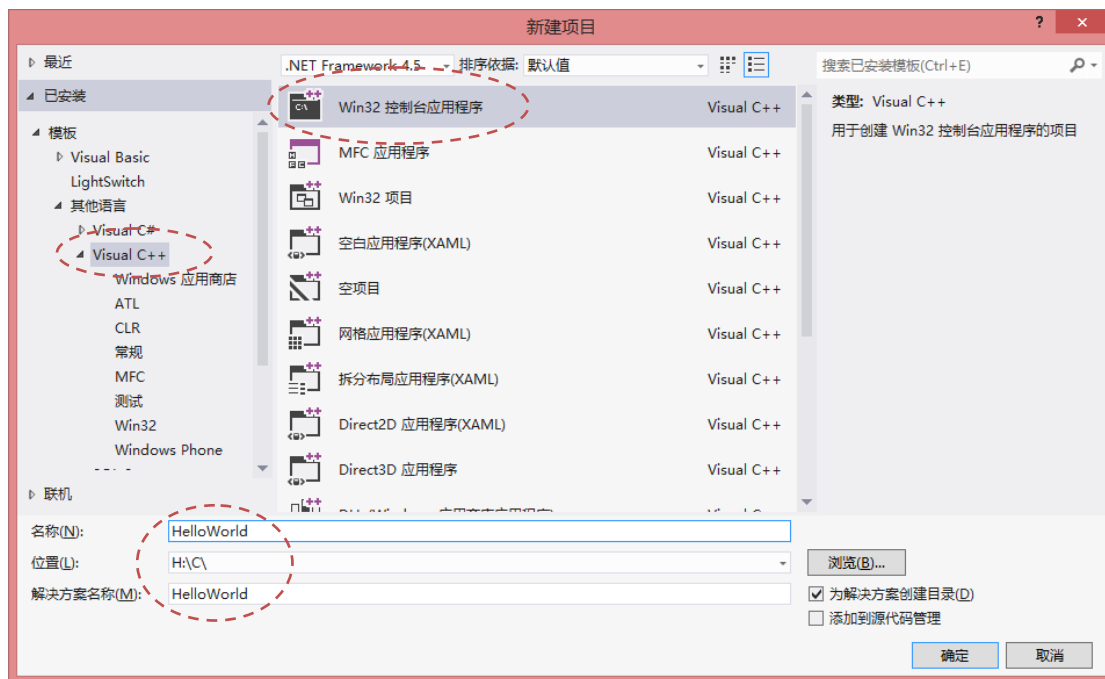


图 3-2 配置新项目相关信息

步骤 4：进入 Win32 应用程序向导界面，单击下一步按钮进行配置，如图 3-3 所示。



图 3-3 Win32 应用程序配置向导界面

步骤 5：配置应用程序类型，如图 3-4 所示，单击完成按钮。



图 3-4 配置应用程序类型界面

步骤 6：添加源文件，在图 3-5 所示的解决方案资源管理器窗口中，找到源文件→添加→新建项。

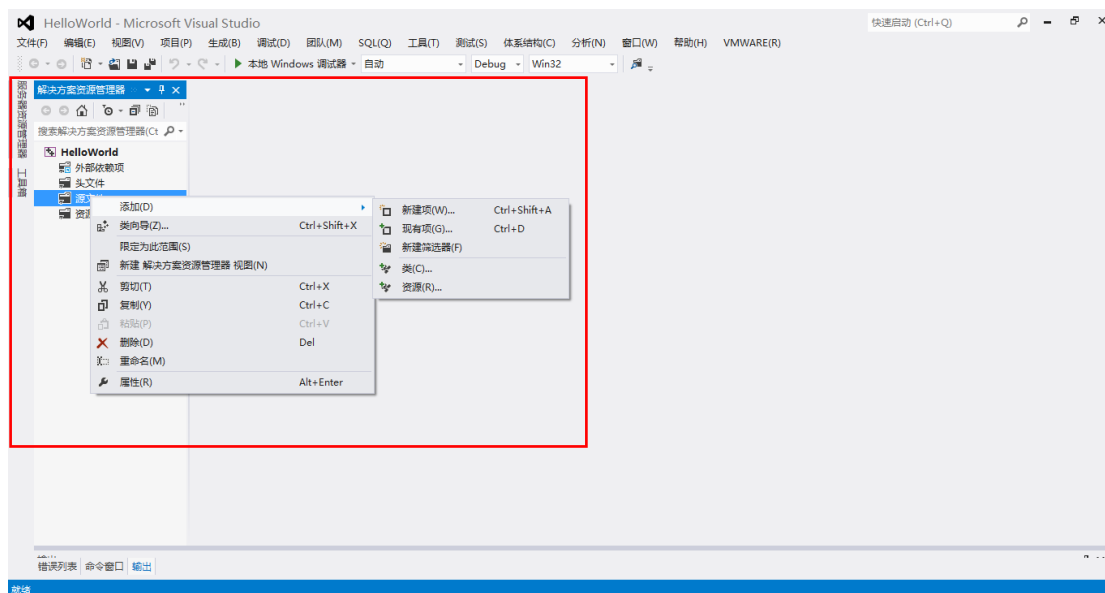


图 3-5 解决方案资源管理器窗口

步骤 7：配置新添加项，如图 3-6 所示，点击添加按钮，该操作使得该项目增加了一个源文件 `helloWorld.c`，内容为空。

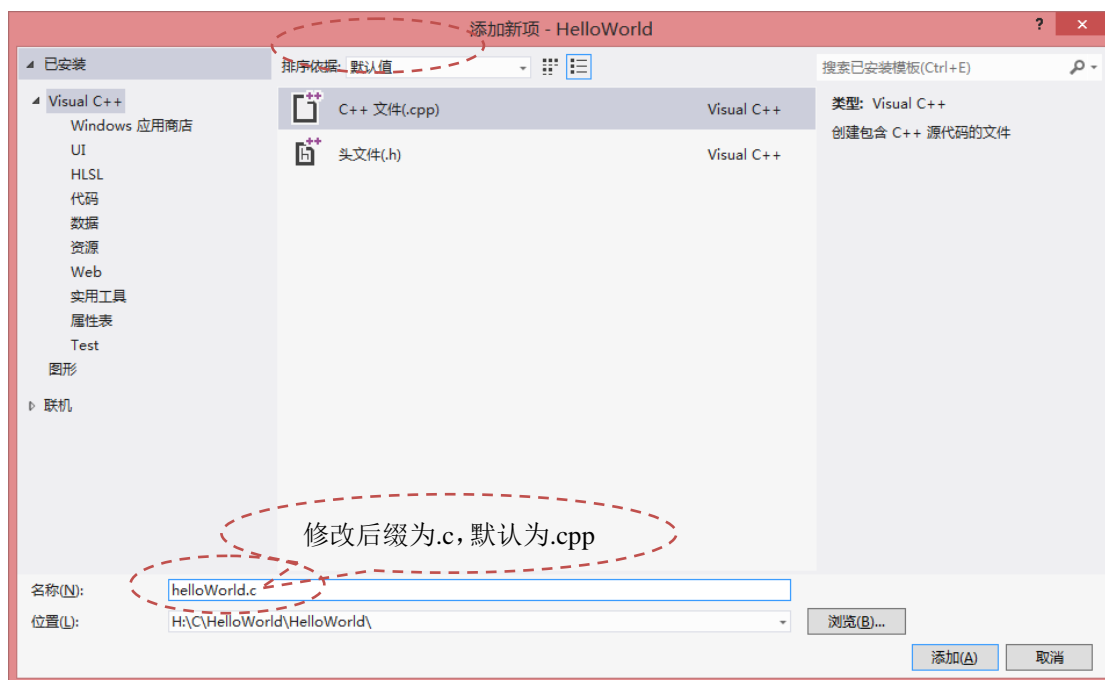


图 3-6 配置新建项

步骤 8: 编辑源文件 helloWorld.c, 如图 3-7 所示, 在“解决方案资源管理器”窗口的“源文件”列表中, 双击 helloWorld.c 打开对 helloWorld.c 的编辑窗口, 在编辑窗口中输入如下内容, 并保存。

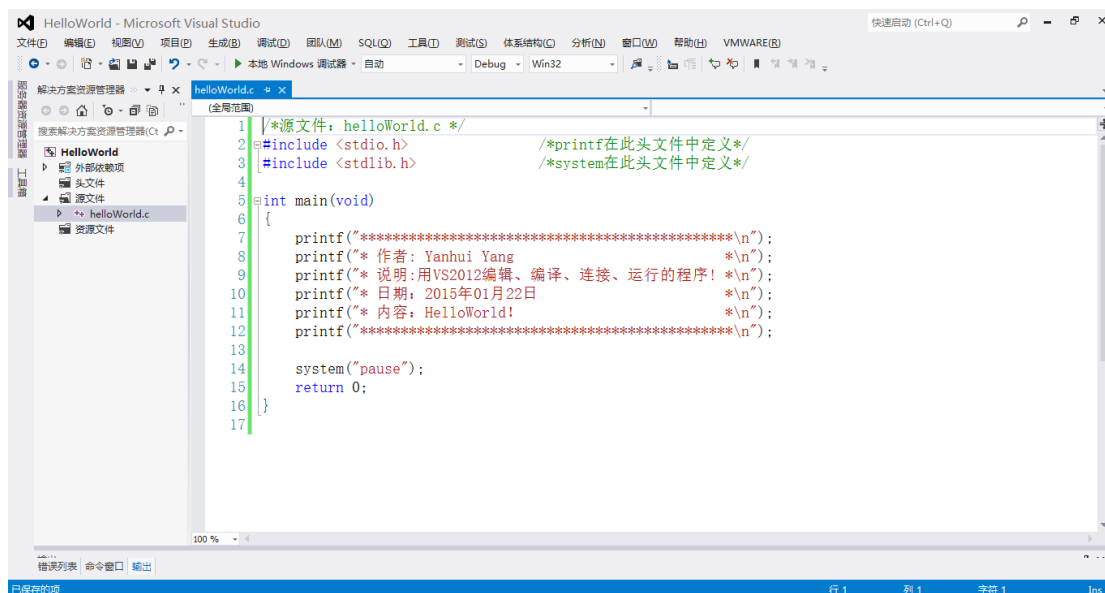


图 3-7 编辑源文件 helloWorld.c

步骤 9：生成可执行文件，如图 3-8 所示。注：这一步骤中先后完成了“编译”、“连接”两个步骤。若在这两个步骤中发生了错误，则 vs2012 将停止生成可执行文件的过程并向程序员汇报错误。

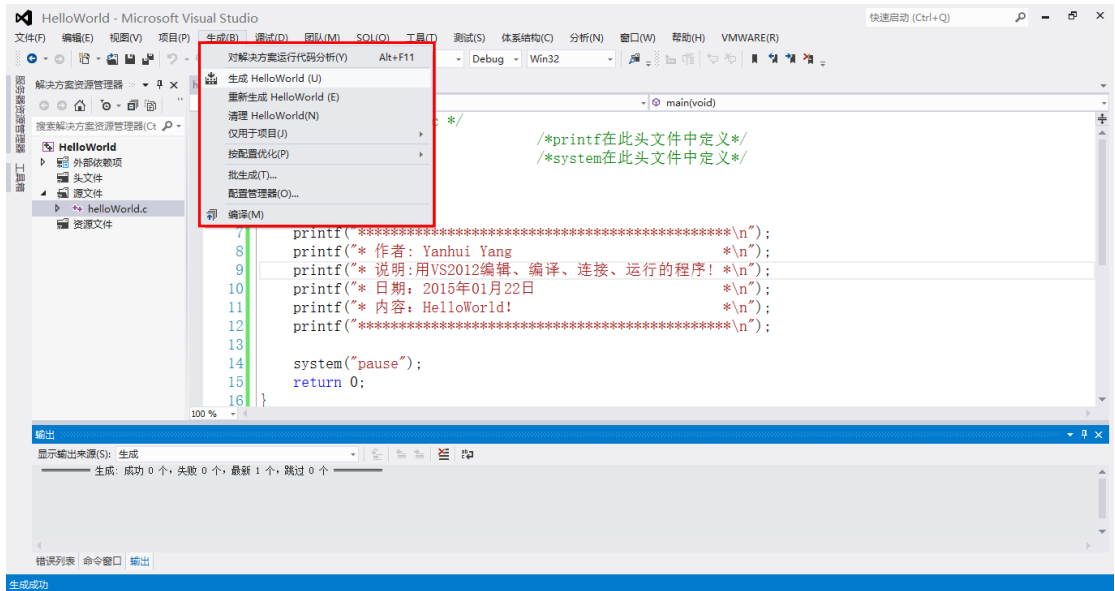


图 3-8 生成源文件 helloWorld.c

步骤 10：进入工程目录：H:\C\HelloWorld\HelloWorld\Debug，查看编译生成的二进制文件为 helloWorld.obj 文件，如图 3-9 所示。

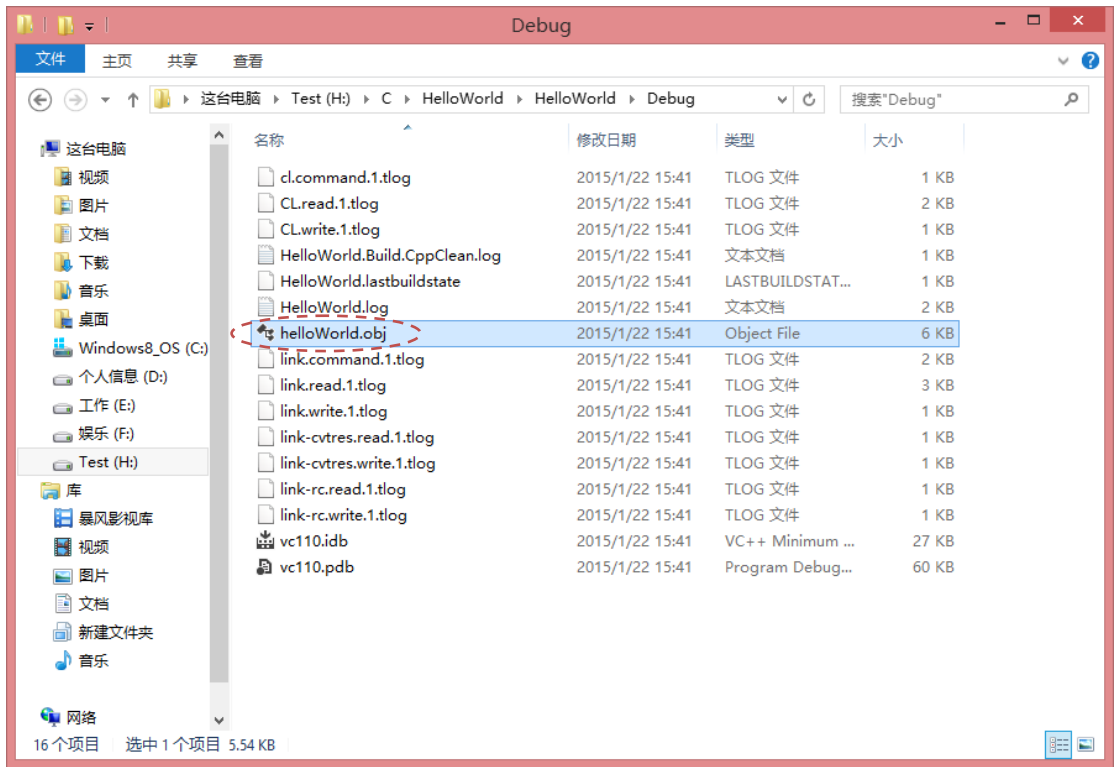


图 3-9 helloWorld.c 编译后的二进制文件 helloWorld.obj

步骤 11：进入工程目录：H:\C\HelloWorld\Debug，查看链接生成的可执行文件 HelloWorld.exe 文件，如图 3-10 所示。注：该项目链接的文件有 stdio.h、stdlib.h、helloworld.c。

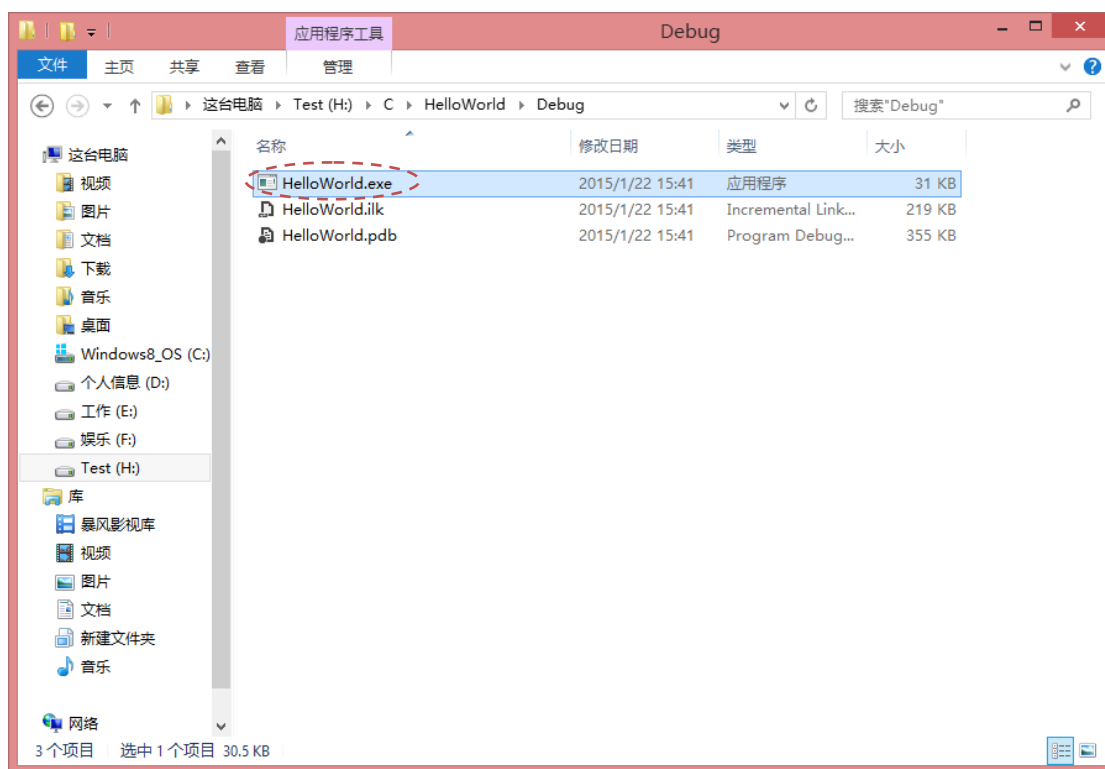


图 3-10 可执行文件 HelloWorld.exe

步骤 12：执行程序/可执行文件，点击开始执行/ctrl+F5，如图 3-11 所示。注：直接双击 HelloWorld.exe 文件也可执行程序。

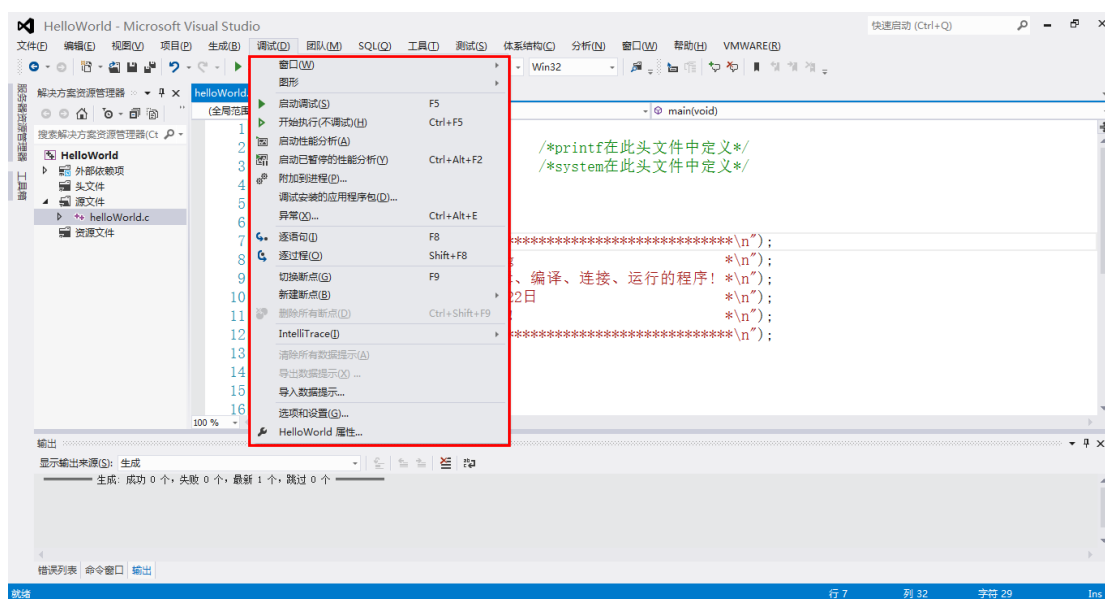



图 3-11 执行程序

实验结果/结论

1. 实验结果

本程序的功能是在屏幕上打印文字，效果如图 3-12 所示。



```
C:\WINDOWS\system32\cmd.exe
*****
* 作者: Yanhui Yang *
* 说明: 用VS2012编辑、编译、连接、运行的程序! *
* 日期: 2015年01月22日 *
* 内容: HelloWorld! *
*****
请按任意键继续. . .
```

图 3-12 程序执行结果

2. 实验结论

- ✓ VS2012 是一个集成开发环境 (IDE)，利用 VS2012 可以“一站式”地完成 C 语言程序的编辑、编译、连接、运行的全部步骤。
- ✓ 一个 C 语言程序可能包含有多个源文件，编译器将这些源文件编译成目标文件，连接器将分布在不同目标文件或不同位置的目标代码“归并”到一个可执行文件中。

四、初步体验 VS2012 调试 C 程序

实验目的

1. 初步了解编译、链接发生错误的原因
2. 初步掌握查看程序错误的方法
3. 根据错误提示修改程序简单错误

实验步骤

该实验在完成实验三的基础上进行。

步骤 1：将 helloWorld.c 的

```
printf(" * 作者: Yanhui Yang * \n");
```

改为

```
printf(" * 作者: Yanhui Yang * \n")
```

步骤 2：按 Ctrl+F5 运行程序，弹出错误对话框，说明程序生成错误，如图 4-1 所示。

点击否。

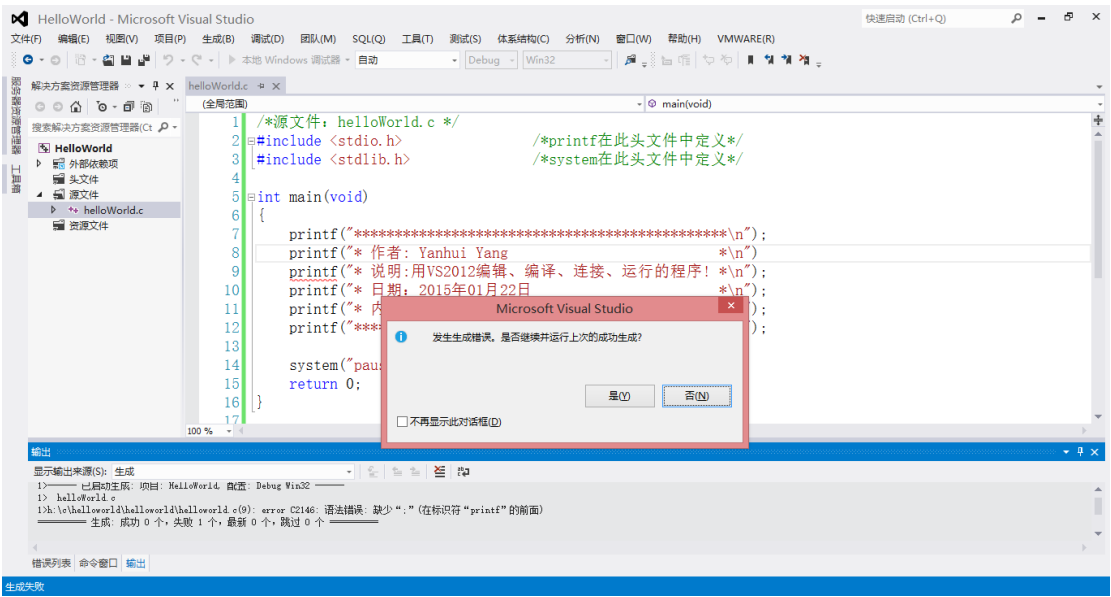


图 4-1 “程序生成”错误提示界面

步骤 3：观察窗口下面的错误提示，如图 4-2 所示。一般在错误列表处会列出源代码的错误说明和位置，可以根据错误提示修改源代码。注：当生成可执行文件，出现错误时，查看错误列表是排查源代码错误的重要途径。

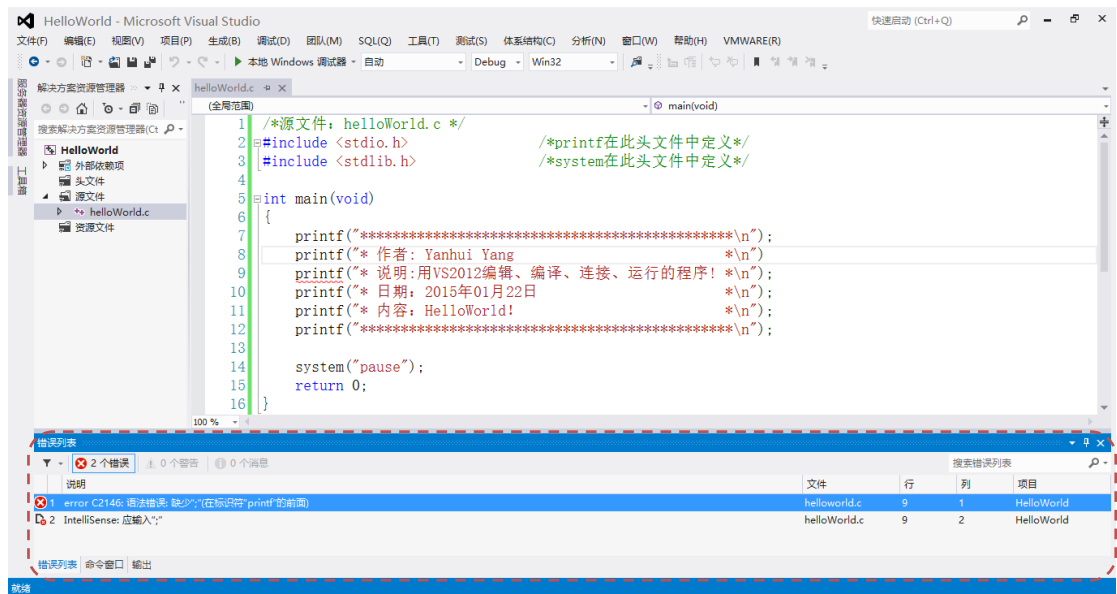


图 4-2 源代码错误列表

步骤 4: 根据提示, 在 `printf(" * 作者: Yanhui Yang * \n")`

后面添加 “;”。

步骤 5: 重新按 Ctrl+F5 运行程序, 程序正常执行。

步骤 6: 将代码 `printf(" * 作者: Yanhui Yang * \n");`

改为 `print(" * 作者: Yanhui Yang * \n");`

步骤 7: 按 Ctrl+F5 运行程序, 弹出错误对话框, 说明程序生成错误, 如图 4-1 所示。

点击否。

步骤 8: 查看错误列表, 如图 4-3, 提示 `print` 未定义, 说明该词 C 程序不认识。

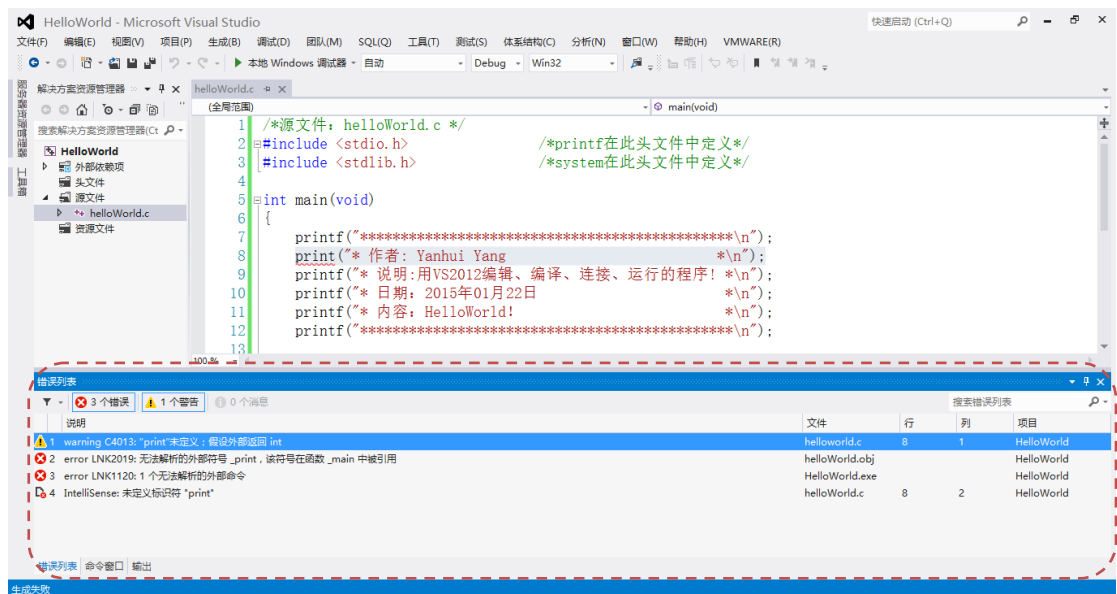


图 4-3 源代码错误列表

步骤 9：根据错误提示将

```
print ("* 作者: Yanhui Yang                *\n");
```

改为

```
printf("* 作者: Yanhui Yang                *\n");
```

步骤 10：重新按 Ctrl+F5 运行程序，程序正常执行。

实验结果/结论

1. 实验结果

参见 vs2012 步骤 3、步骤 8 在编译阶段的报错错误。

2. 实验结论

- ✓ 当源代码出现错误时，程序不能正常编译、链接生成可执行文件，不能正常运行。
- ✓ 出现错误一般的解决办法是根据错误列表提示，找到对应的行修改源代码。
- ✓ 注：有时候错误可能会出现的提示行的上面几行或者下面几行。

五、照猫画虎

实验目的

1. 了解 c 程序的基本结构
2. 根据已有的样例代码，在屏幕上输出不同要求的信息。

实验步骤

步骤 1：参见实验三步骤 1-7，生成新项目。新项目命名为 Tree, 新建源文件

```
printTree.c。
```

步骤 2：编辑源文件 printTree.c，输入如下内容。注：下面代码是 C 程序最基本框架结构，将要实现的功能写在“TODO”位置。

//代码功能：在屏幕上打印信息

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

```
#include <stdlib.h>
```

```
int main(void)
```

```
{
```

```
    //TODO
```

```
    system("pause");  
    return 0;  
}
```

步骤 3：执行程序，结果如图 5-1 所示。

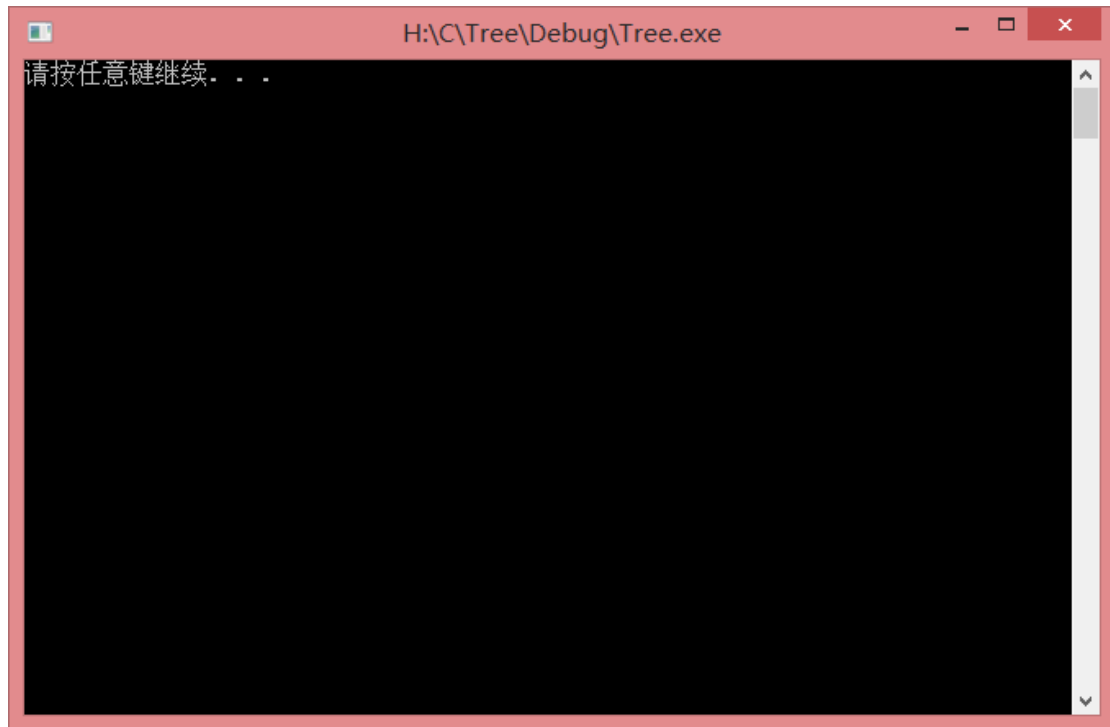


图 5-1 程序执行结果

步骤 4：添加代码，在屏幕上输出如下信息，如图 5-2 所示。注：printf() 函数功能是根据“ ”里边的内容在屏幕上输出信息。

//代码功能：在屏幕上打印信息

```
#include <stdio.h>  
#include <stdlib.h>  
  
int main(void)  
{  
    printf("*****\n");  
    printf("*****\n");  
    printf("*****\n");  
    printf("*****\n");  
  
    system("pause");  
    return 0;  
}
```

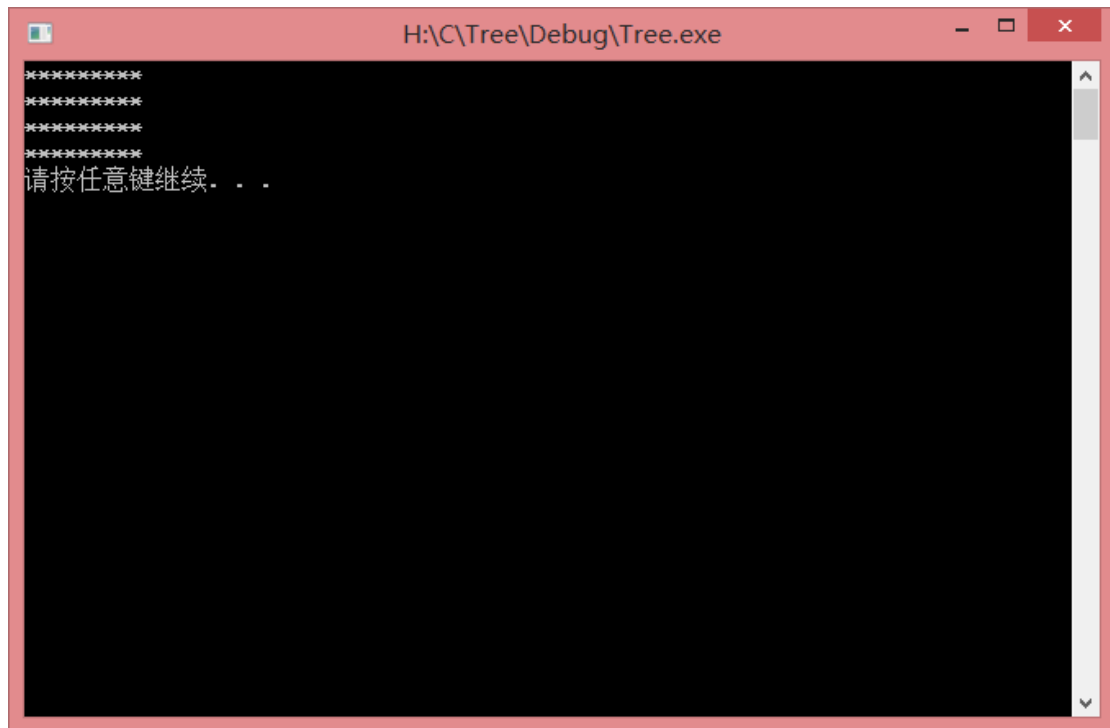


图 5-2 程序运行结果

步骤 5：将上面代码的“*”改为\$，观察程序运行结果。

步骤 6：将上面代码部分“\$”改为“空格”，观察程序运行结果。

实验结果/结论

1. 实验结论

- ✓ C 程序的基本结构是 C 程序的基础。
- ✓ 在 C 程序基本结构中添加/修改代码可以实现多种功能需求。

1.2 理论题

A 类

一、填空题

1. 一个 C 语言源文件就是字符组合的一个序列，_____ 是指允许出现在源文件中的字符集合。
2. C 程序的源文件、头文件、目标代码文件、可执行文件的扩展名分别为：_____、

- _____、_____和_____。
3. C 程序开发的步骤：_____、_____、_____、调试和_____。
 4. 程序的编译，是将_____转换成_____的过程。
 5. 注释在翻译阶段将变成一个_____。
 6. C 语言程序中有且仅有一个_____函数。
 7. 一个 C 程序的执行是从_____函数开始。

二、简答题

1. 要想书写出完美的程序，应该具备哪些知识？C 语言在这些知识中起到什么作用？
2. 标准与实现的关系在我们生活中也无处不在，请列举生活中的一个例子进一步谈谈标准和实现的关系。
3. 谈一谈源文件与可执行文件的联系和区别。
4. 谈一谈源字符集和执行字符集的联系和区别。
5. 在程序中添加注释有哪些好处？
6. 想一想，在程序中包含头文件有什么好处？

三、程序题

1. 请在屏幕上打印如下图所示的金字塔形图案。

```
      *
     ***
    *****
   *********
```

2. 请在屏幕上打印如下图所示的两棵树图案。

```
      *      *
     ***    ***
    ***** *****
   | |      | |
```


B 类

一、填空题

1. 实现分为两种：_____ 和 _____。
2. IDE 是 _____ 的缩写。
3. C 语言中输出操作是由 _____ 库函数完成的，调用该函数应在源程序的首部写上 _____。
4. _____ 和 _____ 间的内容称为语句块。
5. C 源程序的基本单位是_____。
6. 标准 C 语言包括 _____ 和一组标准库。
7. C99 标准有 _____ 个关键词。
8. 标准规定的单词类型有 _____、_____、_____、_____、操作符和分隔符。

二、简答题

1. 头文件为什么不称为一个翻译单元？
2. 简述 IDE。
3. 请分析：C 语言单词是组成 C 语言源文件的最小单位，因此掌握 C 语言单词及其用法是编写 C 语言程序的必经之路。
4. 请分析：C 语言程序中所有的注释符 “/*.....*/” 都可以用注释符 “//.....” 代替。
5. 简述 C 程序的开发步骤。
6. 编译器的主要任务是什么？
7. 连接器的主要任务是什么？
8. 简述语法错误。
9. 简述语义错误。
10. 程序代码如下，请标出下述程序中的单词及其类型。

```
/* 源文件：demo.c*/  
#include <stdio.h>  
#include <stdlib.h>
```

```
int main(void)  
{
```

```

int i, sum = 0;
for(i = 1; i <= 5; ++i)
    sum += i;           /*累计求和*/
printf("%d\n", sum);    /*函数返回值为计算1到5的和*/

system("PAUSE");
return 0;
}

```

三、程序题

1. 打印功能菜单。编程在屏幕上（或说控制台窗口内）打印如下所示的菜单。
功能菜单：

```

-----
[1]  加法      [2]  减法
[3]  乘法      [4]  除法
[5]  平方      [6]  开方
[7]  绝对值    [0]  退出
-----

```

请输入您的选择（0---7）：

2. 杨辉三角。编程在屏幕上（或说控制台窗口内）打印如下所示的菜单。

```

1
1  1
1  2  1
1  3  3  1
1  4  6  4  1
1  5  10 10 5  1

```

本章答案

A 类

一、填空题

1. 源字符集
2. .c .h .obj .exe

3. 编辑、编译、链接、运行
4. 源文件、目标文件
5. 空格
6. main
7. main

二、简答题

略

三、程序题

1. 编码提示

- 1) 编程过程中严格遵循课上和教材中强调的编码风格和规范。
- 2) 代码如下。

```
#include <stdio.h>

#include <stdlib.h>

int main(void)
{

    printf("  *\n");      /*\n代表换行，即输出  *后将转到下一行进行输出*/
    printf("  ***\n");
    printf("  *****\n");
    printf("  *****\n");

    system("pause");

    return 0;

}
```

2. 编码提示

- 1) 编程过程中严格遵循课上和教材中强调的编码风格和规范。

2) 代码如下。

```
#include <stdio.h>

#include <stdlib.h>

int main(void)
{

    printf("    *        *\n");
    printf("   ***      ***\n");
    printf("  *****  *****\n");
    printf(" | |      | |\n");
    printf(" | |      | |\n");

    system("pause");

    return 0;
}
```

B 类

一、填空题

1. 宿主实现、独立实现
2. Integrated Development Environment
3. printf、#include <stdio.h>
4. { 、 }
5. 语句
6. 语言标准
7. 37
8. 关键词、标识符、常量、字符串

二、简答题

略

三、程序题

略