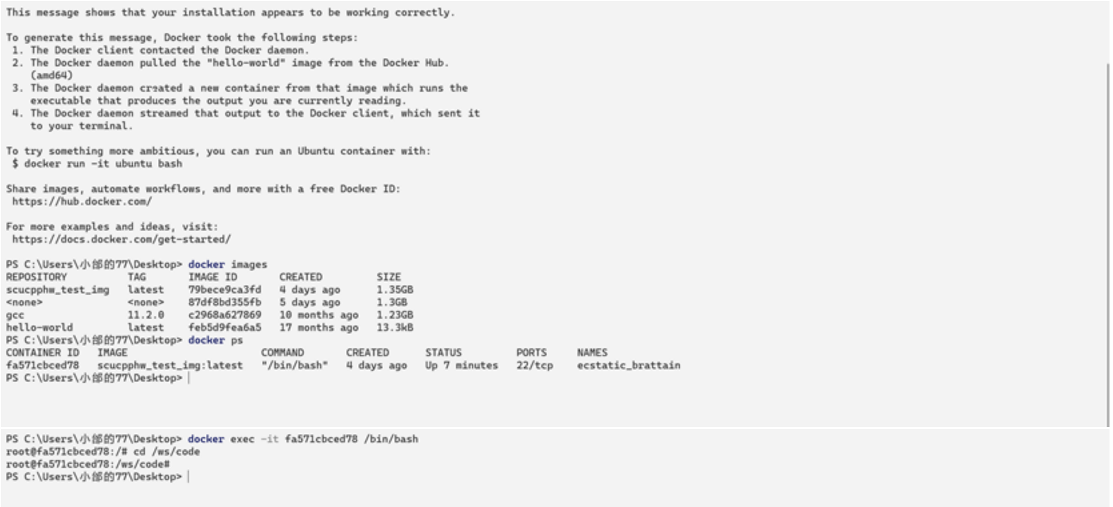
Student Name: GeTianqi
Student ID: 2022141460202

C++ Object-Oriented Programming

Assignment 1

1 Configure docker environment

Here are the results of the image and ps and cd operations

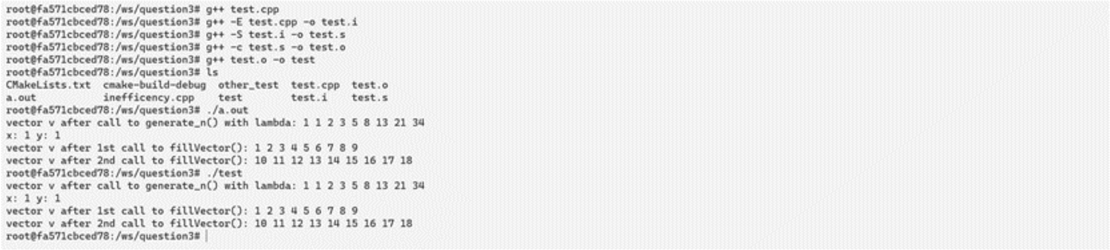


```
This message shows that your installation appears to be working correctly.

To generate this message, Docker took the following steps:
 1. The Docker client contacted the Docker daemon.
 2. The Docker daemon pulled the "hello-world" image from the Docker Hub.
    (amd64)
 3. The Docker daemon created a new container from that image which runs the
    executable that produces the output you are currently reading.
 4. The Docker daemon streamed that output to the Docker client, which sent it
    to your terminal.

To try something more ambitious, you can run an Ubuntu container with:
 $ docker run -it ubuntu bash

Share images, automate workflows, and more with a free Docker ID:
 https://hub.docker.com/

For more examples and ideas, visit:
 https://docs.docker.com/get-started/

PS C:\Users\小郜的77\Desktop> docker images
REPOSITORY          TAG       IMAGE ID       CREATED          SIZE
scucpphw_test_img   latest    79bece9ca3fd   4 days ago       1.35GB
<none>              <none>    87df8bd355fb   5 days ago       1.3GB
gcc                 11.2.0    c2968a627869   10 months ago    1.23GB
hello-world         latest    feb5d9fea6a5   17 months ago    13.3kB
PS C:\Users\小郜的77\Desktop> docker ps
CONTAINER ID   IMAGE                      COMMAND       CREATED      STATUS         PORTS     NAMES
fa571cbced78   scucpphw_test_img:latest   "/bin/bash"   4 days ago   Up 7 minutes   22/tcp    ecstatic_brattain
PS C:\Users\小郜的77\Desktop> |
```

```
PS C:\Users\小郜的77\Desktop> docker exec -it fa571cbced78 /bin/bash
root@fa571cbced78:/# cd /ws/code
root@fa571cbced78:/ws/code#
PS C:\Users\小郜的77\Desktop> |
```

2 Compiling single files with g++

2.1 Single step compilation and divisional compilation



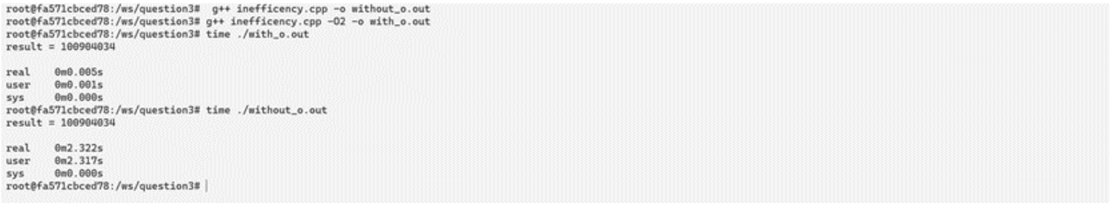
```
root@fa571cbced78:/ws/question3# g++ test.cpp
root@fa571cbced78:/ws/question3# g++ -E test.cpp -o test.i
root@fa571cbced78:/ws/question3# g++ -S test.i -o test.s
root@fa571cbced78:/ws/question3# g++ -c test.s -o test.o
root@fa571cbced78:/ws/question3# g++ test.o -o test
root@fa571cbced78:/ws/question3# ls
CMakeLists.txt  cmake-build-debug  other_test  test.cpp  test.o
a.out           inefficency.cpp    test        test.i    test.s
root@fa571cbced78:/ws/question3# ./a.out
vector v after call to generate_n() with lambda: 1 1 2 3 5 8 13 21 34
x: 1 y: 1
vector v after 1st call to fillVector(): 1 2 3 4 5 6 7 8 9
vector v after 2nd call to fillVector(): 10 11 12 13 14 15 16 17 18
root@fa571cbced78:/ws/question3# ./test
vector v after call to generate_n() with lambda: 1 1 2 3 5 8 13 21 34
x: 1 y: 1
vector v after 1st call to fillVector(): 1 2 3 4 5 6 7 8 9
vector v after 2nd call to fillVector(): 10 11 12 13 14 15 16 17 18
root@fa571cbced78:/ws/question3# |
```

During the compilation process, we generated the following files

- Single step compilation : a.out
- Divisional compilation : test.cpp test.i test.s test.o test.out

Although the compilation process is not all the same, the final executable file will run with the same result

2.2 Optimized vs. non-optimized compilation



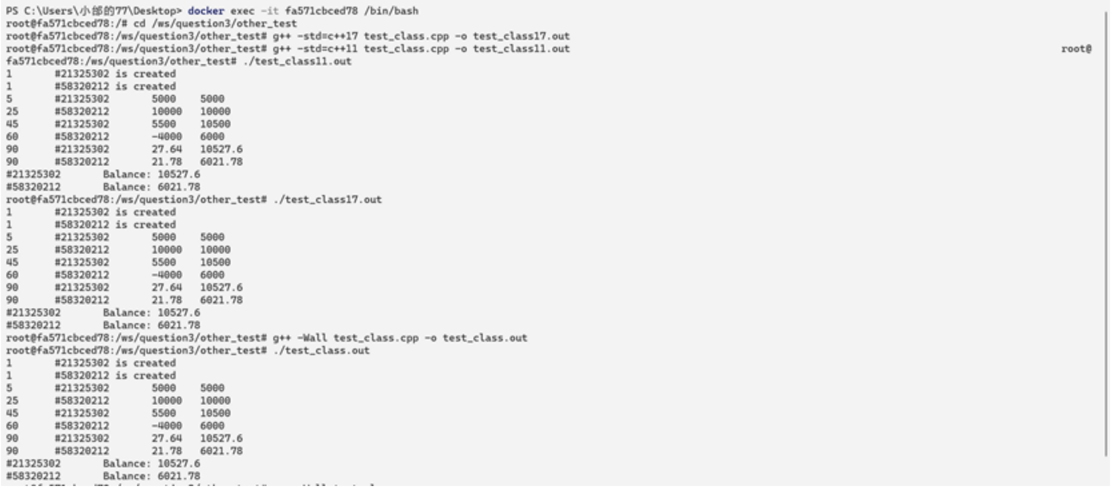
```
root@fa571cbced78:/ws/question3#  g++ inefficency.cpp -o without_o.out
root@fa571cbced78:/ws/question3# g++ inefficency.cpp -O2 -o with_o.out
root@fa571cbced78:/ws/question3# time ./with_o.out
result = 100904034

real    0m0.005s
user    0m0.001s
sys     0m0.000s
root@fa571cbced78:/ws/question3# time ./without_o.out
result = 100904034

real    0m2.322s
user    0m2.317s
sys     0m0.000s
root@fa571cbced78:/ws/question3# |
```

Obviously, you can see that for this code, it runs much faster after optimization

2.3 Compile with different g++ parameters

1. testclass.cpp

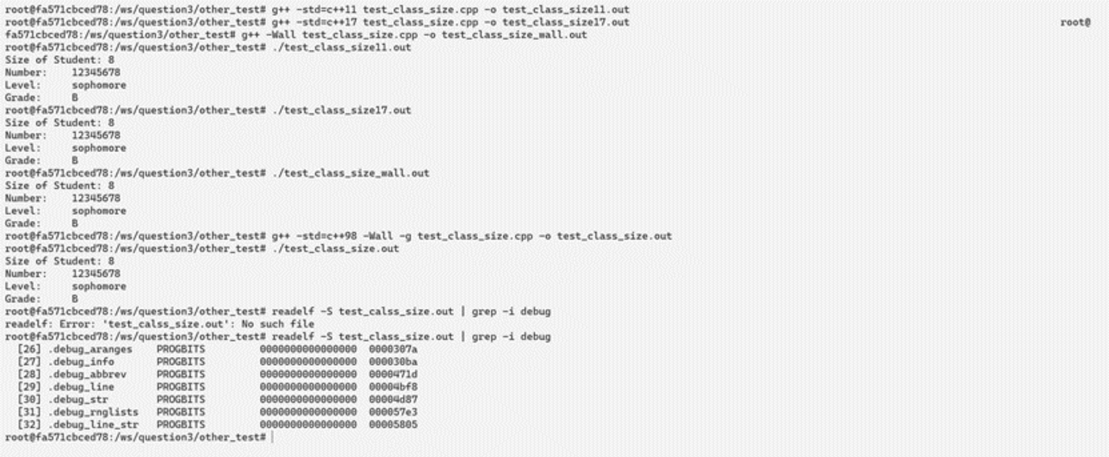
   I compiled the source file and ran it using the c++11 and c++17 standards respectively, and there was no difference between the two; then I tried to print the warning message provided by g++, but there was no warning in this code



```
PS C:\Users\小郜的77\Desktop> docker exec -it fa571cbced78 /bin/bash
root@fa571cbced78:/# cd /ws/question3/other_test
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++17 test_class.cpp -o test_class17.out
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++11 test_class.cpp -o test_class11.out                                                          root@
fa571cbced78:/ws/question3/other_test# ./test_class11.out
1       #21325302 is created
1       #58320212 is created
5       #21325302       5000    5000
25      #58320212       10000   10000
45      #21325302       5500    10500
60      #58320212       -4000   6000
90      #21325302       27.64   10527.6
90      #58320212       21.78   6021.78
#21325302       Balance: 10527.6
#58320212       Balance: 6021.78
root@fa571cbced78:/ws/question3/other_test# ./test_class17.out
1       #21325302 is created
1       #58320212 is created
5       #21325302       5000    5000
25      #58320212       10000   10000
45      #21325302       5500    10500
60      #58320212       -4000   6000
90      #21325302       27.64   10527.6
90      #58320212       21.78   6021.78
#21325302       Balance: 10527.6
#58320212       Balance: 6021.78
root@fa571cbced78:/ws/question3/other_test# g++ -Wall test_class.cpp -o test_class.out
root@fa571cbced78:/ws/question3/other_test# ./test_class.out
1       #21325302 is created
1       #58320212 is created
5       #21325302       5000    5000
25      #58320212       10000   10000
45      #21325302       5500    10500
60      #58320212       -4000   6000
90      #21325302       27.64   10527.6
90      #58320212       21.78   6021.78
#21325302       Balance: 10527.6
#58320212       Balance: 6021.78
```

2. testclasssize.cpp

   - compiled the source file and ran it using the c++11 and c++17 standards respectively
   - print the warning message provided by g++, but there was no warning in this code
   - use the readelf -S command to see the generated debugging information

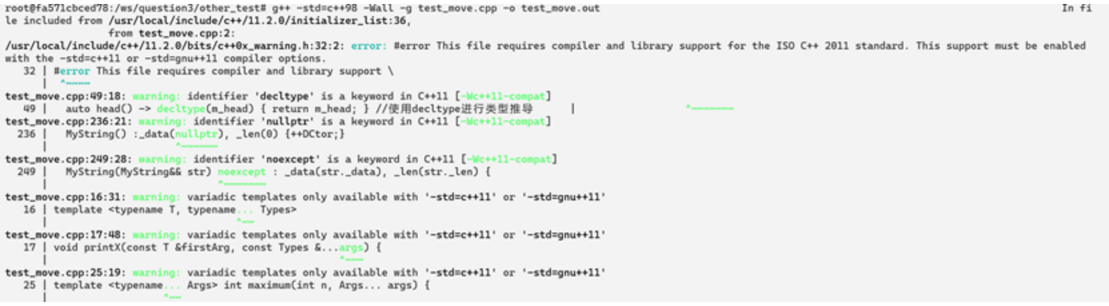


```
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++11 test_class_size.cpp -o test_class_size11.out
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++17 test_class_size.cpp -o test_class_size17.out                          root@
fa571cbced78:/ws/question3/other_test# g++ -Wall test_class_size.cpp -o test_class_size_wall.out
root@fa571cbced78:/ws/question3/other_test# ./test_class_size11.out
Size of Student: 8
Number:    12345678
Level:     sophomore
Grade:     B
root@fa571cbced78:/ws/question3/other_test# ./test_class_size17.out
Size of Student: 8
Number:    12345678
Level:     sophomore
Grade:     B
root@fa571cbced78:/ws/question3/other_test# ./test_class_size_wall.out
Size of Student: 8
Number:    12345678
Level:     sophomore
Grade:     B
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++98 -Wall -g test_class_size.cpp -o test_class_size.out
root@fa571cbced78:/ws/question3/other_test# ./test_class_size.out
Size of Student: 8
Number:    12345678
Level:     sophomore
Grade:     B
root@fa571cbced78:/ws/question3/other_test# readelf -S test_calss_size.out | grep -i debug
readelf: Error: 'test_calss_size.out': No such file
root@fa571cbced78:/ws/question3/other_test# readelf -S test_class_size.out | grep -i debug
  [26] .debug_aranges    PROGBITS         0000000000000000  0000307a
  [27] .debug_info       PROGBITS         0000000000000000  000030ba
  [28] .debug_abbrev     PROGBITS         0000000000000000  0000471d
  [29] .debug_line       PROGBITS         0000000000000000  00004bf8
  [30] .debug_str        PROGBITS         0000000000000000  00004d87
  [31] .debug_rnglists   PROGBITS         0000000000000000  000057e3
  [32] .debug_line_str   PROGBITS         0000000000000000  00005805
root@fa571cbced78:/ws/question3/other_test# |
```

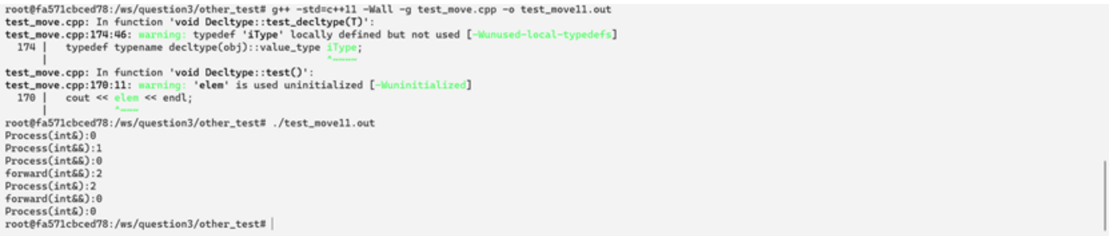
3. testmove.cpp

   I tried to compile the source files separately under the c++98 and c++11 standards and printed out the warnings provided by g++

   - Under the c++98 standard, the file fails to compile and prints dozens of lines of warnings and errors
   - Under the c++11 standard, it compiles and runs normally, and prints the warn provided by g++



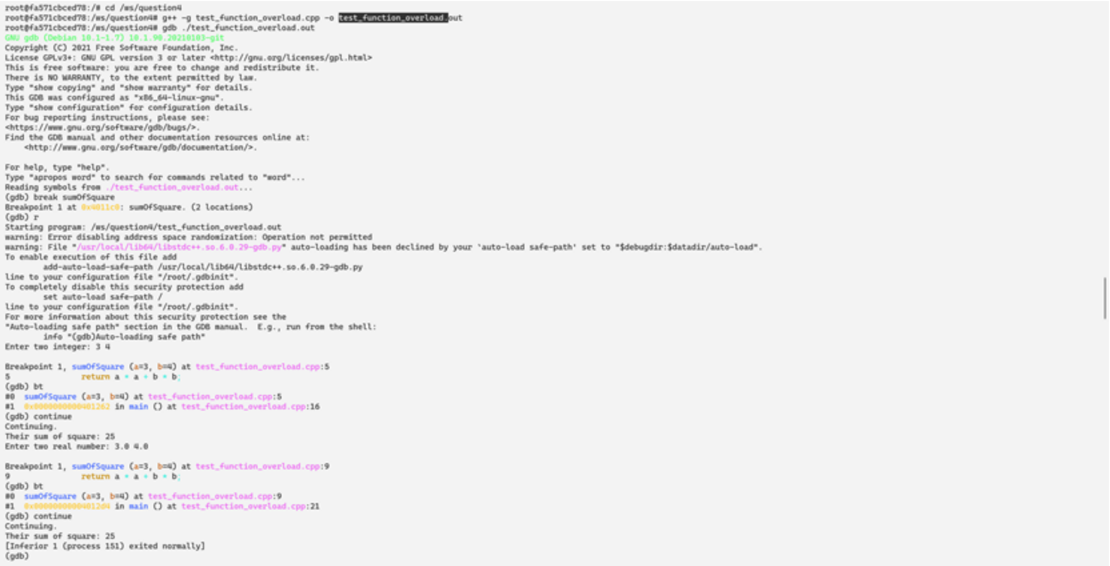
```
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++98 -Wall -g test_move.cpp -o test_move.out                              In fi
le included from /usr/local/include/c++/11.2.0/initializer_list:36,
                 from test_move.cpp:2:
/usr/local/include/c++/11.2.0/bits/c++0x_warning.h:32:2: error: #error This file requires compiler and library support for the ISO C++ 2011 standard. This support must be enabled
with the -std=c++11 or -std=gnu++11 compiler options.
   32 | #error This file requires compiler and library support \
      |  ^~~~~
test_move.cpp:49:18: warning: identifier 'decltype' is a keyword in C++11 [-Wc++11-compat]
   49 |   auto head() -> decltype(m_head) { return m_head; } //使用decltype进行类型推导      |                  ^~~~~~~~
test_move.cpp:236:21: warning: identifier 'nullptr' is a keyword in C++11 [-Wc++11-compat]
  236 |   MyString() :_data(nullptr), _len(0) {++DCtor;}
      |                     ^~~~~~~
test_move.cpp:249:28: warning: identifier 'noexcept' is a keyword in C++11 [-Wc++11-compat]
  249 |   MyString(MyString&& str) noexcept : _data(str._data), _len(str._len) {
      |                            ^~~~~~~~
test_move.cpp:16:31: warning: variadic templates only available with '-std=c++11' or '-std=gnu++11'
   16 | template <typename T, typename... Types>
      |                               ^~~
test_move.cpp:17:48: warning: variadic templates only available with '-std=c++11' or '-std=gnu++11'
   17 | void printX(const T &firstArg, const Types &...args) {
      |                                                ^~~~
test_move.cpp:25:19: warning: variadic templates only available with '-std=c++11' or '-std=gnu++11'
   25 | template <typename... Args> int maximum(int n, Args... args) {
      |                   ^~~
```



```
root@fa571cbced78:/ws/question3/other_test# g++ -std=c++11 -Wall -g test_move.cpp -o test_move11.out
test_move.cpp: In function 'void Decltype::test_decltype(T)':
test_move.cpp:174:46: warning: typedef 'iType' locally defined but not used [-Wunused-local-typedefs]
  174 |   typedef typename decltype(obj)::value_type iType;
      |                                              ^~~~~
test_move.cpp: In function 'void Decltype::test()':
test_move.cpp:170:11: warning: 'elem' is used uninitialized [-Wuninitialized]
  170 |   cout << elem << endl;
      |           ^~~~
root@fa571cbced78:/ws/question3/other_test# ./test_move11.out
Process(int&):0
Process(int&&):1
Process(int&&):0
forward(int&&):2
Process(int&):2
forward(int&&):0
Process(int&):0
root@fa571cbced78:/ws/question3/other_test# |
```

3 GDB debug file

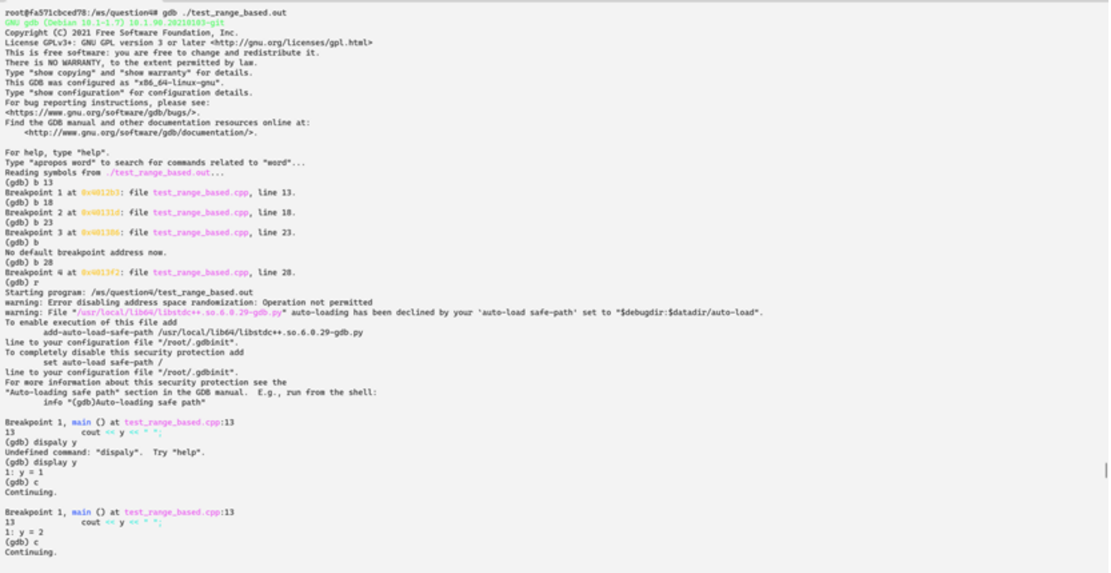
1. Tracing stack frames and hierarchical relationships(test function overload.cpp)



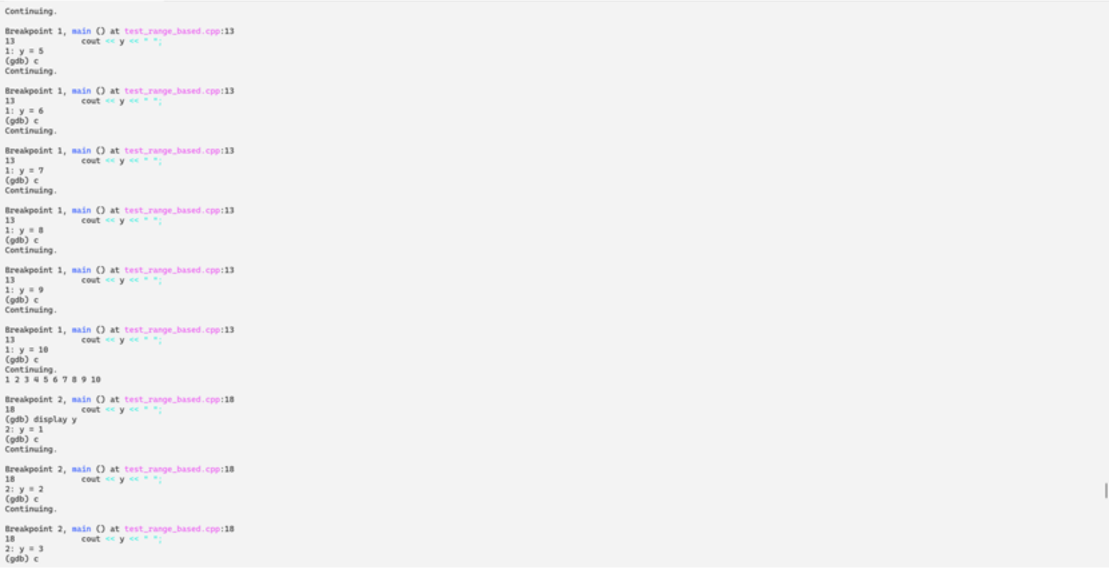
```
root@fa571cbced78:/# cd /ws/question4
root@fa571cbced78:/ws/question4# g++ -g test_function_overload.cpp -o test_function_overload.out
root@fa571cbced78:/ws/question4# gdb ./test_function_overload.out
GNU gdb (Debian 10.1-1.7) 10.1.90.20210103-git
Copyright (C) 2021 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./test_function_overload.out...
(gdb) break sumOfSquare
Breakpoint 1 at 0x4011c0: sumOfSquare. (2 locations)
(gdb) r
Starting program: /ws/question4/test_function_overload.out
warning: Error disabling address space randomization: Operation not permitted
warning: File "/usr/local/lib64/libstdc++.so.6.0.29-gdb.py" auto-loading has been declined by your `auto-load safe-path' set to "$debugdir:$datadir/auto-load".
To enable execution of this file add
        add-auto-load-safe-path /usr/local/lib64/libstdc++.so.6.0.29-gdb.py
line to your configuration file "/root/.gdbinit".
To completely disable this security protection add
        set auto-load safe-path /
line to your configuration file "/root/.gdbinit".
For more information about this security protection see the
"Auto-loading safe path" section in the GDB manual.  E.g., run from the shell:
        info "(gdb)Auto-loading safe path"
Enter two integer: 3 4

Breakpoint 1, sumOfSquare (a=3, b=4) at test_function_overload.cpp:5
5               return a * a + b * b;
(gdb) bt
#0  sumOfSquare (a=3, b=4) at test_function_overload.cpp:5
#1  0x0000000000401262 in main () at test_function_overload.cpp:16
(gdb) continue
Continuing.
Their sum of square: 25
Enter two real number: 3.0 4.0

Breakpoint 1, sumOfSquare (a=3, b=4) at test_function_overload.cpp:9
9               return a * a + b * b;
(gdb) bt
#0  sumOfSquare (a=3, b=4) at test_function_overload.cpp:9
#1  0x00000000004012d4 in main () at test_function_overload.cpp:21
(gdb) continue
Continuing.
Their sum of square: 25
[Inferior 1 (process 151) exited normally]
(gdb)
```

2. Trace the value of y in the loop and point out the meaning of j(test range based.cpp)

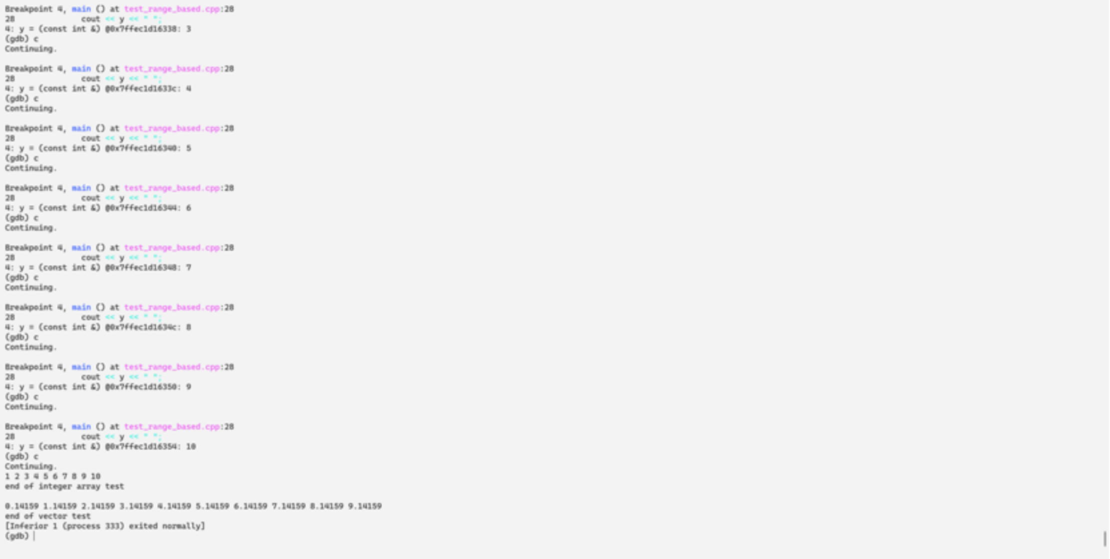
The content is too long, only part of it is intercepted



```
root@fa571cbced78:/ws/question4# gdb ./test_range_based.out
GNU gdb (Debian 10.1-1.7) 10.1.90.20210103-git
Copyright (C) 2021 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.
Type "show copying" and "show warranty" for details.
This GDB was configured as "x86_64-linux-gnu".
Type "show configuration" for configuration details.
For bug reporting instructions, please see:
<https://www.gnu.org/software/gdb/bugs/>.
Find the GDB manual and other documentation resources online at:
    <http://www.gnu.org/software/gdb/documentation/>.

For help, type "help".
Type "apropos word" to search for commands related to "word"...
Reading symbols from ./test_range_based.out...
(gdb) b 13
Breakpoint 1 at 0x4012b3: file test_range_based.cpp, line 13.
(gdb) b 18
Breakpoint 2 at 0x40131d: file test_range_based.cpp, line 18.
(gdb) b 23
Breakpoint 3 at 0x401386: file test_range_based.cpp, line 23.
(gdb) b
No default breakpoint address now.
(gdb) b 28
Breakpoint 4 at 0x4013f2: file test_range_based.cpp, line 28.
(gdb) r
Starting program: /ws/question4/test_range_based.out
warning: Error disabling address space randomization: Operation not permitted
warning: File "/usr/local/lib64/libstdc++.so.6.0.29-gdb.py" auto-loading has been declined by your `auto-load safe-path' set to "$debugdir:$datadir/auto-load".
To enable execution of this file add
        add-auto-load-safe-path /usr/local/lib64/libstdc++.so.6.0.29-gdb.py
line to your configuration file "/root/.gdbinit".
To completely disable this security protection add
        set auto-load safe-path /
line to your configuration file "/root/.gdbinit".
For more information about this security protection see the
"Auto-loading safe path" section in the GDB manual.  E.g., run from the shell:
        info "(gdb)Auto-loading safe path"

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
(gdb) dispaly y
Undefined command: "dispaly".  Try "help".
(gdb) display y
1: y = 1
(gdb) c
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 2
(gdb) c
Continuing.
```



```
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 5
(gdb) c
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 6
(gdb) c
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 7
(gdb) c
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 8
(gdb) c
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 9
(gdb) c
Continuing.

Breakpoint 1, main () at test_range_based.cpp:13
13              cout << y << " ";
1: y = 10
(gdb) c
Continuing.
1 2 3 4 5 6 7 8 9 10

Breakpoint 2, main () at test_range_based.cpp:18
18              cout << y << " ";
(gdb) display y
2: y = 1
(gdb) c
Continuing.

Breakpoint 2, main () at test_range_based.cpp:18
18              cout << y << " ";
2: y = 2
(gdb) c
Continuing.

Breakpoint 2, main () at test_range_based.cpp:18
18              cout << y << " ";
2: y = 3
(gdb) c
```



```
Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d16338: 3
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d1633c: 4
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d16340: 5
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d16344: 6
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d16348: 7
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d1634c: 8
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d16350: 9
(gdb) c
Continuing.

Breakpoint 4, main () at test_range_based.cpp:28
28              cout << y << " ";
4: y = (const int &) @0x7ffec1d16354: 10
(gdb) c
Continuing.
1 2 3 4 5 6 7 8 9 10
end of integer array test

0.14159 1.14159 2.14159 3.14159 4.14159 5.14159 6.14159 7.14159 8.14159 9.14159
end of vector test
[Inferior 1 (process 333) exited normally]
(gdb) |
```

- j corresponds to each element of the vector array

3. test const.cpp

   - cout << &c.NUM << endl; should be replaced with cout << C::NUM << endl;

     Because static member variables belong to classes and not instances of classes

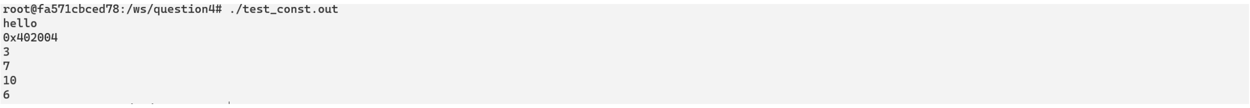
   - Delete cout << &C:NUM1 << endl;

     Because the enum type is a constant without an address

   - The output of MAX(++a, b) for the same input (a=5, b=0) is 7, while the output of Max(++a,b) is 6.

     If you want the result of MAX(++a, b) to be 6 as well, you can change MAX(++a, b) to MAX(a+1, b).

   - The only one with an address is const



```
root@fa571cbced78:/ws/question4# ./test_const.out
hello
0x402004
3
7
10
6
```

The address of the const int NUM is 0x402004