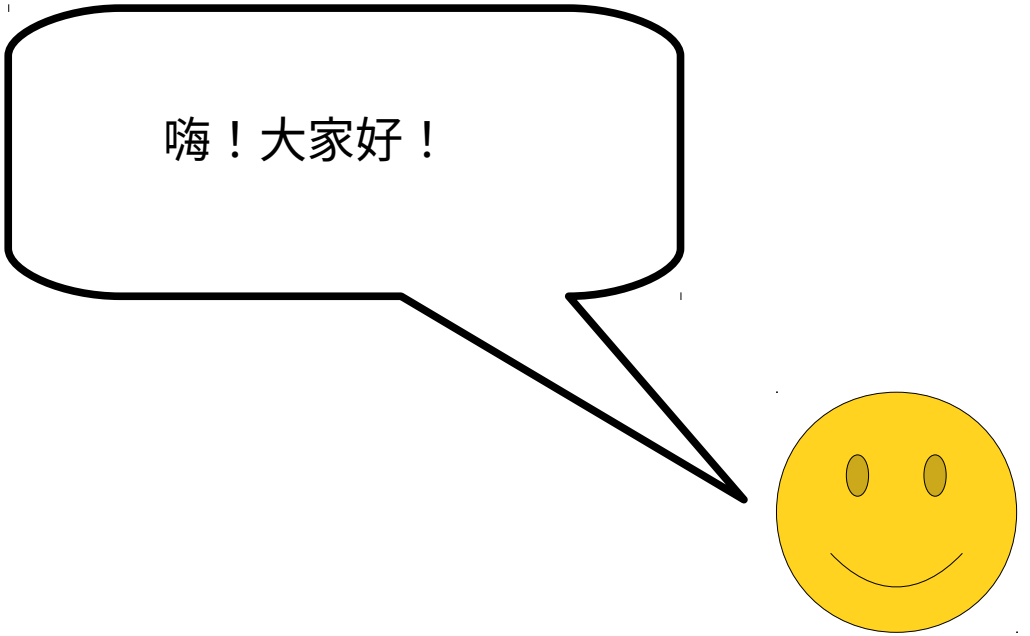
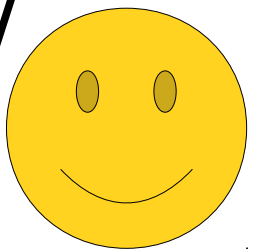
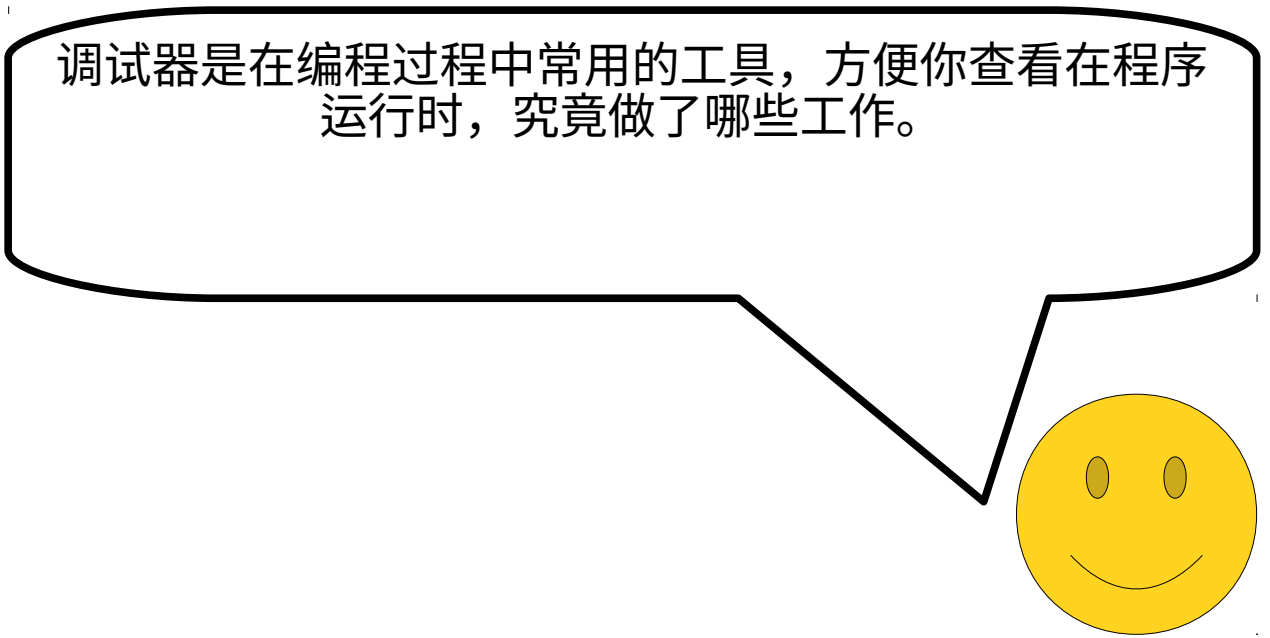
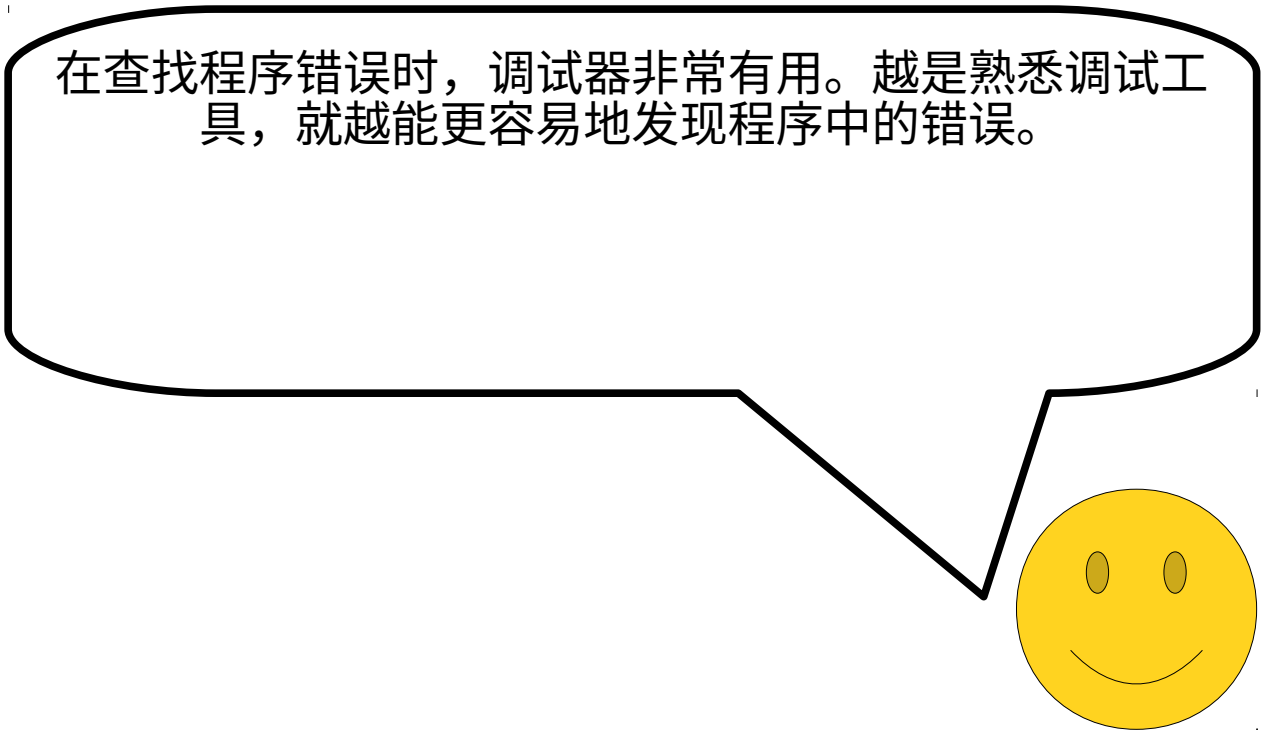
作业 0：使用调试器

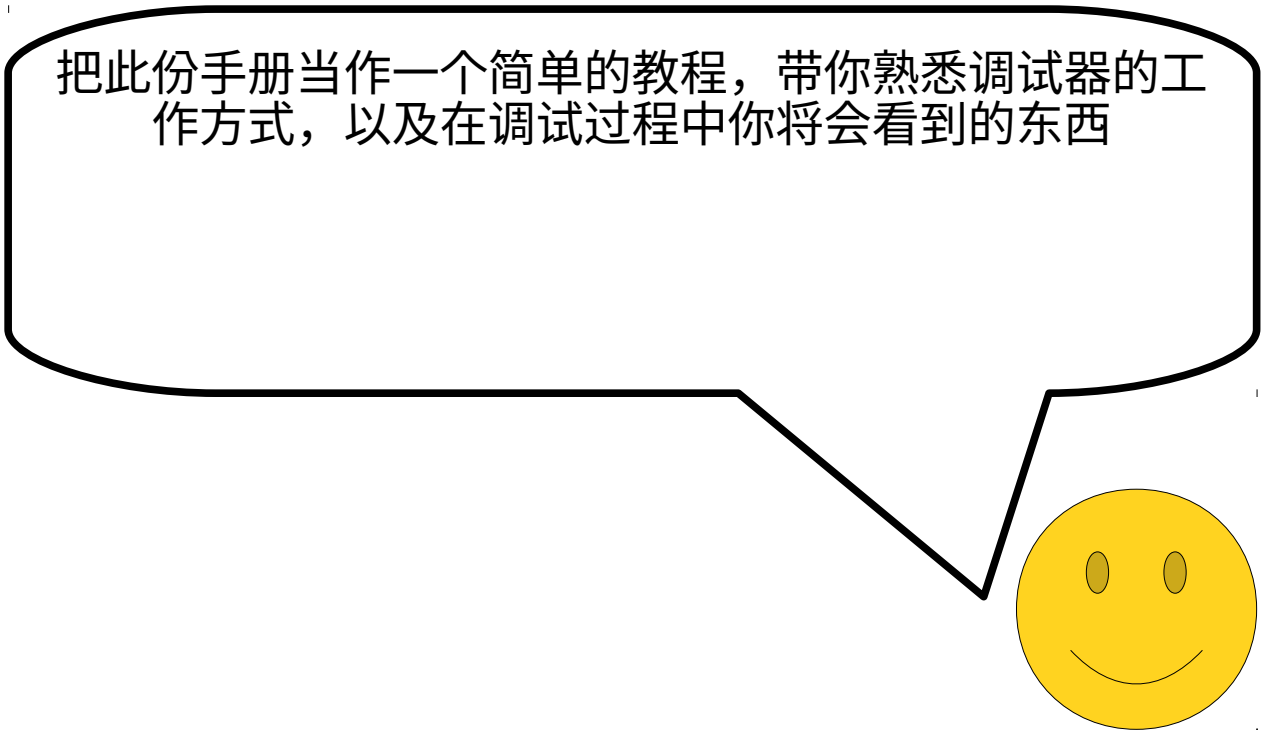

嗨！大家好！

作为作业 0 的一部分，希望大家练习下 Qt Creator 的调试器

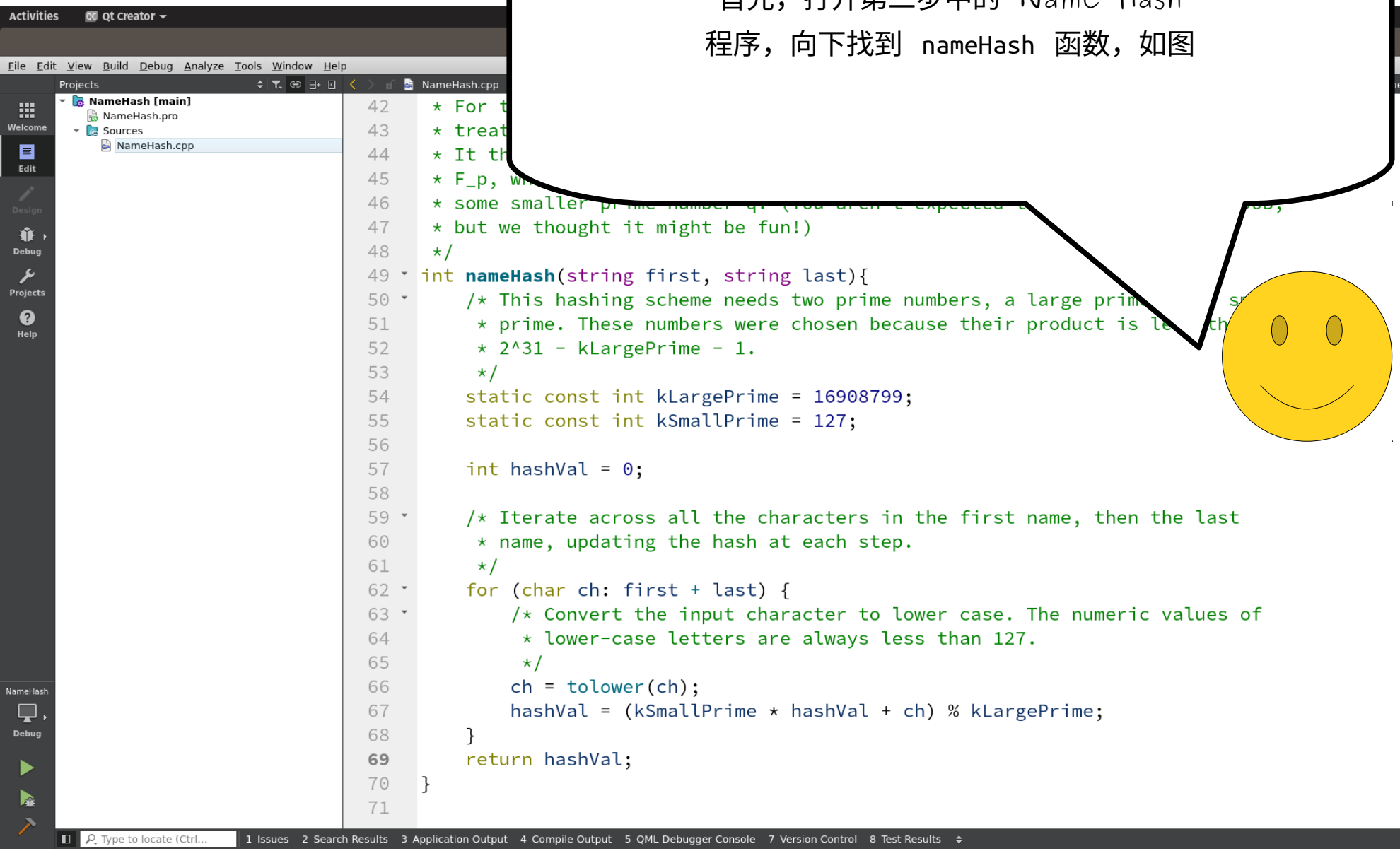



调试器是在编程过程中常用的工具，方便你查看在程序运行时，究竟做了哪些工作。


在查找程序错误时，调试器非常有用。越是熟悉调试工具，就越能更容易地发现程序中的错误。


把此份手册当作一个简单的教程，带你熟悉调试器的工作方式，以及在调试过程中你将会看到的东西

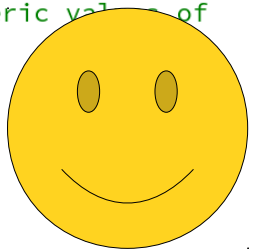
首先，打开第二步中的 Name Hash
程序，向下找到 nameHash 函数，如图




Activities
Qt Creator
Jan 4 3:09 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
NameHash.cpp
nameHash(std::string, std::string) -> int
Unix (LF)
Line: 69, Col: 20
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
把鼠标移动到 66 行前面。
此时，点击一下鼠标左键！
42 * For those of you who are more mathematically inclined, this function
43 * treats each character in the input name as a number between 0 and 128.
44 * It then uses them as coefficients in a polynomial over the finite field
is for CS106B,
and a small
less than
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the
60 * name, updating the hash at each step.
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower case. The
64 * lower-case letters are always less than 127.
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kLargePrime;
68 }
69 return hashVal;
70 }
71
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results

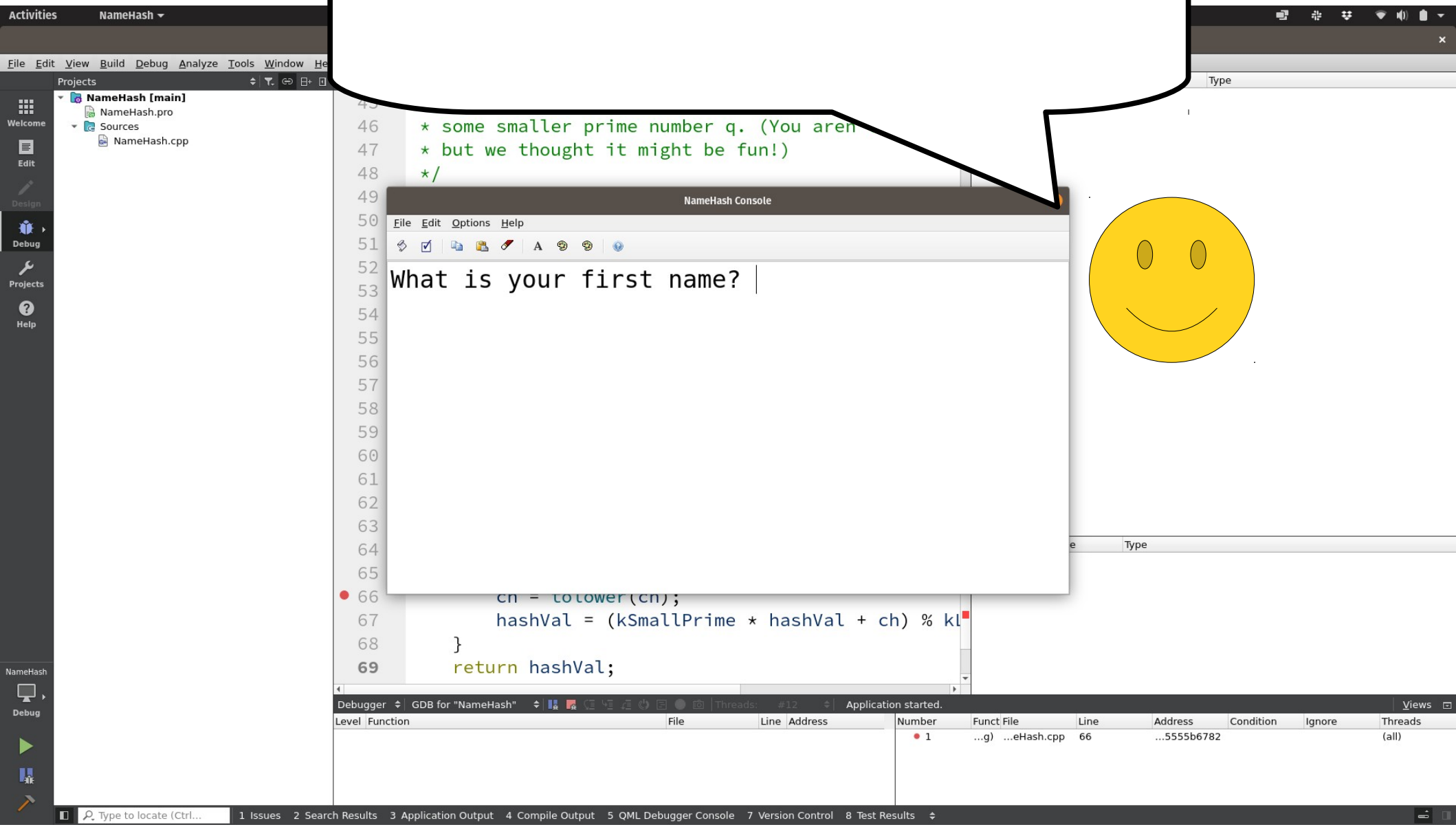
完成后，你会在行号前面看到一个红色圆点

这个圆点就叫做 breakpoint 断点。如果在 Debug 模式下运行程序，程序会这一行代码前暂停，并打开调试器。
通过这样的方式，我们可以追踪程序的运行过程。




Activities
Qt Creator
Jan 4 3:11 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
NameHash.cpp
nameHash(std::string, std::string) -> int
Unix (LF)
Line: 69, Col: 20
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
42 * For those of you who are more mathematically inclined, this function
43 * treats each character in the input name as a number between 0 and 128.
44 * It then uses them as coefficients in a polynomial over the finite field
45 * F_p, where p is a large prime number, and evaluates that polynomial at
46 * some smaller prime number q. (You aren't expected to know this for CS106B,
47 * but we thought it might be fun!)
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers, a large prime and a small
than
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas umeric va of
64 * lower-case letters are always less than 127.
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kLargePrime;
68 }
69 return hashVal;
70 }
71
现在我们要在Debug模式下执行程序。首先点击图中的“ 调试 ”按钮（左下角白色箭头处）。
点击完成后……
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results

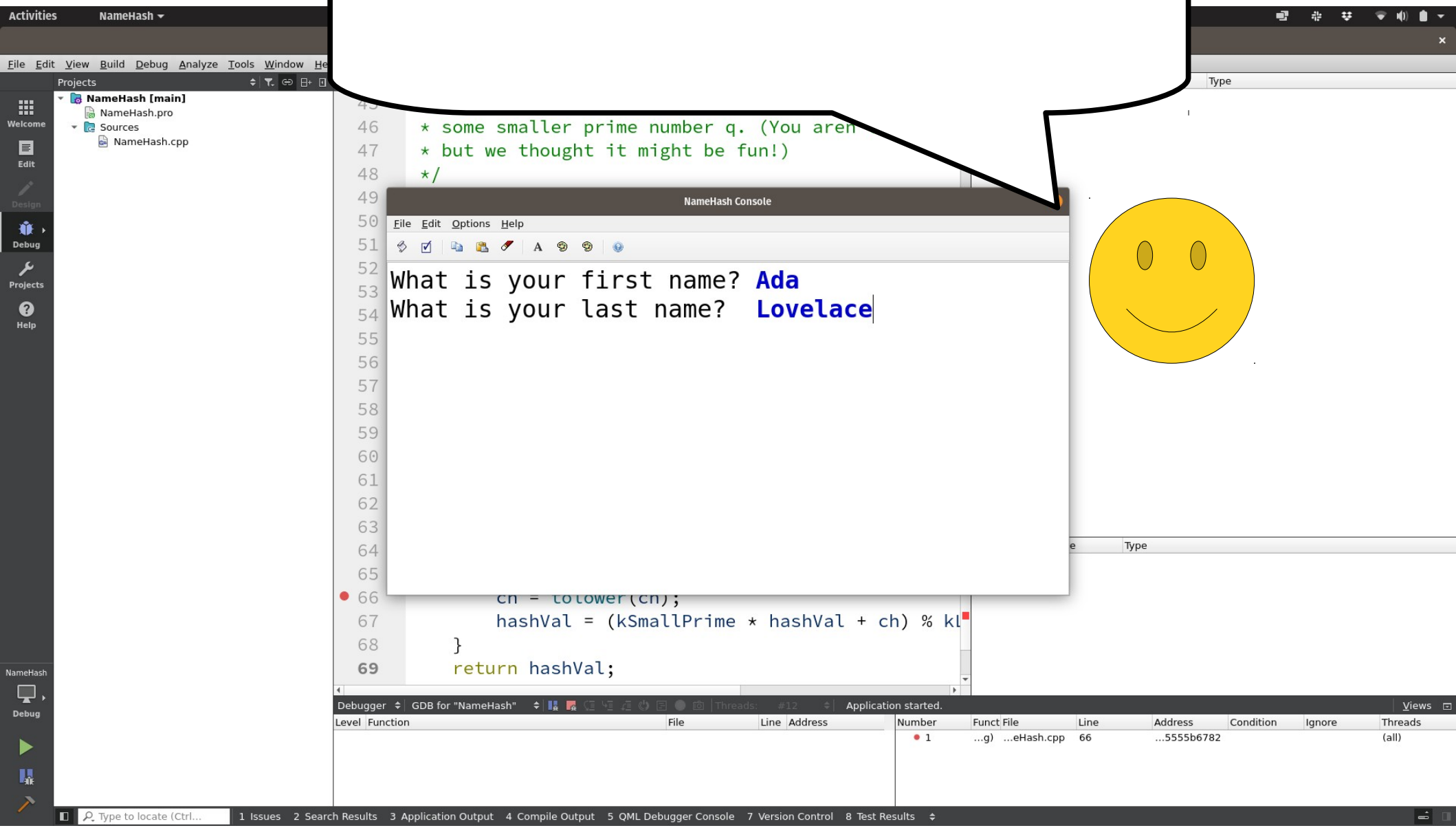
……你将会看到这样的界面。此时，Qt 界面也会弹出一些额外的面板，我们后面会一个个看。



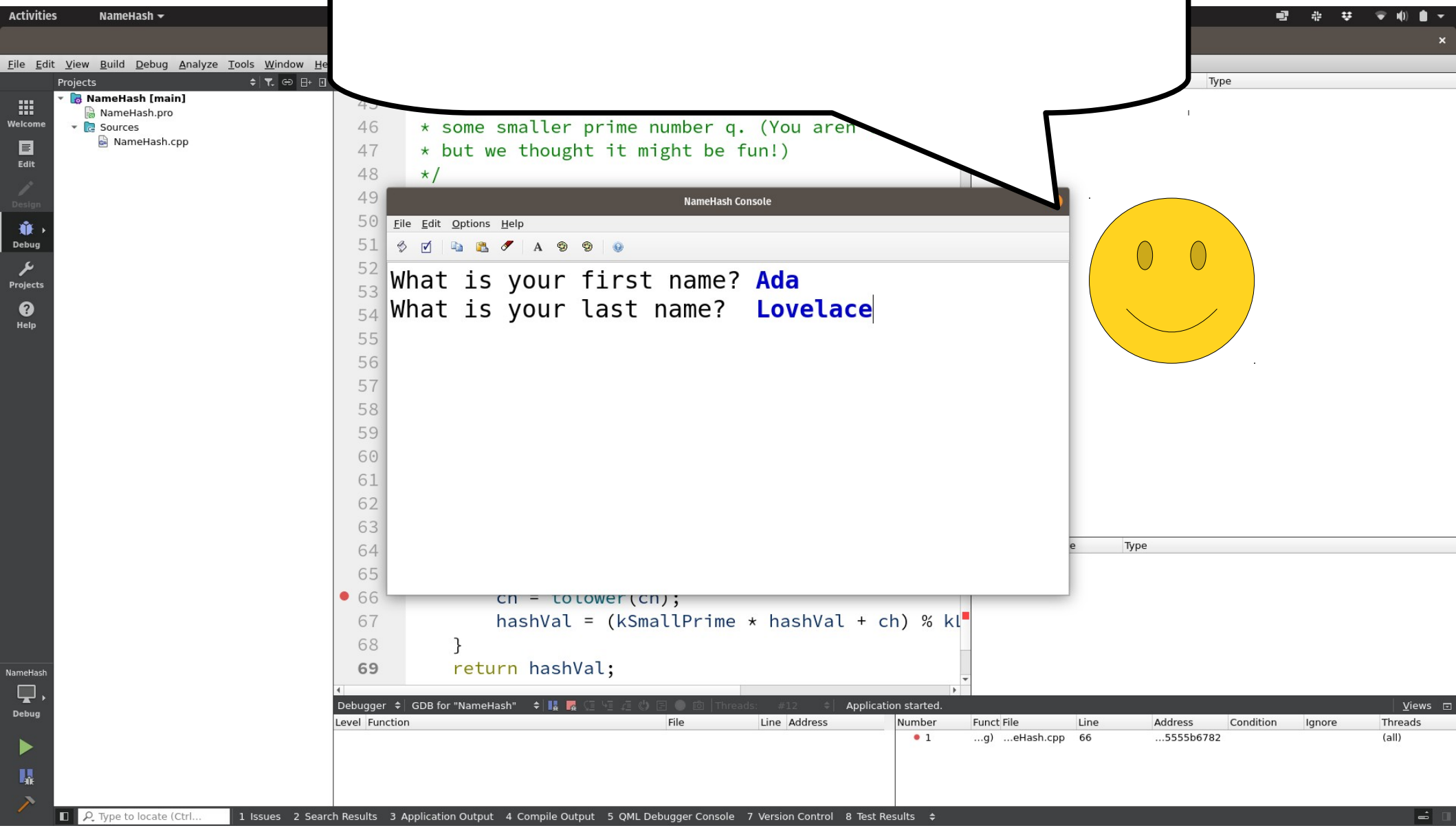
此时，切换到英文输入法，输入 **Ada** 后按回车，**注意不要输入你自己的名字**！

（当然，除非你就叫 Ada）

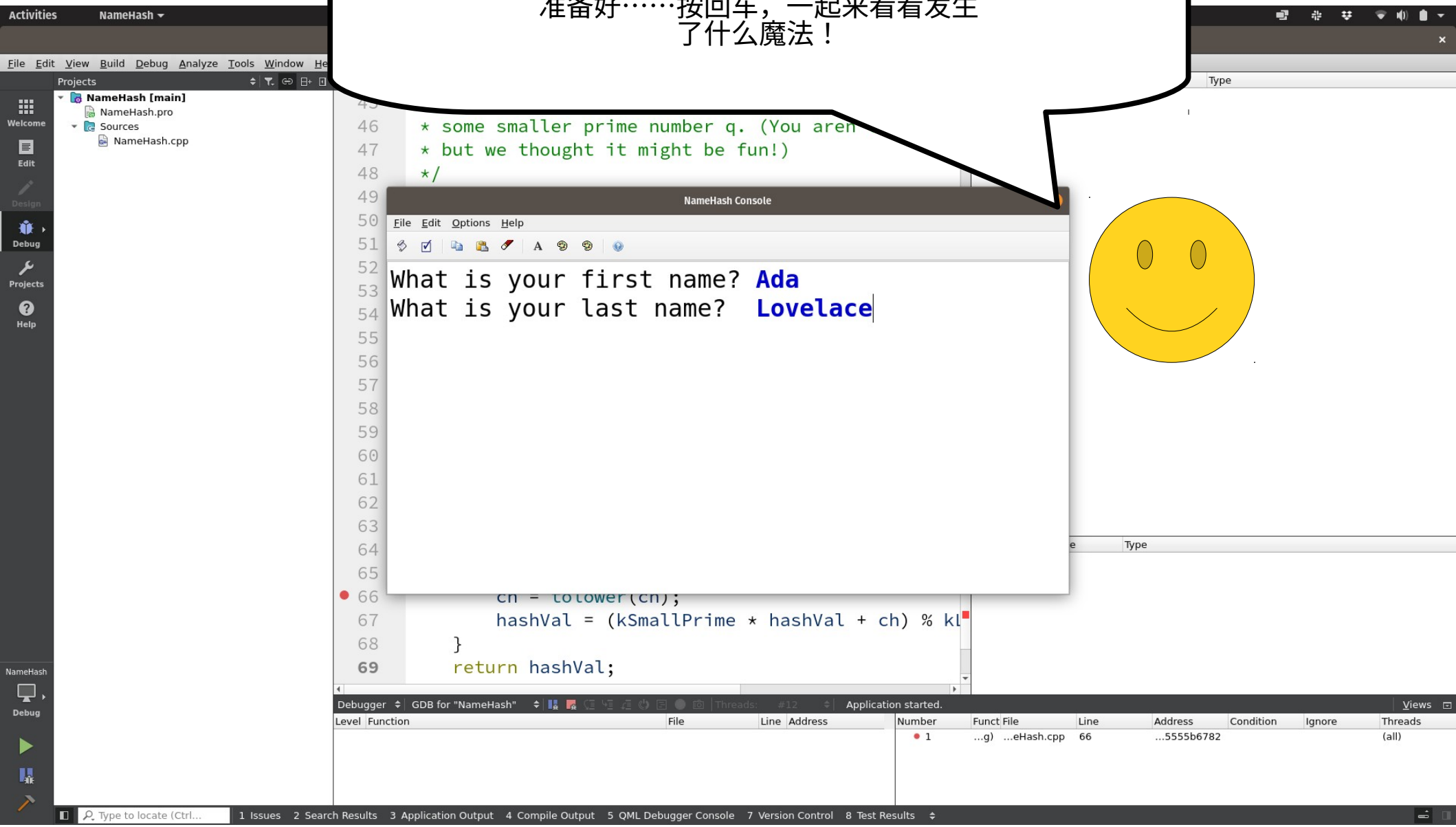
接下来，再输入 **Lovelace** **但是不要再按回车了**



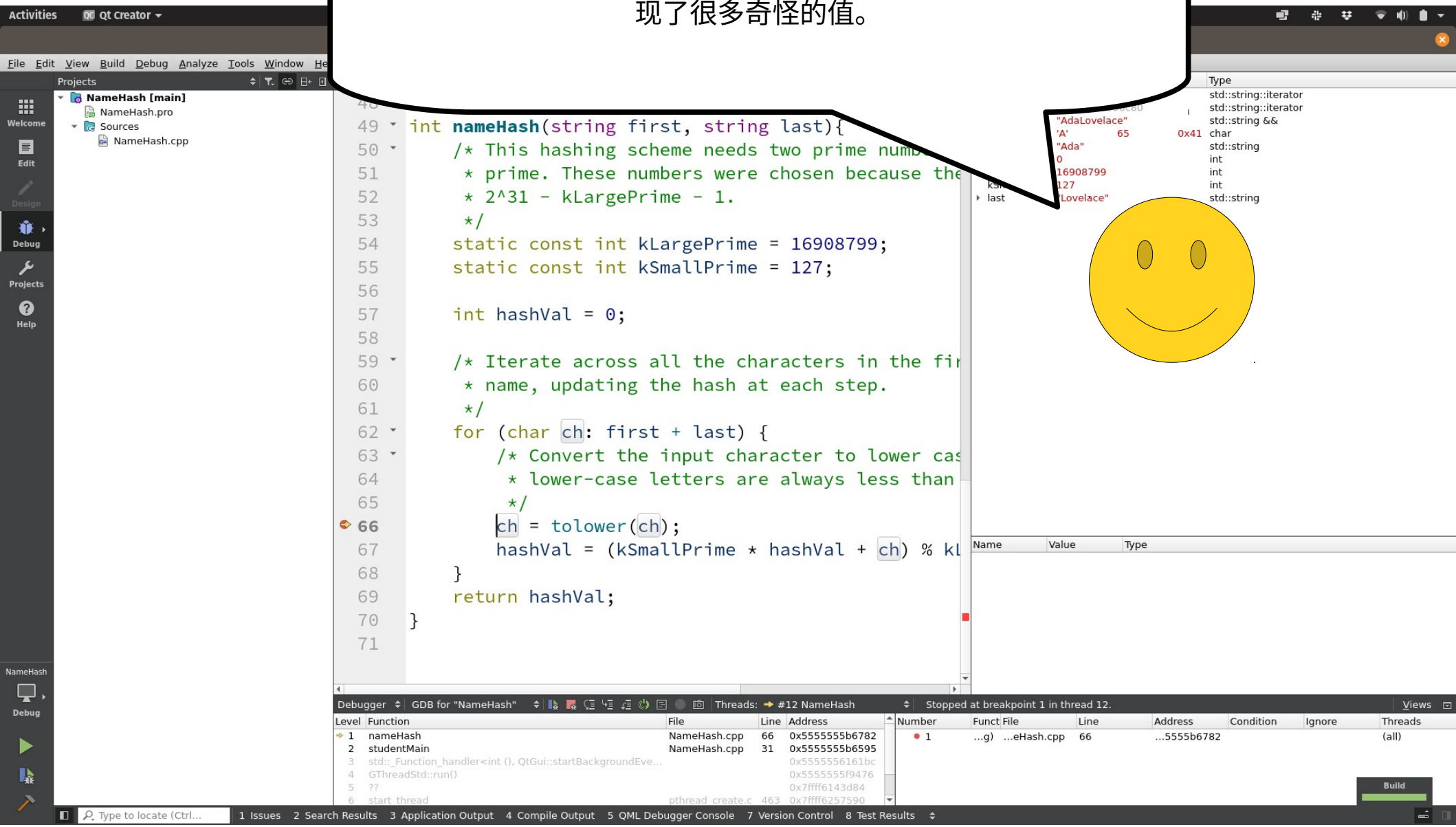
因为一旦你按下回车，Qt 就会弹出一系列对话框。
不要担心，这是正常的。



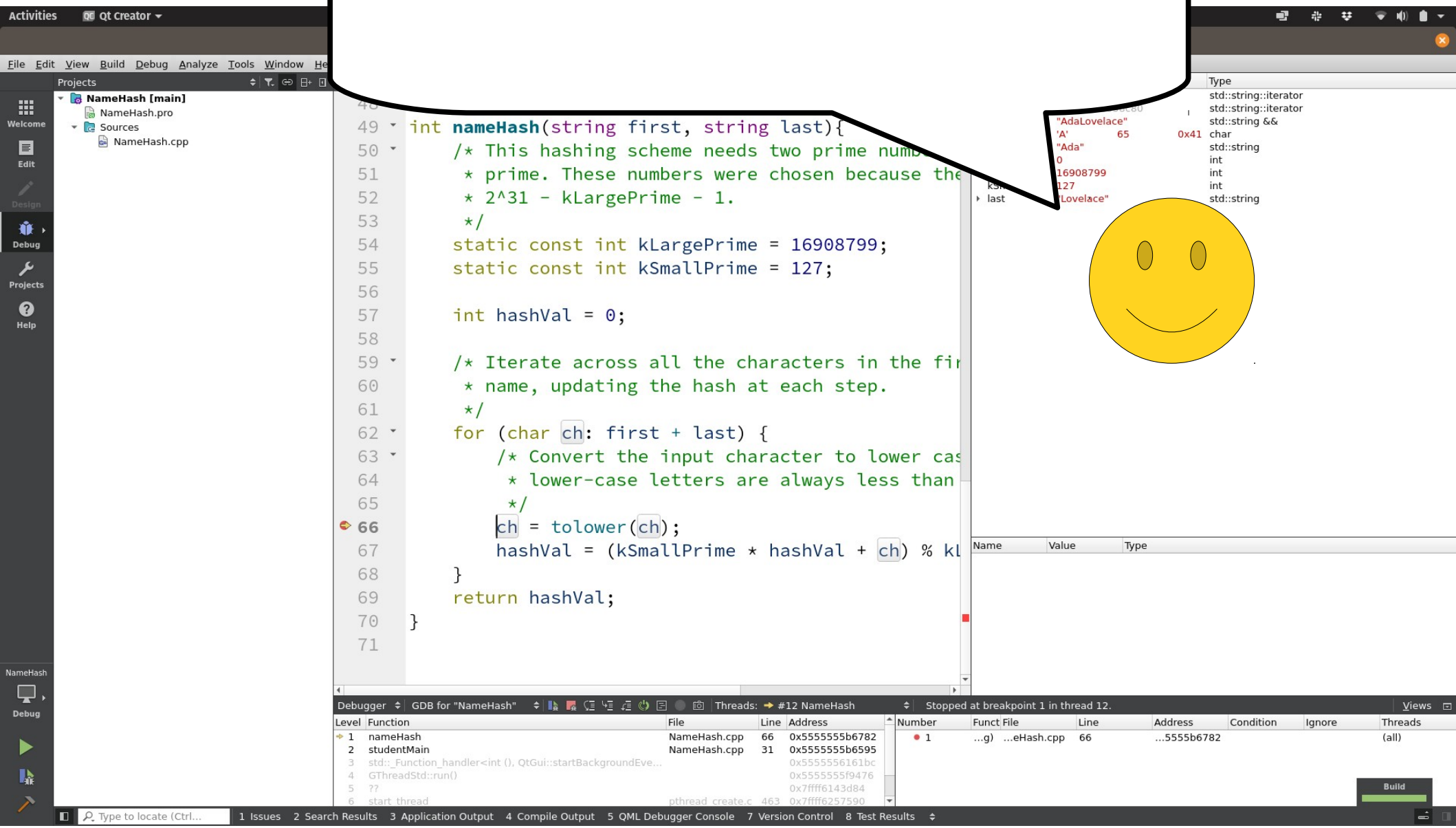
如前所述，你应该做好心理准备了。
准备好……按回车，一起来看看发生了什么魔法！



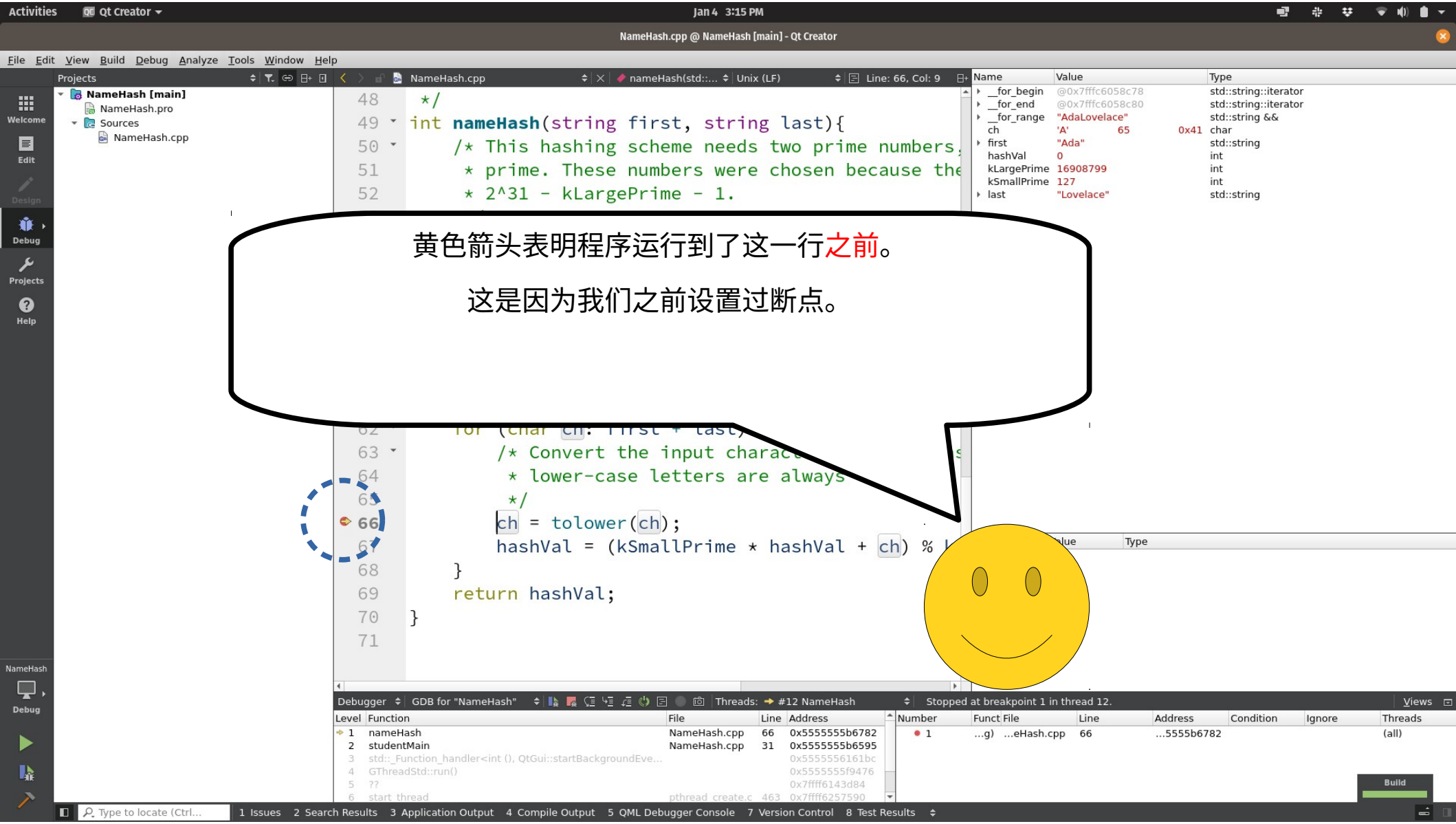
哇哦！我们回到了 Qt Creator 并且在弹出的面板中，出现了很多奇怪的值。

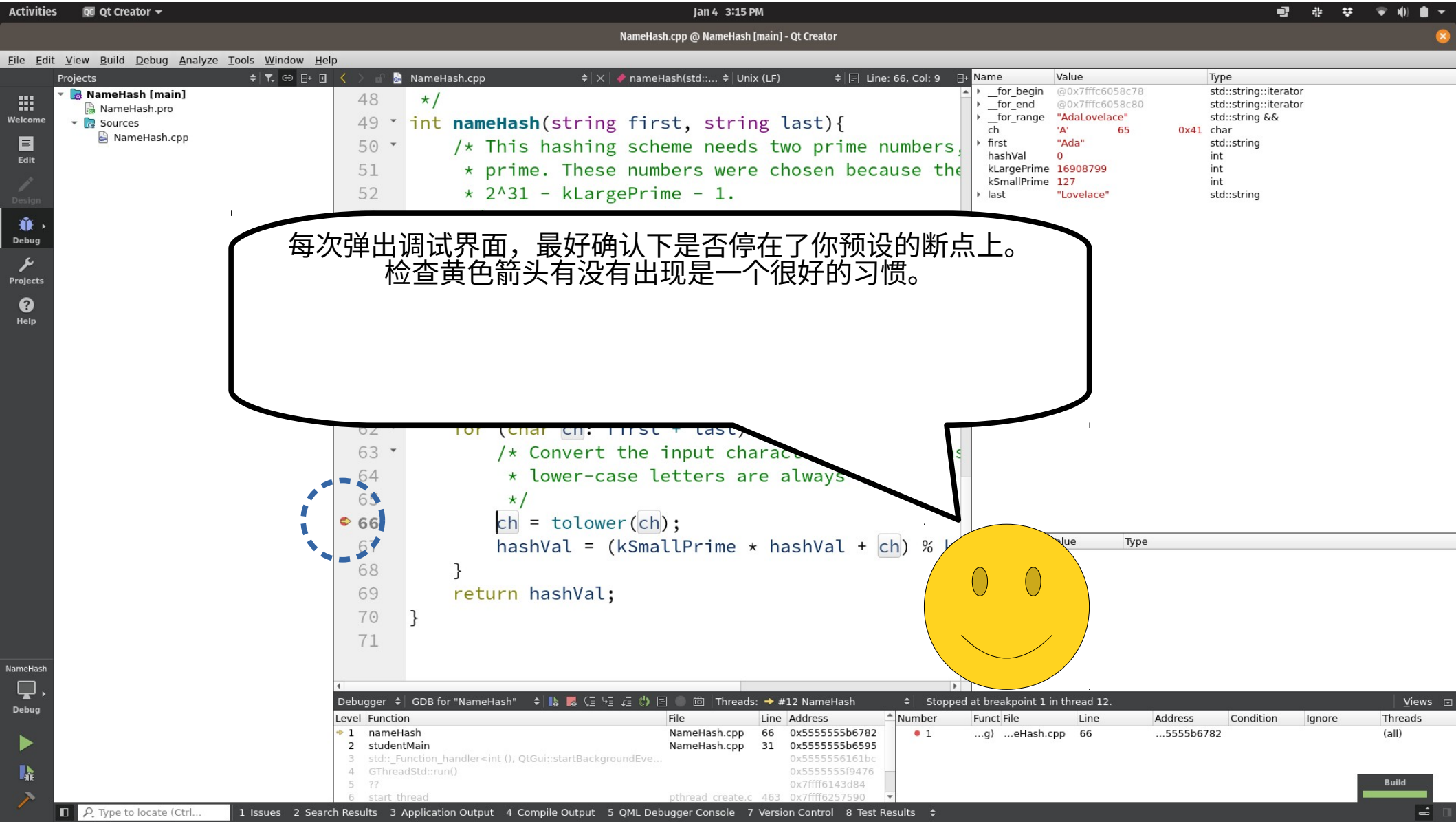


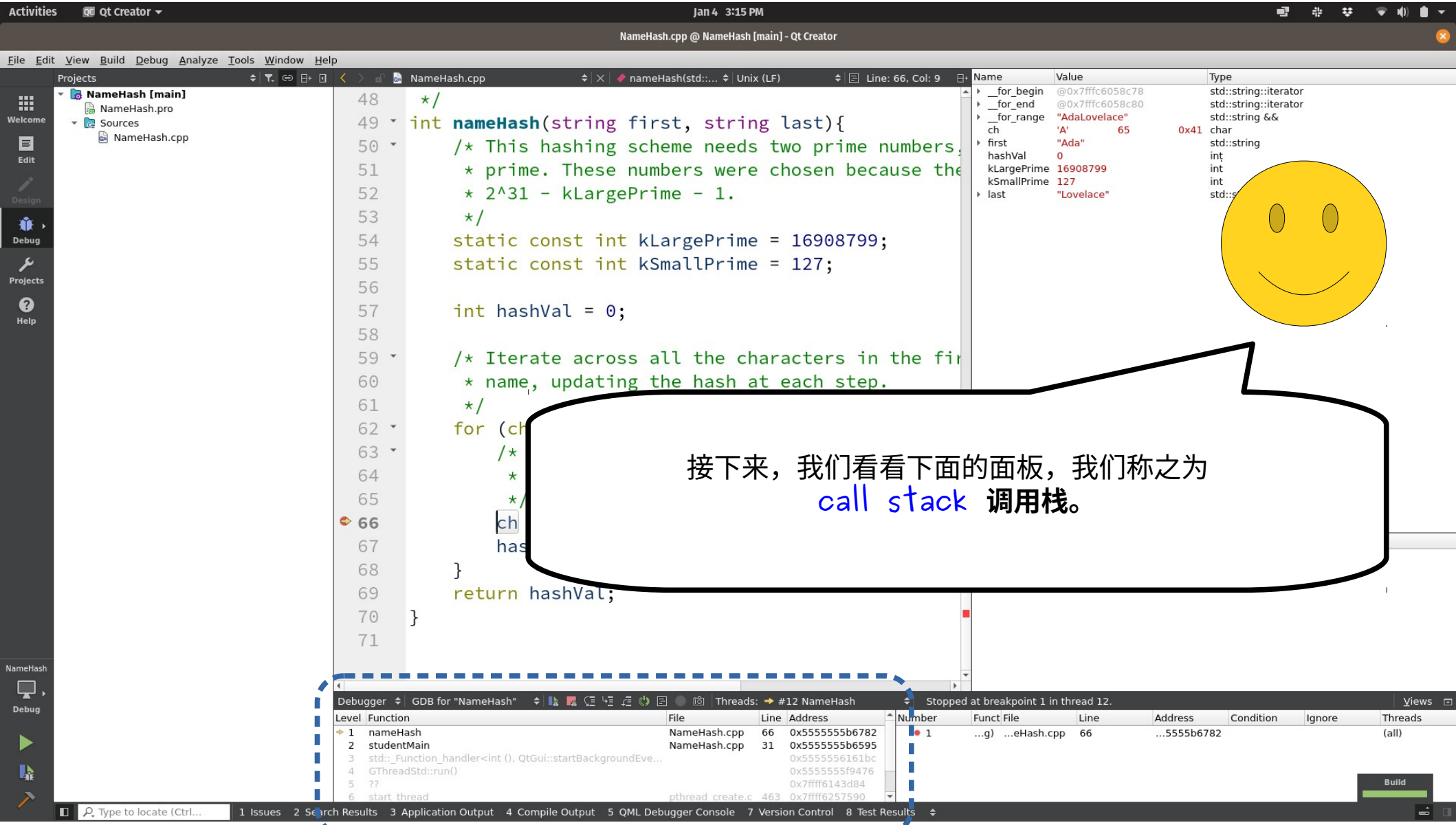
有很多东西值得说道说道，让我们一起来看看吧。

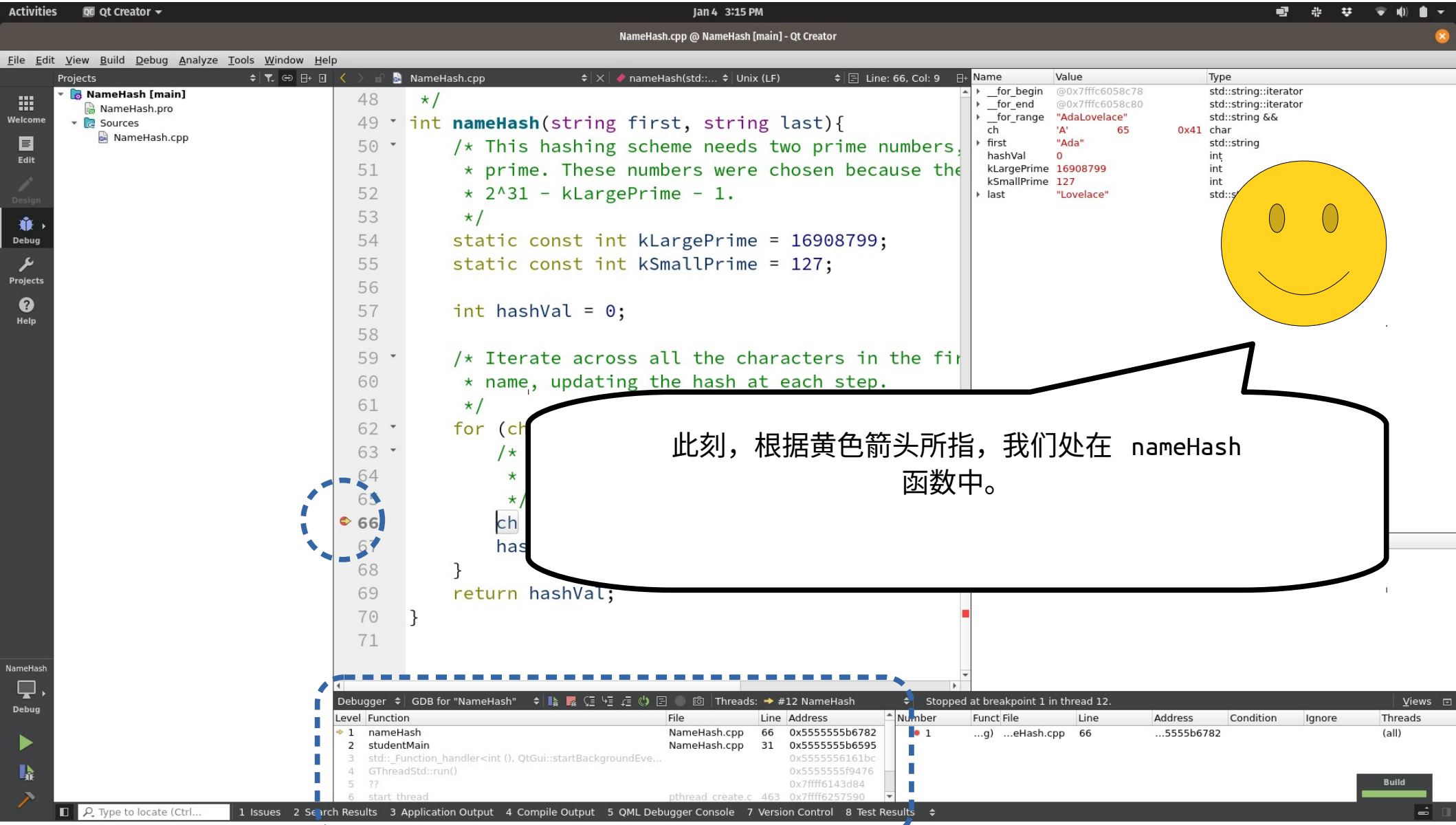


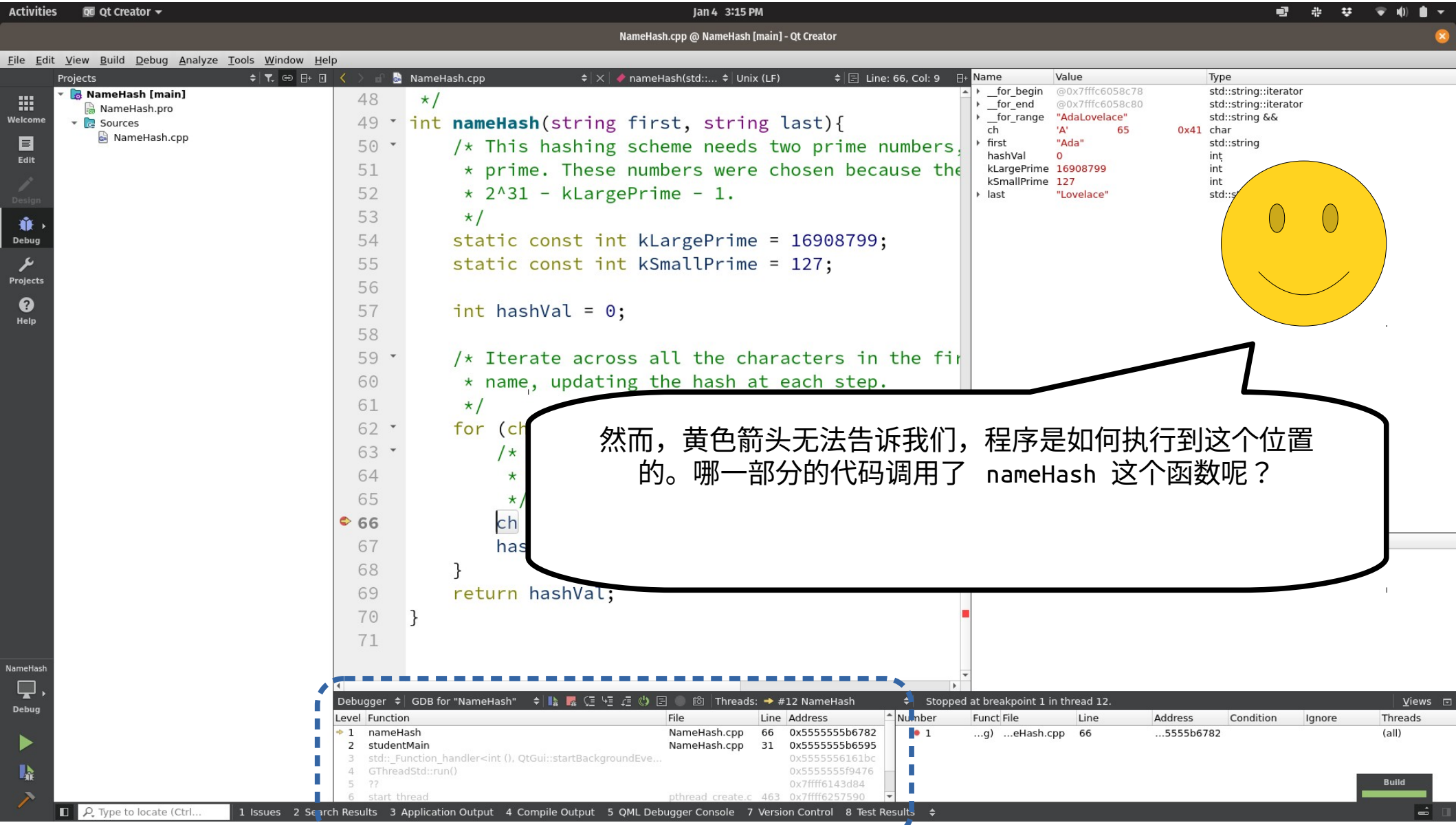

Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::... Unix (LF) Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
63 /* Convert the input charac
64 * lower-case letters are always
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % k
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
首先，注意之前的红色圆点有了一个黄色箭头！
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Build
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

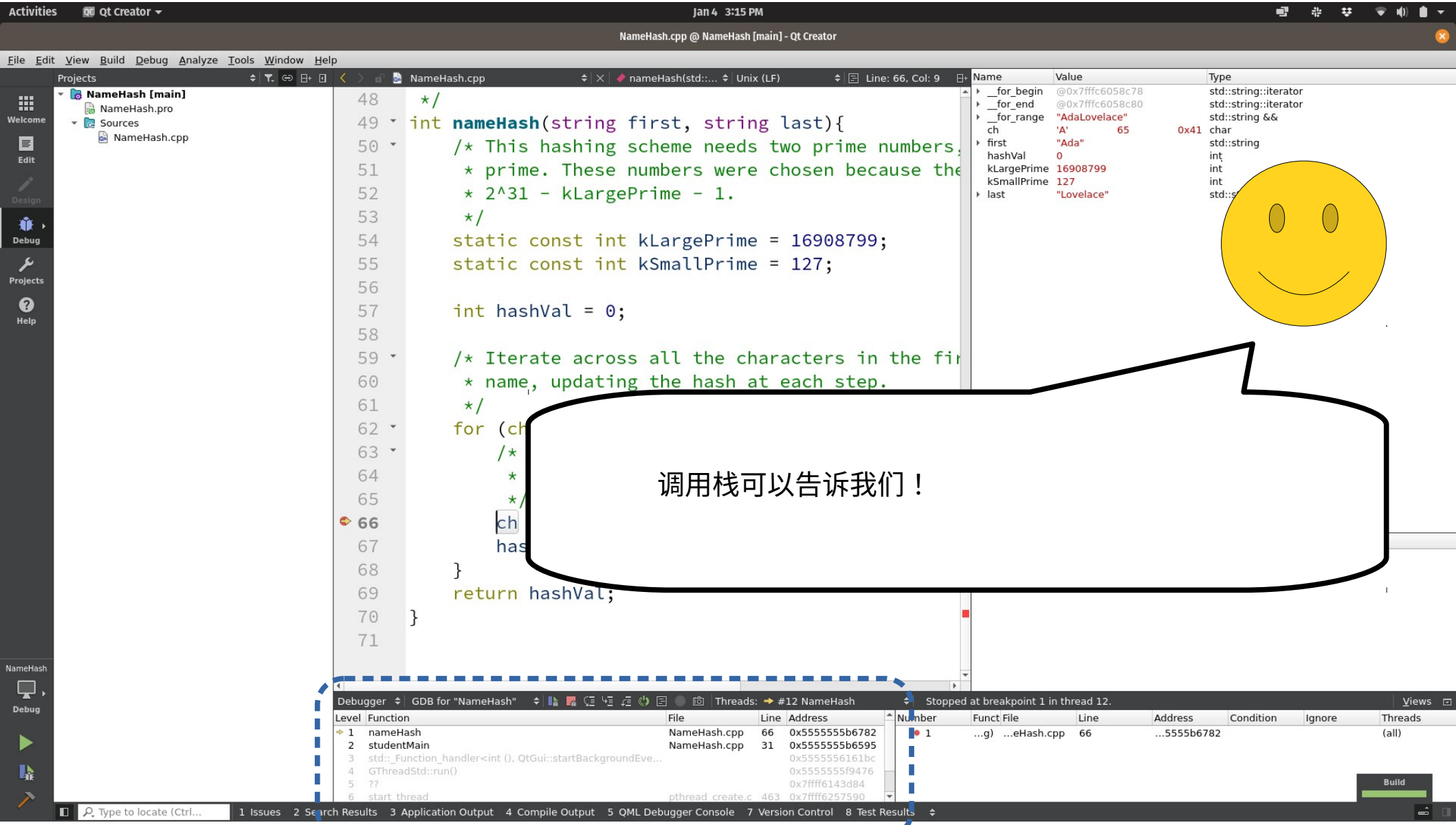

Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
63 /* Convert the input charact
64 * lower-case letters are always
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % k
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
黄色箭头表明程序运行到了这一行之前。
这是因为我们之前设置过断点。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Build
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

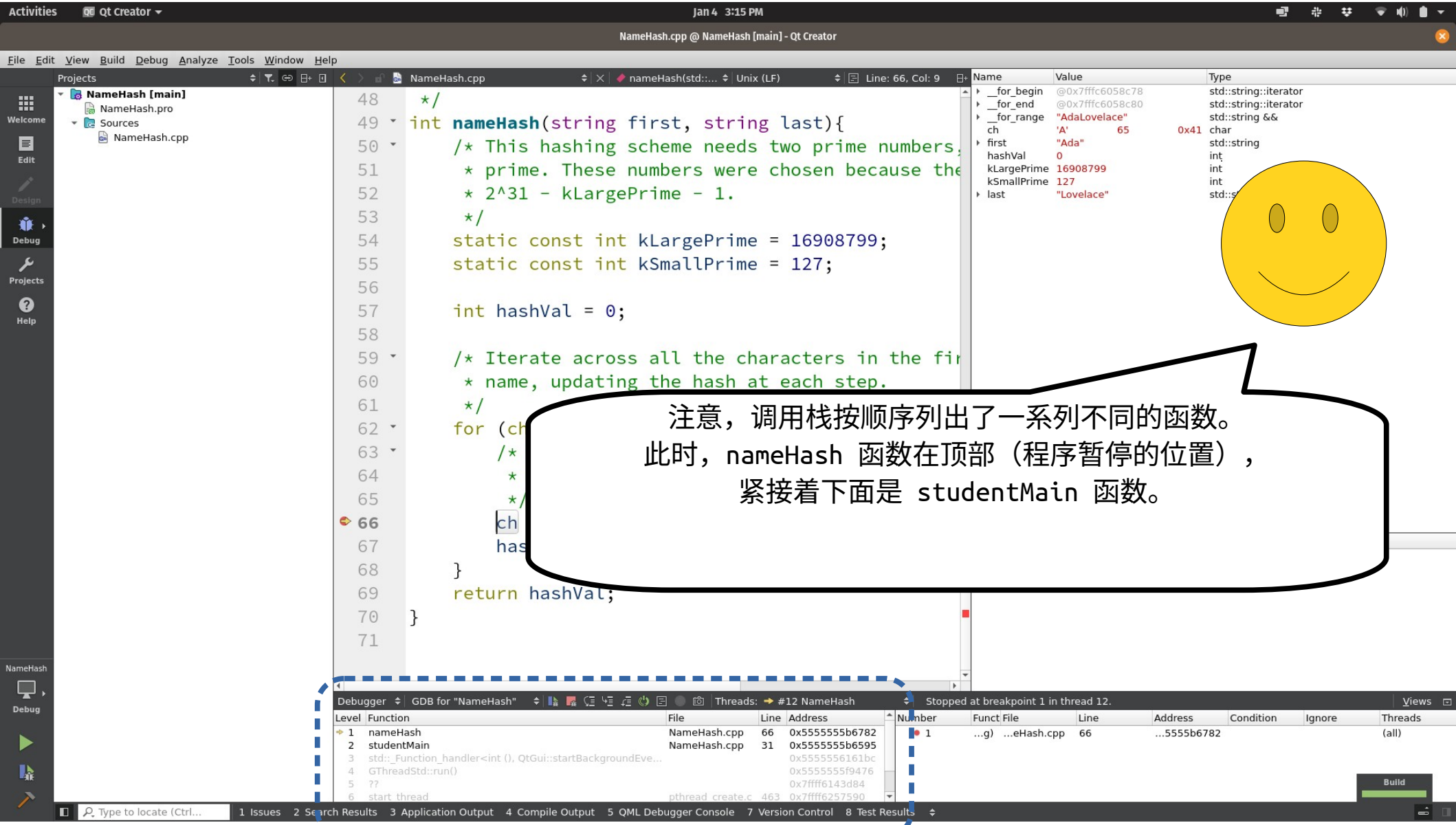

每次弹出调试界面，最好确认下是否停在了你预设的断点上。
检查黄色箭头有没有出现是一个很好的习惯。

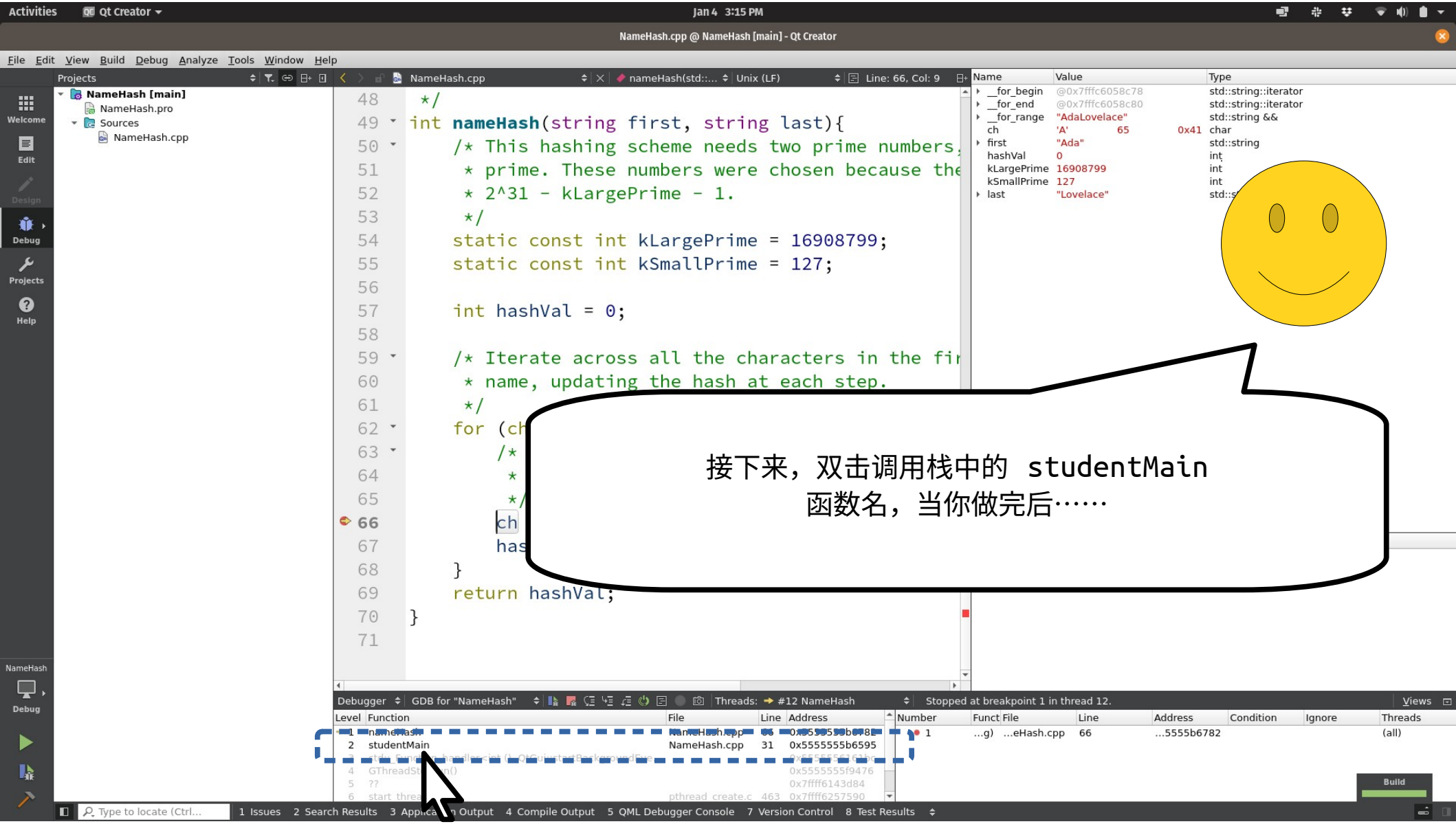

Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::
接下来，我们看看下面的面板，我们称之为
call stack 调用栈。
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Build
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::
此刻，根据黄色箭头所指，我们处在 nameHash 函数中。
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
NameHash
Debug
Build
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::s
然而，黄色箭头无法告诉我们，程序是如何执行到这个位置的。哪一部分的代码调用了 nameHash 这个函数呢？
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Build
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::s
调用栈可以告诉我们！
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Build
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::s
注意，调用栈按顺序列出了一系列不同的函数。
此时，nameHash 函数在顶部（程序暂停的位置），
紧接着下面是 studentMain 函数。
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Build
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results
NameHash
Debug


Activities
Qt Creator
Jan 4 3:15 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::... Unix (LF) Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::
接下来，双击调用栈中的 studentMain
函数名，当你做完后……
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
4 GThreadSt...n() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
NameHash
Debug
Build
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:22 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

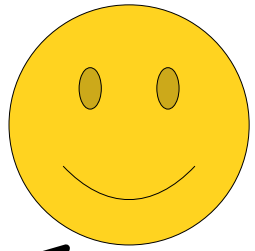
Projects

- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | <Select Symbol> | Unix (LF) | Line: 31, Col: 5

```
19  #include "simpio.h"  // for getLine
20  using namespace std;
21
22  /* Prototype for the nameHash function. This lets u
23   * in main and then define it later in the program.
24   */
25  int nameHash(string first, string last);
26
27  int main() {
28      string first = getLine("What is your first name
29      string last = getLine("What is your last name?
30
31      int hashValue = nameHash(first, last);
32
33      cout << "The hash of your name is: " << hashVal
34      return 0;
35  }
36
37  /* This is the actual function that c
38   * to talk more about what hash funct
39   * the meantime, think of it as a fun
40   * of the input and produces a number
41   *
42   * For those of you who are more mathem
43   * treats each character in the input name as a num
```

Name	Value	Type
first	"Ada"	std::string
hashValue	0	int
last	"Lovelace"	std::string



你会来到这个位置（第33行黄色箭头）

Debugger | GDB for "NameHash" | Threads: #12 NameHash | Stopped at breakpoint 1 in thread 12. | Views

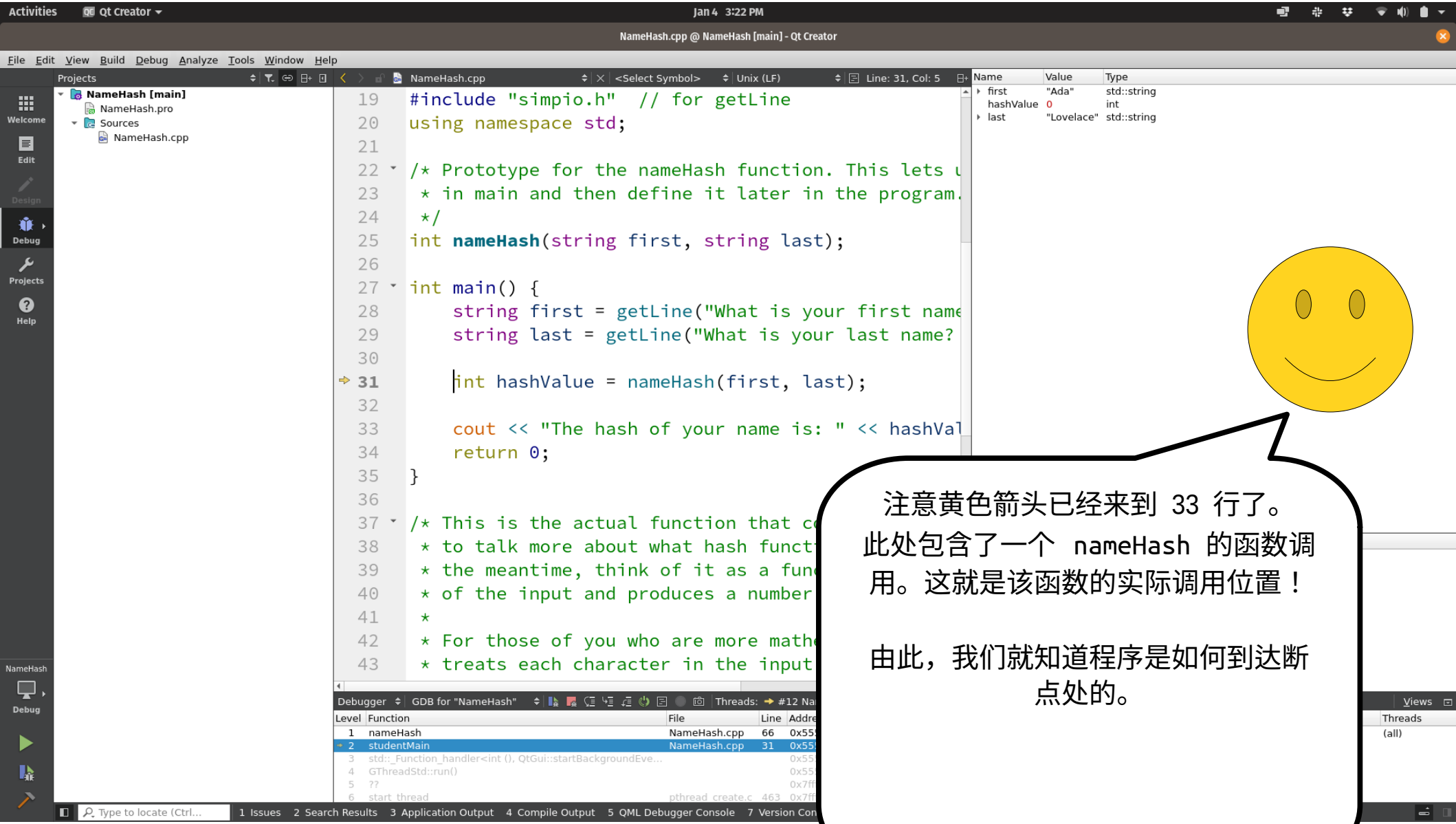
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

Welcome Edit Design Debug Projects Help

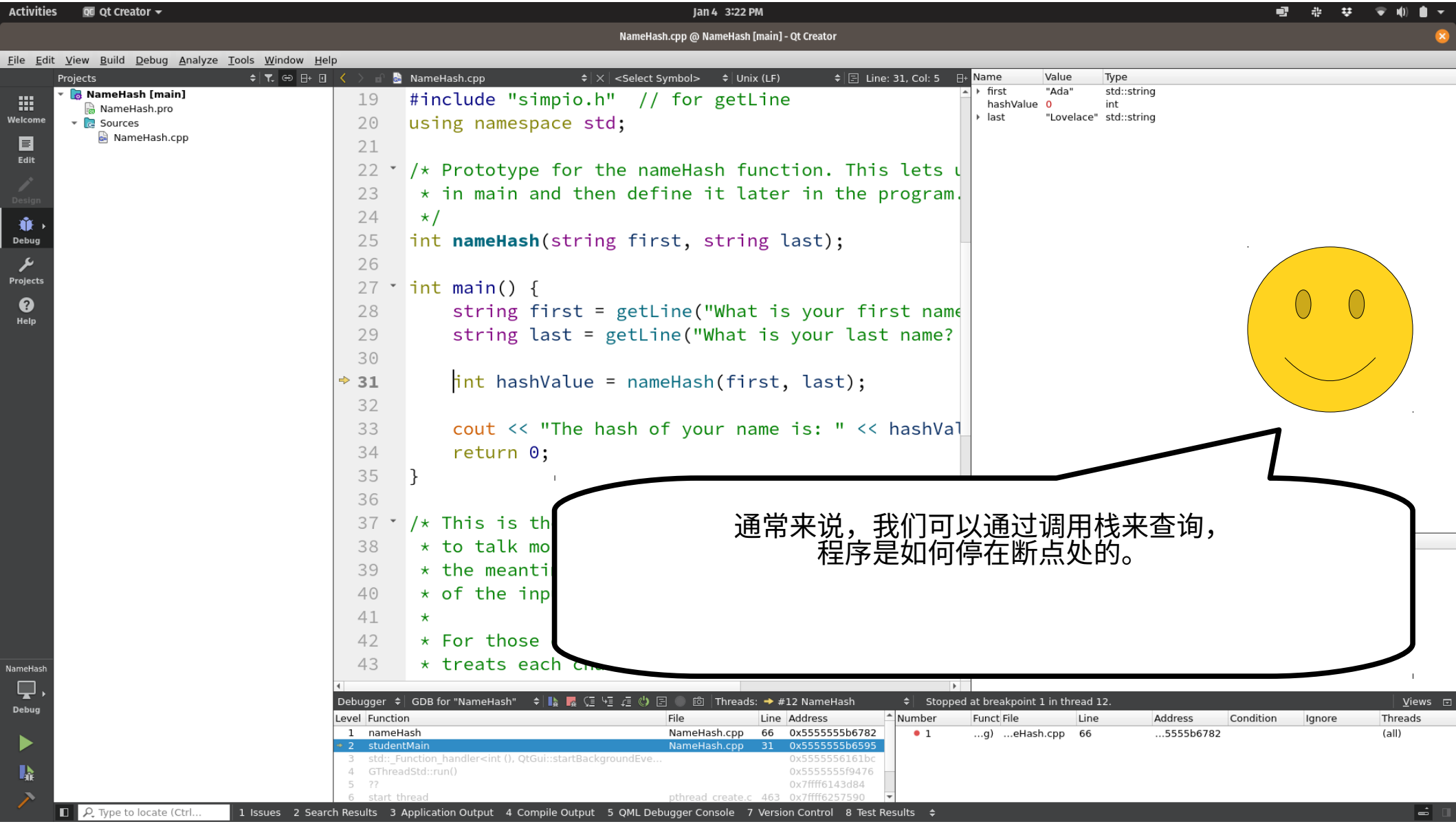
NameHash Debug

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results



注意黄色箭头已经来到 33 行了。此处包含了一个 nameHash 的函数调用。这就是该函数的实际调用位置！

由此，我们就知道程序是如何到达断点处的。


Activities
Qt Creator
Jan 4 3:22 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
<Select Symbol>
Unix (LF)
Line: 31, Col: 5
19 #include "simpio.h" // for getLine
20 using namespace std;
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then define it later in the program.
24 */
25 int nameHash(string first, string last);
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is th
38 * to talk mo
39 * the meanti
40 * of the inp
41 *
42 * For those
43 * treats each
Name Value Type
first "Ada" std::string
hashValue 0 int
last "Lovelace" std::string
通常来说，我们可以通过调用栈来查询，
程序是如何停在断点处的。
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:22 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

NameHash [main]
NameHash.pro
Sources
NameHash.cpp

NameHash.cpp <Select Symbol> Unix (LF) Line: 31, Col: 5

```
19  #include "simpio.h"  // for getLine
20  using namespace std;
21
22  /* Prototype for the nameHash function. This lets u
23   * in main and then define it later in the program.
24   */
25  int nameHash(string first, string last);
26
27  int main() {
28      string first = getLine("What is your first name
29      string last = getLine("What is your last name?
30
31      int hashValue = nameHash(first, last);
32
33      cout <<
34      return
35  }
36
37  /* This is
38   * to talk
39   * the meant
40   * of the input and produces a number.
41   *
42   * For those of you who are more mathematically inc
43   * treats each character in the input name as a num
```

Name	Value	Type
first	"Ada"	std::string
hashValue	0	int
last	"Lovelace"	std::string

根据你使用的操作系统，studentMain 下面还会有一些额外的函数。

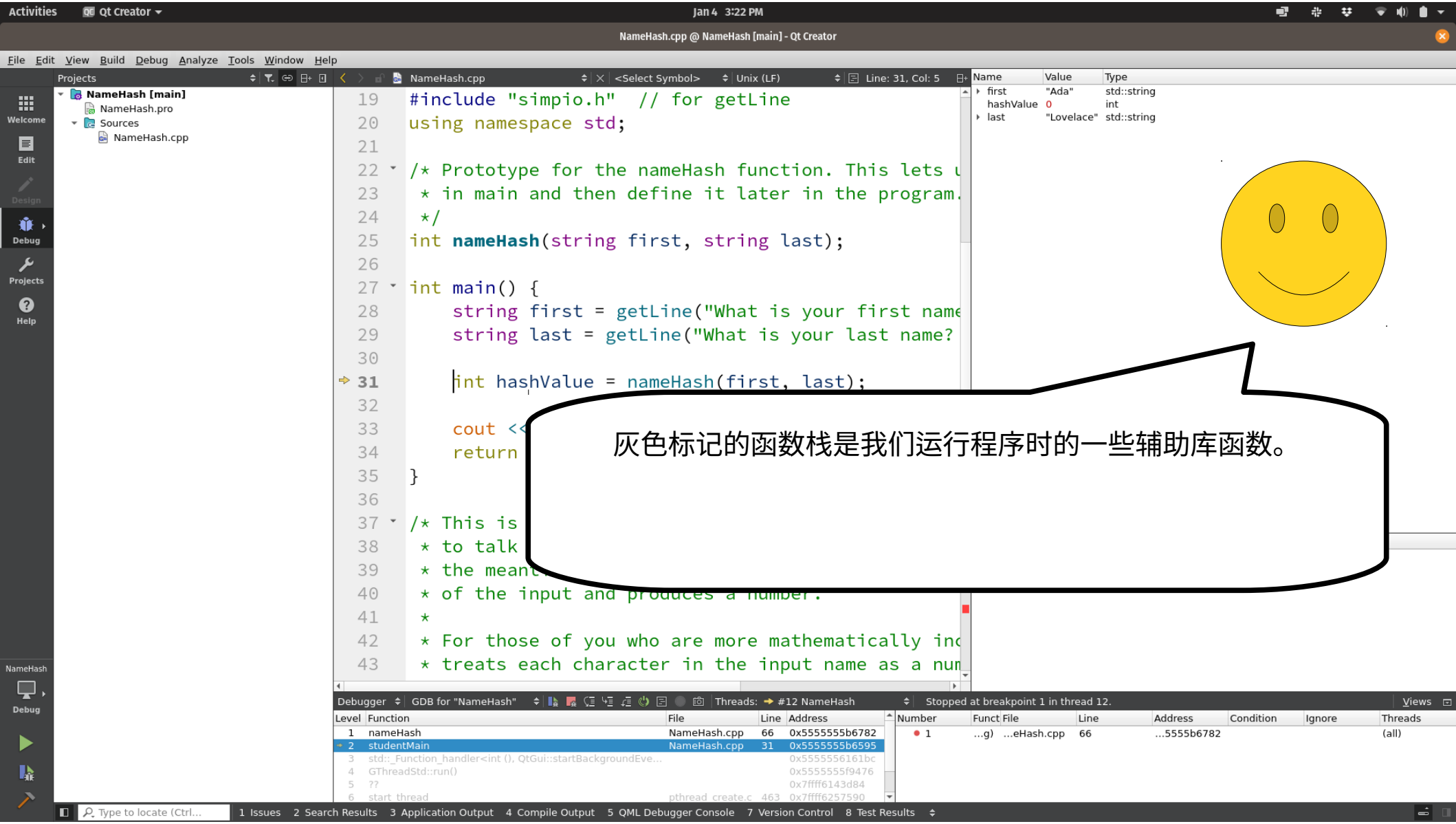
那些是什么呢？

Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views

Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:22 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
<Select Symbol>
Unix (LF)
Line: 31, Col: 5
19 #include "simpio.h" // for getLine
20 using namespace std;
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then define it later in the program.
24 */
25 int nameHash(string first, string last);
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout <<
34 return
35 }
36
37 /* This is
38 * to talk
39 * the meant
40 * of the input and produces a number.
41 *
42 * For those of you who are more mathematically inc
43 * treats each character in the input name as a num
Name Value Type
first "Ada" std::string
hashValue 0 int
last "Lovelace" std::string
灰色标记的函数栈是我们运行程序时的一些辅助库函数。
Debugger
GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results

Activities Qt Creator Jan 4 3:22 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

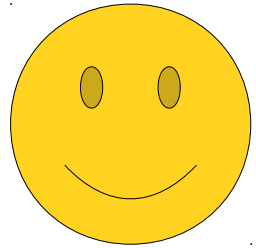
Projects

- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | <Select Symbol> | Unix (LF) | Line: 31, Col: 5

```
19  #include "simpio.h"  // for getLine
20  using namespace std;
21
22  /* Prototype for the nameHash function. This lets u
23   * in main and then define it later in the program.
24   */
25  int nameHash(string first, string last);
26
27  int main() {
28      string first = getLine("What is your first name
29      string last = getLine("What is your last name?
30
31      int hashValue = nameHash(first, last);
32
33      cout <<
34      return
35  }
36
37  /* This is
38   * to talk
39   * the meant
40   * of the input and produces a number.
41   *
42   * For those of you who are more mathematically inc
43   * treats each character in the input name as a num
```

Name	Value	Type
first	"Ada"	std::string
hashValue	0	int
last	"Lovelace"	std::string



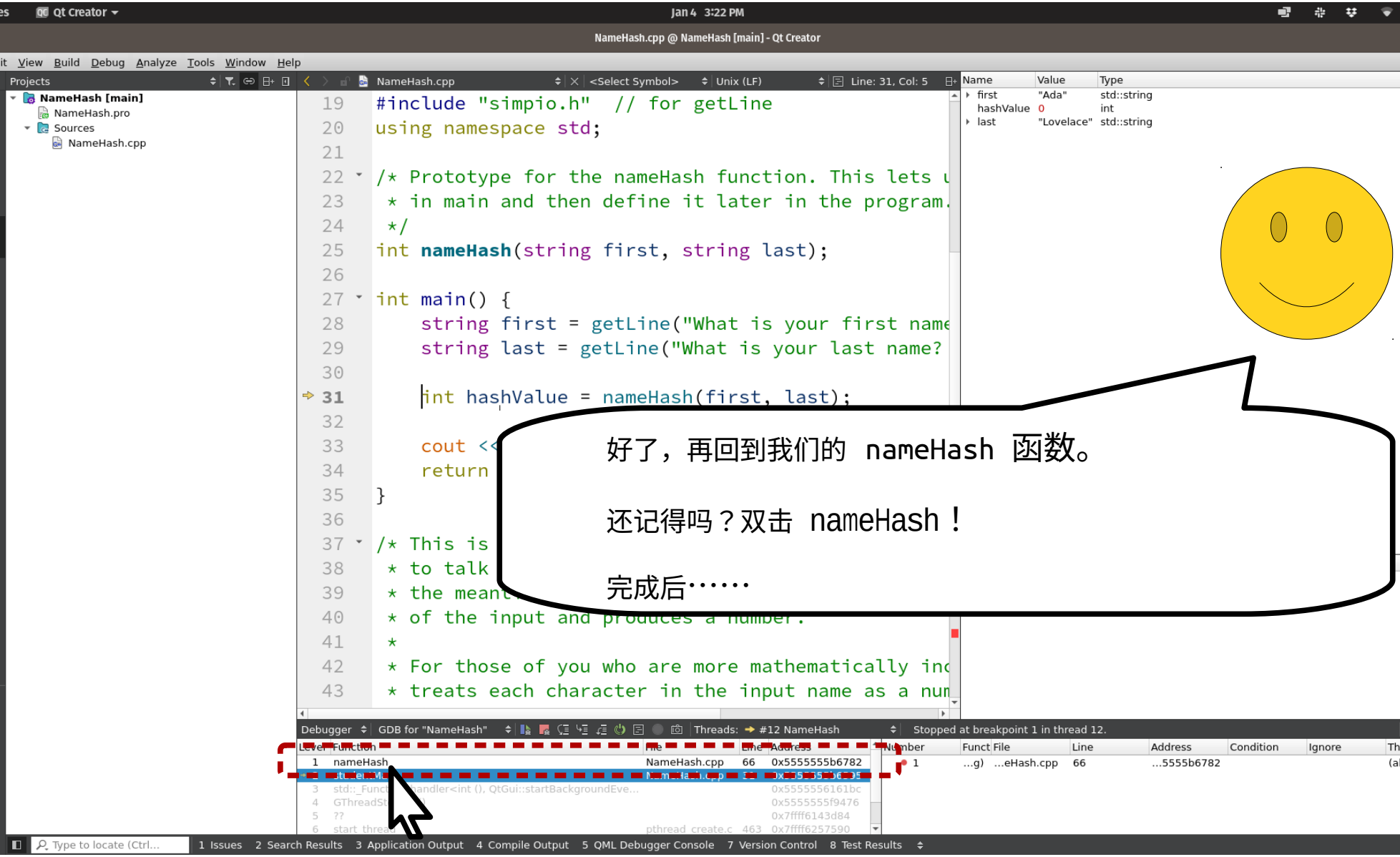
暂时不需要管那些。直接忽略即可。

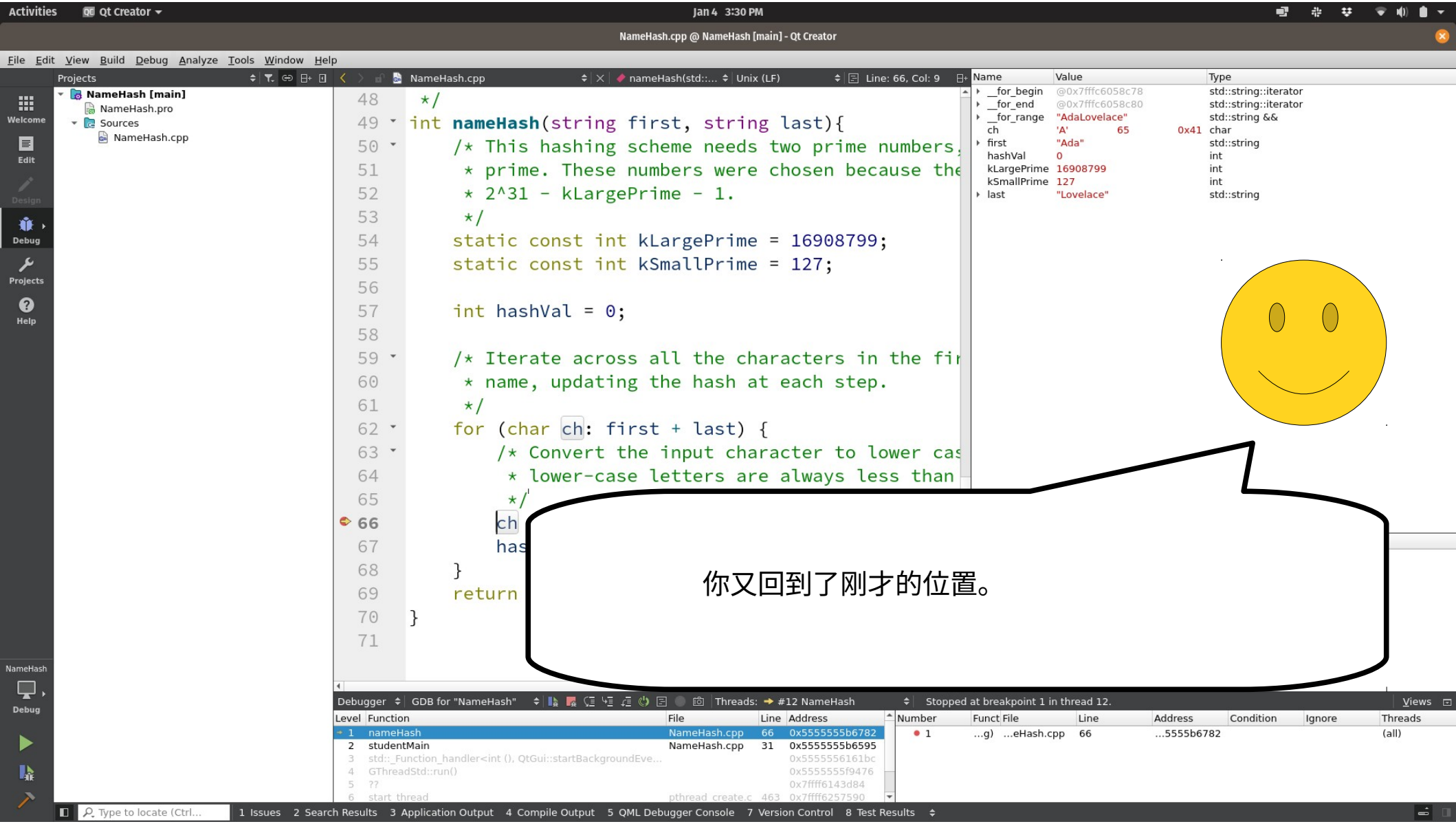
Debugger | GDB for "NameHash" | Threads: #12 NameHash | Stopped at breakpoint 1 in thread 12. | Views

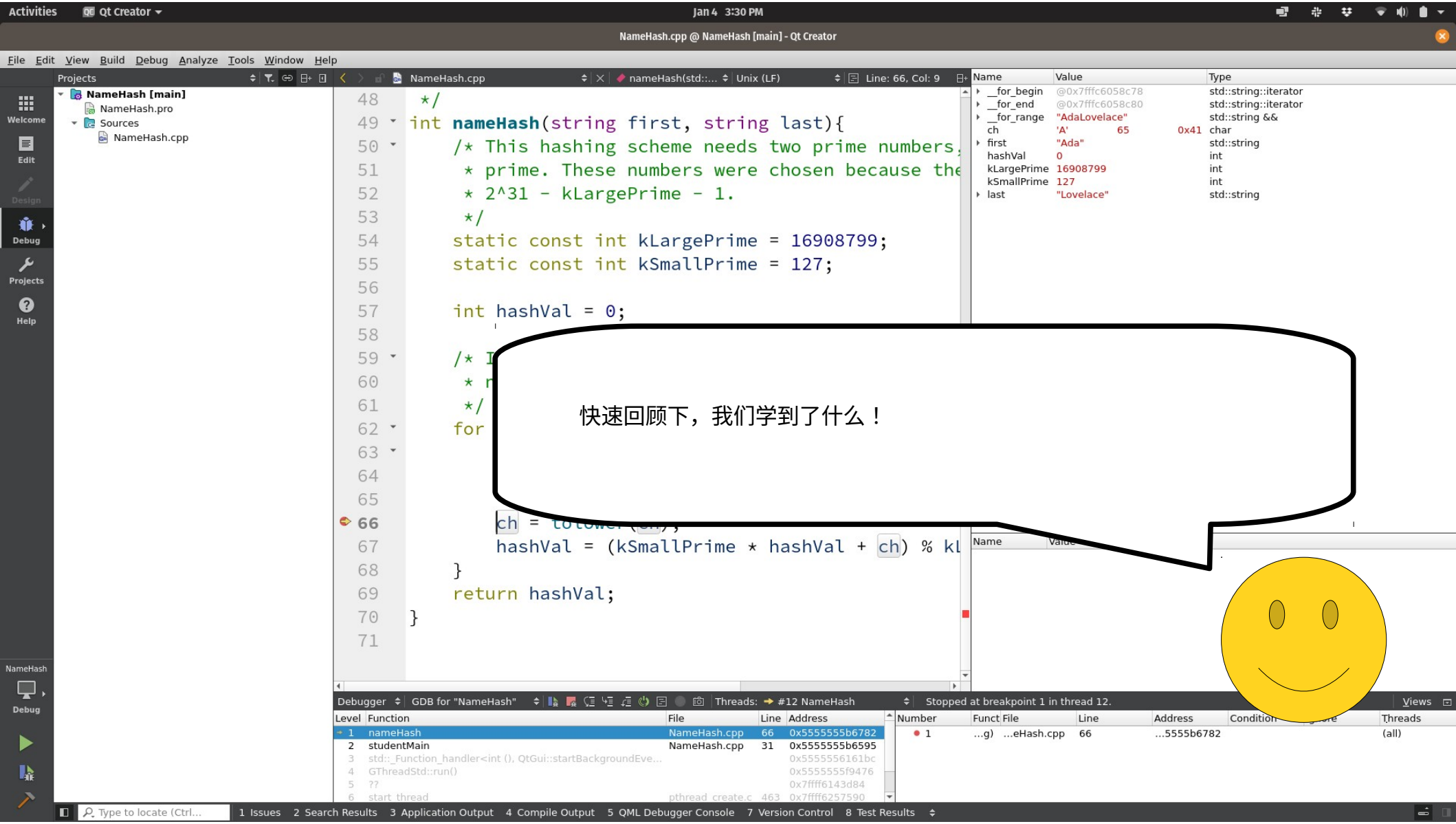
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

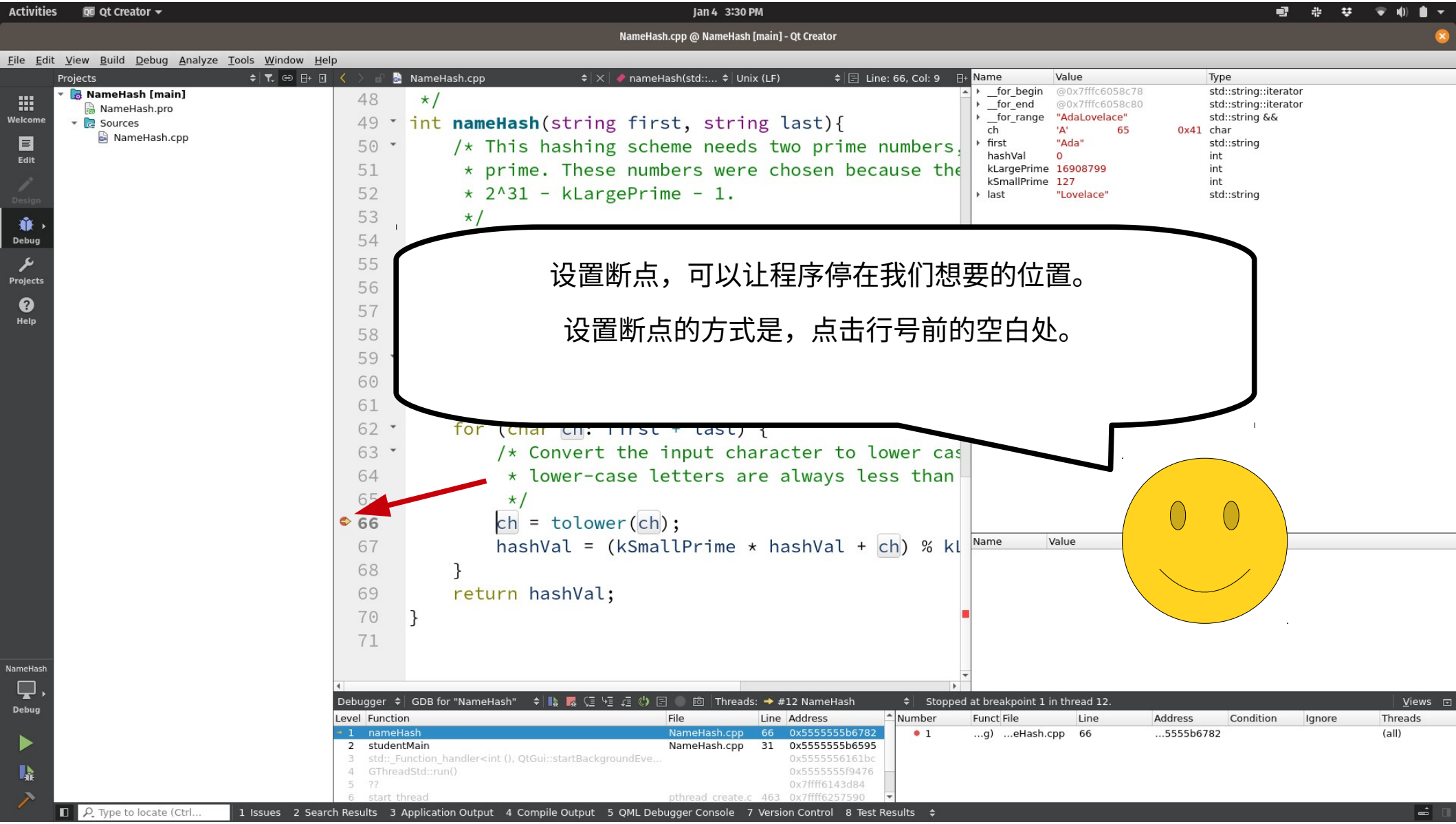
Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results


Activities
Qt Creator
Jan 4 3:22 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
<Select Symbol>
Unix (LF)
Line: 31, Col: 5
19 #include "simpio.h" // for getLine
20 using namespace std;
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then define it later in the program.
24 */
25 int nameHash(string first, string last);
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout <<
34 return
35 }
36
37 /* This is
38 * to talk
39 * the meant
40 * of the input and produces a number.
41 *
42 * For those of you who are more mathematically inc
43 * treats each character in the input name as a num
Name Value Type
first "Ada" std::string
hashValue 0 int
last "Lovelace" std::string
好了，再回到我们的 nameHash 函数。
还记得吗？双击 nameHash！
完成后……
Debugger
GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
3 std::_Funct...handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadSt... 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch
67 has
68 }
69 return
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
你又回到了刚才的位置。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* I
60 * r
61 */
62 for
63
64
65
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kl
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
快速回顾下，我们学到了什么！
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::... Unix (LF) Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
设置断点，可以让程序停在我们想要的位置。
设置断点的方式是，点击行号前的空白处。
Name Value
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:30 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

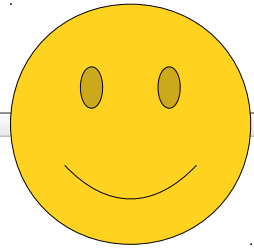
NameHash [main]
NameHash.pro
Sources
NameHash.cpp

NameHash.cpp nameHash(std::... Unix (LF) Line: 66, Col: 9

```
48  */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54
55
56
57
58
59
60
61
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

一旦程序执行到断点处，Qt 会弹出调试面板。

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'A' 65 0x41	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



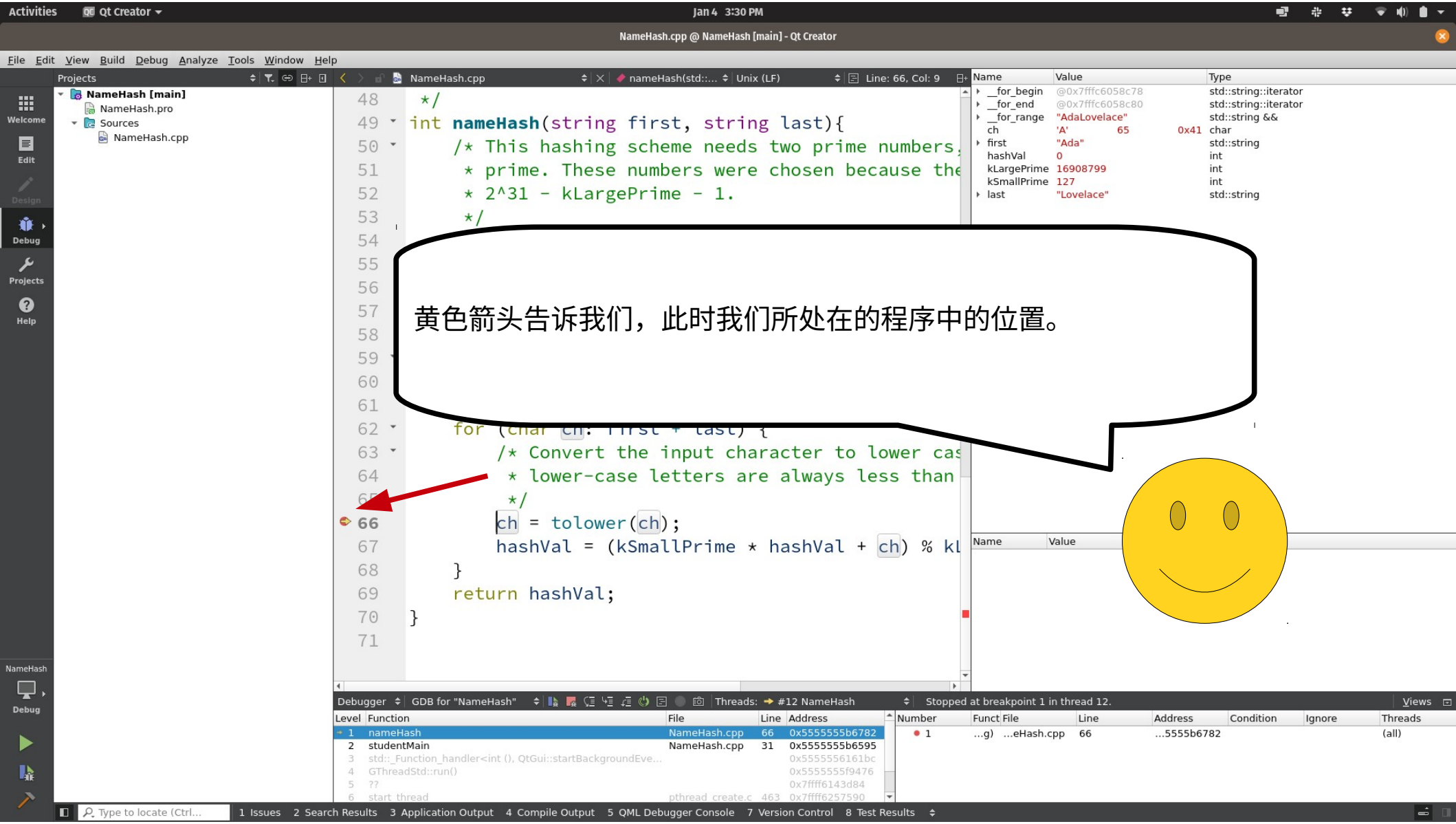
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views

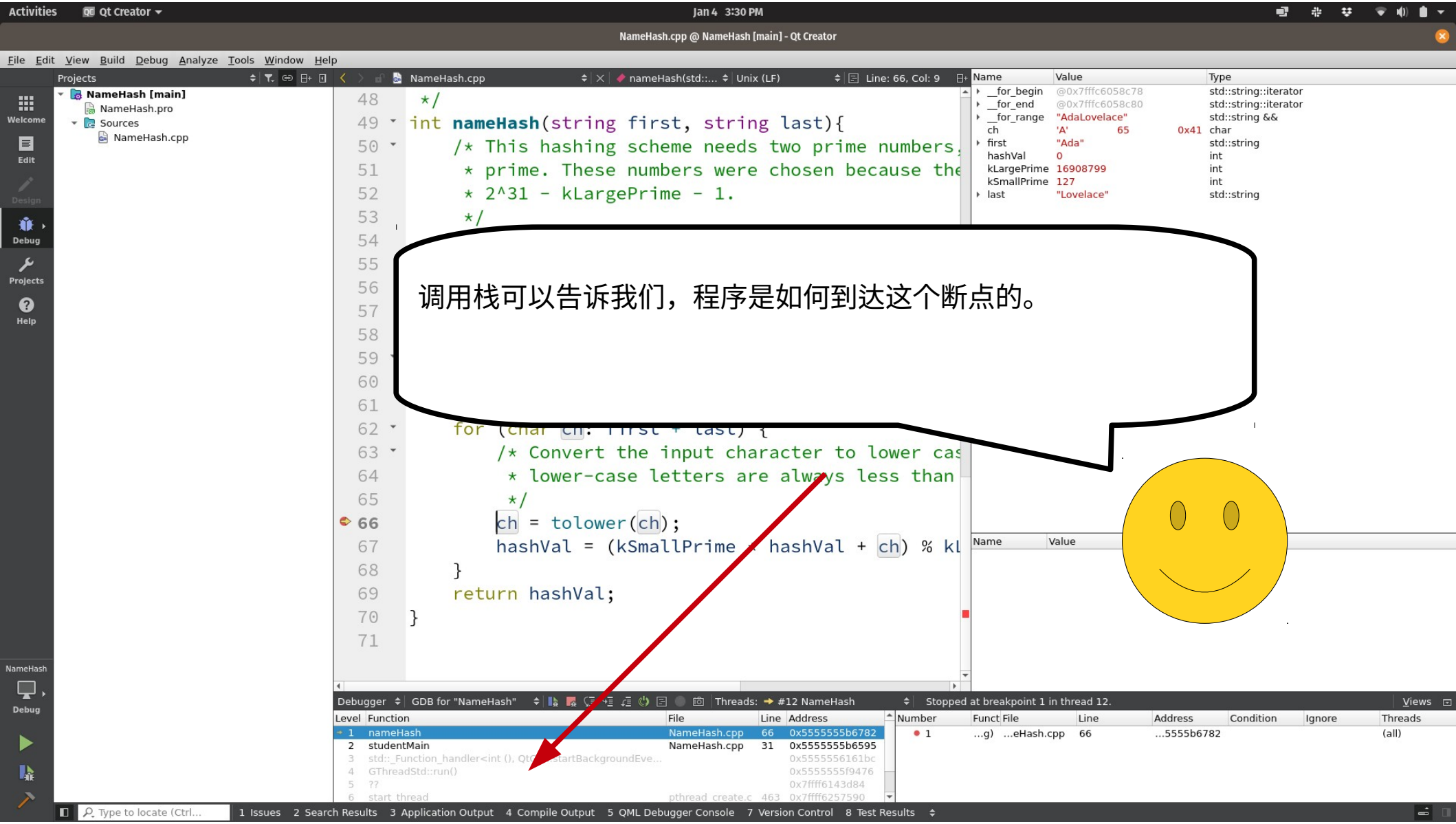
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

Welcome Edit Design Debug Projects Help NameHash Debug

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54
55
56
57
58
59
60
61
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
黄色箭头告诉我们，此时我们所处在的程序中的位置。
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name Value
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
调用栈可以告诉我们，程序是如何到达这个断点的。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:30 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

NameHash [main]
NameHash.pro
Sources
NameHash.cpp

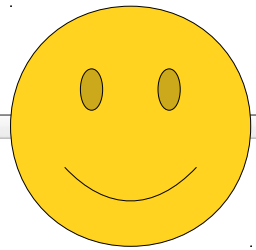
NameHash.cpp nameHash(std::... Unix (LF) Line: 66, Col: 9

```
48  */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54
55
56
57
58
59
60
61
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

接下来，让我们来看看如何查看函数中的变量。

Name	Value			Type
__for_begin	@0x7fffc6058c78			std::string::iterator
__for_end	@0x7fffc6058c80			std::string::iterator
__for_range	"AdaLovelace"			std::string &&
ch	'A'	65	0x41	char
first	"Ada"			std::string
hashVal	0			int
kLargePrime	16908799			int
kSmallPrime	127			int
last	"Lovelace"			std::string

Name	Value



Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views

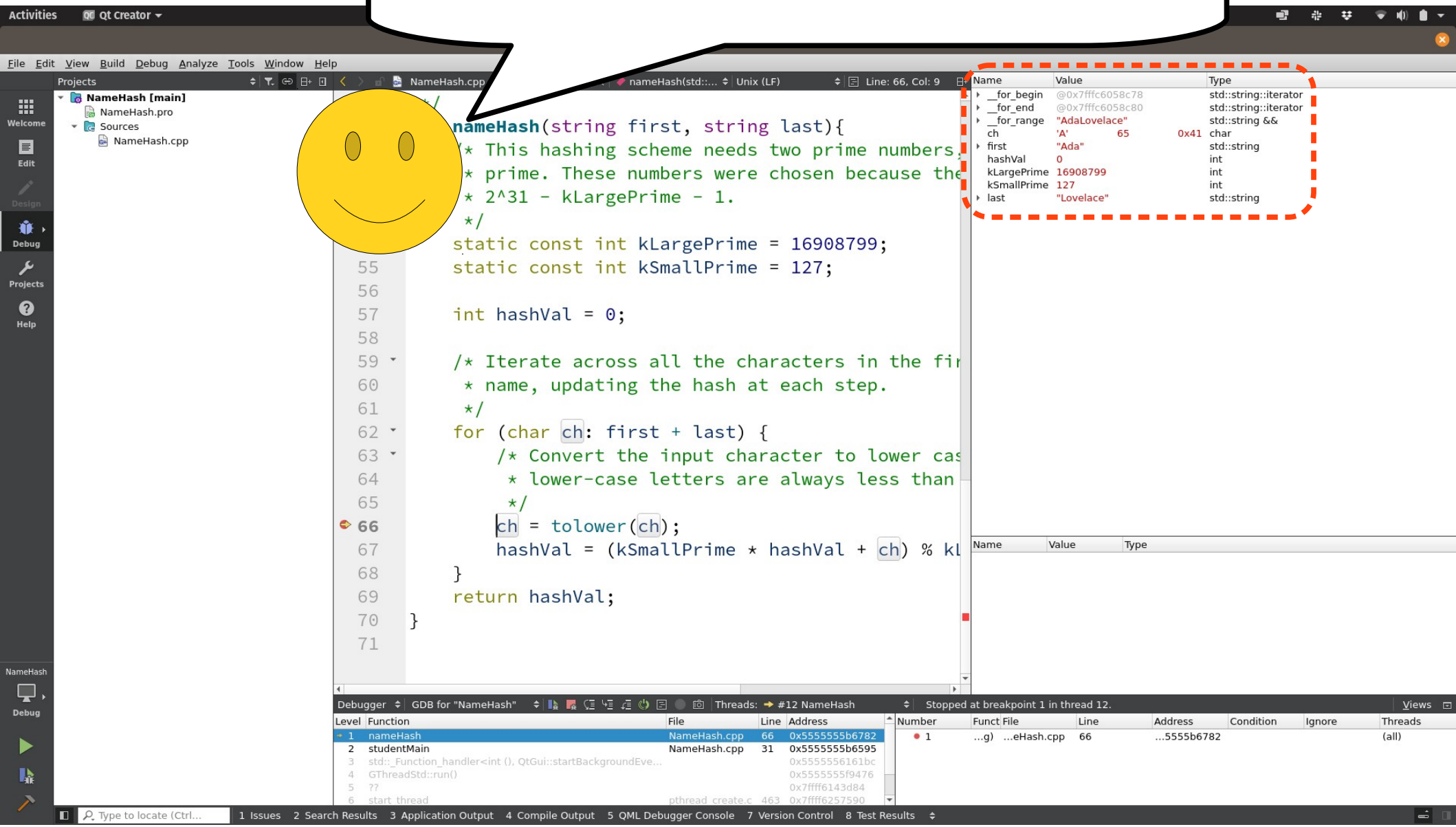
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

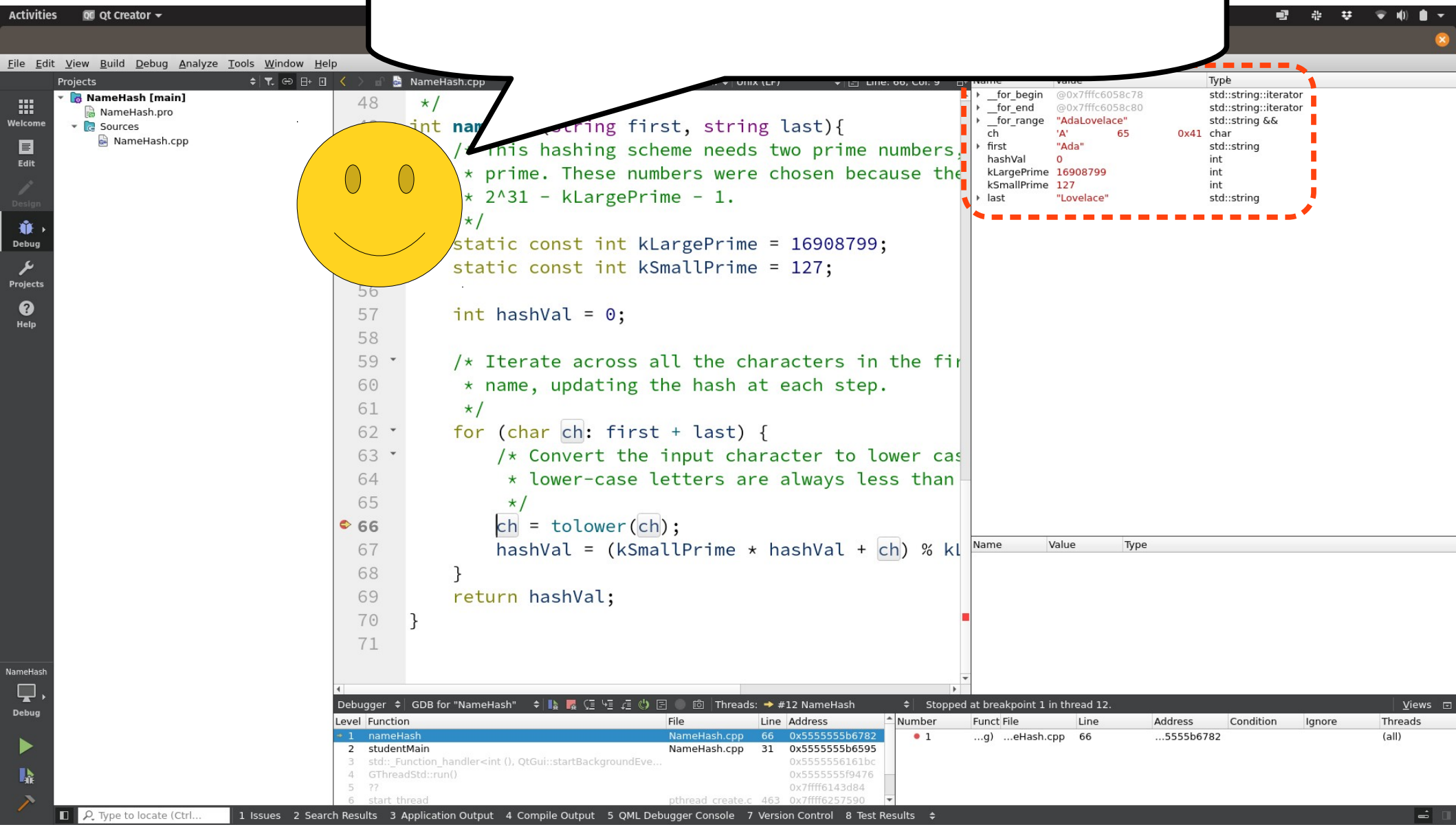

看看右上角的面板！
Activities
Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::... Unix (LF) Line: 66, Col: 9
nameHash(string first, string last){
/* This hashing scheme needs two prime numbers,
* prime. These numbers were chosen because the
* 2^31 - kLargePrime - 1.
*/
static const int kLargePrime = 16908799;
static const int kSmallPrime = 127;
int hashVal = 0;
/* Iterate across all the characters in the fir
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % kL
}
return hashVal;
}
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

这里面列出的是当前函数内的局部变量。

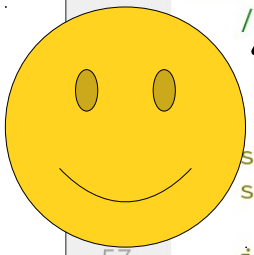


根据不同的操作系统，名称可能不同。

但我们只关注类似 ch 和 hashVal 这样的名称



忽略那些奇奇怪怪的名字，只关注我们熟悉的名字就好了



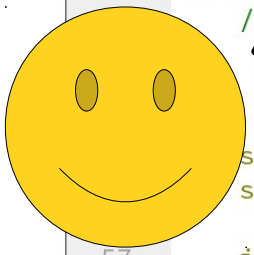
```
48  */
49  int nam                  first, string last){
    /*          hashing scheme needs two prime numbers,
     * prime. These numbers were chosen because the
     * 2^31 - kLargePrime - 1.
     */
    static const int kLargePrime = 16908799;
    static const int kSmallPrime = 127;

57  int hashVal = 0;
58
59  /* Iterate across all the characters in the fir
     * name, updating the hash at each step.
     */
62  for (char ch: first + last) {
63      /* Convert the input character to lower cas
         * lower-case letters are always less than
         */
66      ch = tolower(ch);
67      hashVal = (kSmallPrime * hashVal + ch) % kL
68  }
69  return hashVal;
70  }
71
```

Name	Value			Type
__for_end	@0x7fffc6058c80			std::string::iterator
__for_range	"AdaLovelace"			std::string &&
ch	'A'	65	0x41	char
first	"Ada"			std::string
hashVal	0			int
kLargePrime	16908799			int
kSmallPrime	127			int
last	"Lovelace"			std::string

Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

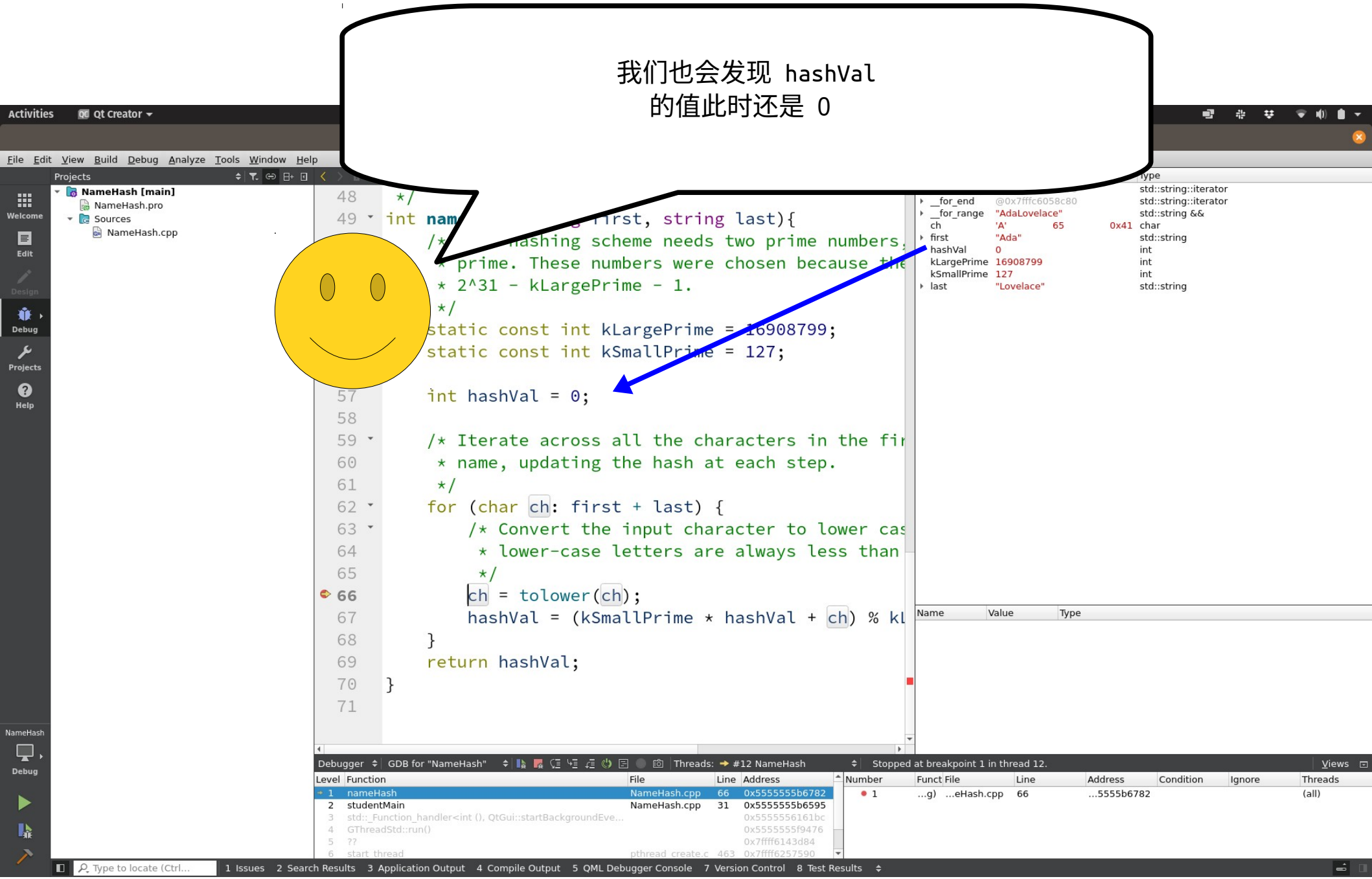
Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

比如，你可以看到 kLargePrime 和 kSmallPrime
跟我们程序中定义的一致。

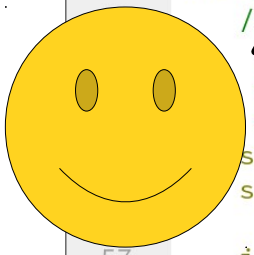


```
 */
int nam                first, string last){
    /*          hashing scheme needs two prime numbers,
     * prime. These numbers were chosen because the
     * 2^31 - kLargePrime - 1.
     */
    static const int kLargePrime = 16908799;
    static const int kSmallPrime = 127;

    int hashVal = 0;

    /* Iterate across all the characters in the fir
     * name, updating the hash at each step.
     */
    for (char ch: first + last) {
        /* Convert the input character to lower cas
         * lower-case letters are always less than
         */
        ch = tolower(ch);
        hashVal = (kSmallPrime * hashVal + ch) % kL
    }
    return hashVal;
}
```

__for_end	@0x7fffc6058c80			std::string::iterator
__for_range	"AdaLovelace"			std::string &&
ch	'A'	65	0x41	char
first	"Ada"			std::string
hashVal	0			int
kLargePrime	16908799			int
kSmallPrime	127			int
last	"Lovelace"			std::string


我们也会发现 hashVal
的值此时还是 0
Activities
Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
48 */
49 int nam first, string last){
/* hashing scheme needs two prime numbers,
* prime. These numbers were chosen because the
* 2^31 - kLargePrime - 1.
*/
static const int kLargePrime = 16908799;
static const int kSmallPrime = 127;
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
* name, updating the hash at each step.
*/
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
* lower-case letters are always less than
*/
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Type
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

当我们一步一步调试程序时，我们会发现这些值的变化。

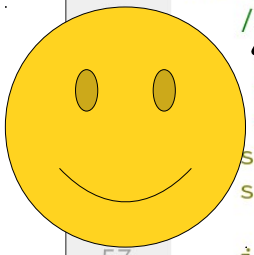


```
 */
int nam                  first, string last){
    /*          hashing scheme needs two prime numbers,
     * prime. These numbers were chosen because the
     * 2^31 - kLargePrime - 1.
     */
    static const int kLargePrime = 16908799;
    static const int kSmallPrime = 127;

    int hashVal = 0;

    /* Iterate across all the characters in the fir
     * name, updating the hash at each step.
     */
    for (char ch: first + last) {
        /* Convert the input character to lower cas
         * lower-case letters are always less than
         */
        ch = tolower(ch);
        hashVal = (kSmallPrime * hashVal + ch) % kL
    }
    return hashVal;
}
```


先看看这个 for 循环。
Activities
Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
48 */
49 int nam first, string last){
/* hashing scheme needs two prime numbers,
* prime. These numbers were chosen because the
* 2^31 - kLargePrime - 1.
*/
static const int kLargePrime = 16908799;
static const int kSmallPrime = 127;
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name Value Type
NameHash
Debug
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

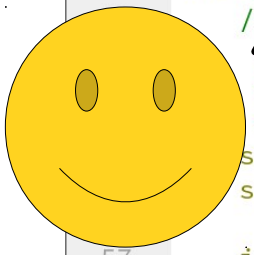
这是一个 range-based for 循环，意思是

“对 first + last 中的每一个字符，执行相应的操作。”



```
 */
int nam                  first, string last){
    /*          hashing scheme needs two prime numbers,
     * prime. These numbers were chosen because the
     * 2^31 - kLargePrime - 1.
     */
    static const int kLargePrime = 16908799;
    static const int kSmallPrime = 127;

    int hashVal = 0;

    /* Iterate across all the characters in the fir
     * name, updating the hash at each step.
     */
    for (char ch: first + last) {
        /* Convert the input character to lower cas
         * lower-case letters are always less than
         */
        ch = tolower(ch);
        hashVal = (kSmallPrime * hashVal + ch) % kL
    }
    return hashVal;
}
```

回忆一下，我们刚刚输入的名字是
Ada Lovelace

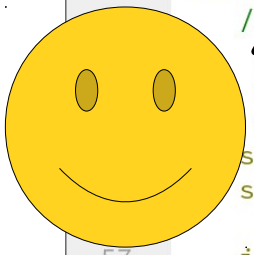


```
48  */
49  int nam                  first, string last){
        /*          hashing scheme needs two prime numbers,
         * prime. These numbers were chosen because the
         * 2^31 - kLargePrime - 1.
         */
        static const int kLargePrime = 16908799;
        static const int kSmallPrime = 127;

57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

如果我们查看 ch 的当前值，会发现此时的值是 A。
这就是 Ada Lovelace 的第一个字母。



```
48  */
49  int nam                  first, string last){
        /*          hashing scheme needs two prime numbers,
         * prime. These numbers were chosen because the
         * 2^31 - kLargePrime - 1.
         */
        static const int kLargePrime = 16908799;
        static const int kSmallPrime = 127;

57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value			Type
__for_end	@0x7fffc6058c80			std::string::iterator
ch	'A'	65	0x41	char
hashVal	0			int
kLargePrime	16908799			int
kSmallPrime	127			int
last	"Lovelace"			std::string

Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

非常好！总结一下：
我们已经知道程序所在的位置（第66行），
还知道程序是如何从 mian 函数执行到 nameHash 的，
还知道此刻程序中的变量值。

Activities Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

NameHash [main]
NameHash.pro
Sources
NameHash.cpp

Welcome Edit Design Debug Projects Help

```
48  */
49  int nam                    first, string last){
        /*          hashing scheme needs two prime numbers,
         * prime. These numbers were chosen because the
         * 2^31 - kLargePrime - 1.
         */
        static const int kLargePrime = 16908799;
        static const int kSmallPrime = 127;

        int hashVal = 0;

        /* Iterate across all the characters in the fir
         * name, updating the hash at each step.
         */
        for (char ch: first + last) {
            /* Convert the input character to lower cas
             * lower-case letters are always less than
             */
            ch = tolower(ch);
            hashVal = (kSmallPrime * hashVal + ch) % kL
        }
        return hashVal;
}
```

Name	Value			Type
__for_end	@0x7fffc6058c80			std::string::iterator
__for_range	"AdaLovelace"			std::string &&
ch	'A'	65	0x41	char
first	"Ada"			std::string
hashVal	0			int
kLargePrime	16908799			int
kSmallPrime	127			int
last	"Lovelace"			std::string

Name Value Type

Debugger GDB for "NameHash" Threads: #12 NameHash Stopped at breakpoint 1 in thread 12. Views

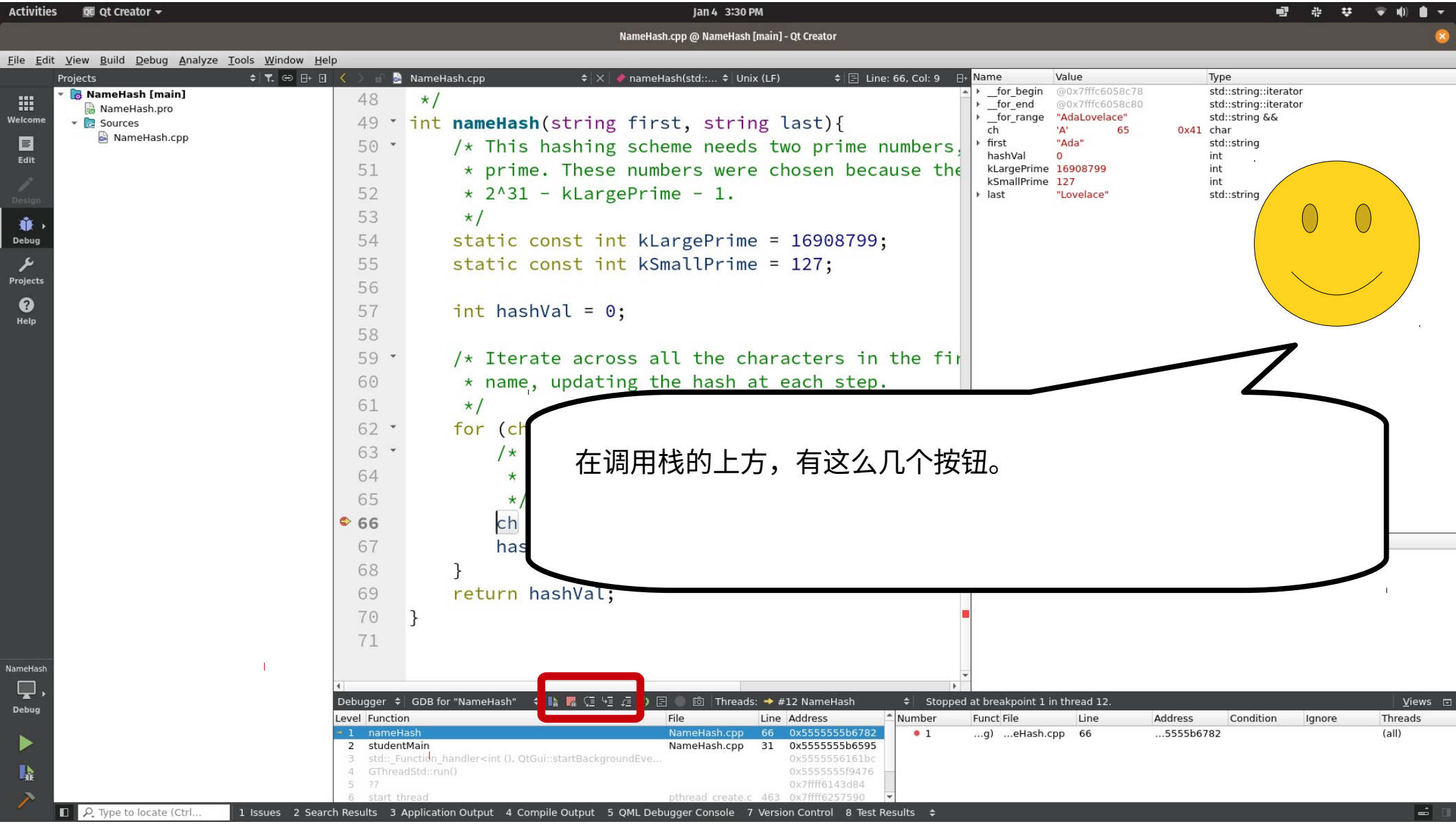
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	66	0x5555555b6782
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

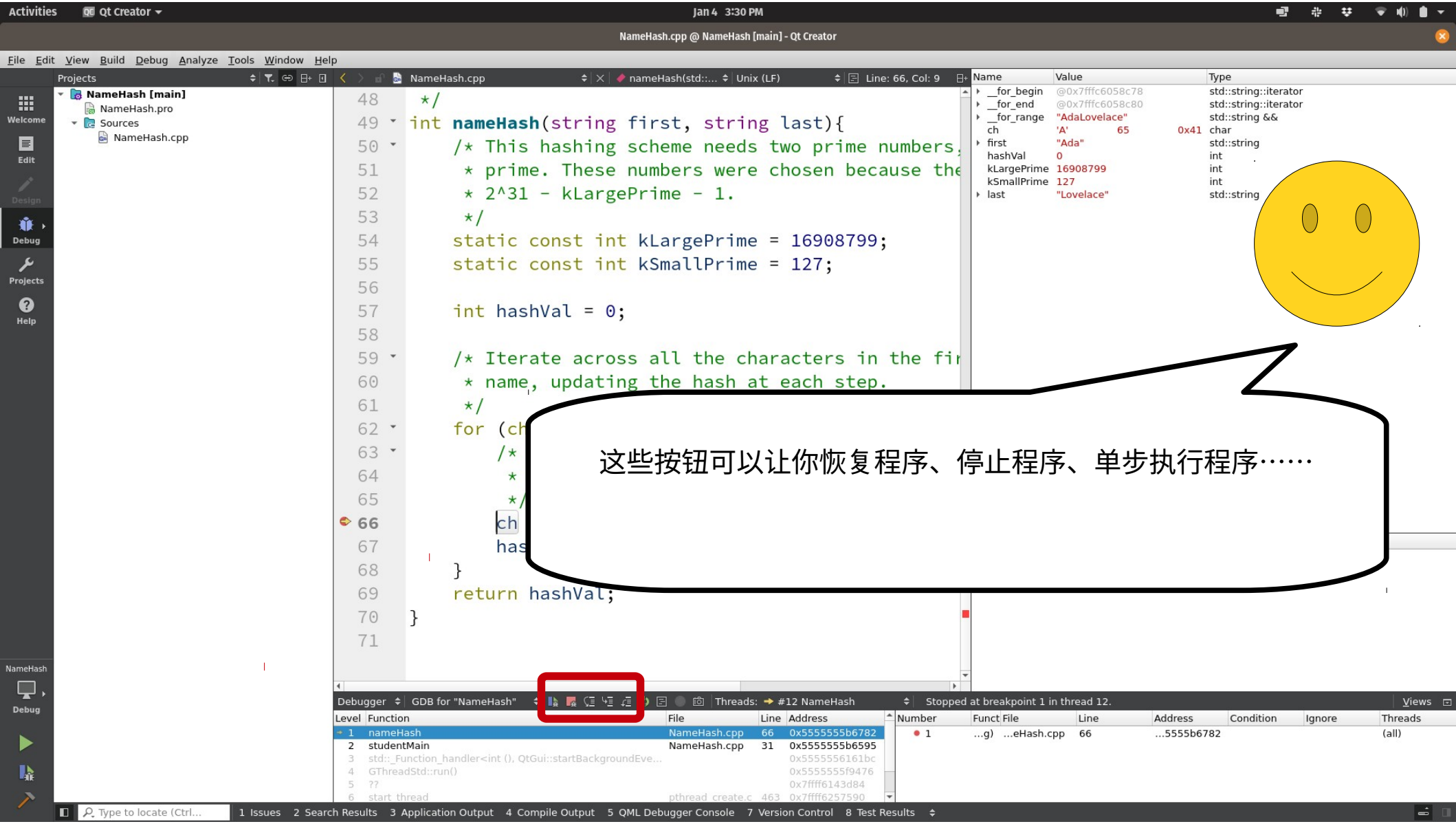
Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

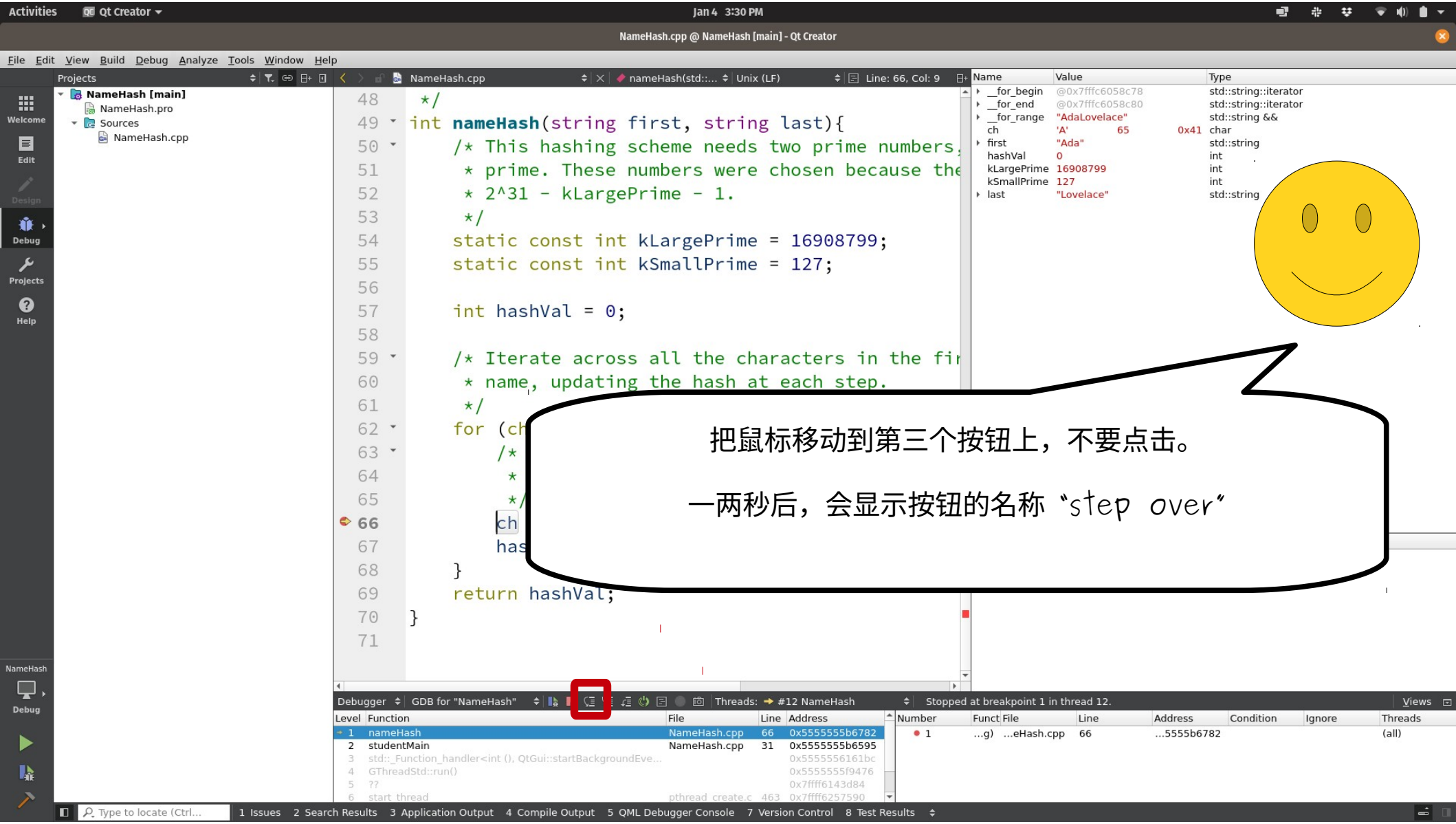
Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

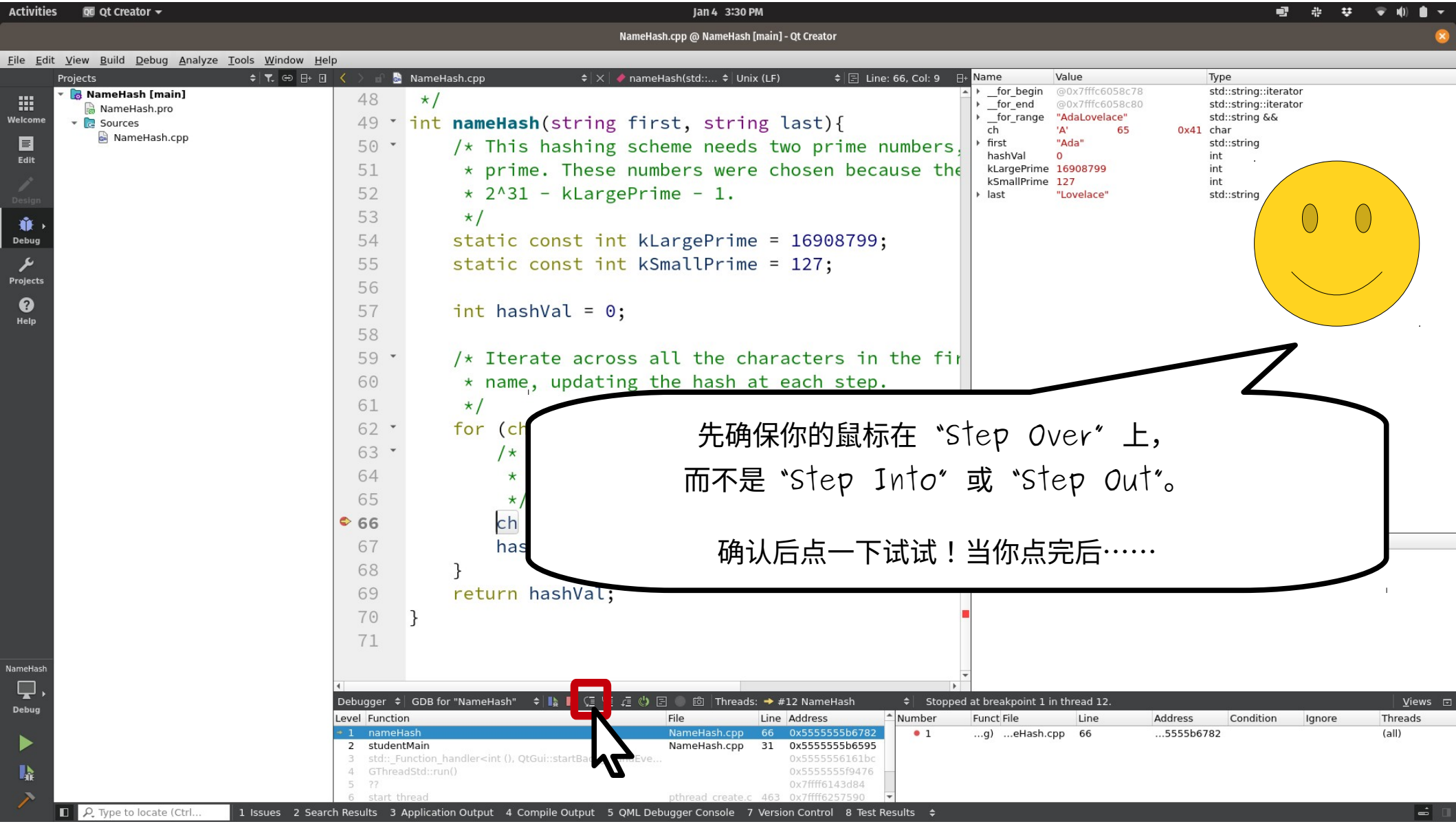
接下来，我们再做一些更酷的事情。
我们将一行一行地执行程序，并观察每一步发生的变化。

```
48  */
49  int nam                  first, string last){
        /*          hashing scheme needs two prime numbers,
         * prime. These numbers were chosen because the
         * 2^31 - kLargePrime - 1.
         */
        static const int kLargePrime = 16908799;
        static const int kSmallPrime = 127;

57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
在调用栈的上方，有这么几个按钮。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
这些按钮可以让你恢复程序、停止程序、单步执行程序……
NameHash
Debug
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
把鼠标移动到第三个按钮上，不要点击。
一两秒后，会显示按钮的名称 "step over"
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:30 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'A' 65 0x41 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
先确保你的鼠标在 "Step Over" 上，
而不是 "Step Into" 或 "Step Out"。
确认后点一下试试！当你点完后……
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBa...nEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:42 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

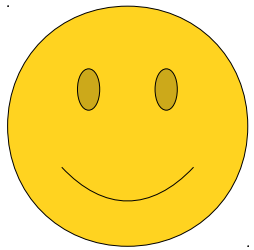
Projects

NameHash [main]
- NameHash.pro
- Sources
  - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 67, Col: 9

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



Name	Value	Type

Debugger | GDB for "NameHash"

Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), G
4	GThreadStd::run()
5	??
6	start_thread

pthread_create.c 463 0x7ffff6257590

…黄色箭头来到了下一行（line 67）

Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results

Activities Qt Creator Jan 4 3:42 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

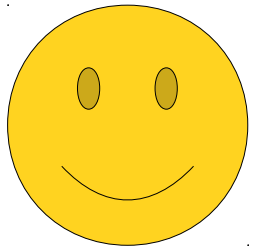
Projects

NameHash [main]
NameHash.pro
Sources
NameHash.cpp

NameHash.cpp nameHash(std::... Unix (LF) Line: 67, Col: 9

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



Name	Value	Type

Debugger GDB for "NameHash"

Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), G
4	GThreadStd::run()
5	??
6	start_thread

pthread_create.c 463 0x7ffff6257590

非常好！已经有一些东西改变了。让我们一起来看看。

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:42 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

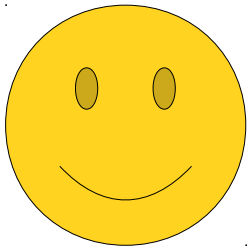
Projects

NameHash [main]
NameHash.pro
Sources
NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 67, Col: 9

```
48  */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



Name	Value	Type

首先，黄色箭头此时在 67 行。

Debugger | GDB for "NameHash"

Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), G
4	GThreadStd::run()
5	??
6	start_thread

pthread_create.c 463 0x7ffff6257590

Views

Welcome Edit Design Debug Projects Help

NameHash Debug

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:42 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

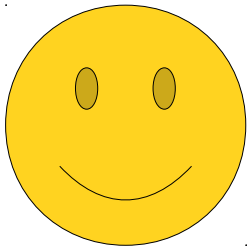
Projects

- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 67, Col: 9

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



Name	Value	Type

你已经成功地运行了一行代码，相当完美！

Debugger | GDB for "NameHash"

Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), G
4	GThreadStd::run()
5	??
6	start_thread

pthread_create.c 463 0x7ffff6257590

Views

Welcome Edit Design Debug Projects Help

NameHash Debug

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:42 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

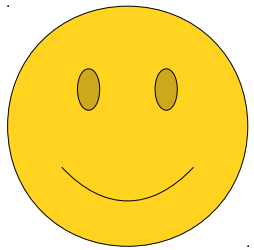
Projects

NameHash [main]
- NameHash.pro
- Sources
  - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 67, Col: 9

```
48  */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



Name	Value	Type

那么这行代码做了什么呢？

Debugger | GDB for "NameHash"

Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), G
4	GThreadStd::run()
5	??
6	start_thread

pthread_create.c 463 0x7ffff6257590

Views

Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results

Activities Qt Creator Jan 4 3:42 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

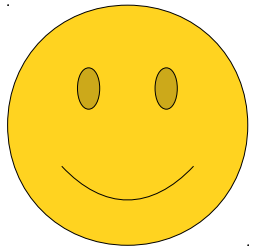
Projects

- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 67, Col: 9

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	0	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string



这行代码把 ch 转换成了小写。函数 tolower 接收一个字符，
并返回它的小写形式。
然后覆盖 ch 值 ，
这样就得到了一个小写的版本。

Debugger | GDB for "NameHash"

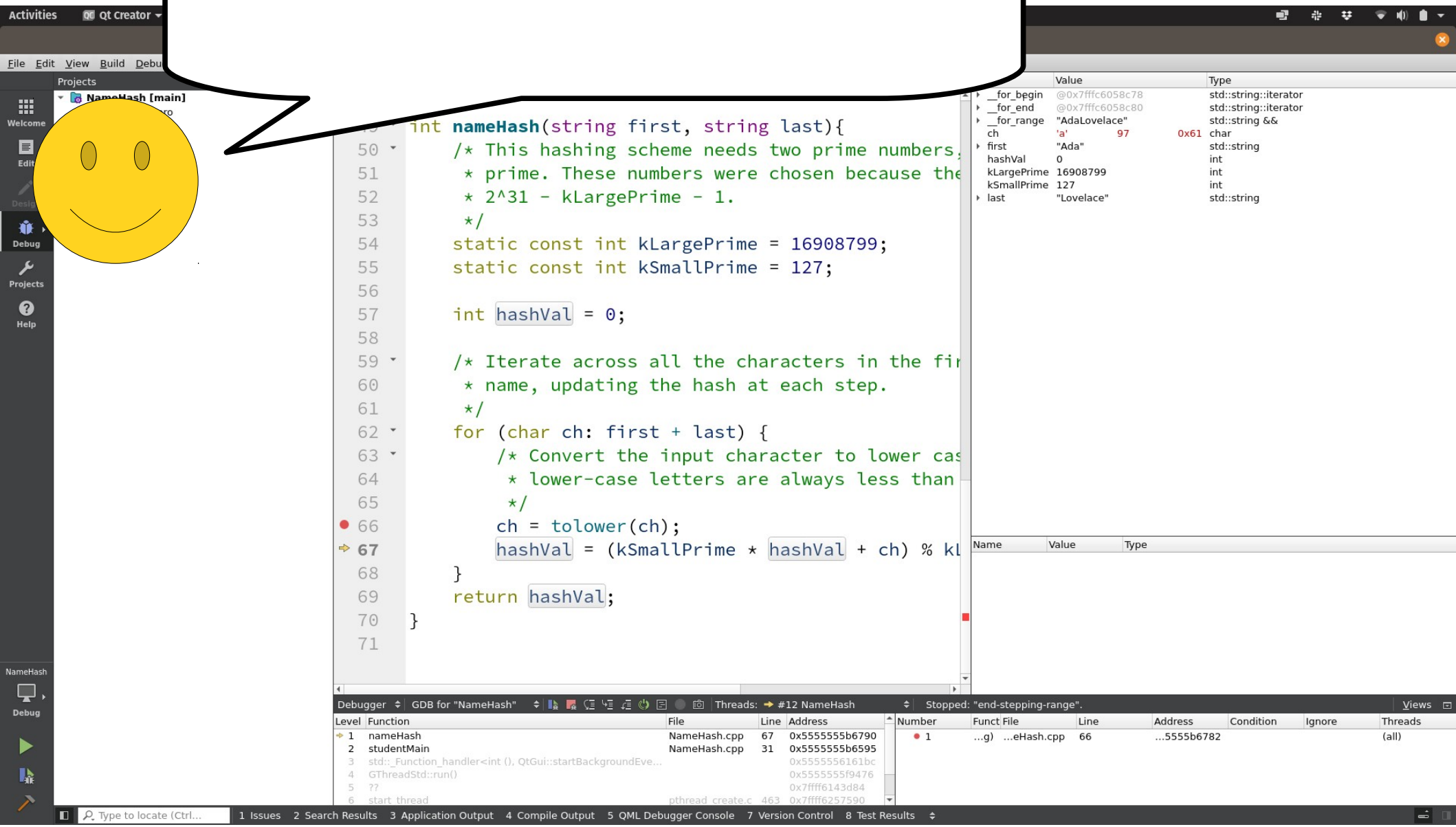
Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), G
4	GThreadStd::run()
5	??
6	start_thread

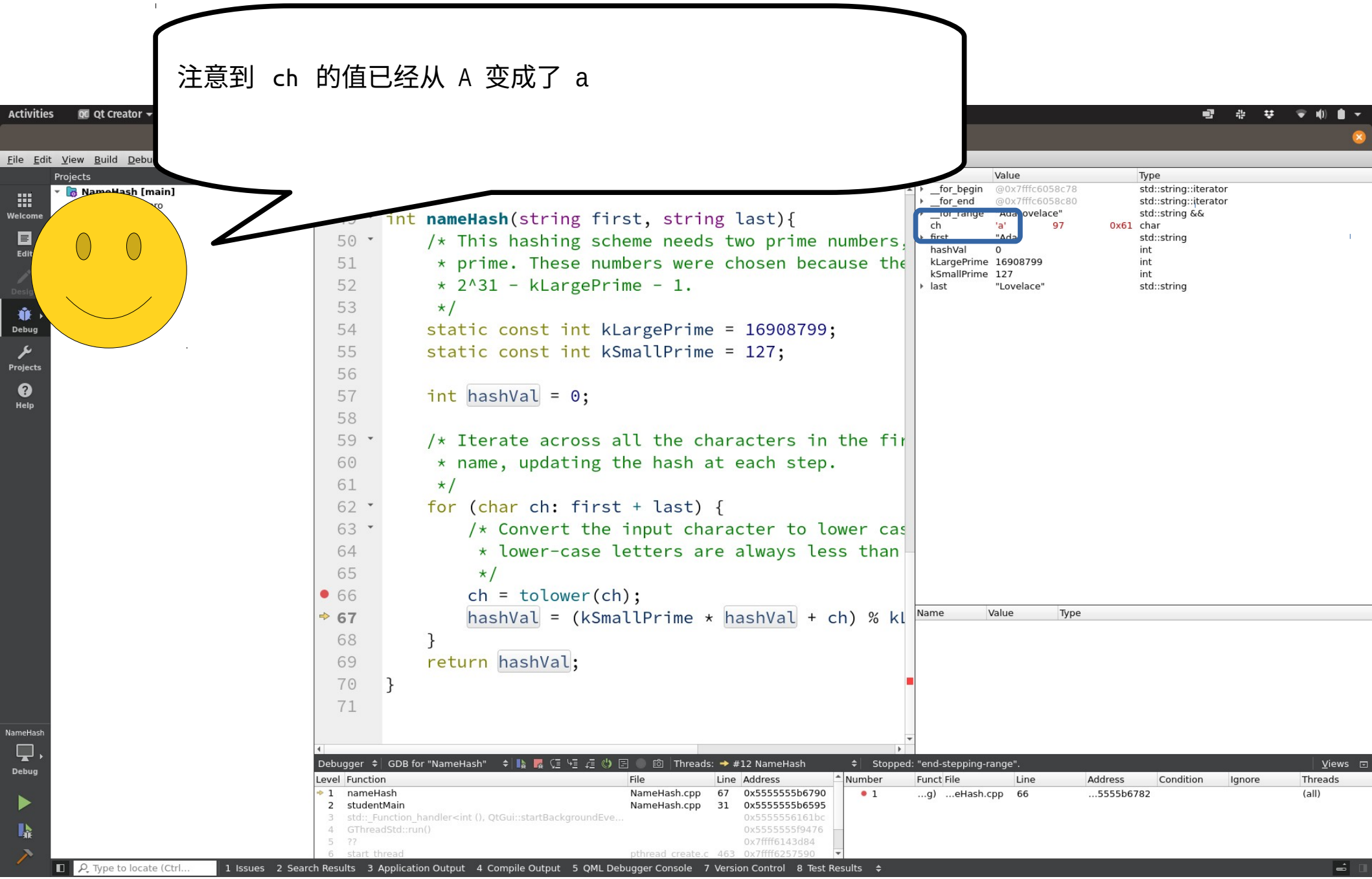
pthread_create.c 463 0x7ffff6257590

Views

Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results

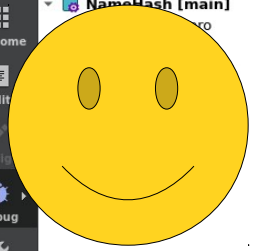
当然，我们也可以通过函数值面板，查看这个过程。




注意到 ch 的值已经从 A 变成了 a
Activities
Qt Creator
File Edit View Build Debu
Projects
NameHash [main]
Welcome
Edit
Debug
Projects
Help
NameHash
Debug
int nameHash(string first, string last){
/* This hashing scheme needs two prime numbers,
* prime. These numbers were chosen because the
* 2^31 - kLargePrime - 1.
*/
static const int kLargePrime = 16908799;
static const int kSmallPrime = 127;
int hashVal = 0;
/* Iterate across all the characters in the fir
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % kL
}
return hashVal;
}
Value
Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "Ada Lovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name
Value
Type
Debugger
GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results


如果你细心的话，可以发现变量一旦发生改变，字体颜色就会标记成红色。
Activities
Qt Creator
File Edit View Build Debu
Projects
NameHash [main]
Welcome
Edit
Debug
Projects
Help
NameHash
Debug
int nameHash(string first, string last){
/* This hashing scheme needs two prime numbers,
* prime. These numbers were chosen because the
* 2^31 - kLargePrime - 1.
*/
static const int kLargePrime = 16908799;
static const int kSmallPrime = 127;
int hashVal = 0;
/* Iterate across all the characters in the fir
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % kL
}
return hashVal;
}
Value
Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "Ada Lovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name
Value
Type
Debugger
GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results

这就表明变量值在刚刚那一步中被改变了，非常的方便。



```
int nameHash(string first, string last){
    /* This hashing scheme needs two prime numbers,
     * prime. These numbers were chosen because the
     * 2^31 - kLargePrime - 1.
     */
    static const int kLargePrime = 16908799;
    static const int kSmallPrime = 127;

    int hashVal = 0;

    /* Iterate across all the characters in the fir
     * name, updating the hash at each step.
     */
    for (char ch: first + last) {
        /* Convert the input character to lower cas
         * lower-case letters are always less than
         */
        ch = tolower(ch);
        hashVal = (kSmallPrime * hashVal + ch) % kL
    }
    return hashVal;
}
```

	Value			Type
__for_begin	@0x7fffc6058c78			std::string::iterator
__for_end	@0x7fffc6058c80			std::string::iterator
__for_range	"Ada Lovelace"			std::string &&
ch	'a'	97	0x61	char
first	"Ada"			std::string
hashVal	0			int
kLargePrime	16908799			int
kSmallPrime	127			int
last	"Lovelace"			std::string

Name	Value	Type

Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	67	0x5555555b6790
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

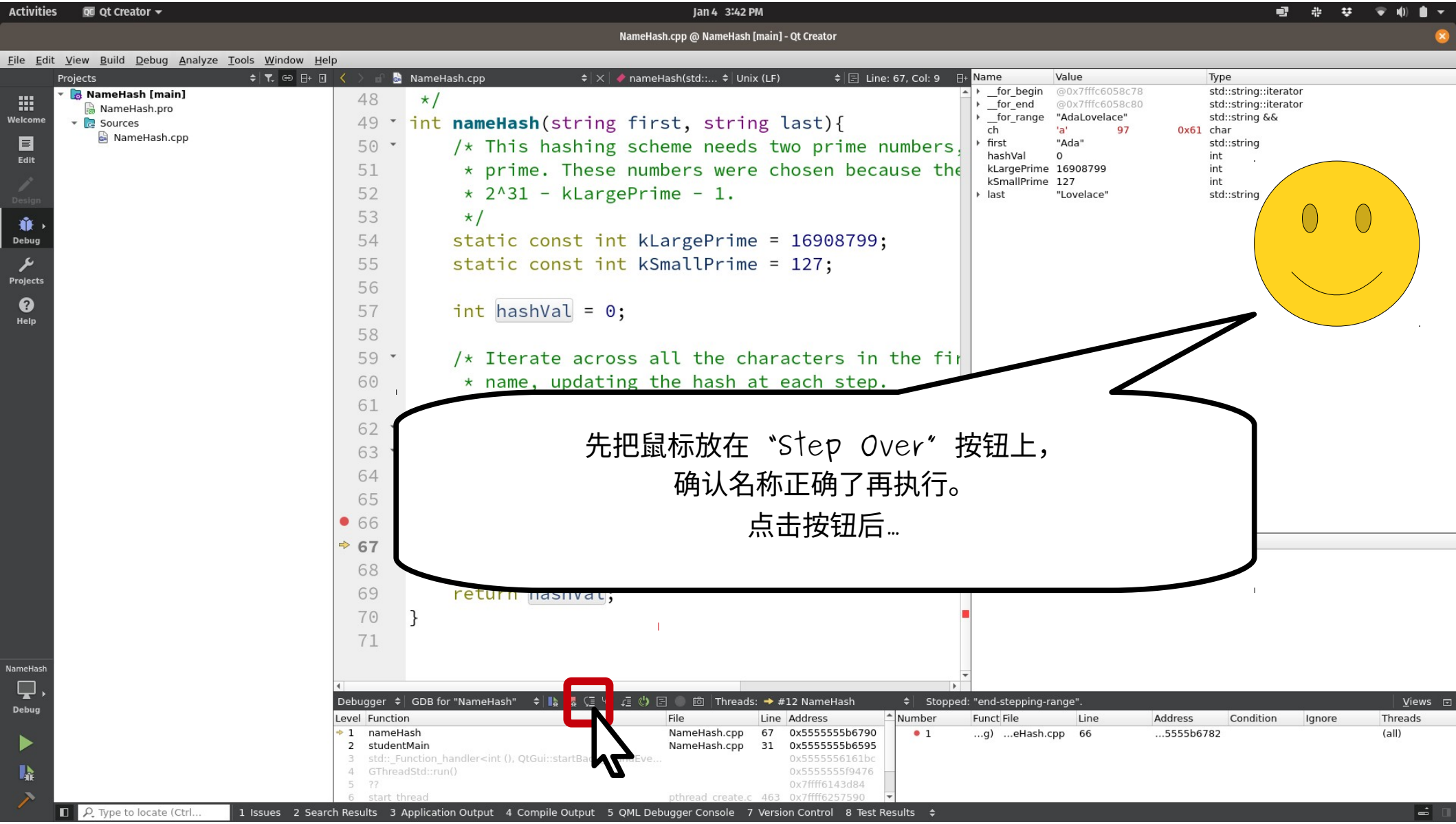
Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)


Activities
Qt Creator
Jan 4 3:42 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
接下来，再看看第 67 行。
int hashVal = 0;
/* Iterate across all the characters in the fi
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % kl
}
return hashVal;
}
Type
::string::iterator
::string::iterator
::string &&
::string
::string
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


不骗你，这是一行比较紧凑的代码。
它执行了一些神奇的数学计算。
Activities
Qt Creator
Jan 4 3:42 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
/* Iterate across all the characters in the fi
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % k
}
return hashVal;
}
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Name Value Type
Type
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


但是我们不必关注那些细节，这里我们只需要知道
计算出的结果保存在 hashVal 就好了。
Activities
Qt Creator
Jan 4 3:42 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
int hashVal = 0;
/* Iterate across all the characters in the fi
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % k
}
return hashVal;
}
Type
::string::iterator
::string::iterator
::string &&
::string
::string
Name
Value
Type
Debugger
GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level
Function
File
Line
Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number
Funct
File
Line
Address
Condition
Ignore
Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results


Activities
Qt Creator
Jan 4 3:42 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
让我们执行一下该行代码，看看会发生什么。
/* Iterate across all the characters in the fi
* name, updating the hash at each step.
*/
for (char ch: first + last) {
/* Convert the input character to lower cas
* lower-case letters are always less than
*/
ch = tolower(ch);
hashVal = (kSmallPrime * hashVal + ch) % k
}
return hashVal;
}
Type
::string::iterator
::string::iterator
::string &&
::string
::string
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:42 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::... Unix (LF) Line: 67, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61
62
63
64
65
66
67
68
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada" std::string
hashVal 0 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
先把鼠标放在 "Step Over" 按钮上，
确认名称正确了再执行。
点击按钮后…
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBa...
4 GThreadStd::run()
5 ??
6 start_thread pthread_create.c 463
0x5555556161bc
0x5555555f9476
0x7ffff6143d84
0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:48 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 62, Col: 5

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int hashVal = 0;
58
59      /* Iterate across all the characters in the fir
60       * name, updating the hash at each step.
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	97	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string

Name	Value	Type

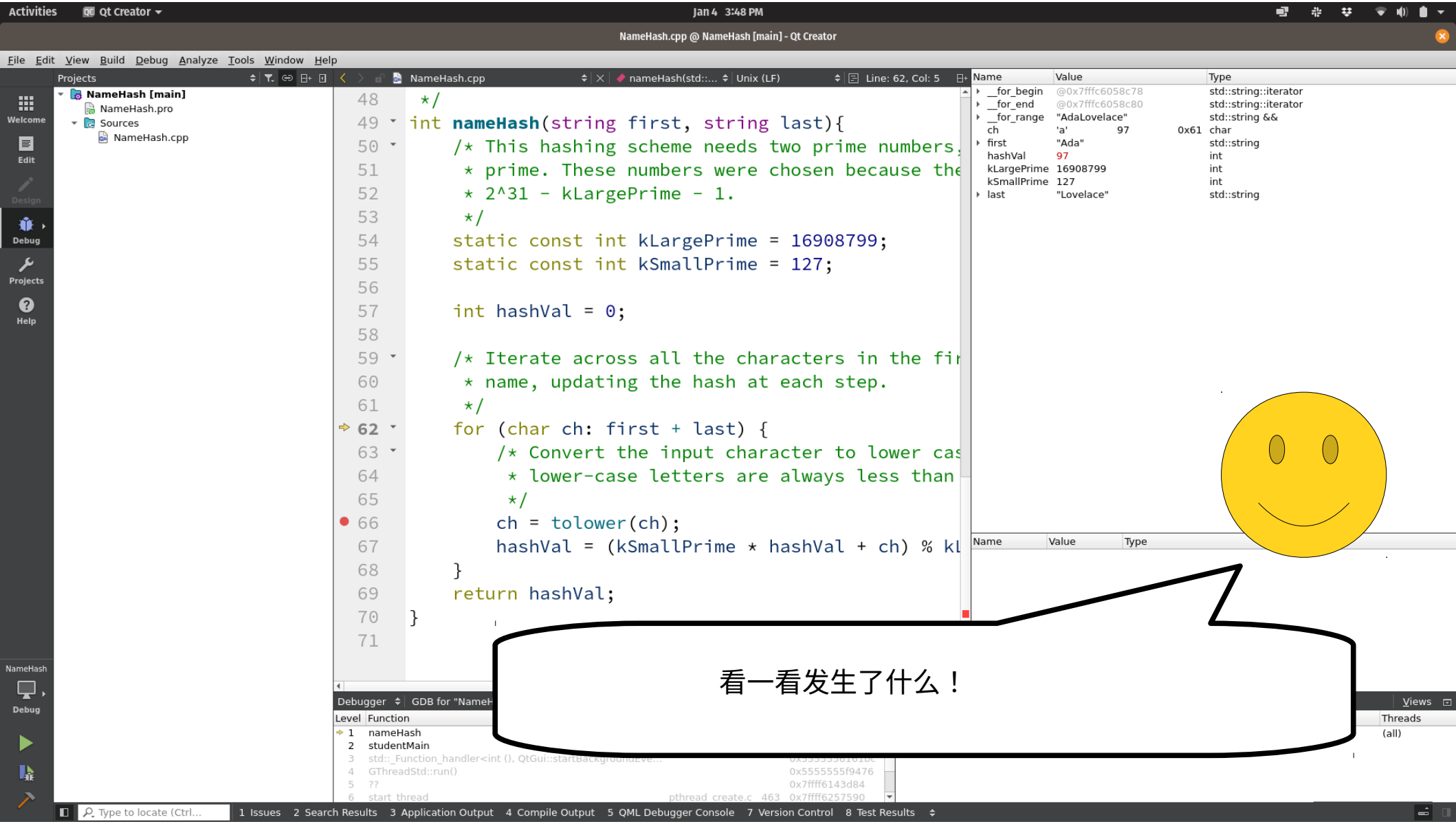
… 你会来到这个界面！

Debugger | GDB for "NameH

Level	Function
1	nameHash
2	studentMain
3	std::_Function_handler<int (), QtGui::startBackgroundEve...
4	GThreadStd::run()
5	??
6	start_thread

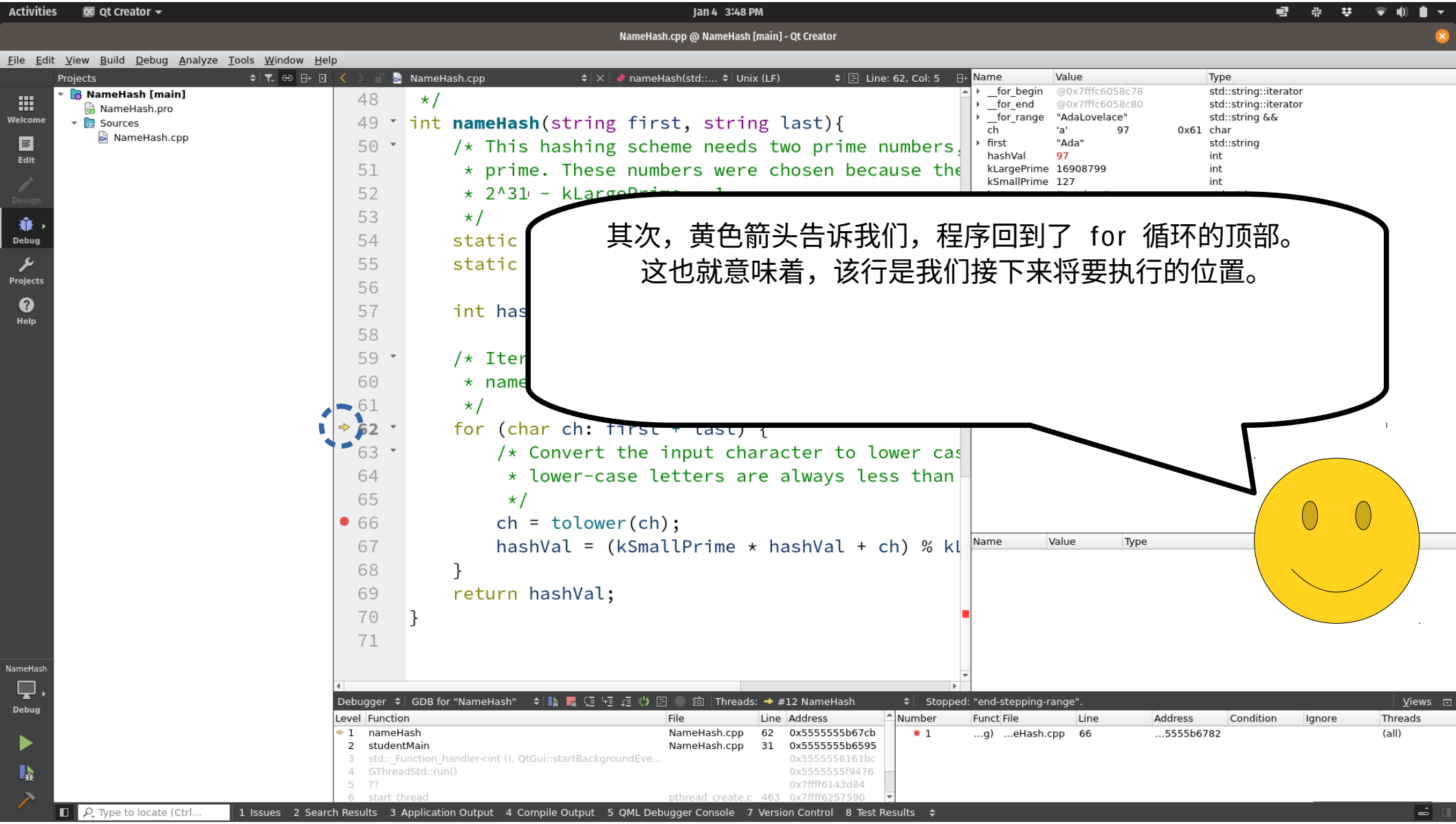
Views | Threads: (all)

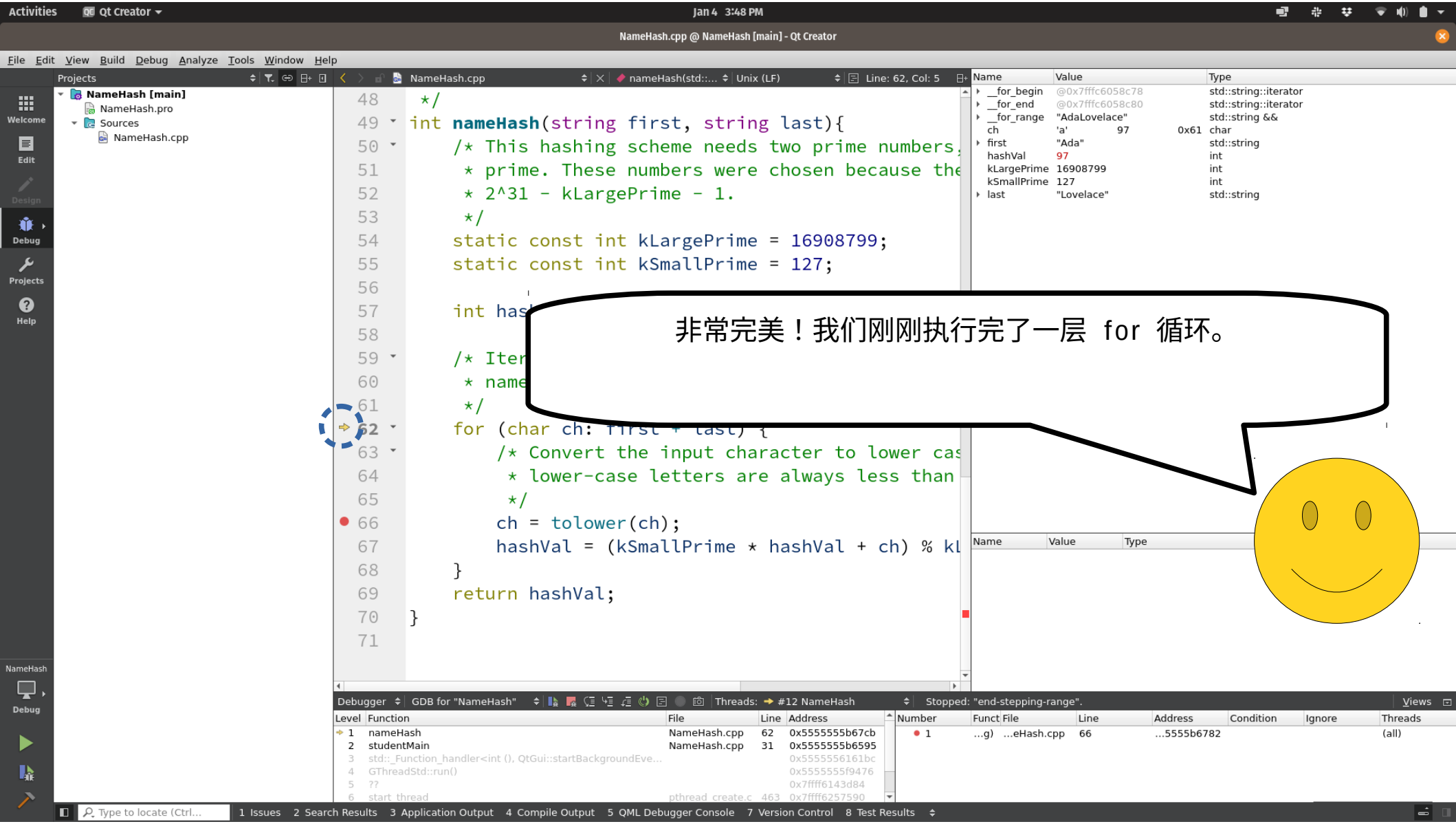
Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results


Activities
Qt Creator
Jan 4 3:48 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
Name Value Type
Debugger GDB for "NameH
Level Function
1 nameHash
2 studentMain
3 std::_Function_handler<int (), QtGui::startBackgroundEve...
4 GThreadStd::run()
5 ??
6 start_thread pthread_create.c 463 0x7ffff6257590
0x5555555f9476
0x7ffff6143d84
Views
Threads
(all)
看一看发生了什么！
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

首先 hashVal 的值变成了 97
因为这个值是红色的，所以我们很容易发现它。
其他值的颜色没变化，所以它们的值也没有改变。

```
/* This hashing scheme needs two prime numbers,
 * prime. These numbers were chosen because the
 * 2^31 - kLargePrime - 1.
 */
static const int kLargePrime = 16908799;
static const int kSmallPrime = 127;

int hashVal = 0;

/* Iterate across all the characters in the fir
 * name, updating the hash at each step.
 */
for (char ch: first + last) {
    /* Convert the input character to lower cas
     * lower-case letters are always less than
     */
    ch = tolower(ch);
    hashVal = (kSmallPrime * hashVal + ch) % kL
}
return hashVal;
}
```


Activities
Qt Creator
Jan 4 3:48 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLarge
53 */
54 static
55 static
56
57 int has
58
59 /* Iter
60 * name
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
其次，黄色箭头告诉我们，程序回到了 for 循环的顶部。
这也就意味着，该行是我们接下来将要执行的位置。
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:48 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int has
58
59 /* Iter
60 * name
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
非常完美！我们刚刚执行完了一层 for 循环。
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 3:48 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

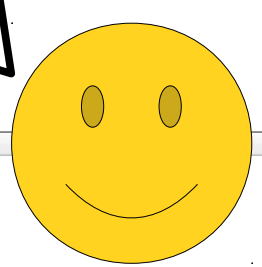
- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 62, Col: 5

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const int kSmallPrime = 127;
56
57      int has
58
59      /* Iter
60       * name
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'a' 97 0x61	char
first	"Ada"	std::string
hashVal	97	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string

继续加油！

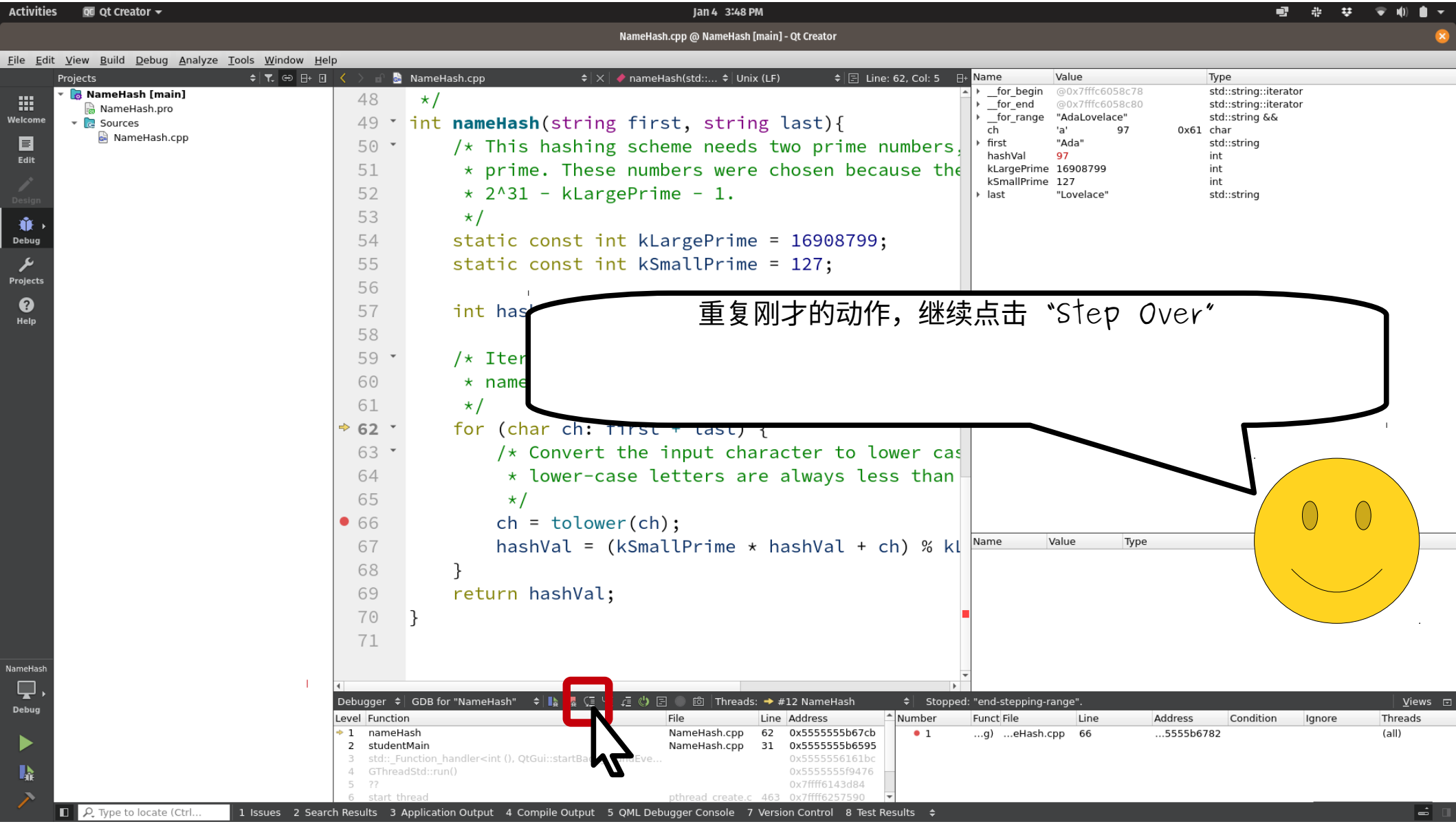


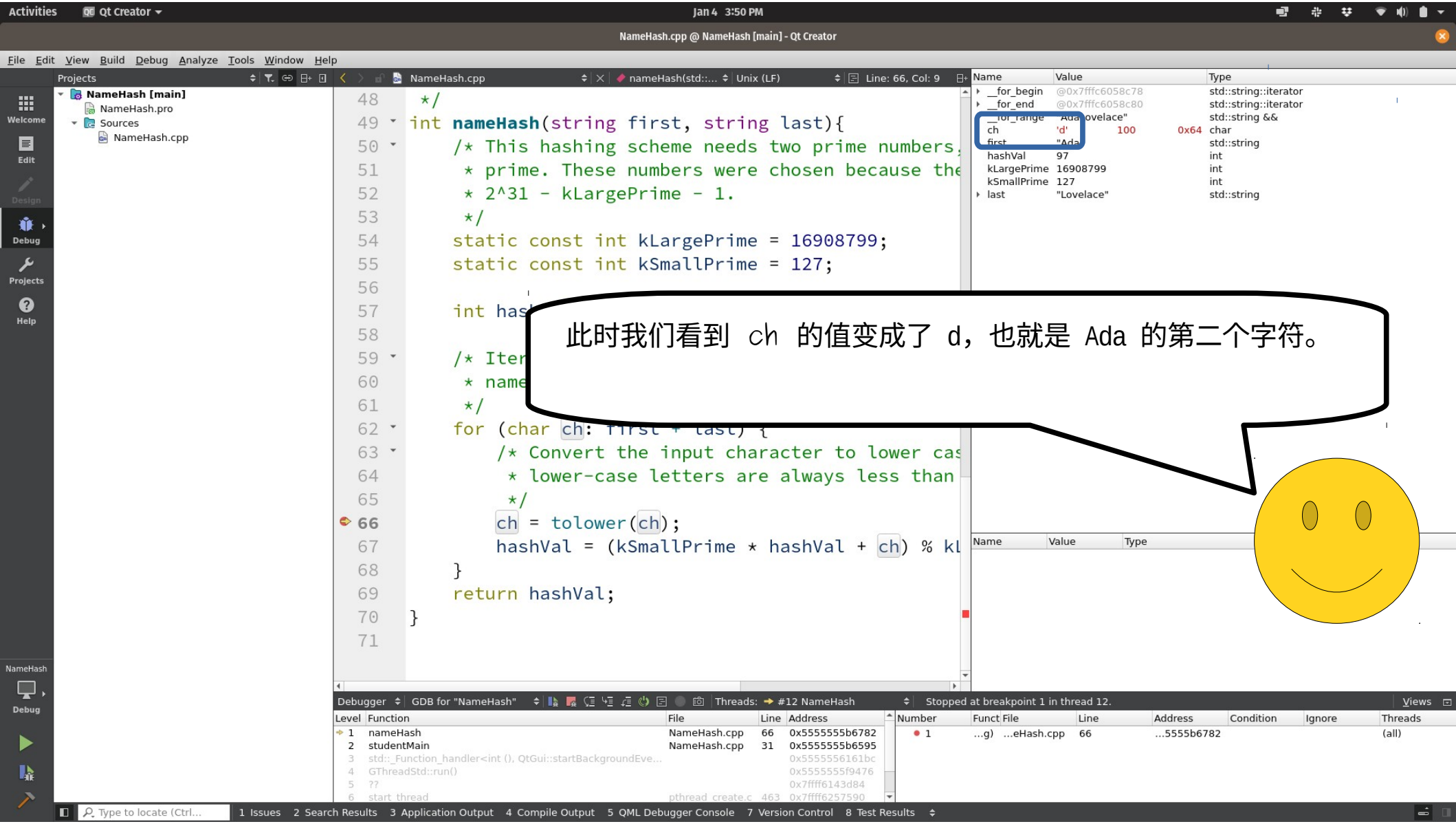
Debugger | GDB for "NameHash" | Threads: #12 NameHash | Stopped: "end-stepping-range". | Views

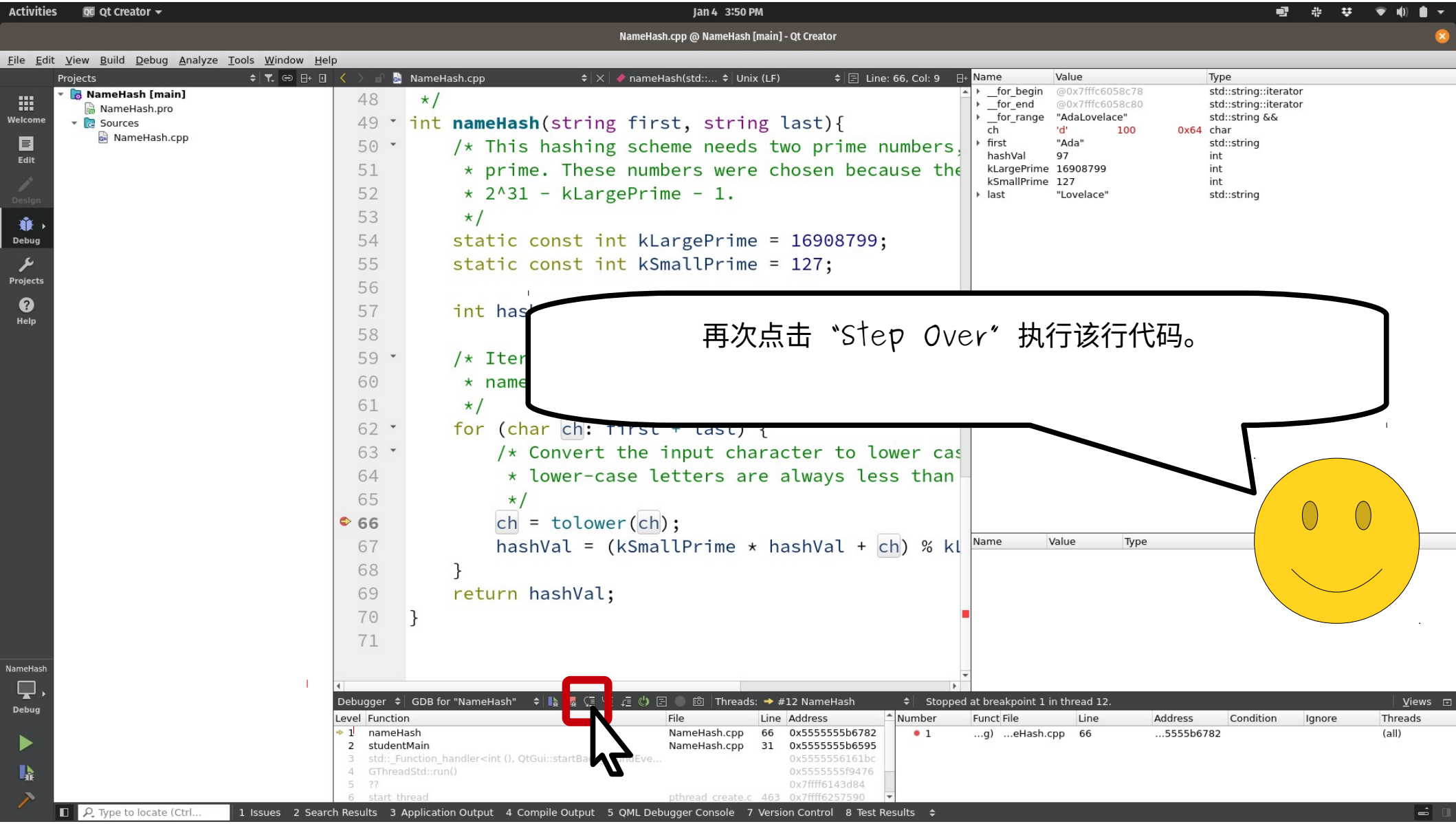
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	62	0x5555555b67cb
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results


Activities
Qt Creator
Jan 4 3:48 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int has
58
59 /* Iter
60 * name
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'a' 97 0x61 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
重复刚才的动作，继续点击 "Step Over"
Name Value Type
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBa...uEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:50 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int has
58
59 /* Iter
60 * name
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "Ada Lovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
此时我们看到 ch 的值变成了 d，也就是 Ada 的第二个字符。
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:50 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 66, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int has
58
59 /* Iter
60 * name
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
再次点击 "Step Over" 执行该行代码。
Name Value Type
Debugger
GDB for "NameHash"
Threads: #12 NameHash
Stopped at breakpoint 1 in thread 12.
Views
Level Function File Line Address
1 nameHash NameHash.cpp 66 0x5555555b6782
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBa...uEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues
2 Search Results
3 Application Output
4 Compile Output
5 QML Debugger Console
7 Version Control
8 Test Results

Activities Qt Creator Jan 4 3:52 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

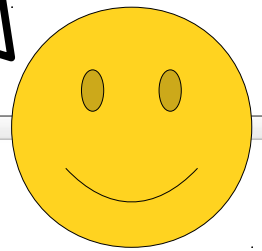
- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | nameHash(std::... | Unix (LF) | Line: 67, Col: 9

```
48   */
49  int nameHash(string first, string last){
50      /* This hashing scheme needs two prime numbers,
51       * prime. These numbers were chosen because the
52       * 2^31 - kLargePrime - 1.
53       */
54      static const int kLargePrime = 16908799;
55      static const
56
57      int has
58
59      /* Iter
60       * name
61       */
62      for (char ch: first + last) {
63          /* Convert the input character to lower cas
64           * lower-case letters are always less than
65           */
66          ch = tolower(ch);
67          hashVal = (kSmallPrime * hashVal + ch) % kL
68      }
69      return hashVal;
70  }
71
```

Name	Value	Type
__for_begin	@0x7fffc6058c78	std::string::iterator
__for_end	@0x7fffc6058c80	std::string::iterator
__for_range	"AdaLovelace"	std::string &&
ch	'd' 100 0x64	char
first	"Ada"	std::string
hashVal	97	int
kLargePrime	16908799	int
kSmallPrime	127	int
last	"Lovelace"	std::string

你应该处在这个位置。注意，没有一个值发生改变。
显然，d 转换成小写还是 d。



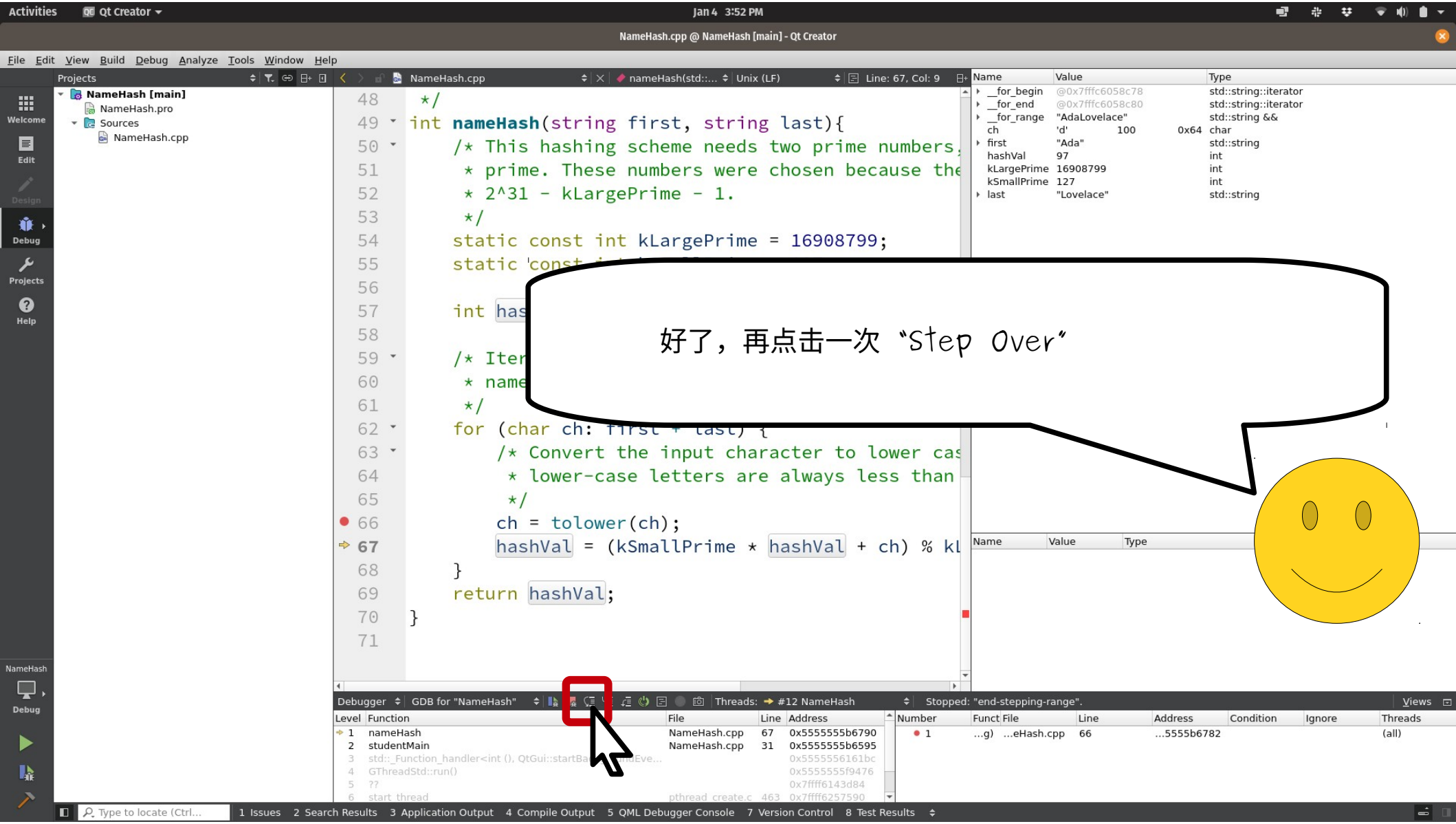
Name	Value	Type

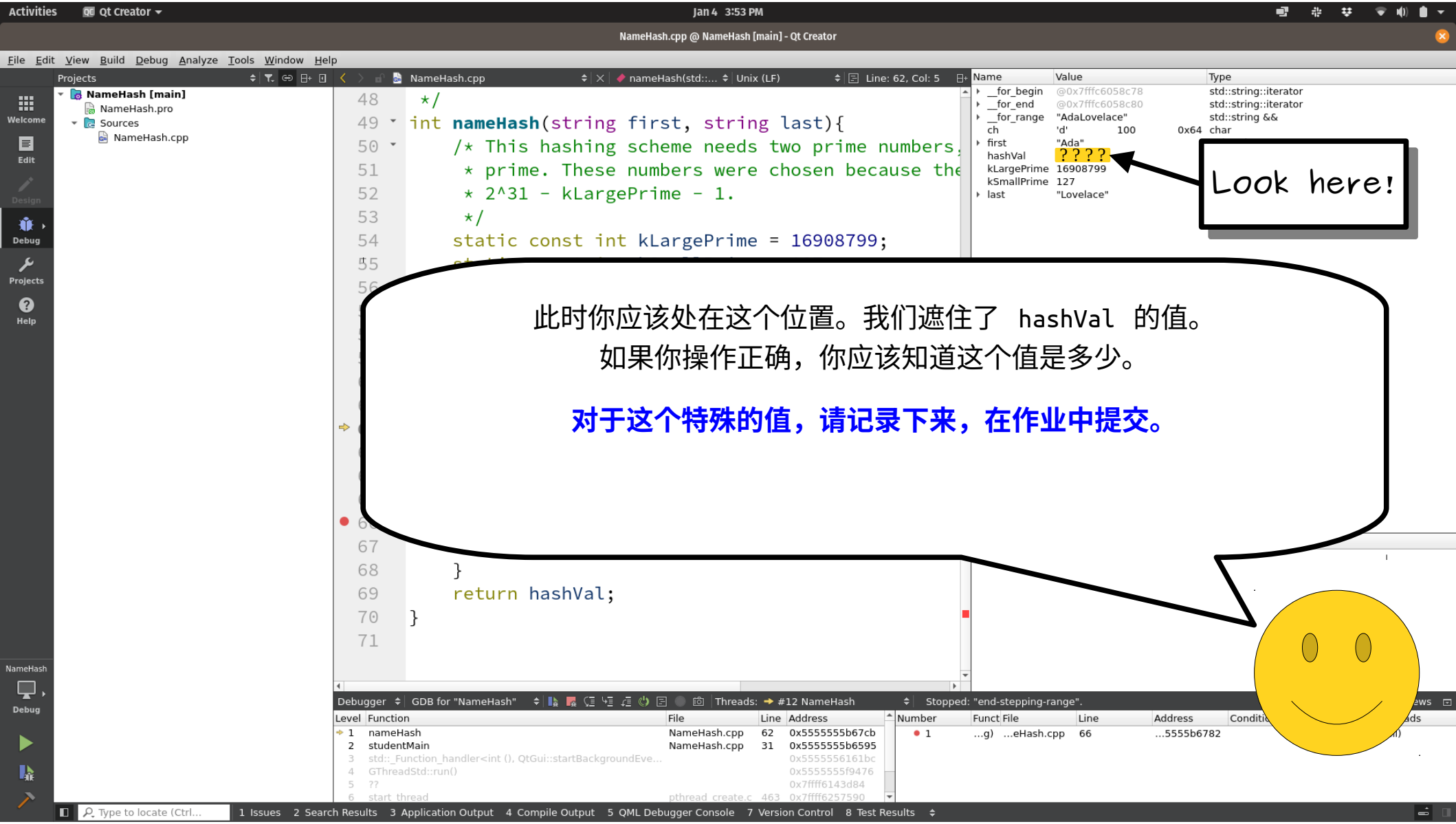
Debugger | GDB for "NameHash" | Threads: #12 NameHash | Stopped: "end-stepping-range". | Views

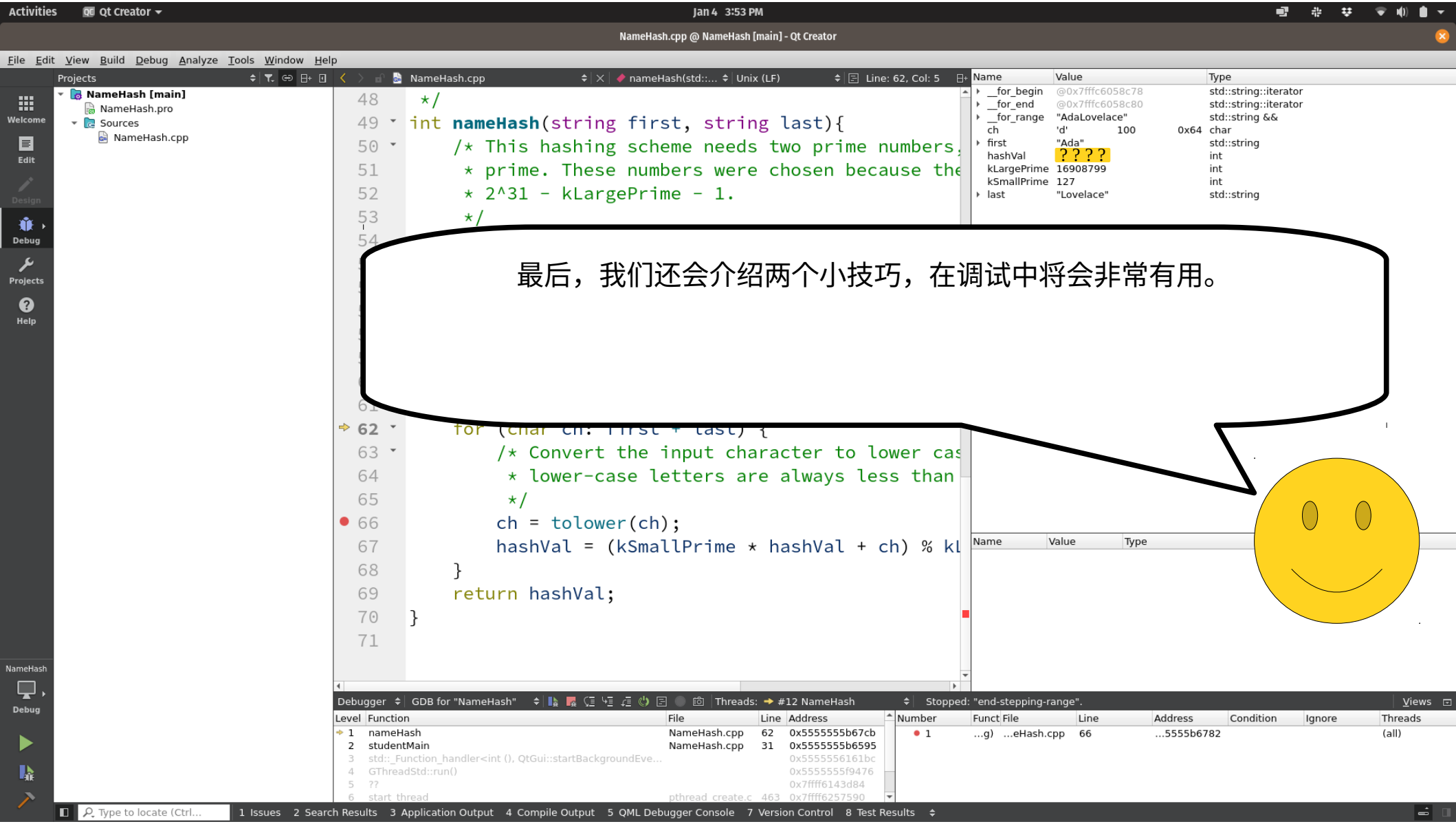
Level	Function	File	Line	Address
1	nameHash	NameHash.cpp	67	0x5555555b6790
2	studentMain	NameHash.cpp	31	0x5555555b6595
3	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
4	GThreadStd::run()			0x5555555f9476
5	??			0x7ffff6143d84
6	start_thread	pthread_create.c	463	0x7ffff6257590

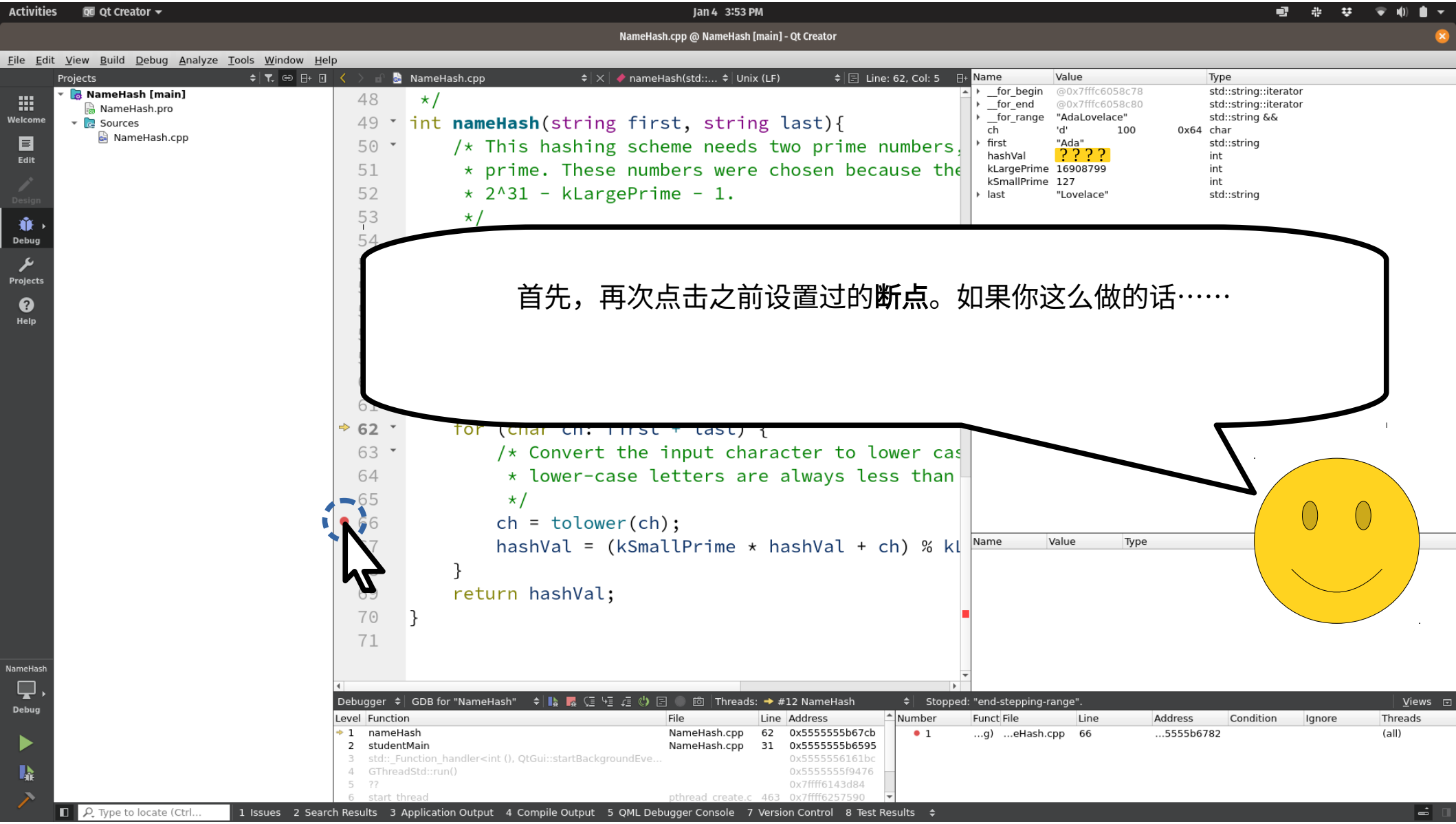
Number	Funct	File	Line	Address	Condition	Ignore	Threads
1	...g)	...eHash.cpp	66	...5555b6782			(all)

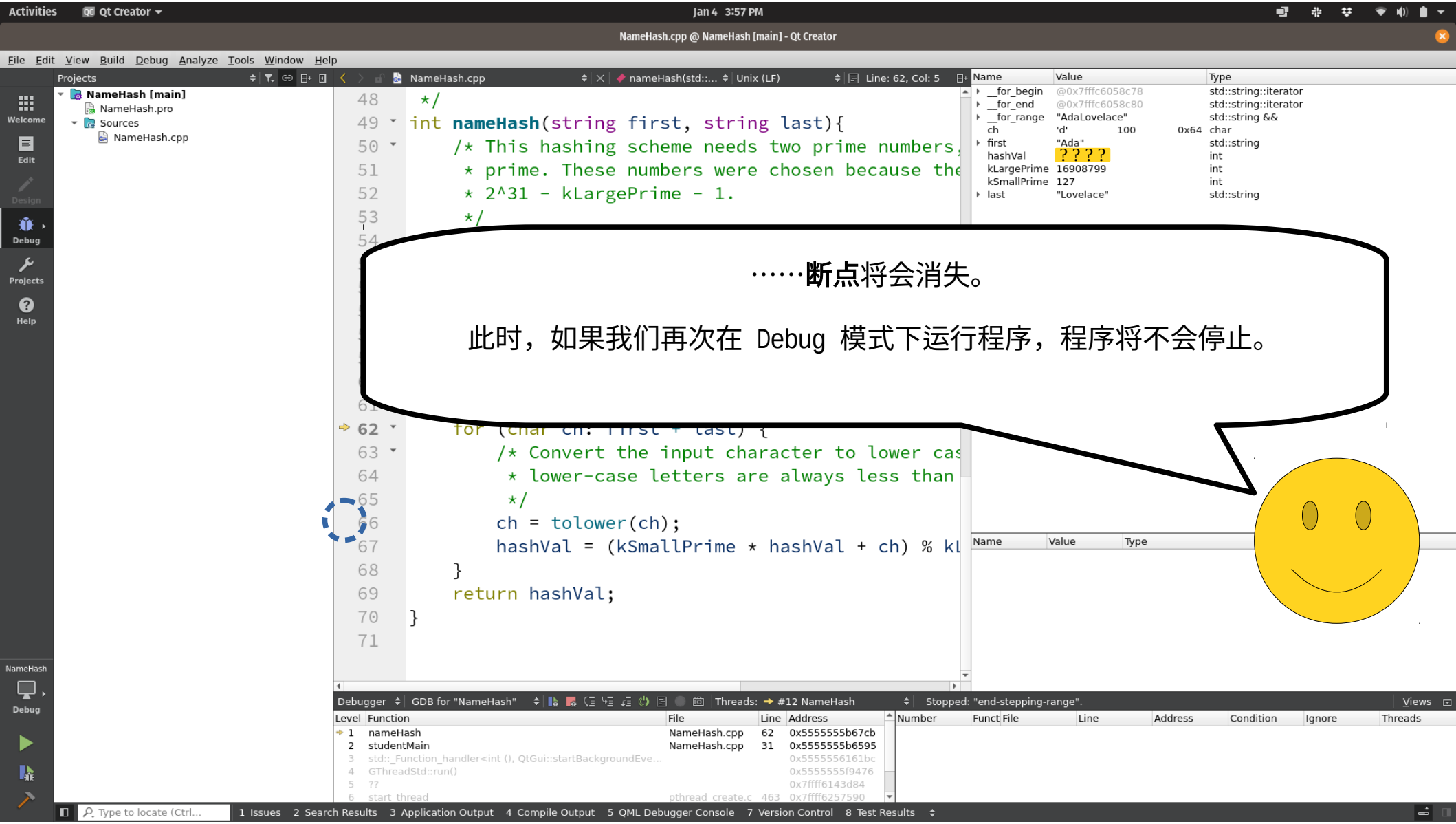
Type to locate (Ctrl... | 1 Issues | 2 Search Results | 3 Application Output | 4 Compile Output | 5 QML Debugger Console | 7 Version Control | 8 Test Results

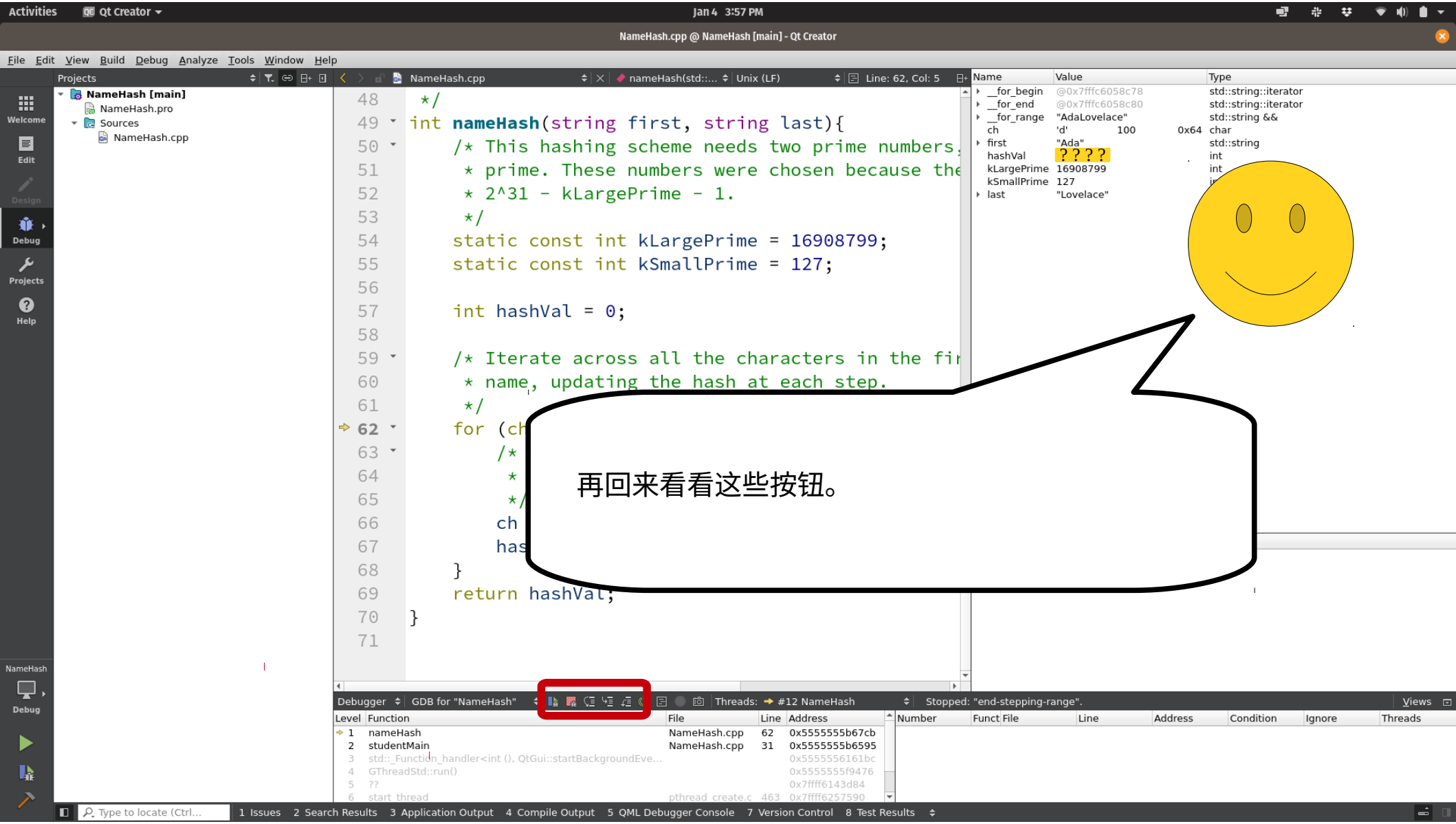

Activities
Qt Creator
Jan 4 3:52 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::... Unix (LF) Line: 67, Col: 9
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const
56
57 int has
58
59 /* Iter
60 * name
61 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal 97 int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
好了，再点击一次 "Step Over"
Name Value Type
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 67 0x5555555b6790
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBa...EventLoopEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:53 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
nameHash(std::... Unix (LF) Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
67
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada"
hashVal ????
kLargePrime 16908799
kSmallPrime 127
last "Lovelace"
Look here!
此时你应该处在这个位置。我们遮住了 hashVal 的值。
如果你操作正确，你应该知道这个值是多少。
对于这个特殊的值，请记录下来，在作业中提交。
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range".
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address
1 ...g) ...eHash.cpp 66 ...5555b6782
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:53 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
最后，我们还会介绍两个小技巧，在调试中将会非常有用。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

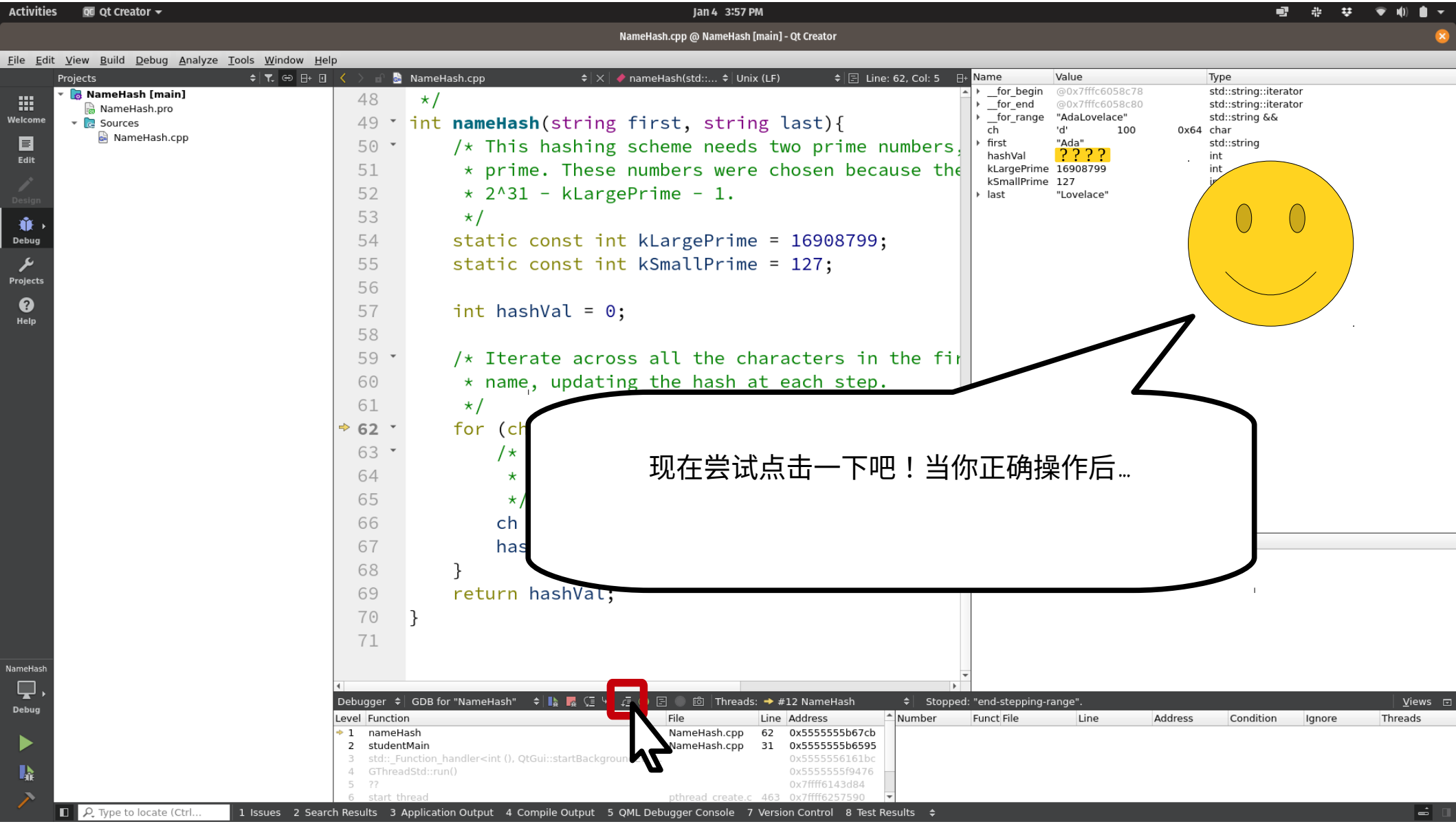

Activities
Qt Creator
Jan 4 3:53 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
首先，再次点击之前设置过的断点。如果你这么做的话……
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
1 ...g) ...eHash.cpp 66 ...5555b6782 (all)
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

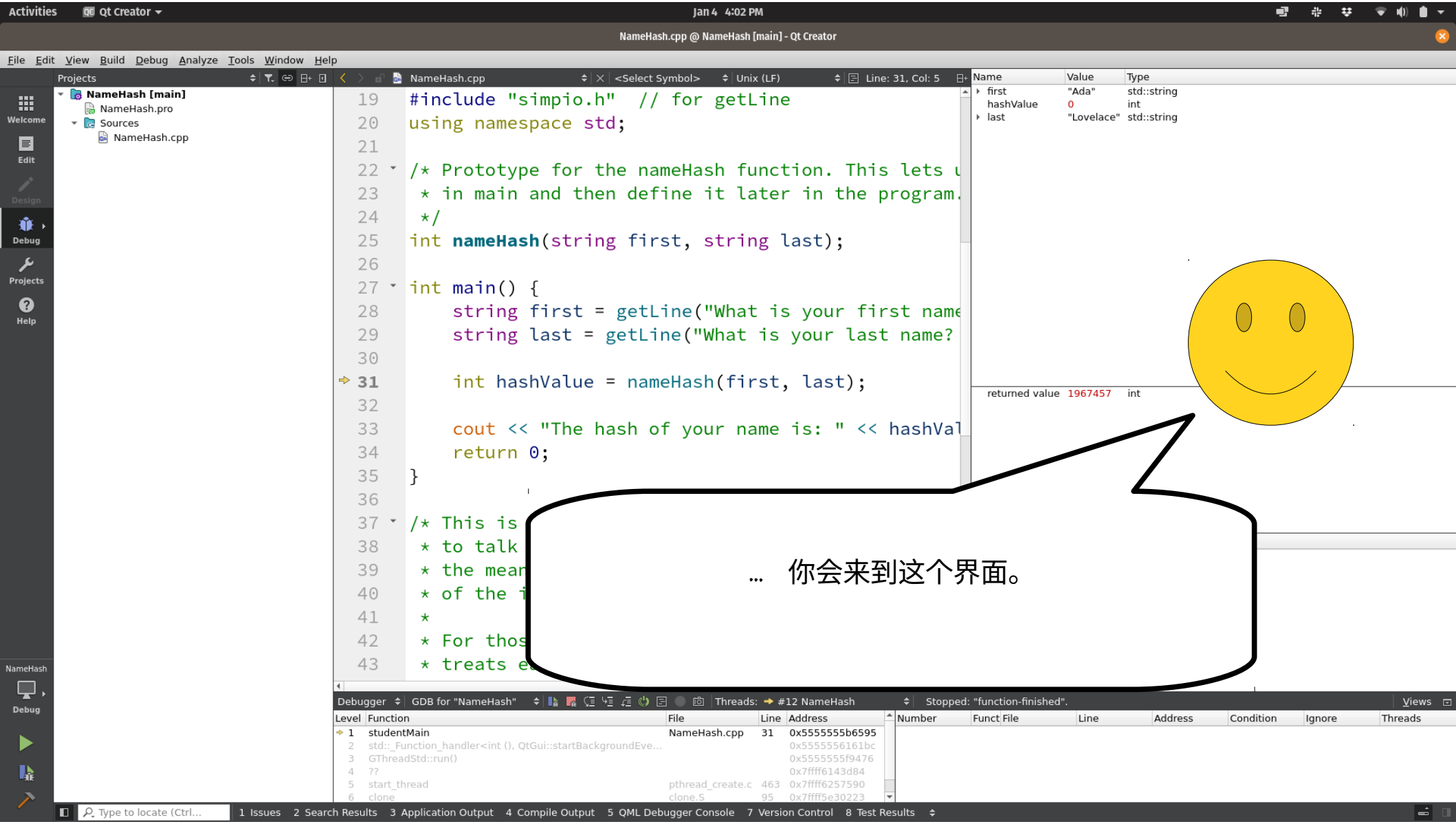

Activities
Qt Creator
Jan 4 3:57 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
62 for (char ch: first + last) {
63 /* Convert the input character to lower cas
64 * lower-case letters are always less than
65 */
66 ch = tolower(ch);
67 hashVal = (kSmallPrime * hashVal + ch) % kL
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace" std::string
……断点将会消失。
此时，如果我们再次在 Debug 模式下运行程序，程序将不会停止。
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:57 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace"
再回来看看这些按钮。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:57 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace"
把鼠标悬停在最右边的按钮上。
此时会提示 "Step Out"
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:57 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace"
先不要点击。
当你决定点击的时候，程序会执行完 nameHash 函数，并返回值。
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 3:57 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
nameHash(std::...
Unix (LF)
Line: 62, Col: 5
48 */
49 int nameHash(string first, string last){
50 /* This hashing scheme needs two prime numbers,
51 * prime. These numbers were chosen because the
52 * 2^31 - kLargePrime - 1.
53 */
54 static const int kLargePrime = 16908799;
55 static const int kSmallPrime = 127;
56
57 int hashVal = 0;
58
59 /* Iterate across all the characters in the fir
60 * name, updating the hash at each step.
61 */
62 for (ch
63 /*
64 *
65 */
66 ch
67 has
68 }
69 return hashVal;
70 }
71
Name Value Type
__for_begin @0x7fffc6058c78 std::string::iterator
__for_end @0x7fffc6058c80 std::string::iterator
__for_range "AdaLovelace" std::string &&
ch 'd' 100 0x64 char
first "Ada" std::string
hashVal ???? int
kLargePrime 16908799 int
kSmallPrime 127 int
last "Lovelace"
现在尝试点击一下吧！当你正确操作后…
Debugger GDB for "NameHash"
Threads: #12 NameHash
Stopped: "end-stepping-range".
Views
Level Function File Line Address
1 nameHash NameHash.cpp 62 0x5555555b67cb
2 studentMain NameHash.cpp 31 0x5555555b6595
3 std::_Function_handler<int (), QtGui::startBackgroun 0x5555556161bc
4 GThreadStd::run() 0x5555555f9476
5 ?? 0x7ffff6143d84
6 start_thread pthread_create.c 463 0x7ffff6257590
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 4:02 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
<Select Symbol>
Unix (LF)
Line: 31, Col: 5
19 #include "simpio.h" // for getLine
20 using namespace std;
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then define it later in the program.
24 */
25 int nameHash(string first, string last);
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is
38 * to talk
39 * the mean
40 * of the
41 *
42 * For thos
43 * treats e
Name Value Type
first "Ada" std::string
hashValue 0 int
last "Lovelace" std::string
returned value 1967457 int
… 你会来到这个界面。
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "function-finished". Views
Level Function File Line Address
1 studentMain NameHash.cpp 31 0x5555555b6595
2 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
3 GThreadStd::run() 0x5555555f9476
4 ?? 0x7ffff6143d84
5 start_thread pthread_create.c 463 0x7ffff6257590
6 clone clone.S 95 0x7ffff5e30223
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 4:02 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

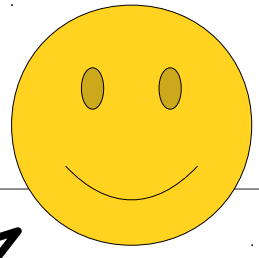
- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | <Select Symbol> | Unix (LF) | Line: 31, Col: 5

```
19  #include "simpio.h"  // for getLine
20  using namespace std;
21
22  /* Prototype for the nameHash function. This lets u
23   * in main and then define it later in the program.
24   */
25  int nameHash(string first, string last);
26
27  int main() {
28      string first = getLine("What is your first name
29      string last = getLine("What is your last name?
30
31      int hashValue = nameHash(first, last);
32
33      cout << "The hash of your name is: " << hashVal
34      return 0;
35  }
36
37  /* This is
38   * to talk
39   * the mean
40   * of the i
41   *
42   * For thos
43   * treats e
```

Name	Value	Type
first	"Ada"	std::string
hashValue	0	int
last	"Lovelace"	std::string

returned value 1967457 int



让我们分析一下。我们究竟处在哪里？

Debugger | GDB for "NameHash" | Threads: #12 NameHash | Stopped: "function-finished". | Views

Level	Function	File	Line	Address
1	studentMain	NameHash.cpp	31	0x5555555b6595
2	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
3	GThreadStd::run()			0x5555555f9476
4	??			0x7ffff6143d84
5	start_thread	pthread_create.c	463	0x7ffff6257590
6	clone	clone.S	95	0x7ffff5e30223

Number	Funct	File	Line	Address	Condition	Ignore	Threads

Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

Activities Qt Creator Jan 4 4:02 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

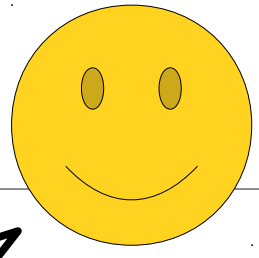
- NameHash [main]
  - NameHash.pro
  - Sources
    - NameHash.cpp

NameHash.cpp | <Select Symbol> | Unix (LF) | Line: 31, Col: 5

```
19  #include "simpio.h"  // for getLine
20  using namespace std;
21
22  /* Prototype for the nameHash function. This lets u
23   * in main and then define it later in the program.
24   */
25  int nameHash(string first, string last);
26
27  int main() {
28      string first = getLine("What is your first name
29      string last = getLine("What is your last name?
30
31      int hashValue = nameHash(first, last);
32
33      cout << "The hash of your name is: " << hashVal
34      return 0;
35  }
36
37  /* This is
38   * to talk
39   * the mean
40   * of the i
41   *
42   * For thos
43   * treats e
```

Name	Value	Type
first	"Ada"	std::string
hashValue	0	int
last	"Lovelace"	std::string

returned value 1967457 int



黄色箭头表明我们已经回到了 main 函数。
很棒！

Debugger | GDB for "NameHash" | Threads: #12 NameHash | Stopped: "function-finished". | Views

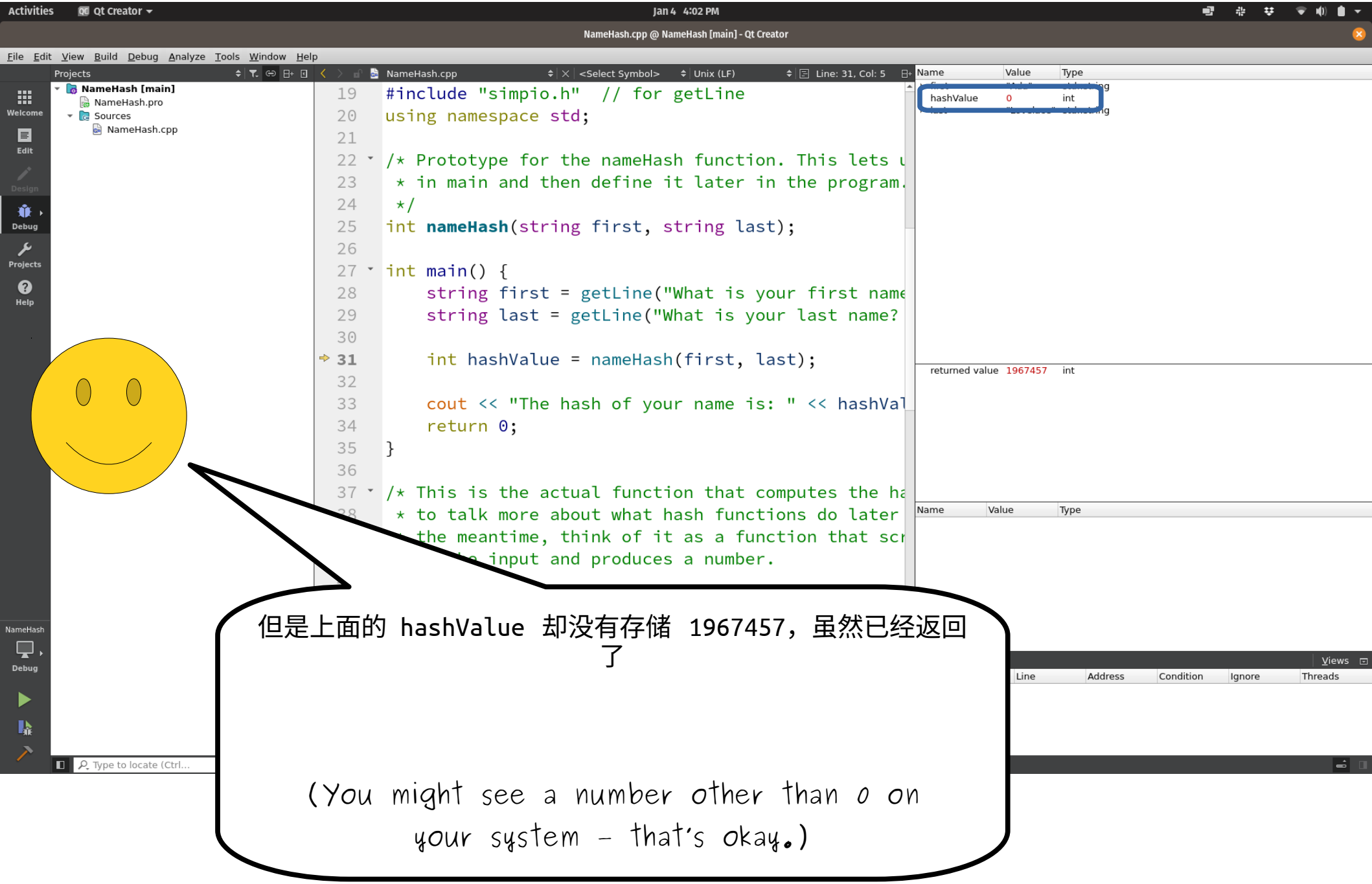
Level	Function	File	Line	Address
1	studentMain	NameHash.cpp	31	0x5555555b6595
2	std::_Function_handler<int (), QtGui::startBackgroundEve...			0x5555556161bc
3	GThreadStd::run()			0x5555555f9476
4	??			0x7ffff6143d84
5	start_thread	pthread_create.c	463	0x7ffff6257590
6	clone	clone.S	95	0x7ffff5e30223

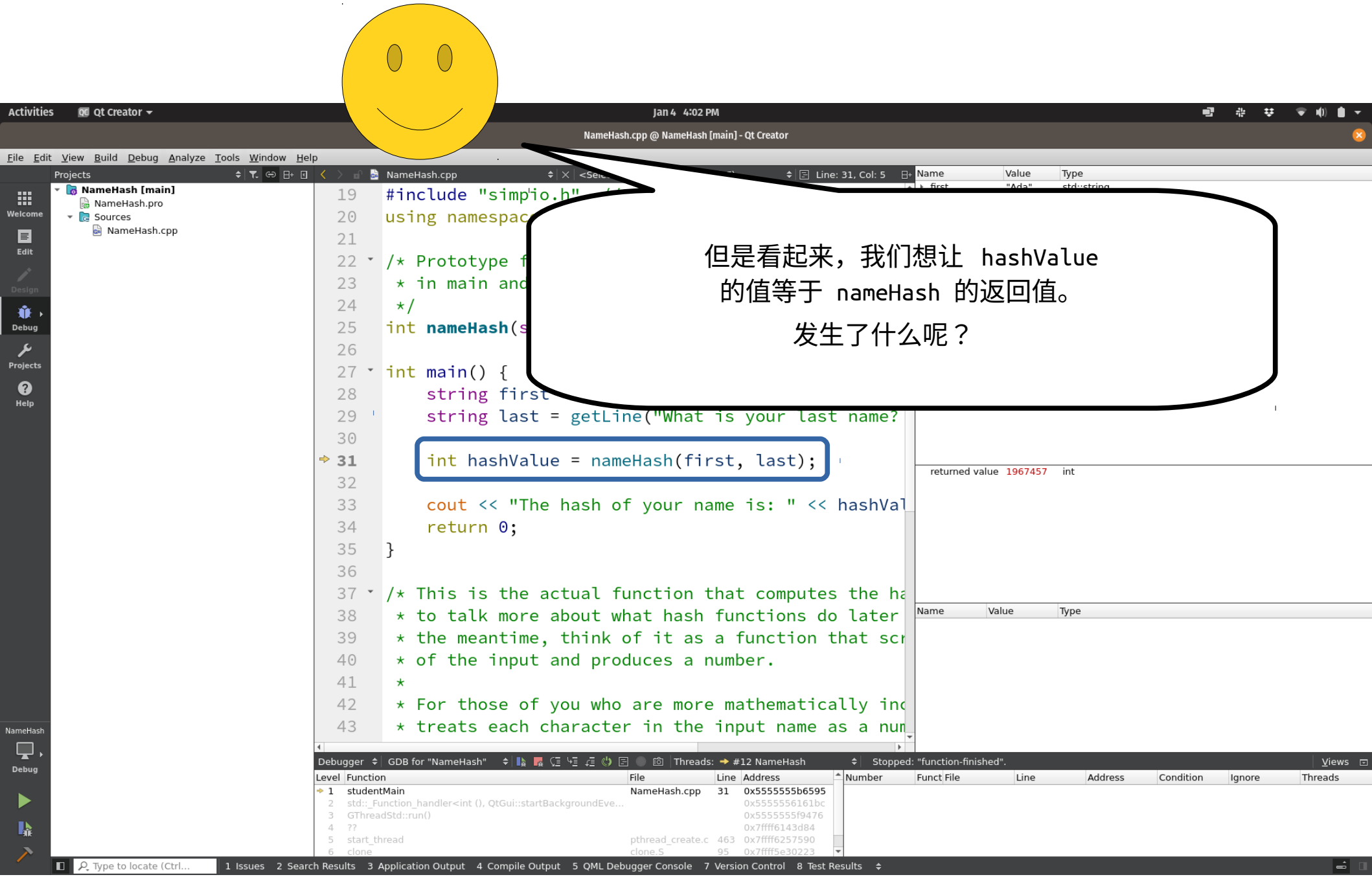
Number	Funct File	Line	Address	Condition	Ignore	Threads

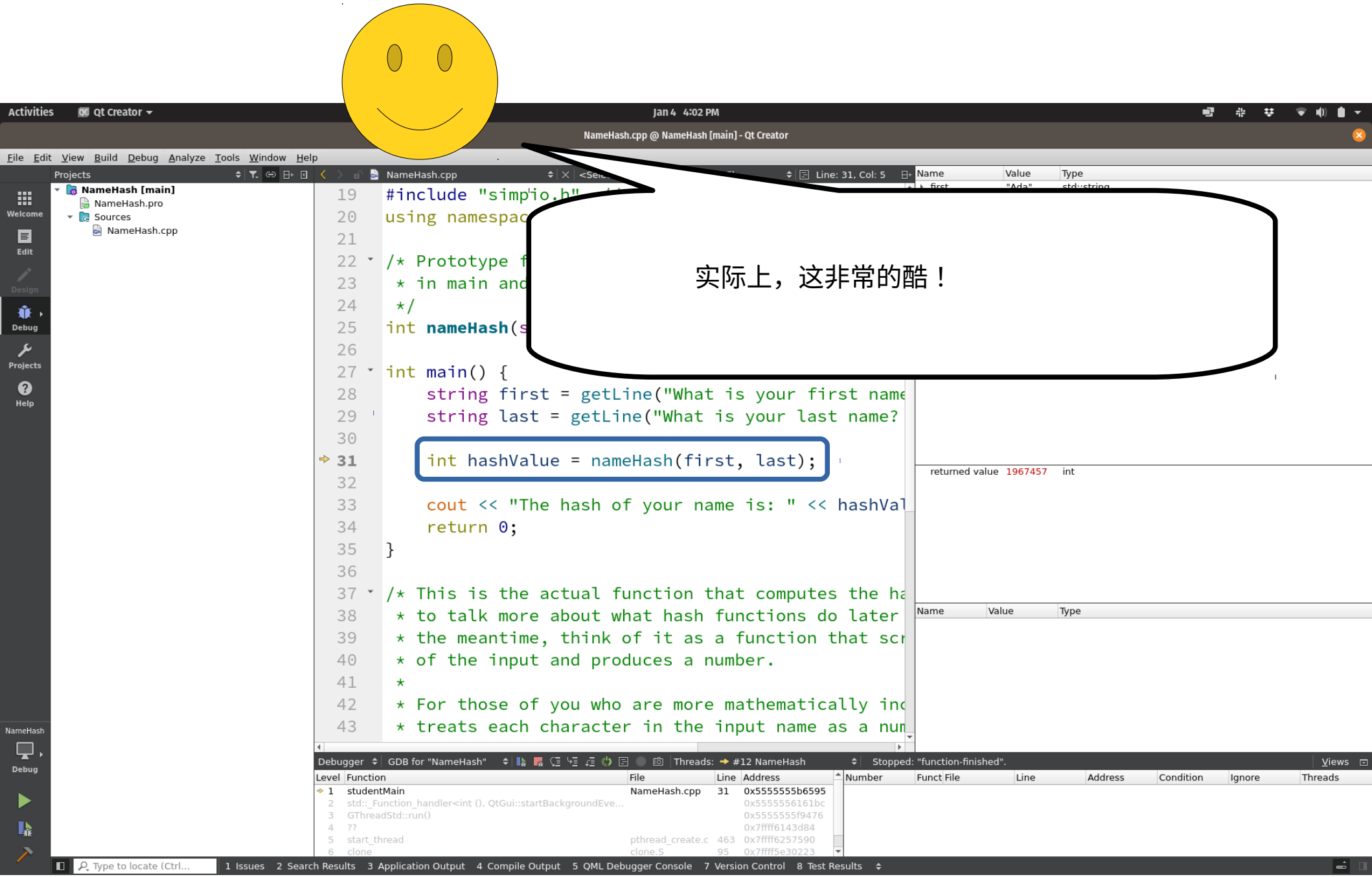
Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results

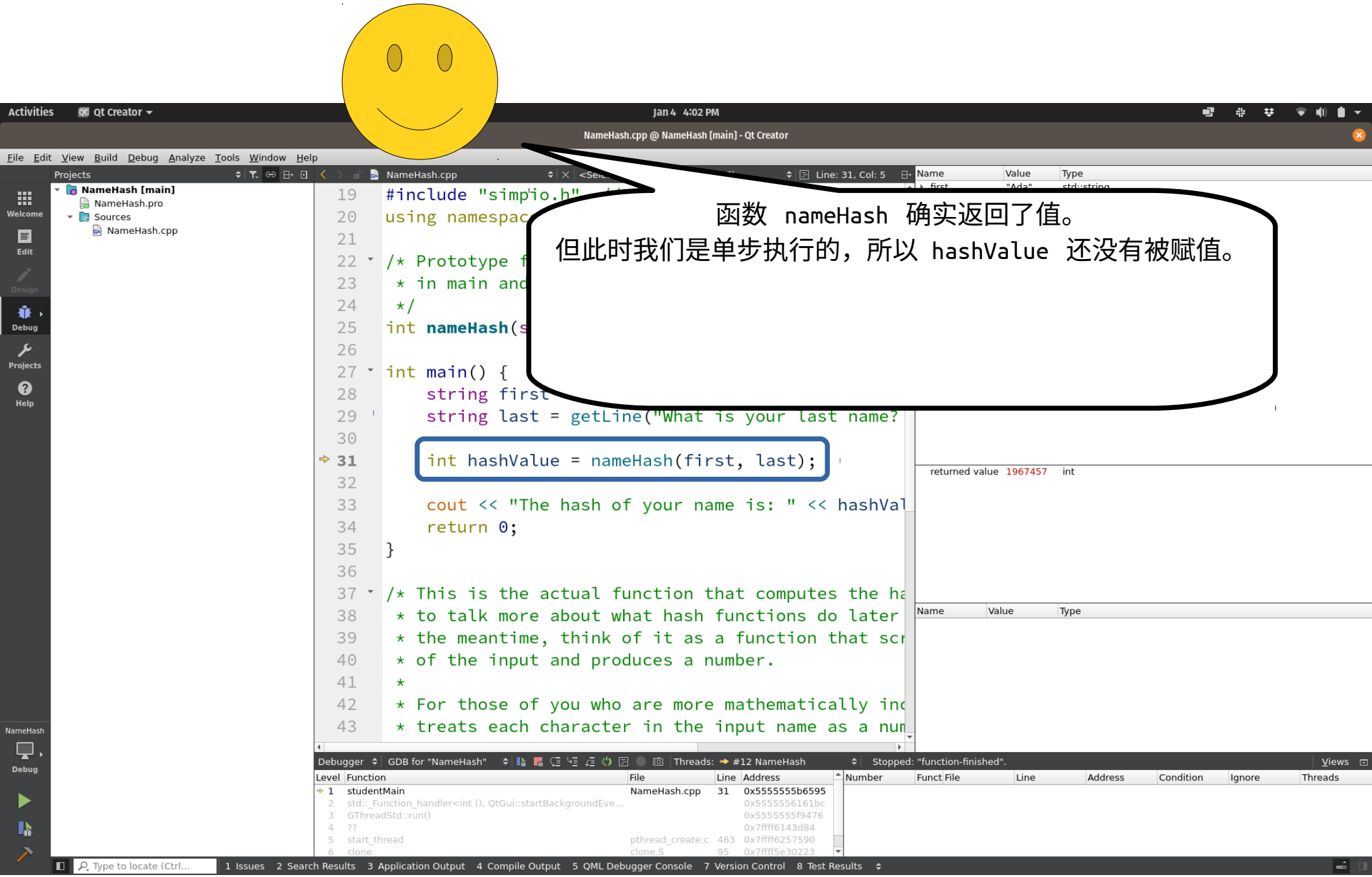
在这个框内我们可以看到 nameHash 函数的返回值 1967457。感谢调试器！

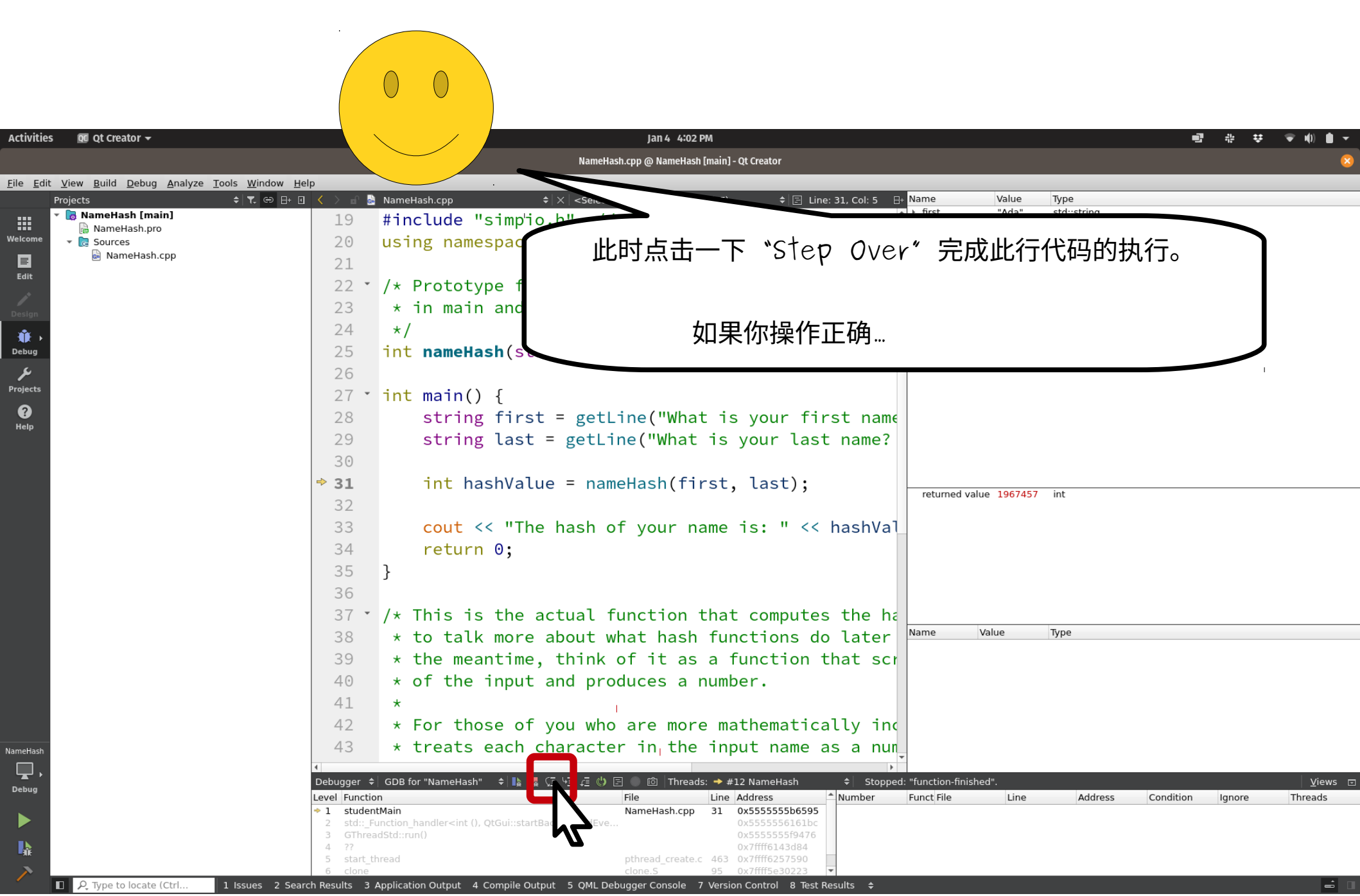
(A note: it seems like on some Macs, this number doesn't display. Don't worry if you don't see it - just continue on as usual.)


Activities
Qt Creator
Jan 4 4:02 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
NameHash.cpp
<Select Symbol>
Unix (LF)
Line: 31, Col: 5
19 #include "simpio.h" // for getLine
20 using namespace std;
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then define it later in the program.
24 */
25 int nameHash(string first, string last);
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is the actual function that computes the ha
* to talk more about what hash functions do later
the meantime, think of it as a function that scr
input and produces a number.
Name Value Type
hashValue 0 int
returned value 1967457 int
Name Value Type
Views
Line Address Condition Ignore Threads
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
Type to locate (Ctrl...
但是上面的 hashValue 却没有存储 1967457，虽然已经返回了
(You might see a number other than 0 on your system - that's okay.)


但是看起来，我们想让 hashValue
的值等于 nameHash 的返回值。
发生了什么呢？


Activities
Qt Creator
Jan 4 4:02 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
NameHash.cpp
Line: 31, Col: 5
19 #include "simpio.h"
20 using namespac
21
22 /* Prototype f
23 * in main and
24 */
25 int nameHash(s
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is the actual function that computes the ha
38 * to talk more about what hash functions do later
39 * the meantime, think of it as a function that scr
40 * of the input and produces a number.
41 *
42 * For those of you who are more mathematically inc
43 * treats each character in the input name as a num
实际上，这非常的酷！
Name Value Type
returned value 1967457 int
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "function-finished". Views
Level Function File Line Address
1 studentMain NameHash.cpp 31 0x5555555b6595
2 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
3 GThreadStd::run() 0x5555555f9476
4 ?? 0x7ffff6143d84
5 start_thread pthread_create.c 463 0x7ffff6257590
6 clone clone.S 95 0x7ffff5e30223
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


Activities
Qt Creator
Jan 4 4:02 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
Line: 31, Col: 5
19 #include "simpio.h"
20 using namespac
21
22 /* Prototype f
23 * in main and
24 */
25 int nameHash(s
26
27 int main() {
28 string first
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is the actual function that computes the ha
38 * to talk more about what hash functions do later
39 * the meantime, think of it as a function that scr
40 * of the input and produces a number.
41 *
42 * For those of you who are more mathematically inc
43 * treats each character in the input name as a num
Name Value Type
first "Ada" std::string
returned value 1967457 int
Name Value Type
函数 nameHash 确实返回了值。
但此时我们是单步执行的，所以 hashValue 还没有被赋值。
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "function-finished". Views
Level Function File Line Address
1 studentMain NameHash.cpp 31 0x5555555b6595
2 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
3 GThreadStd::run() 0x5555555f9476
4 ?? 0x7ffff6143d84
5 start_thread pthread_create.c 463 0x7ffff6257590
6 clone clone.S 95 0x7ffff5e30223
Number Funct File Line Address Condition Ignore Threads
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


此时点击一下 "Step Over" 完成此行代码的执行。
如果你操作正确…


Activities
Qt Creator
Jan 4 4:07 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash
Debug
<Select Symbol>
Unix (LF)
Line: 33, Col: 5
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then de in the program.
24 */
25 int nameHash(s
26
27 int main() {
28 string fir
29 string las
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is the actual function that computes the ha
38 * to talk more about what hash functions do later
39 * the meantime, think of it as a function that scr
40 * of the input and produces a number.
41 *
42 * For those of you who are more mathematically inc
43 * treats each character in the input name as a num
44 * It then uses them as coefficients in a polynomia
45 * F_p, where p is a large prime number, and evalua
Name Value Type
first "Ada" std::string
hashValue 1967457 int
last "Lovelace" std::string
Name Value Type
… 你将会看到正确的值(注意红色字体)。
此时程序也跳到了下一行。
Debugger GDB for "NameHash" Threads: #12 NameHash Stopped: "end-stepping-range". Views
Level Function File Line Address
1 studentMain NameHash.cpp 33 0x5555555b65b3
2 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
3 GThreadStd::run() 0x5555555f9476
4 ?? 0x7ffff6143d84
5 start_thread pthread_create.c 463 0x7ffff6257590
6 clone clone.S 95 0x7ffff5e30223
Number Funct File Line Address Condition Ignore Threads
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


至此，我们已经学到了很多。
接下来我们想让程序持续运行，而不是单步执行。


Activities
Qt Creator
Jan 4 4:07 PM
NameHash.cpp @ NameHash [main] - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects
NameHash [main]
NameHash.pro
Sources
NameHash.cpp
Welcome
Edit
Design
Debug
Projects
Help
NameHash.cpp
<Select Symbol>
Unix (LF)
Line: 33, Col: 5
21
22 /* Prototype for the nameHash function. This lets u
23 * in main and then define it later in the program.
24 */
25 int nameHash(string first, string last);
26
27 int main() {
28 string first = getLine("What is your first name
29 string last = getLine("What is your last name?
30
31 int hashValue = nameHash(first, last);
32
33 cout << "The hash of your name is: " << hashVal
34 return 0;
35 }
36
37 /* This is the actual function that computes the h
38 * to talk more about what hash functions do later
39 * the meantime, think of it as a funct
40 * of the input and
Name Value Type
first "Ada" std::string
hashValue 1967457 int
last "Lovelace" std::string
Type
Stopped: "end-stepping-range".
Views
Funct File Line Address Condition Ignore Threads
NameHash
Debug
Type to locate (Ctrl...
1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results
看一下这个按钮，鼠标放上去会显示 "Continue"
这个按钮的意思是，"不要暂停程序，保持程序运行"

Activities NameHash Jan 4 4:08 PM

NameHash.cpp @ NameHash [main] - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects

NameHash [main]
NameHash.pro
Sources
NameHash.cpp

NameHash.cpp <Select Symbol> Unix (LF) Line: 33, Col: 5

```
/* Prototype for the nameHash function. This lets u
 * in main and then define it later in the program.
 */
```

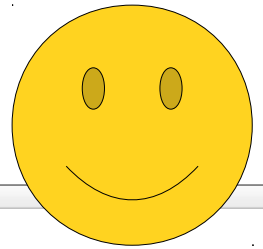
NameHash Console [Completed]

File Edit Options Help

```
What is your first name? Ada
What is your last name?  Lovelace
The hash of your name is: 1967457
```

Name Value Type

Type



点击一下，你会得到这样的结果。

（如果这个窗口没有自动弹出。请手动打开。）

此时，程序也完成了执行。

Debug

Level Address Condition Ignore Threads

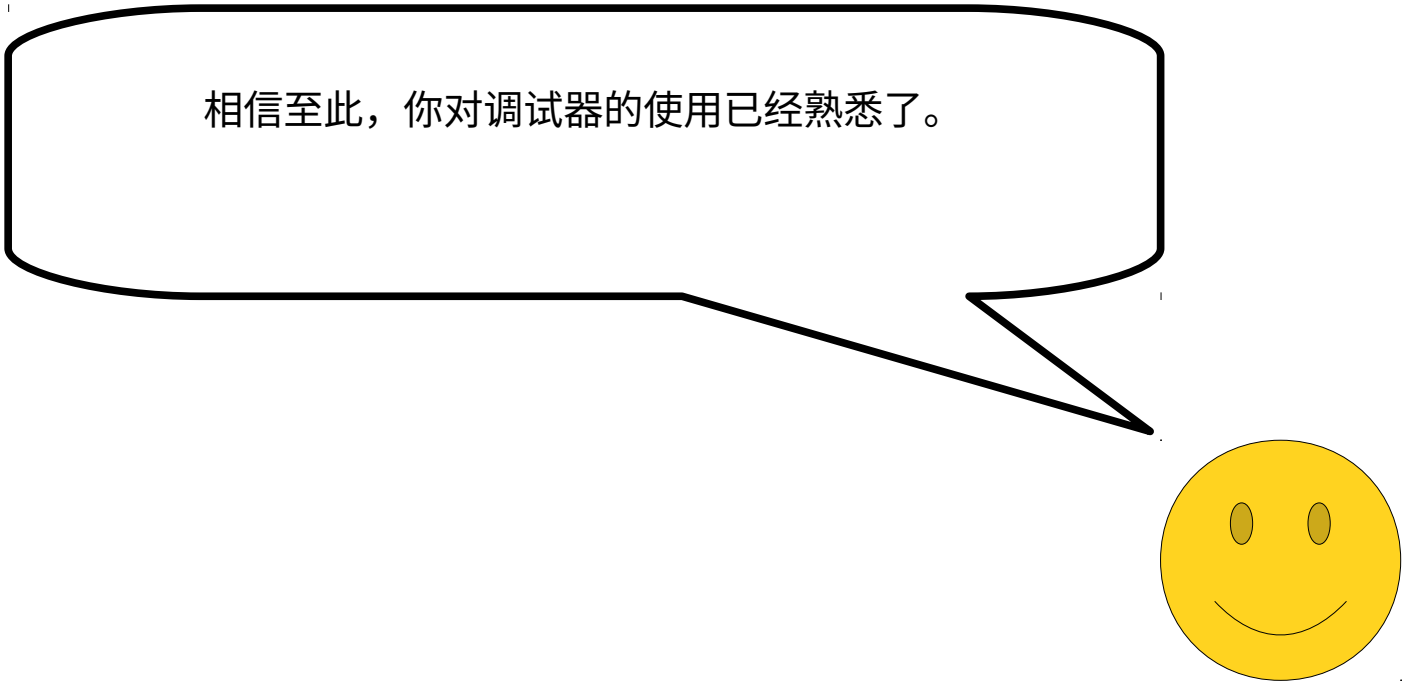
1 student
2 std::_Function_handler<int (), QtGui::startBackgroundEve... 0x5555556161bc
3 GThreadStd::run() 0x5555555f9476
4 ?? 0x7ffff6143d84
5 start_thread pthread_create.c 463 0x7ffff6257590
6 clone clone.S 95 0x7ffff5e30223

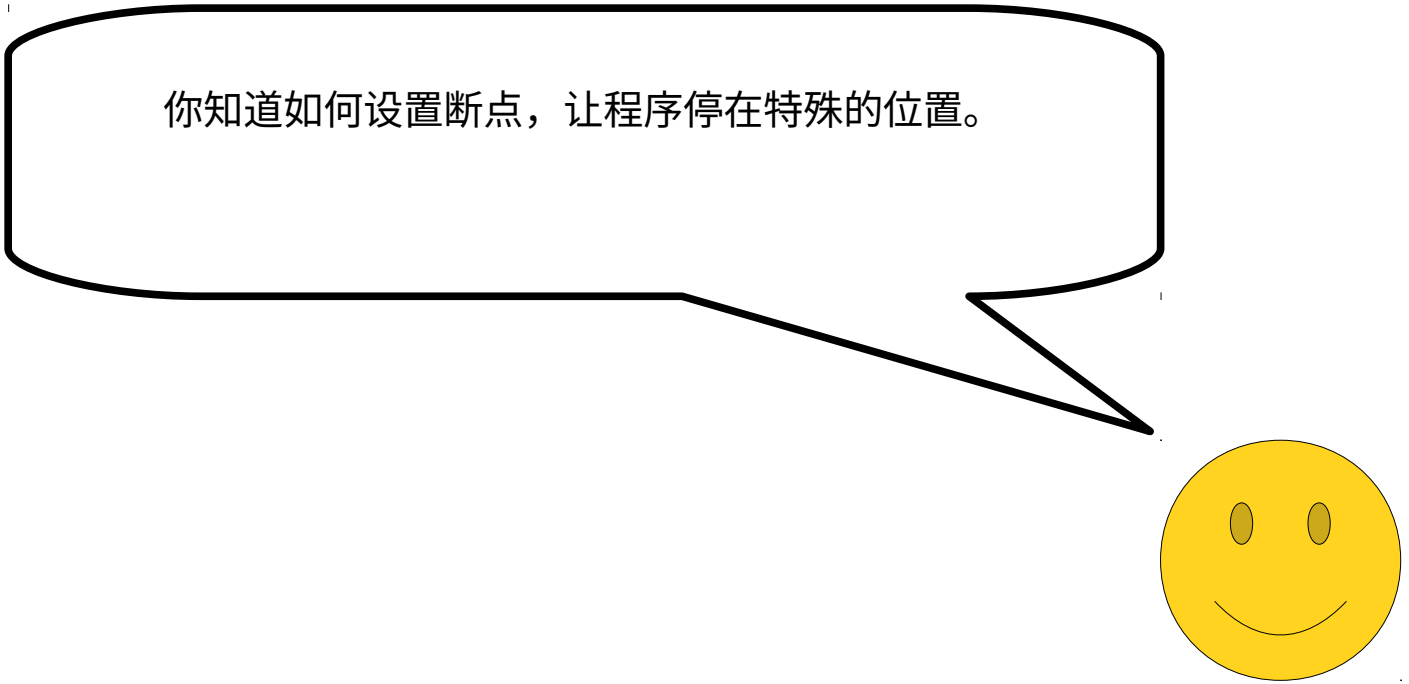
Views

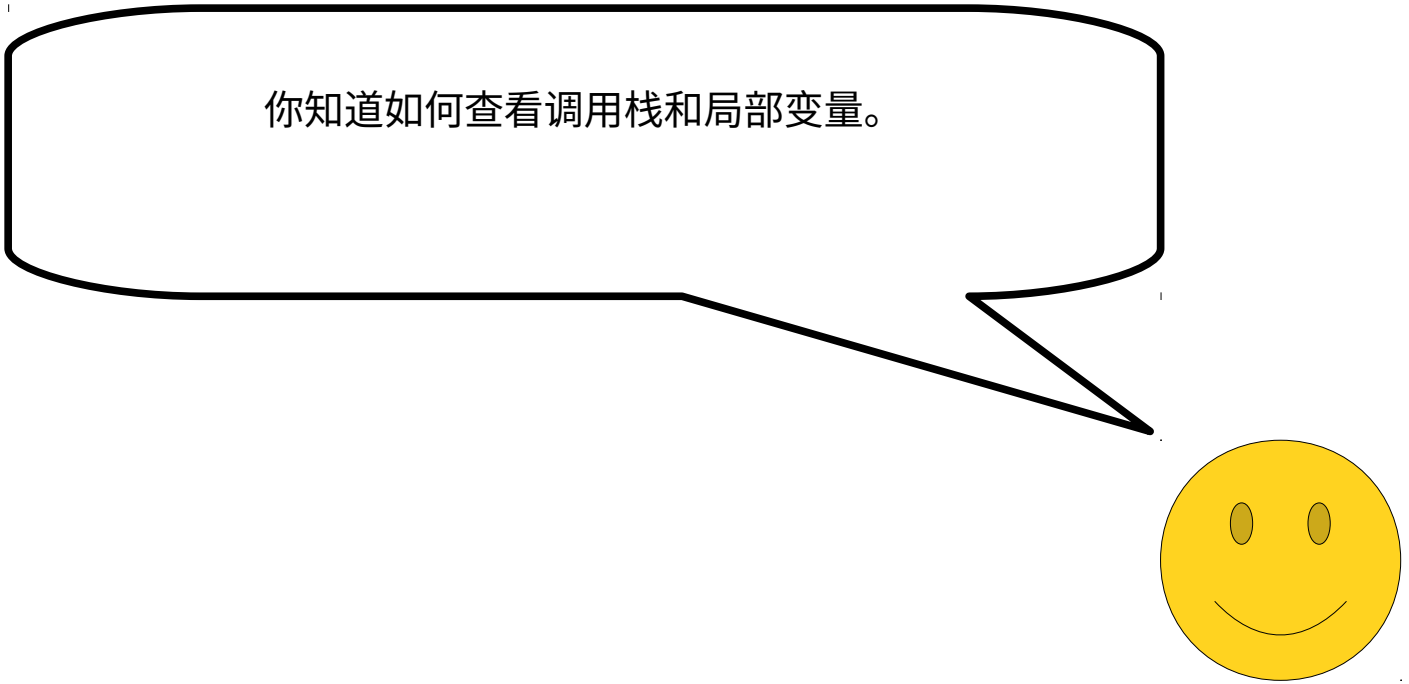
Welcome Edit Design Debug Projects Help

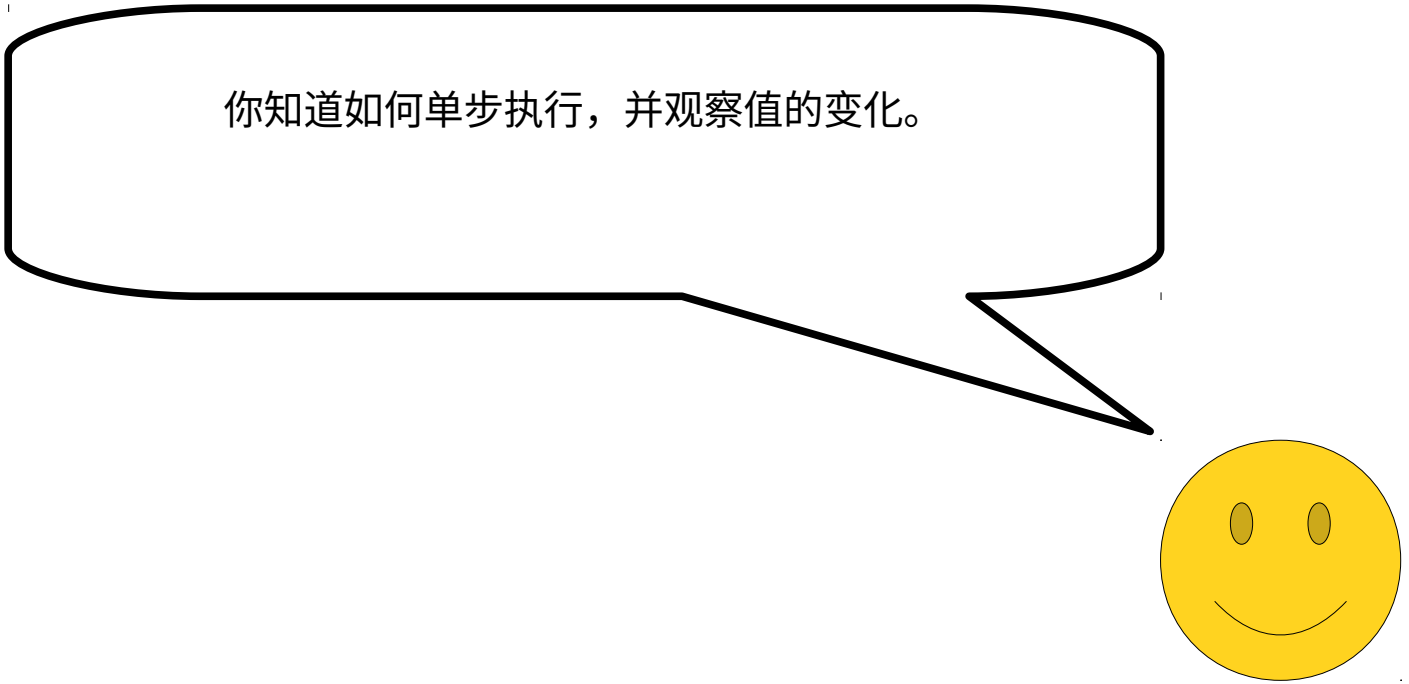
NameHash Debug

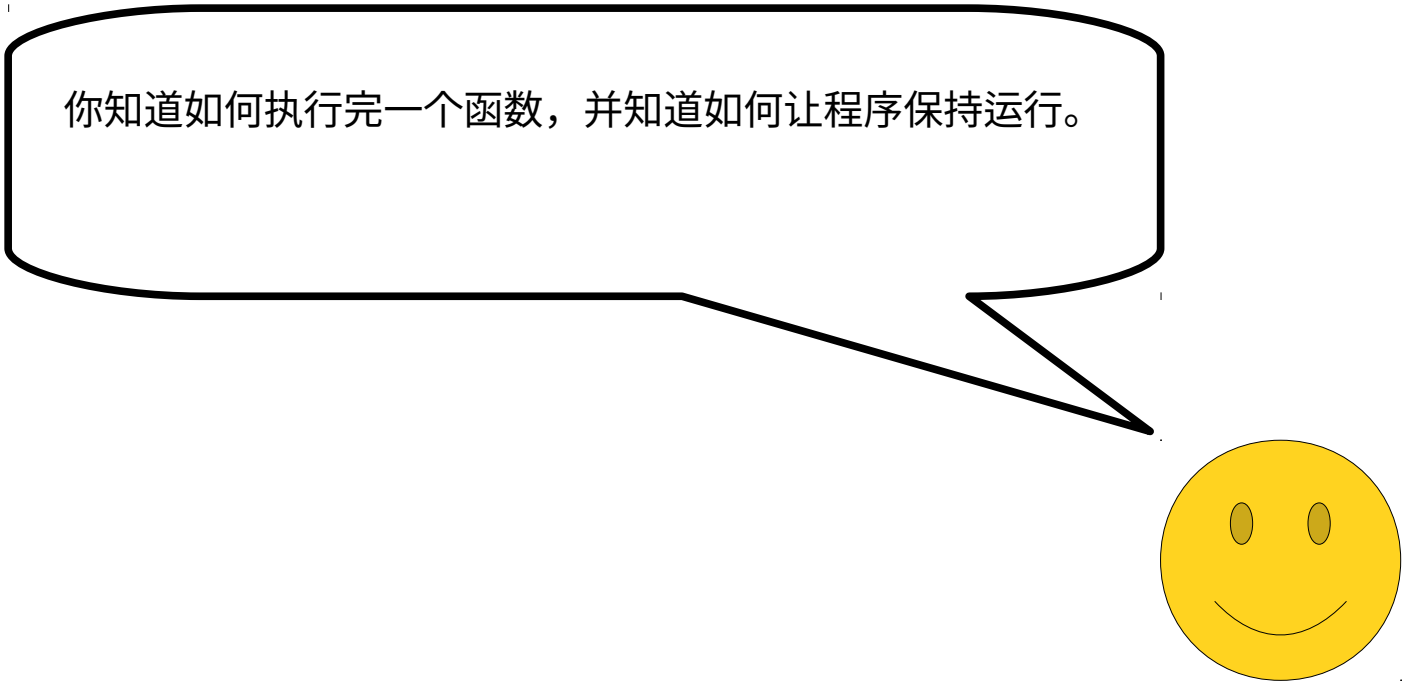
Type to locate (Ctrl... 1 Issues 2 Search Results 3 Application Output 4 Compile Output 5 QML Debugger Console 7 Version Control 8 Test Results


相信至此，你对调试器的使用已经熟悉了。

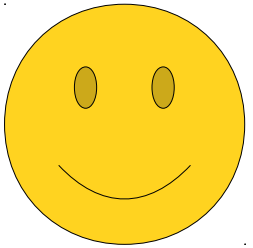

你知道如何设置断点，让程序停在特殊的位置。


你知道如何查看调用栈和局部变量。

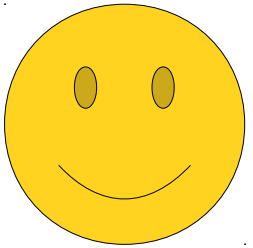

你知道如何单步执行，并观察值的变化。


你知道如何执行完一个函数，并知道如何让程序保持运行。

随着课程的深入，你会对调试器越来越熟悉，对程序的运行方式也会越来越了解。



另外，如果你继续学习软件开发，
你会发现，学会使用调试器是个非常有价值的技能。



希望以上信息，可以给你带来帮助！
欢迎参加本期的课程！