

```
1: // $Id: queue.h,v 1.5 2013-05-01 13:22:44-07 - - $
2:
3: #ifndef __QUEUE_H__
4: #define __QUEUE_H__
5:
6: #include <stdbool.h>
7:
8: typedef struct queue queue;
9: typedef char *queue_item_t;
10:
11: queue *new_queue (void);
12:
13: void free_queue (queue *);
14:
15: void insert_queue (queue *, queue_item_t);
16:
17: queue_item_t remove_queue (queue *);
18:
19: bool isempty_queue (queue *);
20:
21: #endif
22:
```

```
1: // $Id: main.c,v 1.10 2013-02-15 17:17:42-08 - - $
2:
3: #include <assert.h>
4: #include <errno.h>
5: #include <libgen.h>
6: #include <stdio.h>
7: #include <stdlib.h>
8: #include <string.h>
9:
10: #include "queue.h"
11:
12: char *execname = NULL;
13: int exit_status = EXIT_SUCCESS;
14:
15: void putinqueue (queue *the_queue, FILE *input, char *filename) {
16:     char buffer[1024];
17:     for (int linenr = 1; ; ++linenr) {
18:         char *linepos = fgets (buffer, sizeof buffer, input);
19:         if (linepos == NULL) break;
20:         linepos = strchr (buffer, '\n');
21:         if (linepos == NULL) {
22:             fflush (NULL);
23:             fprintf (stderr, "%s: %s[%d]: unterminated line\n",
24:                     execname, filename, linenr);
25:             fflush (NULL);
26:             exit_status = EXIT_FAILURE;
27:         } else {
28:             *linepos = '\0';
29:         }
30:         linepos = strdup (buffer);
31:         assert (linepos != NULL);
32:         insert_queue (the_queue, linepos);
33:     }
34: }
35:
36: void putfileinqueue (queue *the_queue, char *filename) {
37:     FILE *input = fopen (filename, "r");
38:     if (input == NULL) {
39:         fflush (NULL);
40:         fprintf (stderr, "%s: %s: %s\n",
41:                 execname, filename, strerror (errno));
42:         fflush (NULL);
43:         exit_status = EXIT_FAILURE;
44:     } else {
45:         putinqueue (the_queue, input, filename);
46:         fclose (input);
47:     }
48: }
49:
```

```
50:
51: int main (int argc, char **argv) {
52:     execname = basename (argv[0]);
53:     queue *the_queue = new_queue();
54:
55:     if (argc < 2) {
56:         putinqueue (the_queue, stdin, "-");
57:     }else {
58:         for (int argi = 1; argi < argc; ++argi) {
59:             if (strcmp (argv[argi], "-") == 0) {
60:                 putinqueue (the_queue, stdin, "-");
61:             }else {
62:                 putfileinqueue (the_queue, argv[argi]);
63:             }
64:         }
65:     }
66:
67:     while (! isempty_queue (the_queue)) {
68:         printf ("%s\n", remove_queue (the_queue));
69:     }
70:
71:     return exit_status;
72: }
73:
```

```
1: // $Id: queue.c,v 1.9 2013-05-01 13:22:44-07 - - $
2:
3: #include <assert.h>
4: #include <stdio.h>
5: #include <stdlib.h>
6: #include <string.h>
7:
8: #include "queue.h"
9:
10: #define STUBPRINTF(...) fprintf (stderr, __VA_ARGS__);
11:
12: typedef struct queue_node queue_node;
13: struct queue_node {
14:     queue_item_t item;
15:     queue_node *link;
16: };
17:
18: struct queue {
19:     queue_node *front;
20:     queue_node *rear;
21: };
22:
23: queue *new_queue (void) {
24:     STUBPRINTF ("return NULL\n");
25:     return NULL;
26: }
27:
28: void free_queue (queue *this) {
29:     assert (isempty_queue (this));
30:     free (this);
31: }
32:
33: void insert_queue (queue *this, queue_item_t item) {
34:     STUBPRINTF ("item =\n\t\"%s\"\n", item);
35: }
36:
37: queue_item_t remove_queue (queue *this) {
38:     assert (! isempty_queue (this));
39:     STUBPRINTF ("return NULL\n");
40:     return NULL;
41: }
42:
43: bool isempty_queue (queue *this) {
44:     return this->front == NULL;
45: }
46:
```

```
1: # $Id: Makefile,v 1.5 2013-05-01 13:21:00-07 - - $
2:
3: MKFILE      = Makefile
4: DEPSFILE    = ${MKFILE}.deps
5: NOINCLUDE   = ci clean spotless
6: NEEDINCL    = ${filter ${NOINCLUDE}, ${MAKECMDGOALS}}
7:
8: GCC          = gcc -g -O0 -Wall -Wextra -std=gnu99
9: MKDEPS      = gcc -MM
10: GRIND        = valgrind --leak-check=full
11:
12: CSOURCE      = main.c queue.c
13: CHEADER      =          queue.h
14: OBJECTS      = ${CSOURCE:.c=.o}
15: EXECBIN      = catqueue
16: SOURCES      = ${CHEADER} ${CSOURCE} ${MKFILE}
17: LISTSRC      = ${SOURCES} ${DEPSFILE}
18: LISTING      = Listing.catqueue.ps
19: OUTPUT       = output*.lis
20:
21: all : ${EXECBIN}
22:
23: ${EXECBIN} : ${OBJECTS}
24:             ${GCC} -o $@ ${OBJECTS}
25:
26: %.o : %.c
27:         cid + $<
28:         ${GCC} -c $<
29:
30: ci : ${SOURCES}
31:     cid + ${SOURCES} test*.data
32:
33: lis : ${SOURCES} test
34:     mkpspdf ${LISTING} ${LISTSRC} ${OUTPUT}
35:
36: clean :
37:     - rm ${OBJECTS} ${DEPSFILE} core ${OUTPUT}
38:
39: spotless : clean
40:     - rm ${EXECBIN} ${LISTING} ${LISTING:.ps=.pdf}
41:
42: test : ${EXECBIN}
43:     - runprogram.perl -x output1.lis -0test1.data ${EXECBIN}
44:     - runprogram.perl -x output2.lis ${EXECBIN} test*.data
45:     - ${GRIND} ${EXECBIN} <test1.data >output3.lis 2>&1
46:
47: deps : ${CSOURCE} ${CHEADER}
48:     @ echo "# ${DEPSFILE} created `date`" >${DEPSFILE}
49:     ${MKDEPS} ${CSOURCE} | sort | uniq >>${DEPSFILE}
50:
51: ${DEPSFILE} :
52:     @ touch ${DEPSFILE}
53:     ${MAKE} --no-print-directory deps
54:
55:
56: again :
57:     gmake spotless deps ci all lis
58:
```

10/17/13
18:40:14

\$cmps012b-wm/Labs-cmps012m/lab7c-headers-adts/catqueue/
Makefile

2/2

```
59: ifeq "${NEEDINCL}" ""  
60: include ${DEPSFILE}  
61: endif  
62:
```

10/17/13
18:40:14

\$cmpps012b-wm/Labs-cmps012m/lab7c-headers-adts/catqueue/
Makefile.deps

1/1

```
1: # Makefile.deps created Thu Oct 17 18:40:14 PDT 2013
2: main.o: main.c queue.h
3: queue.o: queue.c queue.h
```

```
1:
2: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
3: log: output1.log
4: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
5:
6:      1  Script   : /afs/cats.ucsc.edu/courses/cmps012b-wm/bin/runprogram.
perl
7:      2  limit c :      0 max core file size (KB)
8:      3  limit f : 4194303 max output file size (KB)
9:      4  limit t : 4294967295 max CPU time (sec)
10:     5  stdin   : test1.data
11:     6  stdout  : output1.out
12:     7  stderr  : output1.err
13:     8  log     : output1.log
14:     9  listing : output1.lis
15:    10  Command : catqueue
16:    11  starting: pid 5421: 18:40:14.00
17:    12  finished: pid 5421: 18:40:14.00, real 0.00, user 0.00, sys 0.00
18:    13  pstatus: 0x008B TERMINATED 11: Segmentation fault (core dumped)
19:
20: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
21: stdin: test1.data
22: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
23:
24:     1  $Id: test1.data,v 1.1 2012-02-14 20:32:33-08 - - $
25:     2  Test data 1 line 1.
26:     3  Test data 1 line 2.
27:     4  Test data 1 line 3.
28:
29: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
30: stdout: output1.out
31: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
32:
33:
34: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
35: stderr: output1.err
36: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
37:
38:     1  return NULL
39:     2  item =
40:     3      "$Id: test1.data,v 1.1 2012-02-14 20:32:33-08 - - $"
41:     4  item =
42:     5      "Test data 1 line 1."
43:     6  item =
44:     7      "Test data 1 line 2."
45:     8  item =
46:     9      "Test data 1 line 3."
```



```
1:
2: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
3: log: output2.log
4: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
5:
6:      1  Script   : /afs/cats.ucsc.edu/courses/cmps012b-wm/bin/runprogram.
perl
7:      2  limit c :      0 max core file size (KB)
8:      3  limit f : 4194303 max output file size (KB)
9:      4  limit t : 4294967295 max CPU time (sec)
10:     5  stdin   : /dev/null
11:     6  stdout  : output2.out
12:     7  stderr  : output2.err
13:     8  log     : output2.log
14:     9  listing : output2.lis
15:    10  Command : catqueue test1.data test2.data test3.data
16:    11  starting: pid 5431: 18:40:15.00
17:    12  finished: pid 5431: 18:40:15.00, real 0.00, user 0.00, sys 0.00
18:    13  pstatus: 0x008B TERMINATED 11: Segmentation fault (core dumped)
19:
20: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
21: stdin: /dev/null
22: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
23:
24:
25: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
26: stdout: output2.out
27: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
28:
29:
30: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
31: stderr: output2.err
32: ::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::::
33:
34:     1  return NULL
35:     2  item =
36:     3      "$Id: test1.data,v 1.1 2012-02-14 20:32:33-08 - - $"
37:     4  item =
38:     5      "Test data 1 line 1."
39:     6  item =
40:     7      "Test data 1 line 2."
41:     8  item =
42:     9      "Test data 1 line 3."
43:    10  item =
44:    11      "$Id: test2.data,v 1.1 2012-02-14 20:32:33-08 - - $"
45:    12  item =
46:    13      "Test data 2 line 1."
47:    14  item =
48:    15      "Test data 2 line 2."
49:    16  item =
50:    17      "Test data 2 line 3."
51:    18  item =
52:    19      "$Id: test3.data,v 1.1 2012-02-14 20:32:33-08 - - $"
53:    20  item =
54:    21      "Test data 3 line 1."
55:    22  item =
56:    23      "Test data 3 line 2."
57:    24  item =
```

10/17/13
18:40:15

\$cmps012b-wm/Labs-cmps012m/lab7c-headers-adts/catqueue/
output2.lis

2/2

58: 25 "Test data 3 line 3."

fo

```
1: ==5439== Memcheck, a memory error detector
2: ==5439== Copyright (C) 2002-2012, and GNU GPL'd, by Julian Seward et al.
3: ==5439== Using Valgrind-3.8.1 and LibVEX; rerun with -h for copyright in
4: ==5439== Command: catqueue
5: ==5439==
6: return NULL
7: item =
8:      "$Id: test1.data,v 1.1 2012-02-14 20:32:33-08 - - $"
9: item =
10:      "Test data 1 line 1."
11: item =
12:      "Test data 1 line 2."
13: item =
14:      "Test data 1 line 3."
15: ==5439== Invalid read of size 8
16: ==5439==      at 0x400D38: isempty_queue (queue.c:44)
17: ==5439==      by 0x400C15: main (main.c:67)
18: ==5439==      Address 0x0 is not stack'd, malloc'd or (recently) free'd
19: ==5439==
20: ==5439==
21: ==5439== Process terminating with default action of signal 11 (SIGSEGV)
22: ==5439==      Access not within mapped region at address 0x0
23: ==5439==      at 0x400D38: isempty_queue (queue.c:44)
24: ==5439==      by 0x400C15: main (main.c:67)
25: ==5439==      If you believe this happened as a result of a stack
26: ==5439==      overflow in your program's main thread (unlikely but
27: ==5439==      possible), you can try to increase the size of the
28: ==5439==      main thread stack using the --main-stacksize= flag.
29: ==5439==      The main thread stack size used in this run was 10485760.
30: ==5439==
31: ==5439== HEAP SUMMARY:
32: ==5439==      in use at exit: 111 bytes in 4 blocks
33: ==5439==      total heap usage: 4 allocs, 0 frees, 111 bytes allocated
34: ==5439==
35: ==5439== 111 bytes in 4 blocks are definitely lost in loss record 1 of 1
36: ==5439==      at 0x4A069EE: malloc (vg_replace_malloc.c:270)
37: ==5439==      by 0x3C0E081081: strdup (in /lib64/libc-2.12.so)
38: ==5439==      by 0x400A2A: putinqueue (main.c:30)
39: ==5439==      by 0x400B7C: main (main.c:56)
40: ==5439==
41: ==5439== LEAK SUMMARY:
42: ==5439==      definitely lost: 111 bytes in 4 blocks
43: ==5439==      indirectly lost: 0 bytes in 0 blocks
44: ==5439==      possibly lost: 0 bytes in 0 blocks
45: ==5439==      still reachable: 0 bytes in 0 blocks
46: ==5439==      suppressed: 0 bytes in 0 blocks
47: ==5439==
48: ==5439== For counts of detected and suppressed errors, rerun with: -v
49: ==5439== ERROR SUMMARY: 2 errors from 2 contexts (suppressed: 6 from 6)
```