```
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % lldb ./lab05
(lldb) target create "./lab05"
Current executable set to '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64).
(lldb) run
Process 1619 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)

-- AN EMPTY LINKED LIST HAS BEEN CREATED.

-- THE NEW ENTRY apples AT POSITION 1 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
1

-- THE NEW ENTRY bananas AT POSITION 2 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
2

-- THE NEW ENTRY sweet potatoes AT POSITION 3 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
3

-- THE NEW ENTRY argula AT POSITION 4 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
4

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.

grocery list:
1. apples
2. bananas
3. sweet potatoes
4. argula

Process 1619 exited with status = 0 (0x00000000)
(lldb) run
Process 1634 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)

-- AN EMPTY LINKED LIST HAS BEEN CREATED.

-- THE NEW ENTRY apples AT POSITION 1 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
1

-- THE NEW ENTRY bananas AT POSITION 2 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
2

-- THE NEW ENTRY sweet potatoes AT POSITION 3 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
3

-- THE NEW ENTRY argula AT POSITION 4 HAS BEEN BE INSERTED INTO THE LIST.
```

```
-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
4

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.

grocery list:
1. apples
2. bananas
3. sweet potatoes
4. argula

Process 1634 exited with status = 0 (0x00000000)
(lldb) b main
Available completions:
        main.cpp
        main
        main_image_has_section
(lldb) b main
Available completions:
        main.cpp
        main
        main_image_has_section
(lldb) b main.cpp:16
Breakpoint 1: where = lab05`main + 79 at main.cpp:16:3, address = 0x000000010000181f
(lldb) run
Process 1660 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)

-- AN EMPTY LINKED LIST HAS BEEN CREATED.
Process 1660 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x000000010000181f lab05`main(argc=1, argv=0x00000003040c39e0) at main.cpp:16:3
   13
   14
   15           try {
-> 16                   groceries -> insert(1, "apples");
   17                   std::cout << groceries -> getLength() << std::endl;
   18                   groceries -> insert(2, "bananas");
   19                   std::cout << groceries -> getLength() << std::endl;
Target 0: (lab05) stopped.
(lldb) frame variable --no-args
(LinkedList<std::basic_string<char, std::char_traits<char>, std::allocator<char> > > *) groceries = 0x0000600000004030
(std::exception &) e = 0x0000000000000000
(std::exception &) e = 0x0000000200042517 {}
(lldb)
(LinkedList<std::basic_string<char, std::char_traits<char>, std::allocator<char> > > *) groceries = 0x0000600000004030
(std::exception &) e = 0x0000000000000000
(std::exception &) e = 0x0000000200042517 {}
(lldb) quit
Quitting LLDB will kill one or more processes. Do you really want to proceed: [Y/n] Y
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % lldb ./lab05
(lldb) target create "./lab05"
Current executable set to '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64).
(lldb) break main.cpp:15
error: command 'breakpoint' did not recognize 'main .cpp:15' as valid (subcommand might be invalid).
(lldb) b main.cpp:15
Breakpoint 1: where = lab05`main + 123 at main.cpp:22:14, address = 0x00000001000017db
(lldb) run
Process 1800 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)
Process 1800 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x00000001000017db lab05`main(argc=1, argv=0x00000003040c39e0) at main.cpp:22:14
   19
   20
   21           LinkedList<std::string>* groceries;
```

```
-> 22           groceries = new LinkedList<std::string>();
   23
   24
   25           try {
Target 0: (lab05) stopped.
(lldb) next

-- AN EMPTY LINKED LIST HAS BEEN CREATED.
Process 1800 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = step over
    frame #0: 0x000000010000180c lab05`main(argc=1, argv=0x00000003040c39e0) at main.cpp:26:3
   23
   24
   25           try {
-> 26                   groceries -> insert(1, "apples");
   27                   std::cout << groceries -> getLength() << std::endl;
   28                   groceries -> insert(2, "bananas");
   29                   std::cout << groceries -> getLength() << std::endl;
Target 0: (lab05) stopped.
(lldb) breakpoint list
Current breakpoints:
1: file = 'main.cpp', line = 15, exact_match = 0, locations = 1, resolved = 1, hit count = 1
  1.1: where = lab05`main + 123 at main.cpp:22:14, address = 0x00000001000017db, resolved, hit count = 1

(lldb) breakpoint delete 1
1 breakpoints deleted; 0 breakpoint locations disabled.
(lldb) breakpoint list
No breakpoints currently set.
(lldb) quit
Quitting LLDB will kill one or more processes. Do you really want to proceed: [Y/n] Y
owner@morgan EECS268-Lab05 % make clea n
make: *** No rule to make target `clea'.  Stop.
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % ./lab05

-- AN EMPTY LINKED LIST HAS BEEN CREATED.

-- THE NEW ENTRY apples AT POSITION 1 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
1

-- THE NEW ENTRY bananas AT POSITION 2 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
2

-- THE NEW ENTRY sweet potatoes AT POSITION 3 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
3

-- THE NEW ENTRY argula AT POSITION 4 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
4

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.

grocery list:
1. apples
2. bananas
3. sweet potatoes
4. argula

owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
```

```
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % lldb ./lab05
(lldb) target create "./lab05"
Current executable set to '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64).
(lldb) run
Process 1857 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)

-- AN EMPTY LINKED LIST HAS BEEN CREATED.

-- THE NEW ENTRY apples AT POSITION 1 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
1

-- THE NEW ENTRY bananas AT POSITION 2 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
2

-- THE NEW ENTRY sweet potatoes AT POSITION 3 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
3

-- THE NEW ENTRY argula AT POSITION 4 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
4

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.

grocery list:
1. apples
2. bananas
3. sweet potatoes
4. argula

Process 1857 exited with status = 0 (0x00000000)
(lldb) list
(lldb) breakpoint list
No breakpoints currently set.
(lldb) b main.cpp:14
Breakpoint 1: where = lab05`main + 89 at main.cpp:14:18, address = 0x00000001000017b9
(lldb) breakpoint list
Current breakpoints:
1: file = 'main.cpp', line = 14, exact_match = 0, locations = 1
  1.1: where = lab05`main + 89 at main.cpp:14:18, address = 0x00000001000017b9, unresolved, hit count = 0

(lldb) frame variable
error: invalid thread
(lldb) frame variable
error: invalid thread
(lldb) frame
Available completions:
        diagnose   -- Try to determine what path path the current stop location used to get to a register or address
        info       -- List information about the current stack frame in the current thread.
        recognizer -- Commands for editing and viewing frame recognizers.
        select     -- Select the current stack frame by index from within the current thread (see 'thread backtrace'.)
        variable   -- Show variables for the current stack frame. Defaults to all arguments and local variables in scope. Names
 of argument, local, file static and file global variables can be specified. Children of aggregate variables can be specified s
uch as 'var->child.x'.  The -> and [] operators in 'frame variable' do not invoke operator overloads if they exist, but directl
y access the specified element.  If you want to trigger operator overloads use the expression command to print the variable ins
tead.
It is worth noting that except for overloaded operators, when printing local variables 'expr local_var' and 'frame var local_va
r' produce the same results.  However, 'frame variable' is more efficient, since it uses debug information and memory reads dir
ectly, rather than parsing and evaluating an expression, which may even involve JITing and running code in the target program.
(lldb) frame variable --no-args
error: invalid thread
(lldb) aliksdj
error: 'aliksdj' is not a valid command.
(lldb) frame info
```

```
error: invalid thread
(lldb) breakpoint list
Current breakpoints:
1: file = 'main.cpp', line = 14, exact_match = 0, locations = 1
  1.1: where = lab05`main + 89 at main.cpp:14:18, address = 0x00000001000017b9, unresolved, hit count = 0

(lldb) quit
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % lldb ./lab05
(lldb) target create "./lab05"
Current executable set to '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64).
(lldb) run
Process 2133 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)

-- AN EMPTY LINKED LIST HAS BEEN CREATED.

-- THE NEW ENTRY apples AT POSITION 1 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
1

-- THE NEW ENTRY bananas AT POSITION 2 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
2

-- THE NEW ENTRY sweet potatoes AT POSITION 3 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
3

-- THE NEW ENTRY argula AT POSITION 4 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
4

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.

grocery list:
1. apples
2. bananas
3. sweet potatoes
4. argula

Process 2133 exited with status = 0 (0x00000000)
(lldb) breakpoint list
No breakpoints currently set.
(lldb) b main.cpp:14
Breakpoint 1: where = lab05`main + 89 at main.cpp:14:18, address = 0x00000001000017b9
(lldb) run
Process 2147 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)
Process 2147 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x00000001000017b9 lab05`main(argc=1, argv=0x00000003040c39e0) at main.cpp:14:18
   11          int size = 5;
   12          int* pointer = new int[size];
   13          for (int i = 0; i < size; i++) {
-> 14                  pointer[i] = 10*(i);
   15          }
   16
   17          LinkedList<std::string>* groceries;
Target 0: (lab05) stopped.
(lldb) frame variable
```

```
(int) argc = 1
(const char **) argv = 0x00000003040c39e0
(int) size = 5
(int *) pointer = 0x0000600000201140
(LinkedList<std::basic_string<char, std::char_traits<char>, std::allocator<char> > > *) groceries = 0x0000000100001760
(std::exception &) e = 0x000000020008c3a0 {}
(std::exception &) e = 0x0000000000000001 {}
(int) i = 0
(lldb) quit
Quitting LLDB will kill one or more processes. Do you really want to proceed: [Y/n] Y
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % make clean
rm *.o lab05
owner@morgan EECS268-Lab05 % make
g++ -g -std=c++11 -Wall -c main.cpp
g++ -g -std=c++11 -Wall main.o -o lab05
owner@morgan EECS268-Lab05 % lldb ./lab05
(lldb) target create "./lab05"
Current executable set to '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64).
(lldb) run
Process 2194 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)

-- AN EMPTY LINKED LIST HAS BEEN CREATED.

-- THE NEW ENTRY apples AT POSITION 1 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
1

-- THE NEW ENTRY bananas AT POSITION 2 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
2

-- THE NEW ENTRY sweet potatoes AT POSITION 3 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
3

-- THE NEW ENTRY argula AT POSITION 4 HAS BEEN BE INSERTED INTO THE LIST.

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.
4

-- RETURNING THE AMOUNT OF NODE OBJECTS IN THE LIST.

grocery list:
1. apples
2. bananas
3. sweet potatoes
4. argula

Process 2194 exited with status = 0 (0x00000000)
(lldb) breakpoint list
No breakpoints currently set.
(lldb) b main.cpp:14
Breakpoint 1: where = lab05`main + 82 at main.cpp:14:18, address = 0x00000001000017c2
(lldb) breakpoint list
Current breakpoints:
1: file = 'main.cpp', line = 14, exact_match = 0, locations = 1
  1.1: where = lab05`main + 82 at main.cpp:14:18, address = 0x00000001000017c2, unresolved, hit count = 0

(lldb) run
Process 2213 launched: '/Users/owner/Desktop/lab05/EECS268-Lab05/EECS268-Lab05/lab05' (x86_64)
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x00000001000017c2 lab05`main at main.cpp:14:18
   11          int size = 5;
```

```
   12          int* pointer = new int[size];
   13          for (int i = 0; i < size; i++) {
-> 14                  pointer[i] = 10*(i);
   15          }
   16
   17          LinkedList<std::string>* groceries;
Target 0: (lab05) stopped.
(lldb) frame variable
(int) size = 5
(int *) pointer = 0x0000600000201140
(LinkedList<std::basic_string<char, std::char_traits<char>, std::allocator<char> > > *) groceries = 0x00000003040c38a0
(std::exception &) e = 0x0000000000000000
(std::exception &) e = 0x00000003040c3850 {}
(int) i = 0
(lldb) next
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = step over
    frame #0: 0x00000001000017d1 lab05`main at main.cpp:13:29
   10
   11          int size = 5;
   12          int* pointer = new int[size];
-> 13          for (int i = 0; i < size; i++) {
   14                  pointer[i] = 10*(i);
   15          }
   16
Target 0: (lab05) stopped.
(lldb) next
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x00000001000017c2 lab05`main at main.cpp:14:18
   11          int size = 5;
   12          int* pointer = new int[size];
   13          for (int i = 0; i < size; i++) {
-> 14                  pointer[i] = 10*(i);
   15          }
   16
   17          LinkedList<std::string>* groceries;
Target 0: (lab05) stopped.
(lldb) frame variable
(int) size = 5
(int *) pointer = 0x0000600000201140
(LinkedList<std::basic_string<char, std::char_traits<char>, std::allocator<char> > > *) groceries = 0x00000003040c38a0
(std::exception &) e = 0x0000000000000000
(std::exception &) e = 0x00000003040c3850 {}
(int) i = 1
(lldb) print
Enter expressions, then terminate with an empty line to evaluate:
  1: size
(int) $0 = 5
(lldb)
Enter expressions, then terminate with an empty line to evaluate:
  1: pointer
(int *) $1 = 0x0000600000201140
(lldb) print
Enter expressions, then terminate with an empty line to evaluate:
  1: i
(int) $2 = 1
(lldb) print
Enter expressions, then terminate with an empty line to evaluate:
  1: *pointer
(int) $3 = 0
(lldb) next
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = step over
    frame #0: 0x00000001000017d1 lab05`main at main.cpp:13:29
   10
   11          int size = 5;
   12          int* pointer = new int[size];
-> 13          for (int i = 0; i < size; i++) {
   14                  pointer[i] = 10*(i);
   15          }
   16
```

```
Target 0: (lab05) stopped.
(lldb) next
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x00000001000017c2 lab05`main at main.cpp:14:18
   11          int size = 5;
   12          int* pointer = new int[size];
   13          for (int i = 0; i < size; i++) {
-> 14                  pointer[i] = 10*(i);
   15          }
   16
   17          LinkedList<std::string>* groceries;
Target 0: (lab05) stopped.
(lldb) next
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = step over
    frame #0: 0x00000001000017d1 lab05`main at main.cpp:13:29
   10
   11          int size = 5;
   12          int* pointer = new int[size];
-> 13          for (int i = 0; i < size; i++) {
   14                  pointer[i] = 10*(i);
   15          }
   16
Target 0: (lab05) stopped.
(lldb) next
Process 2213 stopped
* thread #1, queue = 'com.apple.main-thread', stop reason = breakpoint 1.1
    frame #0: 0x00000001000017c2 lab05`main at main.cpp:14:18
   11          int size = 5;
   12          int* pointer = new int[size];
   13          for (int i = 0; i < size; i++) {
-> 14                  pointer[i] = 10*(i);
   15          }
   16
   17          LinkedList<std::string>* groceries;
Target 0: (lab05) stopped.
(lldb) print *pointer
(int) $4 = 0
(lldb) print
Enter expressions, then terminate with an empty line to evaluate:
  1: pointer->
Available completions:
        size
        pointer
        groceries
        e
        i
        length
  1: pointer
Available completions:
        size
        pointer
        groceries
        e
        i
        length
  1: pointer -> pointer
error: <user expression 6>:1:9: member reference base type 'int' is not a structure or union
pointer -> pointer
~~~~~~~ ^  ~~~~~~~
(lldb)
Enter expressions, then terminate with an empty line to evaluate:
  1: *pointer
(int) $5 = 0
(lldb) quit
Quitting LLDB will kill one or more processes. Do you really want to proceed: [Y/n] Y
owner@morgan EECS268-Lab05 %
```