```python
#!/usr/bin/env python3
import argparse
from Bio import SeqIO

def interleave_fastq(forward_file, reverse_file, output_file):
    # Open the forward and reverse FASTQ files for reading and the output file for
writing
    with open(forward_file, "r") as forward, open(reverse_file, "r") as reverse,
open(output_file, "w") as output:
        # Use SeqIO to parse the FASTQ files
        forward_iter = SeqIO.parse(forward, "fastq")
        reverse_iter = SeqIO.parse(reverse, "fastq")

        # Interleave reads from the forward and reverse iterators
        for fwd_record, rev_record in zip(forward_iter, reverse_iter):
            # Write each forward and reverse read to the output file
            SeqIO.write(fwd_record, output, "fastq")
            SeqIO.write(rev_record, output, "fastq")

        # Check for any remaining records in either file to handle differing
numbers of records
        try:
            # Attempt to get another record from the forward file
            extra_record = next(forward_iter)
            print(f"Extra record found in forward file: {extra_record.id}")
        except StopIteration:
            # No more records in the forward file
            pass

        try:
            # Attempt to get another record from the reverse file
            extra_record = next(reverse_iter)
            print(f"Extra record found in reverse file: {extra_record.id}")
        except StopIteration:
            # No more records in the reverse file
            pass

if __name__ == '__main__':
    parser = argparse.ArgumentParser(description='Interleave Paired-End FASTQ
Files')
    parser.add_argument('forward_file', help='Forward reads FASTQ file')
    parser.add_argument('reverse_file', help='Reverse reads FASTQ file')
    parser.add_argument('output_file', help='Output interleaved FASTQ file')
    args = parser.parse_args()
    interleave_fastq(args.forward_file, args.reverse_file, args.output_file)
```