

Iteration Callback Takes Two `void*`, Returns a Number

And here's the definition for the callback function:

```
typedef dl_execute_response_t
(*dl_execute_func_t)
(void* dl, void* arg);
```

37

Iteration Requires Three Arguments

Now we can also write the function signature for iteration:

```
void* dl_execute_on_all
(double_list_t* head,
dl_execute_func_t func,
void* arg);
```

Pointer to head of list.

Pointer to callback function.

Argument passed to (all) callback invocations.

38

Iteration Returns a Pointer to a "Thing"

Now we can also write the function signature for iteration:

```
void* dl_execute_on_all
(double_list_t* head,
dl_execute_func_t func,
void* arg);
```

Returns a pointer to a "thing."

39

Local Variables for our Iteration Function

And here's the function:

```
double_list_t* dl;
double_list_t* remove;
dl_execute_response_t result;
```

iteration over list elements

element to be removed

result returned from callback

40