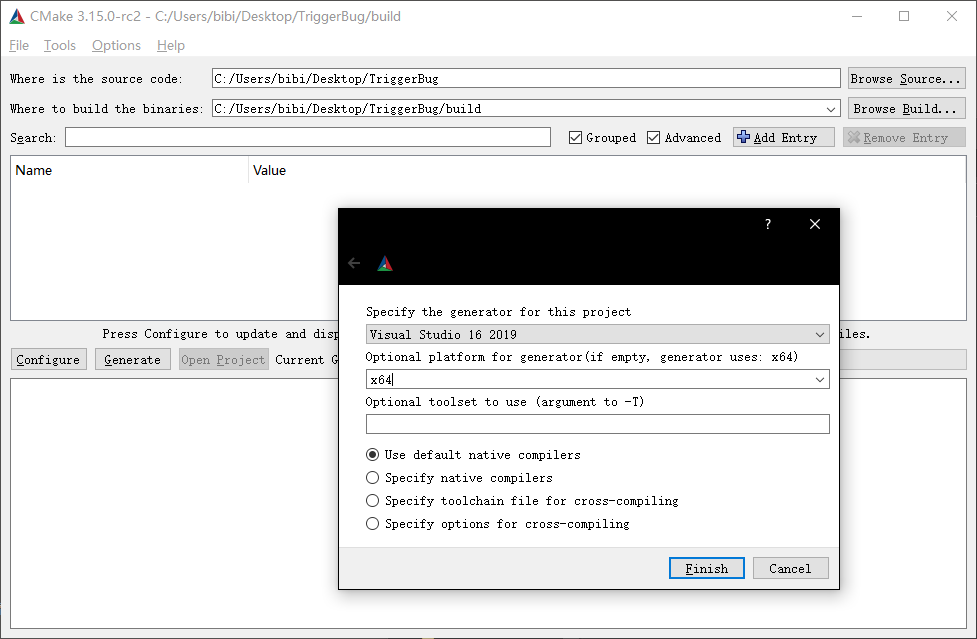
# 编译环境

1.先装Visual studio 2019

2.再装parallel\_studio\_xe\_2019\_update5\_cluster\_edition（作为vs2019编译器）

<https://software.intel.com/en-us/parallel-studio-xe/choose-download/free-trial-cluster-windows-c-fortran>

3.Cmake gui 3.15.0(Gui虽low，但香)





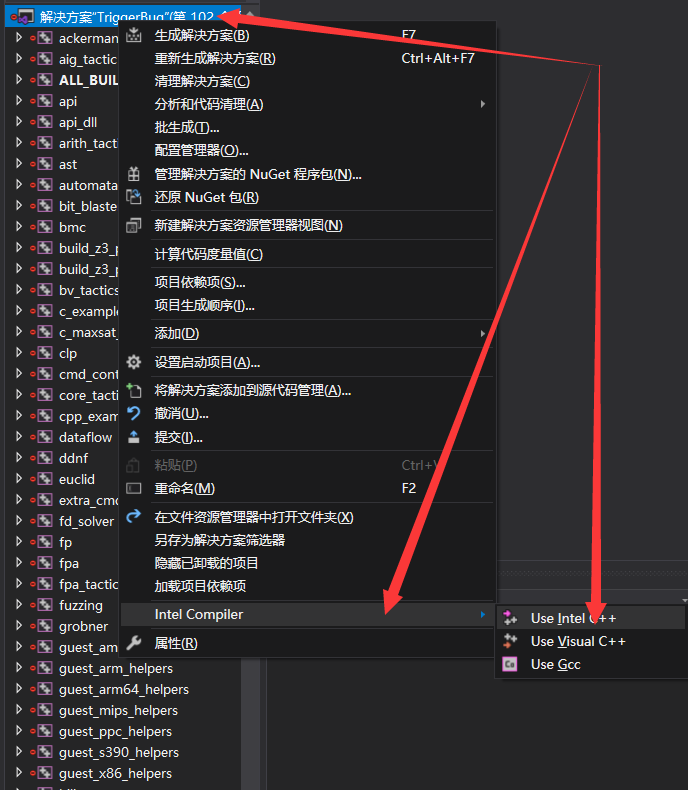
4.然后Generate,先不要open project

cd TriggerBug

python '.\support Qopenmp for Intel C++ compiler projects .py'（一键配置参数，手动配置非常费时间）

再打开项目

5.解决方案全部使用intel编译器进行编译，为了支持at&t风格汇编和高级语法

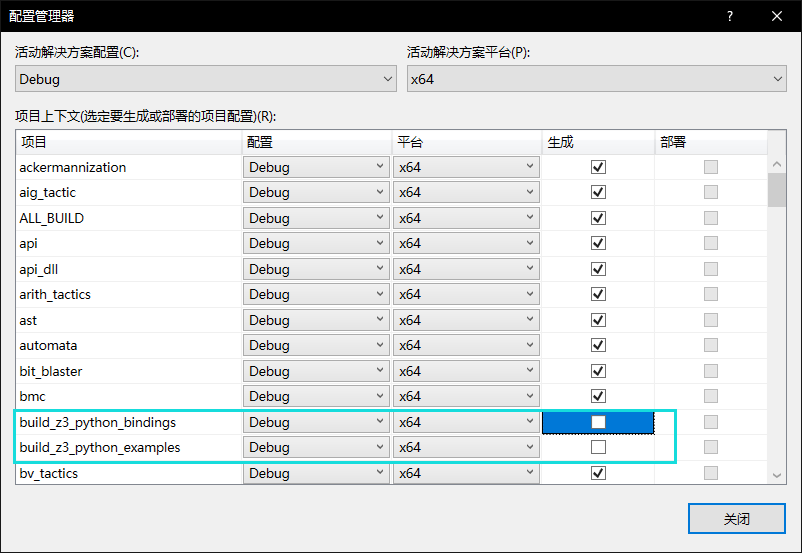


再关闭vs, 再次修改配置

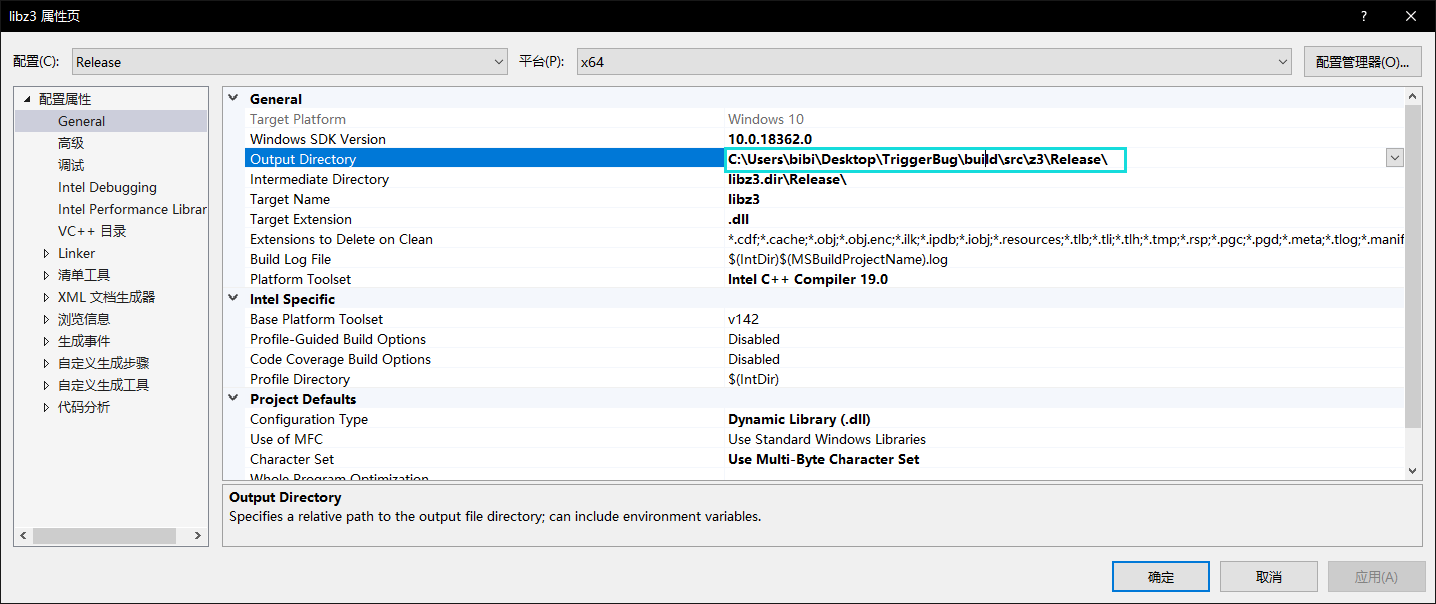
cd TriggerBug

python '.\support Qopenmp for Intel C++ compiler projects .py'

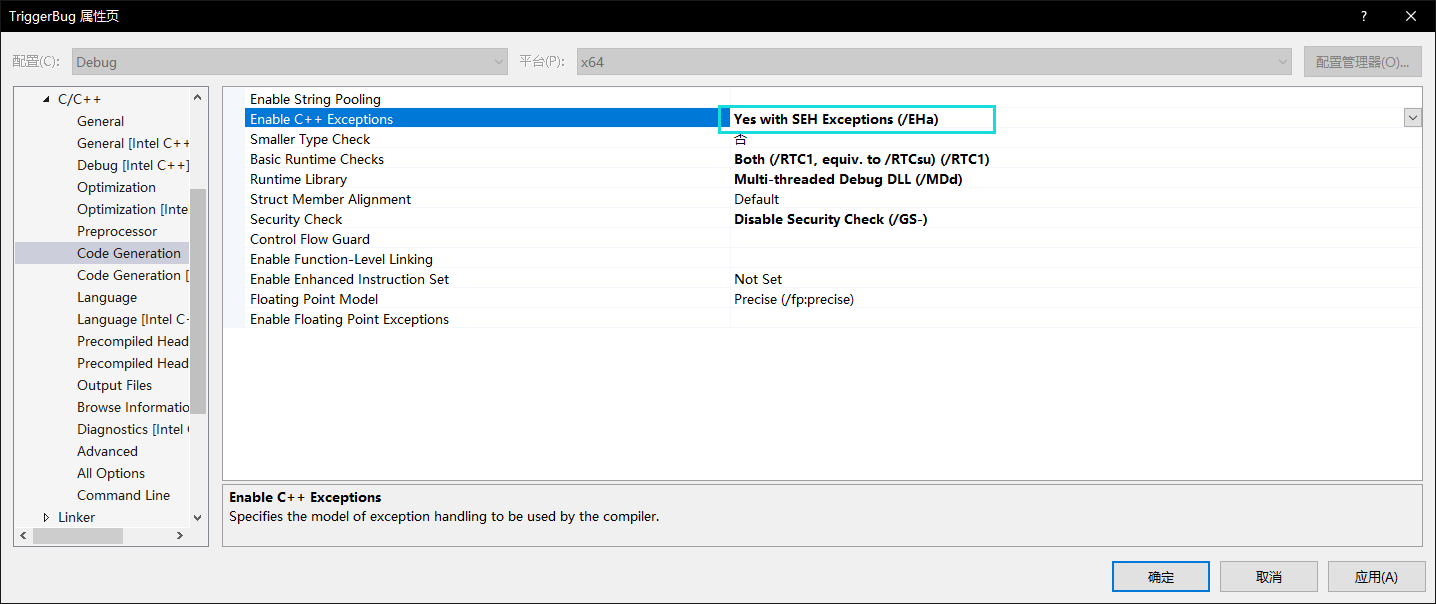
6.生成->配置管理项，不需要了就去掉吧，加快速度



7.注意修改libz3.dll输出路径，一般放..\..\Engine\[Release，Debug]



8.为TriggerBug、SimulationEngine项目设置异常模式（非必要，而且linux不支持）



注：设置Debug参数不要忘了Release也要配置

# 开发提示

\src\Engine\SimulationEngine\[Unop, Binop, Triop, Qop].hpp 中还有很多接口没有实现，遇到引擎报错可以自行完善该op,

如：

模拟机器码中有pshufb xmm0, xmm1

引擎报错unsupport ir op: PermOrZero8x16

宁可以去<https://software.intel.com/sites/landingpage/IntrinsicsGuide/>查找pshufb对应的sse函数\_mm\_shuffle\_epi8

然后添加到switch下

case Iop\_PermOrZero8x16: {dassert(a.bitn == 128); dassert(b.bitn == 128); return Vns(m\_ctx, \_mm\_shuffle\_epi8(a, b)); }//ok pshufb mm, mm

注 万分确认无误就加上//ok 机器码 到末尾。

方便我进行验证，因为op的实现不正确，该错误会导致模拟求解得不到理想解，而且查证非常麻烦，往往会花几天才能查出来bug.