

调试技巧

`freopen_s`

使用命令行比对文件

方式一：PowerShell 的 `Compare-Object`

方式二： `diff` 命令

默认输出格式

`unified` 输出格式

完整流程示例

参考

调试技巧

对于一些需要从标准输入中读入数据和/或向标准输出中写入数据的程序，每次运行都手动输入和/或人工检查控制台输出是很麻烦的，本文介绍：

- 如何使用库函数 `freopen_s` 进行 I/O 重定向，以及
- 如何使用命令行比对文件

`freopen_s`

Visual Studio 中默认禁用 `freopen`，我们需要使用 `freopen_s`，函数签名如下：

```
errno_t freopen_s(  
    FILE      ** stream,  
    const char * fileName,  
    const char * mode,  
    FILE      * oldStream  
);
```

其中 `errno_t` 是错误码的类型。0 表示成功，其他值表示某种错误，详见文档。

`mode` 参数表示打开文件的模式，在我们的场景下通常会用到：

- 只读 `r`，文件必须存在

- 读写 `w`，如果文件不存在则创建文件，如果文件存在则将其原有内容丢弃

可以像下面这样使用：

```
#include <iostream>

using namespace std;

int main() {
#define DEBUG
#ifdef DEBUG
    FILE *fp = nullptr;
    freopen_s(&fp, "test.in", "r", stdin);
#endif
}
```

这样就把标准输入重定向到了 `test.in` 这个文件。

注意

这里使用了宏来做条件编译。通过插件运行测试时一定要至少将

`#define DEBUG` 这一行注释掉。

使用命令行比对文件

下面的脚本代码全部假设当前工作目录下有两个文件 `actual.out` 和 `expect.out`。`actual.out` 的内容为：

```
This is
a
example
```

`expect.out` 的内容为：

```
This
is
an
example
```

下面的方式都是按行比较的。

方式一：PowerShell 的 `Compare-Object`

```
$actual = "actual.out"
$expect = "expect.out"
Compare-Object -ReferenceObject $(Get-Content $expect) -
DifferenceObject $(Get-Content $actual)
```

输出结果如下：

```
InputObject SideIndicator
-----
This is      =>
a            =>
This         <=
is           <=
an           <=
```

其中 `InputObject` 表示输入的行，`SideIndicator` 表示这一行是哪个文件独有的：箭头指向右侧表示这一行是第二个参数（即，`actual.out`）独有的，指向左侧表示这一行是第一个参数（即，`expect.out`）独有的。

方式二： `diff` 命令

在 Git Bash 或 WSL 中使用

`diff` 命令实际上是在告诉使用者如何将第一个文件变成第二个文件

默认输出格式

```
diff expect.out actual.out
```

输出结果如下：

```
1,3c1,2
< This
< is
< an
---
> This is
> a
```

其中第一行的 `1,3c1,2`

第一行由三部分组成：

- 与第一个文件相对应的行号
- 字母（`a` 表示添加 `add`，`c` 表示更改 `change`，`d` 表示删除 `delete`）
- 与第二个文件相对应的行号

在上面的输出中，`1,3c1,2` 表示“需要更改第一个文件中的第 1 至 3 行以匹配第二个文件中的第 1 至 2 行”。

以 `<` 开头的行是第一个文件中的行，以 `>` 开头的行是第二个文件中的行，`---` 将两个文件的行分开。

unified 输出格式

```
diff -u expect.out actual.out
```

输出为：

```
--- expect.out 2021-04-23 14:58:44.157006000 +0800
+++ actual.out 2021-04-23 15:00:37.766213700 +0800
@@ -1,4 +1,3 @@
-This
-is
-an
+This is
+a
example
```

前两行的时间表示文件的最后一次修改时间。文本行前的 `-` 表示需要删除的行，`+` 表示需要添加的行。

完整流程示例

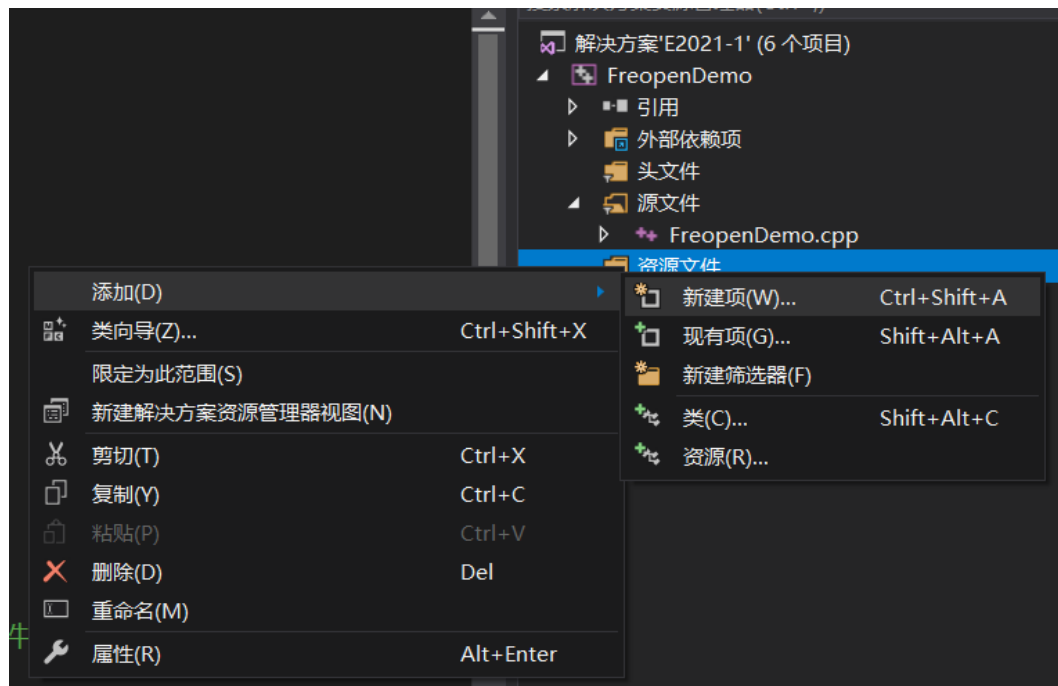
以 A + B 问题为例。

1. 创建测试用例

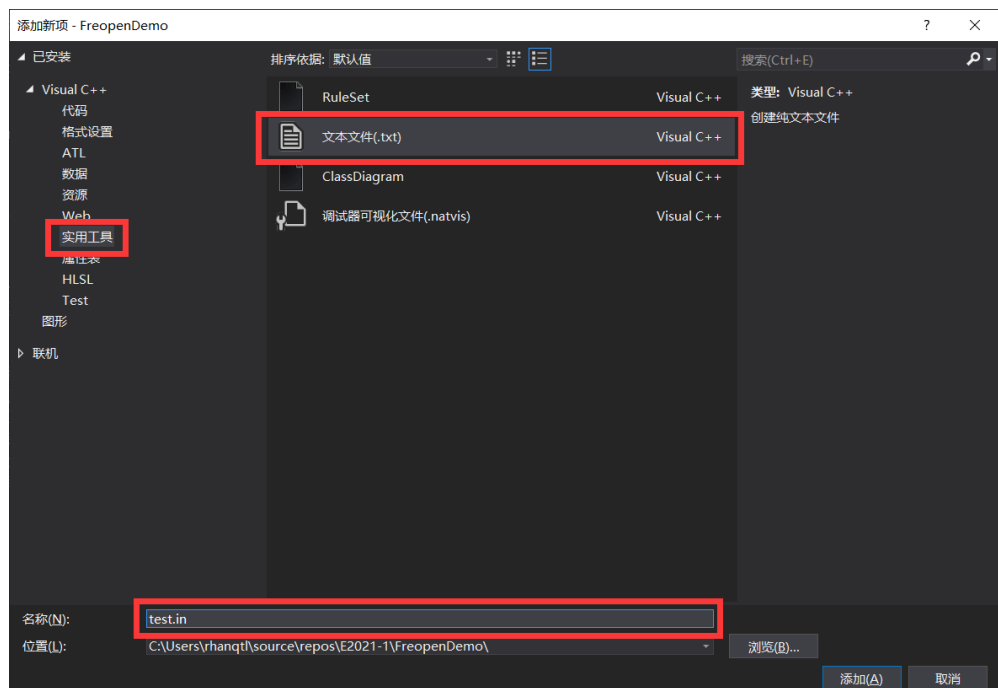
- 测试输入，文件名为 `test.in`，内容为 `2 3`
- 期望输出，文件名为 `expect.out`，内容为 `5`

添加文件参考步骤

- 右键单击项目的“资源文件”，添加 > 新建项



2. 创建文件



2. 编写代码

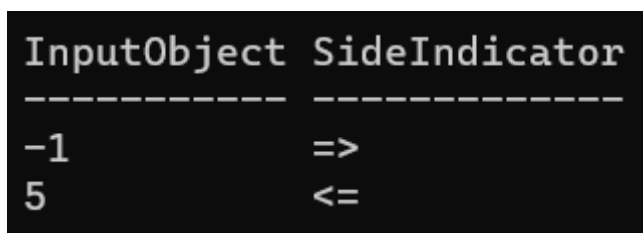
```
#include <iostream>

using namespace std;

int main() {
#define DEBUG
#ifdef DEBUG
    FILE *ifp = nullptr, *ofp = nullptr;
    freopen_s(&ifp, "test.in", "r", stdin);
    freopen_s(&ofp, "actual.out", "w", stdout);
#endif
    int a, b;
    cin >> a >> b;
    // 故意留下错误
    cout << (a - b) << endl;
}
```

3. 通过 VS 的“本地 Windows 调试器”按钮运行程序

4. 使用 `Compare-Object` 命令比对 `actual.out` 和 `expect.out`



InputObject	SideIndicator
-1	=>
5	<=

5. 根据测试情况修改代码

参考

[1] <https://docs.microsoft.com/en-us/cpp/c-runtime-library/reference/freopen-s-wfreopen-s?view=msvc-160>

[1] <https://devblogs.microsoft.com/scripting/use-powershell-to-compare-two-files/>

[2] <https://www.computerhope.com/unix/udiff.htm>