

The image shows two separate runs of a C++ program named 'Queue' in the CLion IDE. Both runs show the state of a dynamic queue as keys are inserted and read.

**Run 1 (Top):**

- Read key b. Inserting b in rear. Queue: b->
- Read key t. Inserting t in rear. Queue: b->t->
- Read key s. Inserting s in rear. Queue: b->t->s->
- Read key a. Inserting a in rear. Queue: b->t->s->a->
- Read key c. Queue is full, removing front. Inserting c in rear. Queue: t->s->a->c->
- Read key c. Key c is already rear. Queue: t->s->a->c->
- Read key s. Moving s to rear. Queue: t->a->c->s->

Process finished with exit code 0

**Run 2 (Bottom):**

- Read key a. Inserting a in rear. Queue: a->
- Read key b. Inserting b in rear. Queue: a->b->
- Read key w. Inserting w in rear. Queue: a->b->w->
- Read key b. Key already in queue, moving b to rear. Queue: a->w->b->
- Read key z. Inserting z in rear. Queue: a->w->b->z->
- Read key e. Queue is full, removing front. Inserting e in rear. Queue: w->b->z->e->

Process finished with exit code 0

Both runs were completed in approximately 1 second, with 548 ms (16 minutes ago).

File Edit View Navigate Code Refactor Build Run Tools VCS Window Help Queue - main.cpp

Queue / main.cpp

Project Queue C:/Users/Jeffr/CLionProjects/Queue

cmake-build-debug

CMakeLists.txt

DynamicQueue.cpp

DynamicQueue.h

main.cpp

Screenshot 2023-02-27 155828.png

Screenshot 2023-02-27 160208.png

External Libraries

Scratches and Consoles

#include ...

```
int main() {
    DynamicQueue Queue;
    Queue.insert('a');
    Queue.insert('a');
    Queue.insert('b');
    Queue.insert('b');
    Queue.insert('e');
    Queue.insert('e');
    Queue.insert('f');
    Queue.insert('f');
    Queue.insert('a');
    Queue.insert('g');
    return 0;
}
```

Run: Queue

↑ Read key a. Inserting a in rear. Queue: a->
↓ Read key a. Key a is already rear. Queue: a->
Read key b. Inserting b in rear. Queue: a->b->
Read key b. Key b is already rear. Queue: a->b->
Read key e. Inserting e in rear. Queue: a->b->e->
Read key e. Key e is already rear. Queue: a->b->e->
Read key f. Inserting f in rear. Queue: a->b->e->f->
Read key f. Key f is already rear. Queue: a->b->e->f->
Read key a. Key already in queue, moving a to rear. Queue: b->e->f->a->
Read key g. Queue is full, removing front. Inserting g in rear. Queue: e->f->a->g->

Process finished with exit code 0

Version Control Run TODO Problems Terminal Python Packages Services CMake Messages

Process finished with exit code 0

15:20 LF UTF-8 4 spaces C++ Queue | Debug