

src/abi2fq

April 26, 2025

## **Contents**

This module provides a command-line tool for converting ABIF files to FASTQ format with optional quality trimming.

The abi2fq tool extracts sequence and quality data from ABIF files, applies quality trimming to remove low-quality regions, and outputs in the standard FASTQ format.

Command-line usage:

```
abi2fq [options] <input.ab1> [output.fq]
```

Options:

<b>-h, --help</b>	Show help message
<b>-w, --window=INT</b>	Window size for quality trimming (default: 10)
<b>-q, --quality=INT</b>	Quality threshold 0-60 (default: 20)
<b>-n, --no-trim</b>	Disable quality trimming
<b>-v, --verbose</b>	Print additional information
<b>--version</b>	Show version information

Examples:

```
# Convert with default quality trimming
abi2fq input.ab1 output.fastq

# Convert without quality trimming
abi2fq -n input.ab1 output.fastq

# Convert with custom quality parameters
abi2fq -w 20 -q 30 input.ab1 output.fastq
```

## 1 Imports

abif

## 2 Types

```
Config = object
    inFile*: string          ## Path to the input ABIF file
    outFile*: string         ## Path to the output FASTQ file (or empty for stdout)
    windowSize*: int         ## Window size for quality trimming (default: 10)
    qualityThreshold*: int   ## Quality threshold 0-60 (default: 20)
    noTrim*: bool            ## Whether to disable quality trimming
    verbose*: bool           ## Whether to show verbose output
    showVersion*: bool       ## Whether to show version information
```

Configuration for the abi2fq tool. Contains command-line options and settings.

## 3 Procs

```
proc main() {.raises: [ValueError], tags: [ReadIOEffect, WriteIOEffect],
             forbids: [].}
```

Main entry point for the abi2fq program.

Handles command-line parsing, reads the input ABIF file, performs quality trimming if enabled, and outputs the result in FASTQ format.

```
proc parseCommandLine(): Config {.raises: [ValueError], tags: [ReadIOEffect],  
                                forbids: []}.
```

Parses command-line arguments and returns a Config object.

This procedure:

- Initializes Config with default values
- Processes command-line arguments
- Validates parameter values
- Handles special flags like `-version` and `-help`

Returns: A Config object with settings based on command-line arguments

```
proc printHelp() {.raises: [], tags: [], forbids: []}.
```

Displays the help message for the abi2fq tool. Exits the program after displaying the message.

```
proc trimSequence(sequence: string; qualities: seq[int]; windowSize: int;  
                  threshold: int): tuple[seq: string, qual: seq[int]] {.  
    raises: [], tags: [], forbids: []}.
```

Trims low-quality regions from the beginning and end of a sequence.

Uses a sliding window approach to identify regions where the average quality score is below the threshold.

Parameters: sequence: The DNA sequence to trim qualities: Quality scores for each base in the sequence windowSize: Size of the sliding window for quality assessment threshold: Quality threshold (bases with qualities below this are trimmed)

Returns: A tuple containing the trimmed sequence and its quality values

```
proc writeFastq(sequence: string; qualities: seq[int]; name: string;  
                outFile: string = "") {.raises: [ValueError, IOError],  
                                       tags: [WriteIOEffect], forbids: []}.
```

Writes sequence and quality data to a FASTQ file.

If outFile is empty, the FASTQ data is written to stdout.

Parameters: sequence: The DNA sequence to write qualities: Quality scores for each base in the sequence name: The sample name for the FASTQ header outFile: Path to the output file (empty string for stdout)