

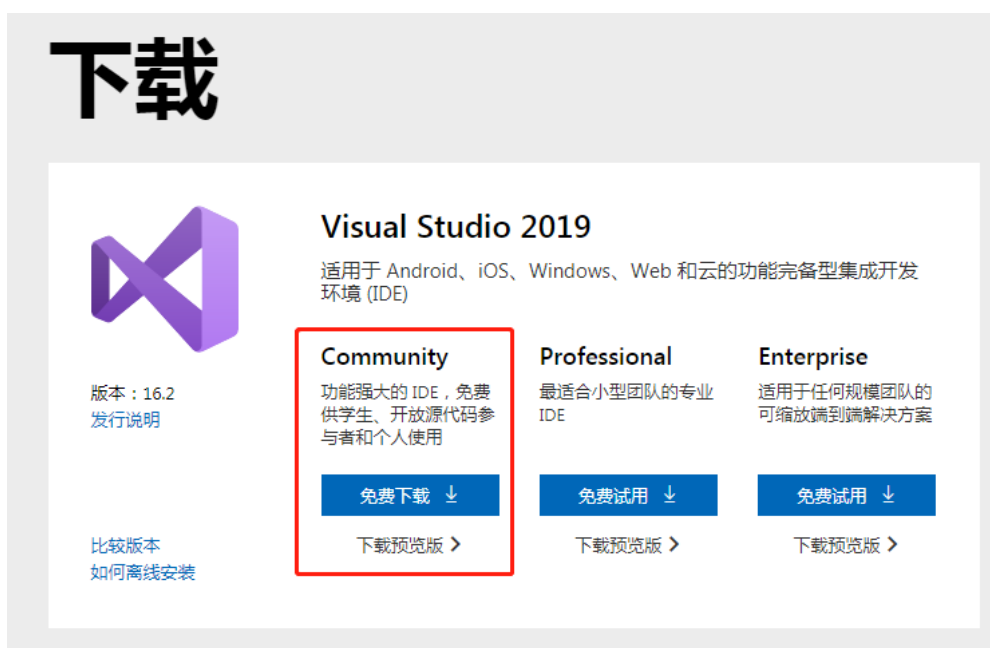
数据结构专题实验课前准备

一 编译工具

推荐使用 Visual Studio IDE，各版本都可以在实验中使用。在文档中将以 Visual Studio 2019 Community 为例进行说明。如果你已经是 Code::Block 或 Dev C++ 的深度使用者，可以选择自己喜欢的编译工具，将不在本文档中单独介绍 Code::Block 和 Dev C++ 的使用方法。

可以在官网中下载 Visual Studio 2019 Community 的 Windows 版和 Mac 版：

<https://visualstudio.microsoft.com/zh-hans/downloads/>



安装过程不再详述，选择默认参数即可。参考如下链接：

<https://docs.microsoft.com/zh-cn/cpp/build/vscpp-step-0-installation?view=vs-2019>

二 创建工程

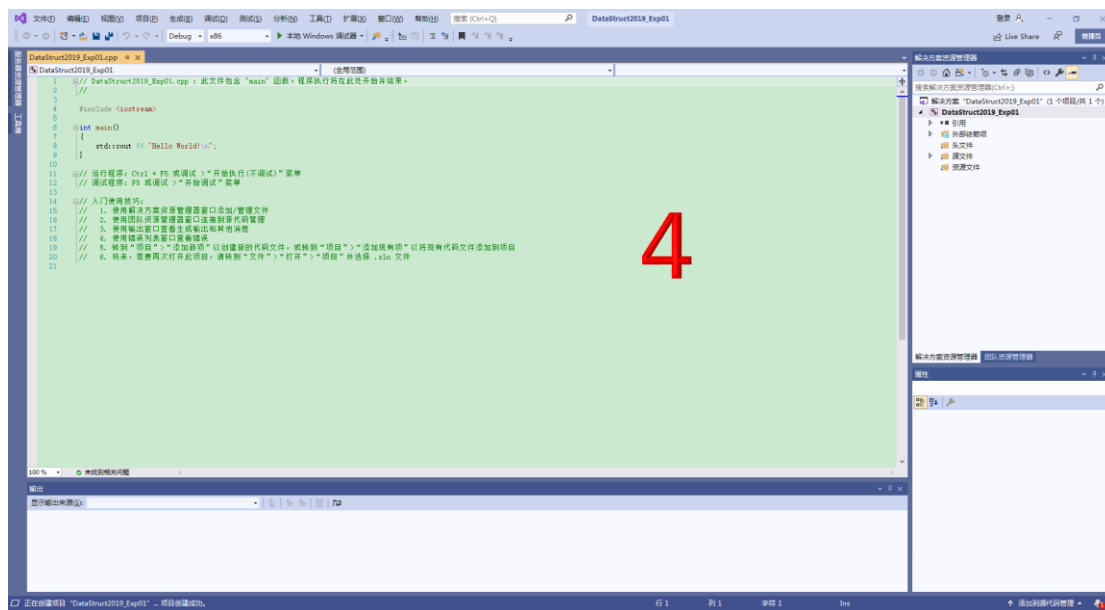
Visual Studio 2019 的基本操作说明参考如下链接：

<https://docs.microsoft.com/zh-cn/cpp/get-started/tutorial-console-cpp?view=vs-2019>

本学期数据结构与算法专题实验均不涉及界面开发，因此创建“控制台程序应用”即可。

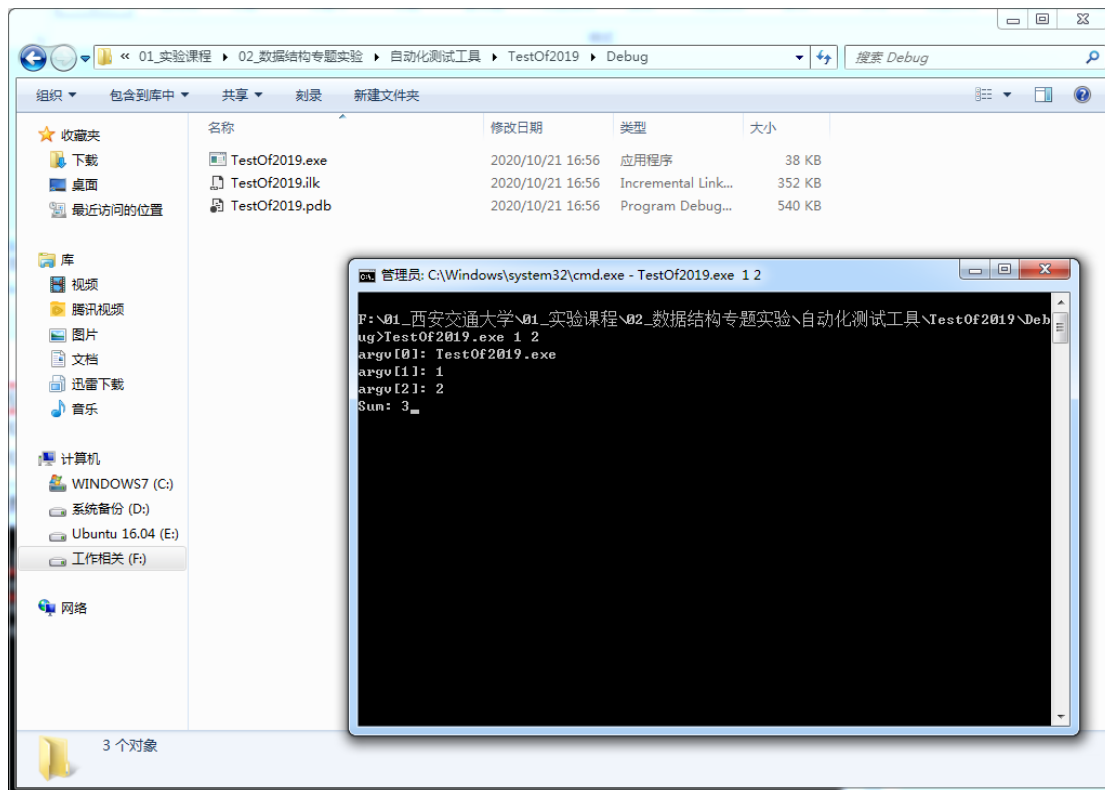


选择好项目类型后，为项目命名并选择需要的保存路径，接着点“创建”按钮，项目就创建完成了。



三 接口定义

为了便于验收，统一在命令行参数接受用户输入并输出运行结果。如下图所示可执行文件 TestOf2019.exe 通过命令行调用时传入 1 和 2 两个参数，计算两个整数参数的和并输出：



通过命令行获取参数的示例代码如下：

```
#include <stdio.h>
int main(int argc, char* argv[])
{
    //参数数量错误
    if (argc != 3)
    {
        printf("parameter input error.\n");
        printf("ERROR_01");
        getchar(); //等待用户输入
        return -1;
    }
    printf("argv[0]: %s\n", argv[0]); //二进制文件名称在argv[0]字符数组中
    printf("argv[1]: %s\n", argv[1]); //第一组命令行参数存在argv[1]字符数组中
    printf("argv[2]: %s\n", argv[2]); //第二组命令行参数存在argv[2]字符数组中

    int param1 = 0;
    int param2 = 0;
    sscanf_s(argv[1], "%d", &param1);
    sscanf_s(argv[2], "%d", &param2);
    printf("Sum: %d", param1+param2); //输出计算结果
    getchar(); //等待用户输入
    return 0;
}
```

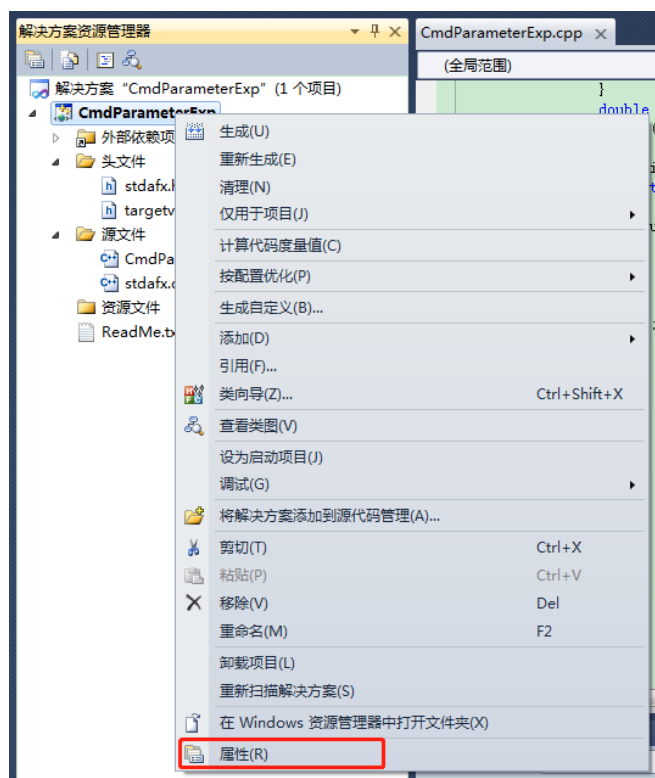
四 FAQ

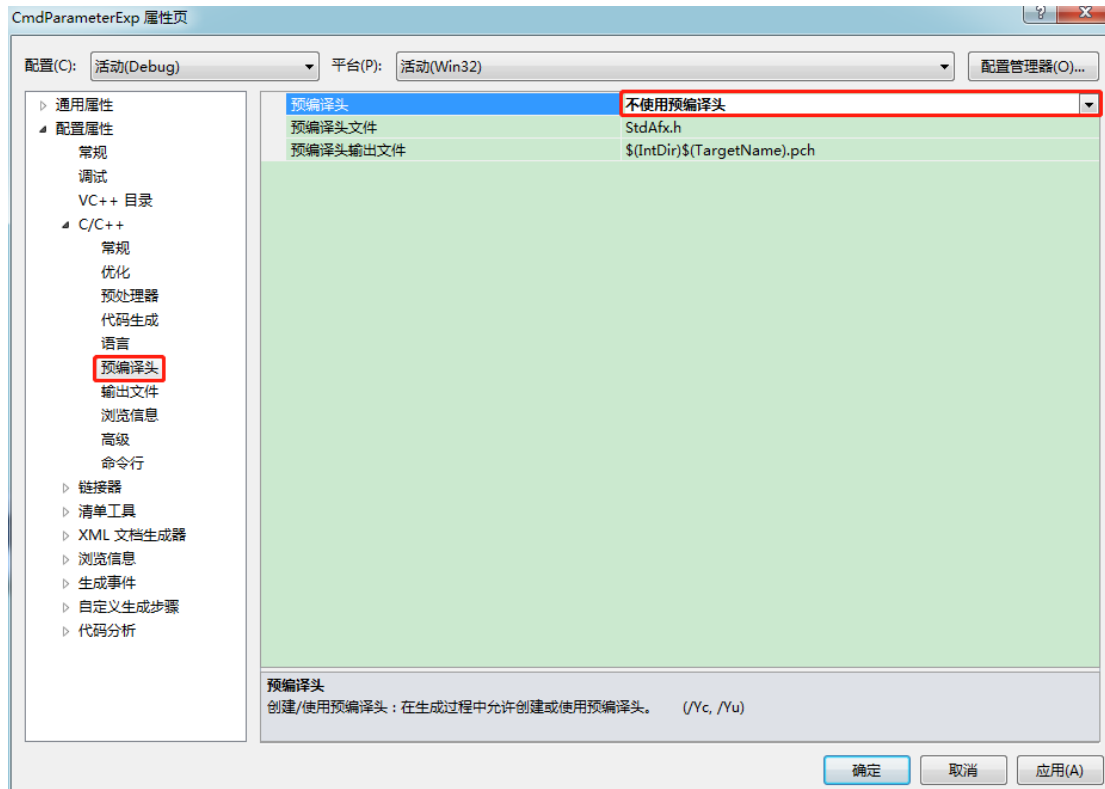
1 程序在 Visual Studio 早期版本中运行不了？

由于早期版本创建的项目中默认启用了“预编译头”选项，项目编译过程需要使用 stdafx.h 等文件。

可以通过以下步骤关闭项目中的“预编译头”选项：

(1) 在项目名称上单击鼠标右键->属性->C/C++->预编译头->不使用预编译头；





(2) 在代码中注释掉 stdafx.h 引用；

修改前： `#include "stdafx.h"`

修改后： `//#include "stdafx.h"`

(3) 修改主函数定义。

修改前： `int _tmain(int argc, _TCHAR* argv[])`

修改后： `int main(int argc, char* argv[])`

2 调试过程中命令行参数如何传入？

在调试程序过程中可以通过如下步骤将定制的命令参数传递给程序：在项

目名称上单击鼠标右键->属性->调试->命令行参数。

