ITK学习笔记

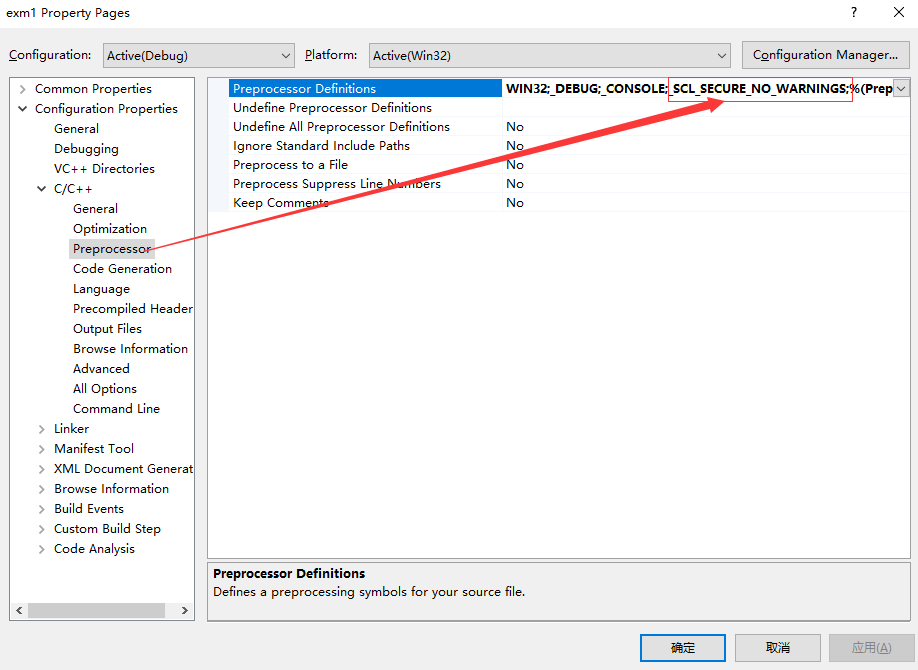
1. **报错提示：**

error C4996: 'std::\_Copy\_impl': Function call with parameters that may be unsafe - this call relies on the caller to check that the passed values are correct. To disable this warning, use -D\_SCL\_SECURE\_NO\_WARNINGS. See documentation on how to use Visual C++ 'Checked Iterators'

1> c:\program files (x86)\microsoft visual studio 11.0\vc\include\xutility(2157) : see declaration of 'std::\_Copy\_impl'

错误原因：安全检查

解决方法：在预处理中添加\_SCL\_SECURE\_NO\_WARNING;



1. **ITK中类的创建**

typedef unsigned char PixelType;

const unsigned int Dimension = 3;

typedef itk::Image<PixelType, Dimension> ImageType;

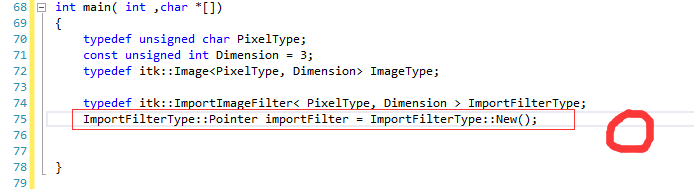
typedef itk::ImportImageFilter< PixelType, Dimension > ImportFilterType;

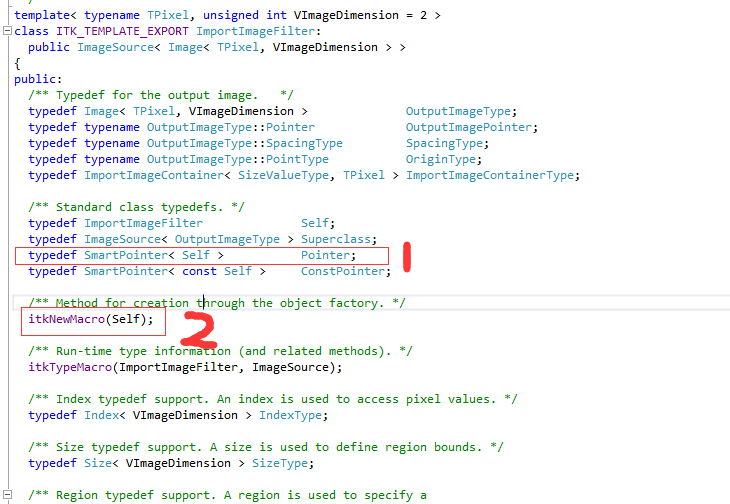
ImportFilterType::Pointer importFilter = ImportFilterType::New();

// 这一句是什意思？

在ITK中,这个New()函数是如何创建对象的？

源代码摘抄如下：







1. **对象工场模式初探**

#include<iostream>

#include<stdexcept>

#include<cstddef>

#include<string>

#include<vector>

using namespace std;

class Shape

{

public:

virtual void draw() = 0;

virtual void erase() = 0;

virtual ~Shape(){}

%%% 这个子类是如何访问的？

class BadShapeCreation : public logic\_error

{

public:

BadShapeCreation(string type):logic\_error("Cannot create type"){}

};

static Shape \*factory(const string &type)

throw(BadShapeCreation);

};

class Circle:public Shape

{

private:

Circle() {}

friend class Shape; %%% 为什么申明为友元类？

public:

void draw(){cout<<"Circle::draw"<<endl;}

void erase(){cout<<"Circle::erase"<<endl;}

~Circle(){cout<<"Circle::~Circle"<<endl;}

};

class Square:public Shape

{

Square(){}

friend class Shape;

public:

void draw(){cout<<"Square::draw"<<endl;}

void erase(){cout<<"Square::erase"<<endl;}

~Square(){cout<<"Square::~Square"<<endl;}

};

Shape \*Shape::factory(const string &type)

throw(Shape::BadShapeCreation) %% 这里为什么会加这个throw

{

if(type == "Circle")

return new Circle;

if(type == "Square")

return new Square;

throw BadShapeCreation(type);

}

char \*sl[]={"Circle","Square","Square","Circle","Circle","Square"};

int main()

{

vector<Shape \*> shapes;

try

{

for(size\_t i =0; i<sizeof sl/sizeof sl[0] ;i++)

shapes.push\_back(Shape::factory(sl[i]));

//cout<<sizeof(sl[i])<<typeid(sl[i]).name()<<endl;

}

catch(Shape::BadShapeCreation e)

{

cout<<e.what()<<endl;

system("pause");

return -1;

}

for(size\_t i = 0;i<shapes.size();i++)

{

shapes[i]->draw();

shapes[i]->erase();

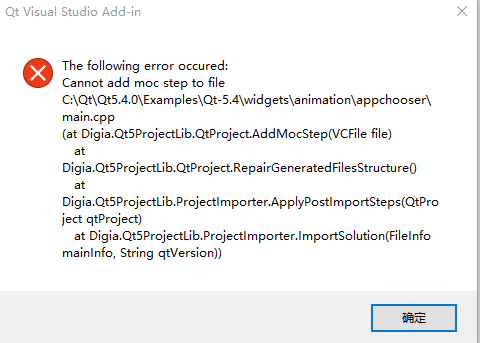
}

system("pause");

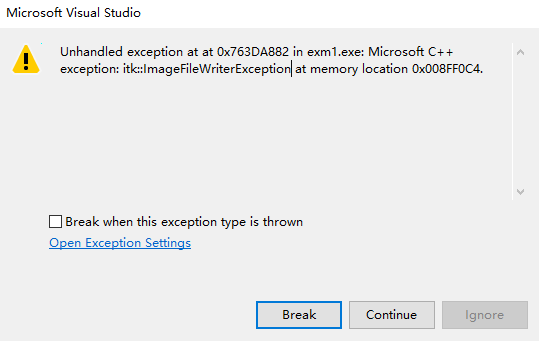
return 0;

}

1. **Qt addin VS2012 出错**



1. **运行ITK例子时出错**



错误原因在 ITK中没有注册 对象工场

解决办法：在里面添加一句话

#include"itkTIFFImageIOFactory.h"

itk::TIFFImageIOFactory::RegisterOneFactory();

1. **CMake构建一个ITK工程简单文档**

cmake\_minimum\_required(VERSION 2.8)

project(itkTest4)

Find\_package(ITK)

if(ITK\_FOUND )

include( ${ITK\_USE\_FILE} )

else( ITK\_FOUND )

message(FATAL\_ERROR"cannt build without itk. please set itk\_dir")

endif(ITK\_FOUND)

add\_executable( Image4 Image4.cpp )

target\_link\_libraries(Image4 ${ITK\_LIBRARIES})

1. 创建一个ImageRegionIterator的迭代器

typedef itk::ImageRegionIteratorWithIndex< ImageType > IteratorType;

IteratorType it( image, image->GetBufferedRegion() );

1. 在ITK中设置GaussianOperator参数时有个Maximum error ，这个参数代表啥意思？

原文中的说明

**[GaussianOperator](https://itk.org/Doxygen/html/classitk_1_1GaussianOperator.html" \o "A NeighborhoodOperator whose coefficients are a one dimensional, discrete Gaussian kernel...)** takes two parameters:

(1) The floating-point variance of the desired Gaussian function.

(2) The "maximum error" allowed in the discrete Gaussian function. "Maximum errror" is defined as the difference between the area under the discrete Gaussian curve and the area under the continuous Gaussian. Maximum error affects the Gaussian operator size. Care should be taken not to make this value too small relative to the variance lest the operator size become unreasonably large.

References: The Gaussian kernel contained in this operator was described by Tony Lindeberg (Discrete Scale-Space Theory and the Scale-Space Primal Sketch. Dissertation. Royal Institute of Technology, Stockholm, Sweden. May 1991.).

DiscreteGaussianImageFilter()

