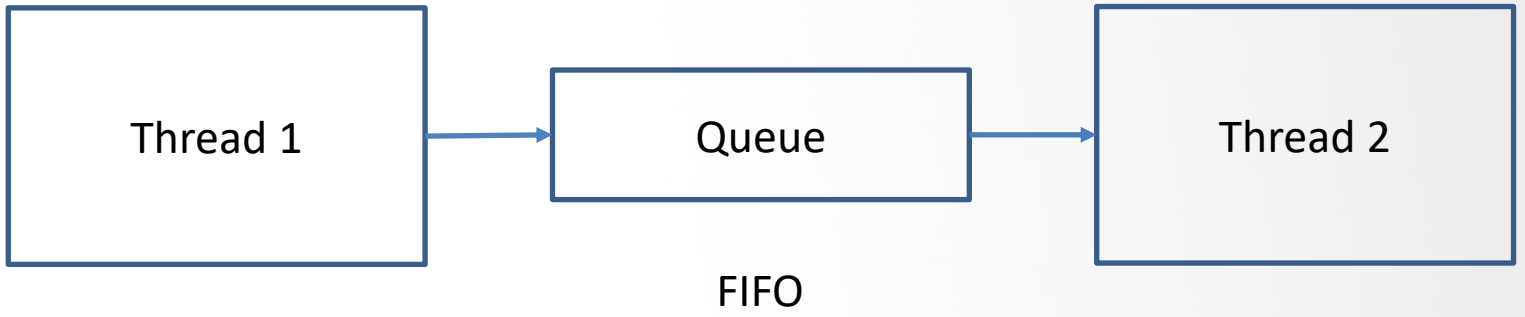
# Tutorial 5
# Thread and Queue

Kai CHEN

# Example

## Producer-Consumer Model

A simplified version



# Example

## Producer-Consumer Model

```python
import time
from threading import Thread
from queue import Queue

queue = Queue()


def produce():
    start = time.time()
    i = 1
    while True:
        time.sleep(1)  # simulate some processing
        t = time.time() - start
        print('{:.1f} s, produces {}'.format(t, i))
        queue.put(i)
        i += 1


def consume():
    start = time.time()
    while True:
        i = queue.get()
        time.sleep(0.5)  # simulate some processing
        t = time.time() - start
        print('{:.1f} s, consumes {}'.format(t, i))


if __name__ == '__main__':
    produce_thread = Thread(target=produce)
    consume_thread = Thread(target=consume)
    produce_thread.start()
    consume_thread.start()
```

Thread 1 (producer)

Thread 2 (consumer)

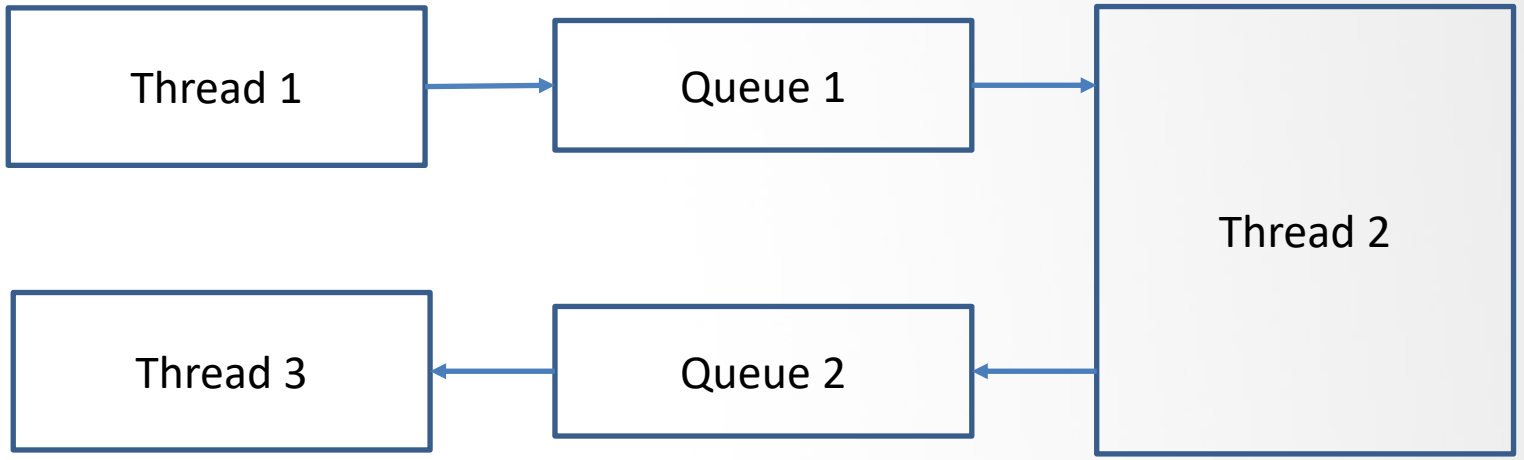
# Example

## Producer-Consumer Model

```
$ python ~/Desktop/tmp.py
1.0 s, produces 1
1.5 s, consumes 1
2.0 s, produces 2
2.5 s, consumes 2
3.0 s, produces 3
3.5 s, consumes 3
4.0 s, produces 4
4.5 s, consumes 4
5.0 s, produces 5
5.5 s, consumes 5
6.0 s, produces 6
```

# Example

More threads and queues



# Example

## Original version

```python
import time
import telepot
from telepot.loop import MessageLoop

def handle(msg):
    """
    A function that will be invoked when a message is
    recevied by the bot
    """
    content_type, chat_type, chat_id = telepot.glance(msg)

    if content_type == "text":
        content = msg["text"]
        reply = "You said: {}".format(content)
        bot.sendMessage(chat_id, reply)

if __name__ == "__main__":

    # Provide your bot's token
    bot = telepot.Bot("YOUR_TELEGRAM_BOT_TOKEN")
    MessageLoop(bot, handle).run_as_thread()

    while True:
        time.sleep(10)
```

# Example

## Thread 1 (receive thread)

```
def receive_thread(msg):
    """
    A function that will be invoked when a message is
    recevied by the bot
    """
    content_type, chat_type, chat_id = telepot.glance(msg)

    if content_type == "text":
        content = msg["text"]
        data = {
            "content": content,
            "chat_id": chat_id
        }
        queue1.put(data)
```

# Example

## Thread 2 (process thread)

```
def processing_thread():
    while True:
        data = queue1.get()
        content = data["content"]
        chat_id = data["chat_id"]
        reply = "You said: {}".format(content)
        data_response = {
            "reply": reply,
            "chat_id": chat_id
        }
        queue2.put(data_response)
```

# Example

## Thread 3 (response thread)

```
def response_thread():
    while True:
        data = queue2.get()
        reply = data["reply"]
        chat_id = data["chat_id"]
        bot.sendMessage(chat_id, reply)
```

# Example

Start threads

```
if __name__ == "__main__":

    threading.Thread(target=client_thread).start()
    threading.Thread(target=response_thread).start()

    # Provide your bot's token
    bot = telepot.Bot("YOUR_TELEGRAM_BOT_TOKEN")
    MessageLoop(bot, receive_thread).run_as_thread()

    while True:
        time.sleep(10)
```

# Thank you!