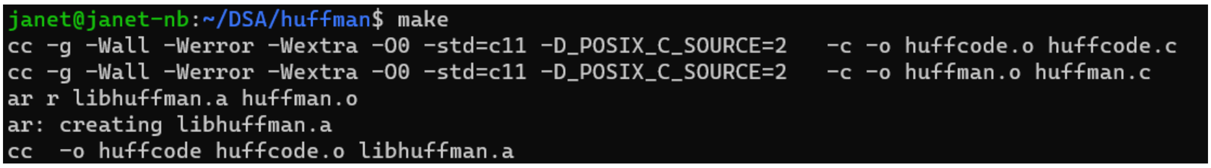
1.Usage

Huffman

- **MakeFile**

  $make



```
janet@janet-nb:~/DSA/huffman$ make
cc -g -Wall -Werror -Wextra -O0 -std=c11 -D_POSIX_C_SOURCE=2   -c -o huffcode.o huffcode.c
cc -g -Wall -Werror -Wextra -O0 -std=c11 -D_POSIX_C_SOURCE=2   -c -o huffman.o huffman.c
ar r libhuffman.a huffman.o
ar: creating libhuffman.a
cc   -o huffcode huffcode.o libhuffman.a
```

- **Run for encode**

  $./huffcode -i (input_filename) -o (output_filename) -c

- **Run for decode**

  $./huffcode -i (input_file_name) -o (output_file_name) -d

Arcd

- **Makefile**

  $make

- **Run for encode**

- **Run for decode**

2.Result

Conclusion

1.Huffman coding runs faster when processing big size data.
2.Arithmetic coding runs faster when processing small size data.
3.Both coding runs faster when decoding, slower when encoding.