

Visual Studio调试注意事项

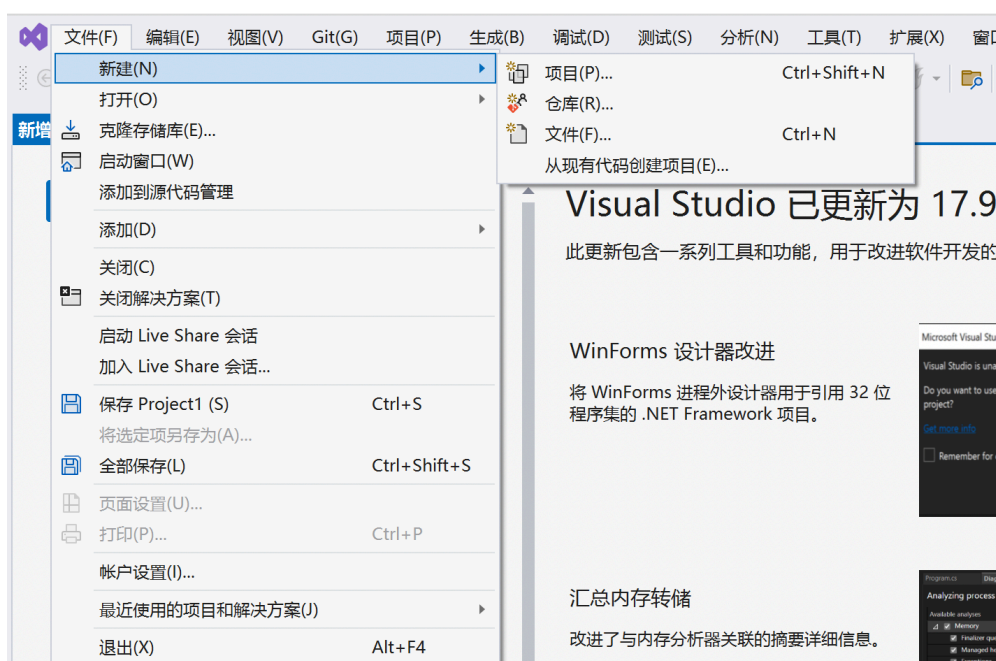
以Visual Studio 2022为例，运行调试都建议使用这种流程，以免出错。其中1-4步为代码编写，第5步为调试。请各位使用Visual Studio仍有困难或未曾调试过的同学尝试并熟悉这种流程。如遇到问题欢迎联系助教。

助教：王骏骁

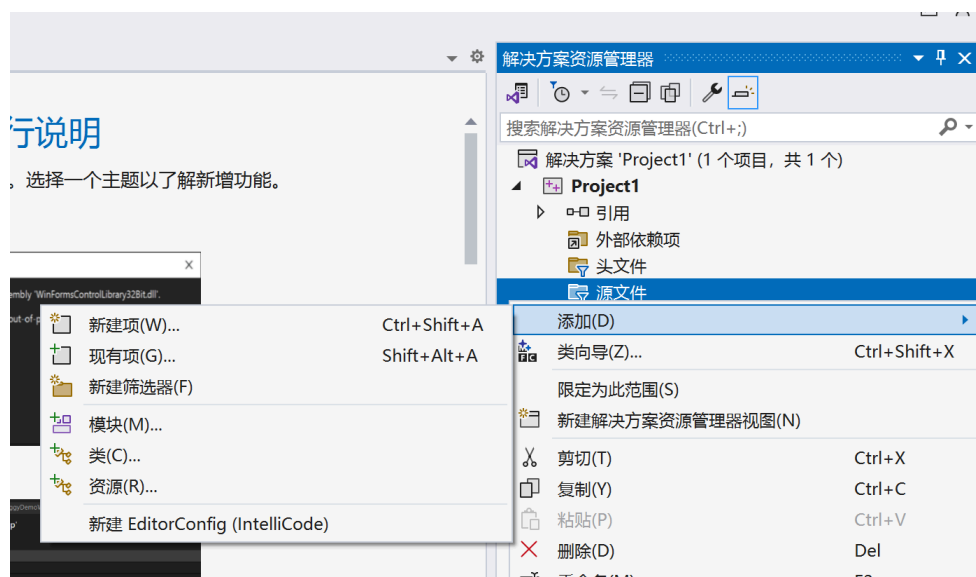
邮箱：wjyy1@126.com

2024年4月4日

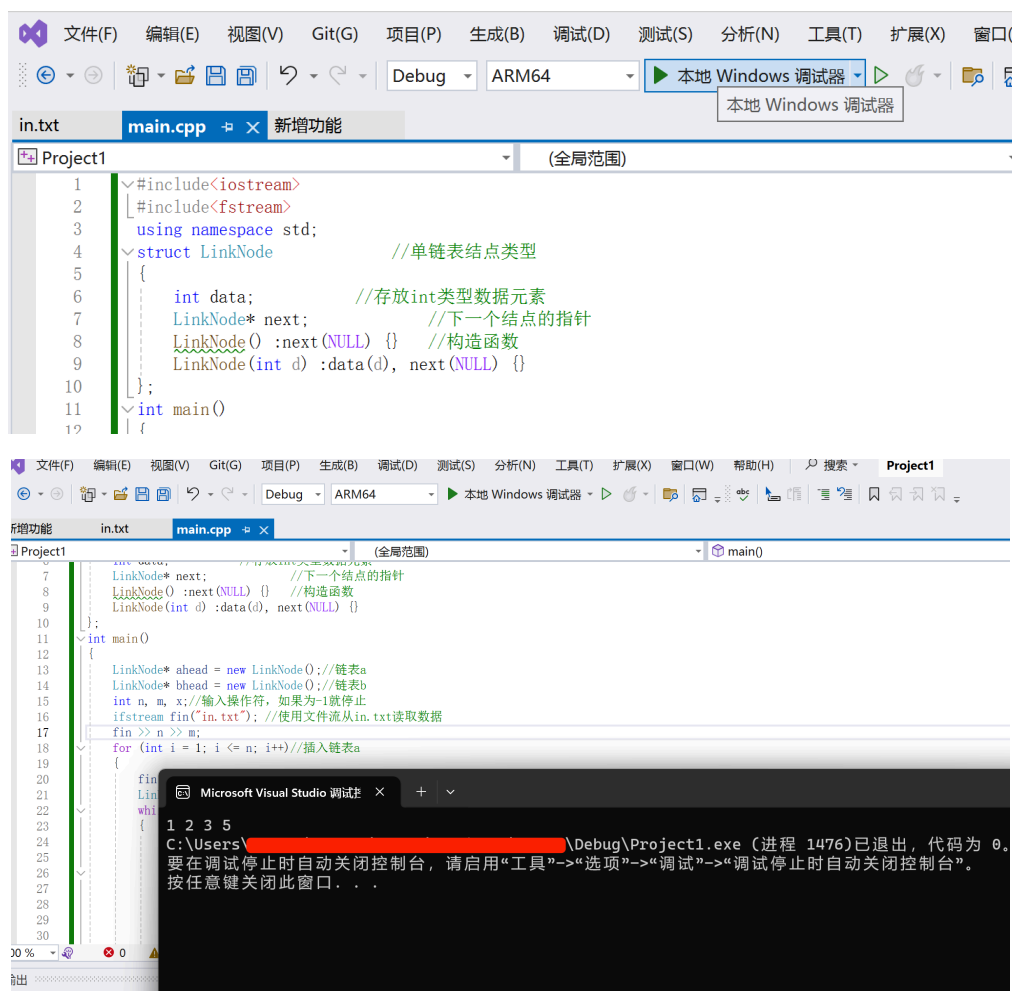
1. 打开Visual Studio，左上角点击文件，新建，项目，选择“空项目”。



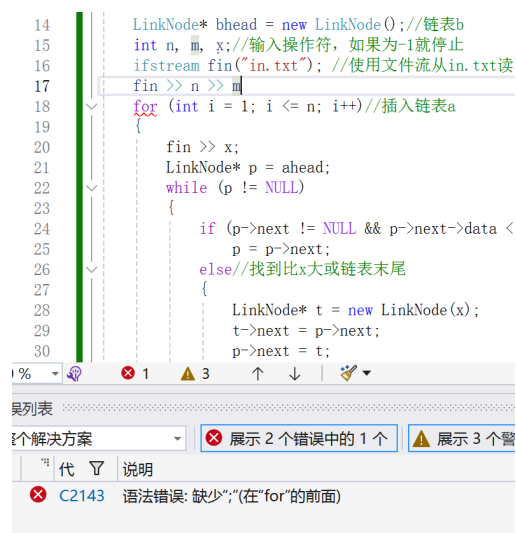
2. 此时屏幕右侧会出现解决方案资源管理器，右键源文件，添加，新建项。在这里可以创建你的源文件，在本课程中以 `main.cpp` 为主。同时你可以在这里创建 `in.txt` 作为输入文件，以免在文件夹中与其他目录混淆。



3. 在 `main.cpp` 中写完代码后, 向 `in.txt` 中写入样例输入或自定义输入, 并点击绿色箭头运行。如果编译正确此时会弹出运行结果, 并告知程序是否正常结束。如果程序出现异常, 则返回代码可能不为0。

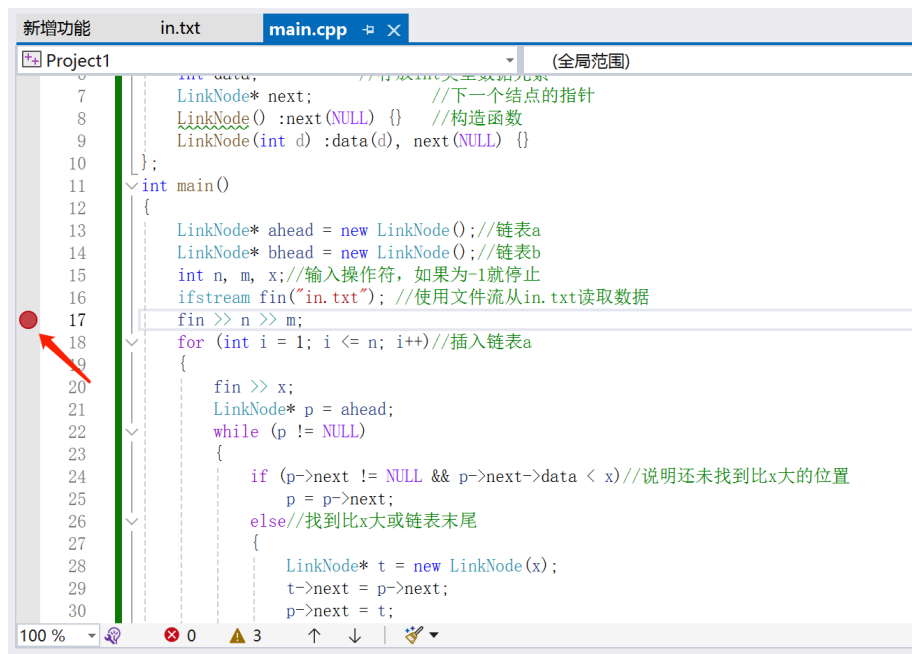


4. 如果未通过编译, 程序会询问是否继续执行上次代码, 选择否后下方会报错哪里出现了问题, 应对照修改。

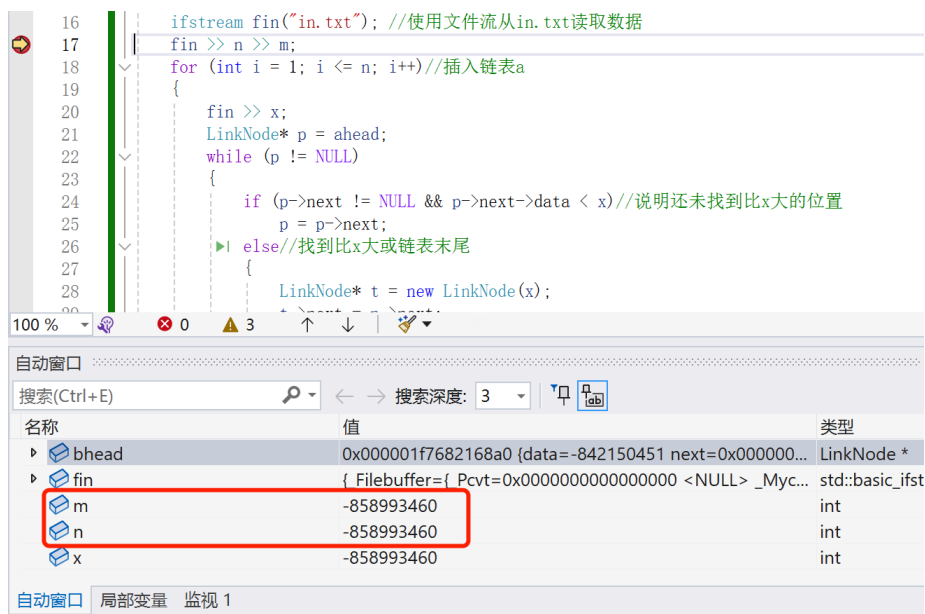


5. 如果程序正常运行, 但结果不对, 可以先自行思考算法是否有问题。如果无法确定算法是否有问题可以使用调试来检查程序运行的每一步是否和预想中的一致。

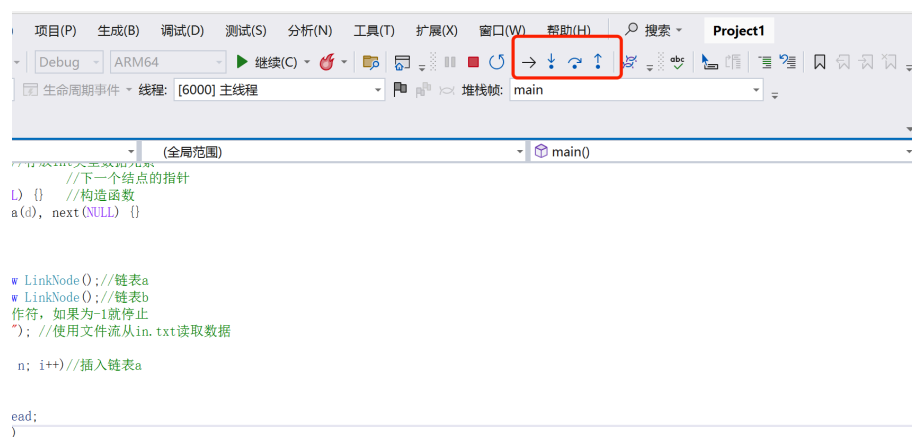
(a) 点击行号前面的空白位置 (空灰位置), 出现红色圆点说明成功设置断点, 然后继续点击绿色箭头开始运行。

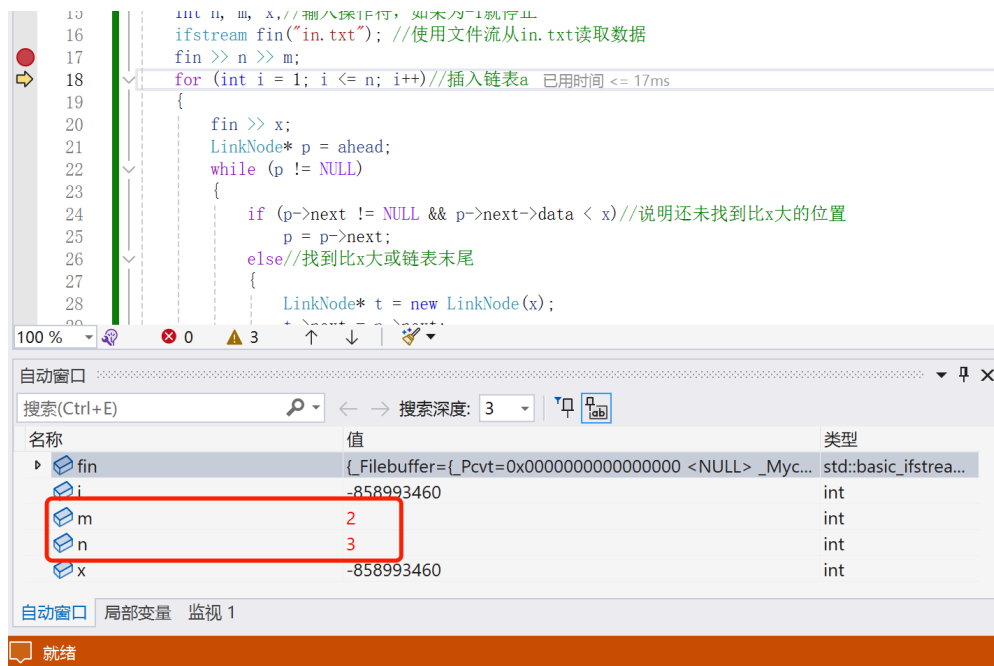


(b) 运行到断点时会停下来，下面显示当前各变量的情况。



(c) 点击图上的按钮，程序会进行相应操作。如“逐过程”，程序会执行到下一行，如果当前行对变量做出了修改，相应变量在此处显示的值也会发生变化。





补充：（基本同第二次作业解析）对于输入流的常见用法，可以引入 `<fstream>`，使用 `ifstream` `fin("in.txt")` 定义 `in.txt` 对应的流。

使用 `fin>>x` 可以向 `x`（可为数值或字符串）读入一段数据，直到下一个空格/换行为止，因此可以读入以空格分开的数字或字符串。

使用 `getline(fin, line)` 可以向 `line`（应为字符串）读入一段数据，直到换行为止，这时一整行数据都在 `line` 中。

此时可以直接将 `line` 作为字符串利用，如果要再读入 `line` 中的数据，可以引入 `<sstream>`，使用 `stringstream` `ss(line)` 定义 `line` 对应的流。

还有一种情况是，对于一行未知数量的数据，可以用 `while(fin>>x){}`，当这一行读入结束后就会跳出 `while` 循环。

强烈建议大家自行尝试！如果有不懂的欢迎群里/私信/邮件问助教！如果回消息不及时或者忘记回了请直接催，谢谢大家。