Tania Akter
V00810640

6.1

```
void func() {
    initialize(); // perform initialization
    do_work(); // do some work
    cleanup(); // perform any necessary cleanup
}
```

It is not exception safe. Because if *do_work()* throws an exception, *cleanup()* is never called and cleanup operation is not performed, which may cause resource leak.

6.2
   a) die3, die, countdown, hello, i, bjarne, herb, dv, u, z
   b) s

6.3
   a. Func() can throw while allocating memory for buf1 and buf2, and it is not handled. The allocate memory parts should be in try-catch block.

   a. It can throw during saving formatting flag or restoring formatting flags. It can also throw during the print statement in line 11. The entire function body should be kept in a try-catch block, or use RAII.

   b. Put() can throw an exception since std::deque::push_back may throw still throw an exception. The function body should be kept in a try-catch block.

6.5
analyze() does not throw exception because it has nothrow guarantee with *noexcept*.

doWork() may throw exception because it does not have the *noexcept* filter. Moreover, while passing the argument *globalThing,* since it is not a built in type, copy/move constructor is called, which may throw, causing *doWork()* to throw.