Lab 2 Q4_Debugging report
Xikai Xu
400440917

//this doc records how to fix the question 4's coding problem

1. Bug Observation 1:
   i) the first compile shows line 31 have written problem, which is lack of ';', add it to fix the c grammar. (used to line 43)

```
38          temp.charData = array[i].charData;
39          //store large
40          array[i].intData = array[i + 1].intData;   //give small to current
41          array[i].charData = array[i + 1].charData;
42
43          array[i + 1].intData = temp.intData   //put large to next
44          array[i + 1].charData = temp.charData;
```

   ii) Analysis: ';' is missing which means the GCC will not make the file

   iii) Bug Fix Validation:
   By adding the missing ';'GCC success start to make the file.

```
C:\Users\Xikai Xu\Coding\COE2SH4\classs week 6\lab2-intelpentium4extreme-xu503>make
gcc -c -o AllTests.o AllTests.c -I. -lm -g
gcc -c -o testCases.o testCases.c -I. -lm -g
gcc -c -o CuTest.o CuTest.c -I. -lm -g
gcc -c -o Question1.o Question1.c -I. -lm -g
gcc -c -o Question2.o Question2.c -I. -lm -g
gcc -c -o Question3.o Question3.c -I. -lm -g
gcc -c -o Question4.o Question4.c -I. -lm -g
gcc -o Lab2 AllTests.o testCases.o CuTest.o Question1.o Question2.o Question3.o Question4.o -I. -lm -g
```

2. Bug observation 2:
   i) By using GDB 3'd times, the logic of swap has problem. The line from 37 to 44 have order problem:

   //A bubble swap suppose to be like:
   if element current > element next{
   temp = next; || temp = current  // store small value or large
   next = current;  || current  = next  // put large to next or put little to current
   current = temp; || next = temp  //put small to current
   the order of swaption is messed up.

```
There were 2 failures:
1) TestQ4_BubbleSort_1: testCases.c:348: expected <0> but was <6>
2) TestQ4_BubbleSort_2: testCases.c:364: expected <6823936> but was <0>
```

   ii) Analysis: major problem is under the swapping loop, by adding a breakpoint at line 29 of the loop, and then run the loop to find the var problems.

```
(gdb) b Question4.c:29
Breakpoint 2 at 0x404fdc: file Question4.c, line 29.
(gdb)
```

```
Command name abbreviations are allowed if unambiguous.
(gdb) i local
temp = {intData = 78, charData = 32 ' '}
i = 7
curr = 78
next = -9
done = 1
(gdb) p temp.intData
$1 = 78
(gdb) p array
$2 = (struct Q4Struct *) 0x61fe08
(gdb) p *array
$3 = {intData = -23, charData = 61 '='}
(gdb)
```

```
(gdb) p *array
$4 = {intData = 10, charData = 99 'c'}
(gdb) r
The program being debugged has been started already.
Start it from the beginning? (y or n) n
Program not restarted.
(gdb) c
Continuing.

Breakpoint 2, sortDatabyBubble (array=0x61fe18, size=6) at Question4.c:29
29              for(i = 0; i <= size - 2; i++)      // <= ---> <, since if we use size - 1 to mapping next element of the array, the arr
will over bound.
(gdb) p *array
$5 = {intData = 10, charData = 99 'c'}
(gdb)
```

By after two times of continuous the loop, the element is not swapped, which means the loop have logic problem.

iii) Bug Fix Validation:

By applying correct logic of bubble sorting, let line 41,42,44,45 swap the location, make the program work.

```
    array[i + 1].intData = temp.intData;   //put large to next
    array[i + 1].charData = temp.charData;

    array[i].intData = array[i + 1].intData;   //give small to current
    array[i].charData = array[i + 1].charData;
```

3. Bug Observation 3:

i) I meet the sig fault problem:

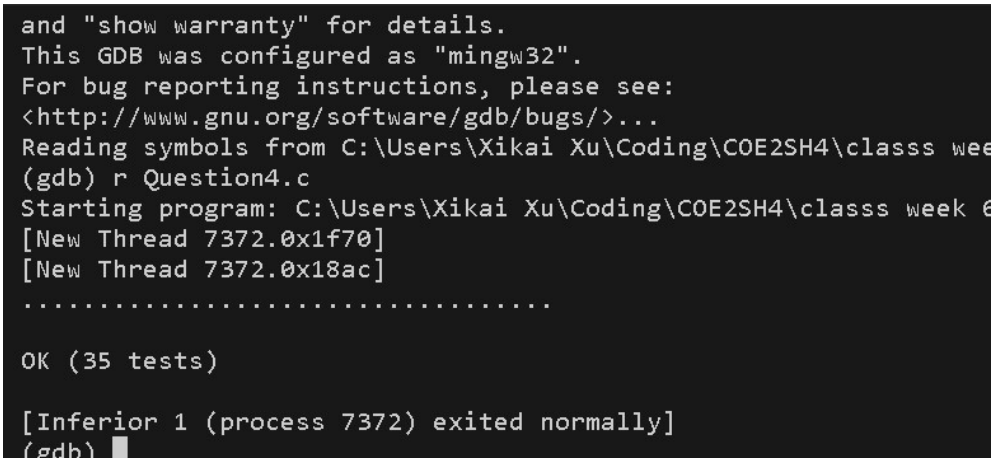
```
Starting program: C:\Users\Xikai Xu\Coding\COE2SH4\classs week 6\lab2-intelpentium4extreme-xu503/Lab2.exe
[New Thread 7084.0xa94]
[New Thread 7084.0x17a0]

Program received signal SIGSEGV, Segmentation fault.
0x004020a5 in TestQ4_BubbleSort_1 (tc=0x742390) at testCases.c:160
160                     CuAssertIntEquals(tc, expected[i].intData, input[i].intData);
(gdb) quit
A debugging session is active.

        Inferior 1 [process 7084] will be killed.
```

ii) Analysis:

After solving the logic of the Q4 bubble sort, the next element of the array question forms as i+1. Since the max size of the array is limited through - 1; so, we need to change the size of the array. Make a size – 2:

```
    for(i = 0; i <= size - 2; i++)      // <= ---> <, since if we u
    {
        curr = array[i].intData;
        next = array[i + 1].intData;    // loose ;
```

iii) Bug Fix Validation:




Successfully passes all the tests.