

# Test report

## Test level 1

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta
Error, invalid number of arguments input.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta 13 1 123
Error, invalid number of arguments input.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>_
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level12 16S.fasta 13 1 123
Error, invalid level input.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

If the file name does not exist in current folder, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16Ss.fasta 13 1 123
Error, cannot find data file 16Ss.fasta
No data inputted.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

If the line number is an even number, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta 14 1
Error, the line number must be an odd number.
```

If the line number or the number of sequences users input exceeds the lines of searching file, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta 14 1000000000000000
Error, line number or number of lines is too large.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta -14 1000000
Error, the line number and number of lines must be positive integer.
```

If the third or fourth argument is not an integer, the program will raise an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta a b
Error, please enter integers for line number and number of lines.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>_
```

Display result:

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level1 16S.fasta 3 1
>NR_118899.1 Actinomyces bovis strain DSM 43014 16S ribosomal RNA, partial sequence
GGGTGAGTAACACGTGAGTAACCTGCCCNACTTCTGGATAACCGCTTGAAGGGTNGCTAATACGGGATATTTGGCCTGCTCGCATGGGTGGGTTTGAAGGTTCTTTTCTGGTT
GGGGATGGGCTCGCGGCTATCAGCTTGTGGTGGGTGATGGCTACCAAGGCTTTGACGGGTAGCCGGCTGAGAGGGTGGACGGCCACACTGGGACTGAGACACGGCCAGACTCCT
ACGGGAGGACAGCAGTNGGAATATTGCAATGGGCGCAAGCCTGATGCAGCAGCCGCGTGAAGGATGAGGCCTTCGGGTTGTGAACCTTTTCGCCAGTGAAGCAGGCCGCTCTT
TGTGGGGTGGTTGACGGTAGCTGGATAAAGAGCGCCGCTAACTACGTGCCAGCAGCCGCGGTAATACGTAGGGCGCNAGCGTTGTCCGGAATTATTGGGCGTAAAGAGCTNGTAGGCGG
CTNTCGGCTCTNTCGTNAATCCTCTNGCTNAACTGGAGGCTTNCGGTGGGTACGGGCGGCTNAGATGCGGTAGGGGAGACTGGAATTCCTGGTGTAGCGGTGGAATGCCAGATATCA
GGAGAAACACCGGTGGCGAAGCGGGTCTCTGGCCGGTACTGACGCTGAGGAGCGAAAGCGTGGGGAGCGAAGGATTAGATACCTCGGTAGTCCAGCCGTAAACGTTGGGCACTAG
GTGGGGGCTCTTTCCGGGGTTTCCGGCGGTAGCTAACGCATTAAAGTCCCGGCTGGGGAGTACGGCCGAAGGCTAAAGCTCAAAGGAATTGACGGGNNCCGCAACGCGCGGAGG
ATGCGGATTAATTCGATGNAACGCGAAGAACCTTACCAAGGCTTGACATGTGGGTGTTGATCAGCGAGACGTGGTTTCTCTCGGAGCGCCTTACAGGTGGTGCATGTTGTCTGTCAGC
TCGTGTCGTGAGATGTTGGGTTNAGTCCCGAACGAGCGCAACCTNGTCNCGTGTGTCAGCAGCTGTTGGTGGGACTCGCGGAGACTGCCGGGTNNACTCGGAGGAAGGTGGGGA
TGACGTCGAACCATCATGCAACCTTATGCTTGGGCTTCAAGTGTCTCAATGGCTGATCAGAGGGTGGGATGTCTNAGGCGGAGCGAATCCCTTAAAGCCAGTCTCAGTTCCGATC
GGTGTCTGCAACTCGACACCGGTGAAGTNGGAGTGCCTAGTAAATCGAGATCAGCNACGCTGCGGTGAATACGTTCTCGGGCTTGTACACACCGCCCTCAGCNNAATAAGCTGGCAAC
ACCCGAAGCCGCTGGCCTTATGGGGAGCGGTGAAGGTNGGGCTNGT
```

Data in 16S.fasta file:

```
1 |>NR_118899.1 Amycolatopsis azurea strain NRRL 11412 16S ribosomal RNA, partial sequence
2 |GGTCTNATACCGGATATAACACCTCATGGCATGGTGTAGTGGAAAGCTCCGGCGGTACGGGATAGCCCGGCTATCAGCTTGTGGTGGGTAATGGCTACCAAGGCGACAGCGGTAGCCGCTGAGAGGGTGAACGGCNACACTGGGA
3 |CTGAGACACGGCCNAGACTCTACNGGAGGAGCAGTGGGGAATATTGCNCAATGGGCGAAAGCCTNATGCAGCAGCCGCGTGAAGGATGACGGCTTCGGGTTGTAACCTTTTCGCCAGGAGCAAGCGCAAGTGACGGTACCTGGAGAAGAA
4 |GCACCGGCTAAGCTAGCTGCCAGCAGCGCGGTAATACGTAGGGTGCAGCGTTGTCCGGAATTACCGGGCTAAAGAGCTNGTAGGCGGTTGTCCGCTNGTTCGTGAAACTCCACGCTNAACTGTAGCGTGCGGCGATACGGGAGACTN
5 |TCCGTTAGGGGAGACTGGAATTCCTNGTGTAGCGGTGAATGCGCAGATATCAGGAGGAACACCGGTTTGAAGGCGGNNCTCTGGGCGGACTGACGCTGAGGAGCGAAGCGTGGGAGCGAAGGATTAGATACCTGGTAGTCCAGCTGT
6 |AAACGTTGGGCGCTAGGTGTGGCGACATCCAGCTGTGCGGTAGCTAACGCATTAAAGCGCCGCTGGGGAGTACGGCCGAAGGCTAAAGCTCAAAGGAATTGACGGGNNCCGCAACGCGCGGAGGATGGAATTAATTCGATGNA
7 |ACCGCAAGCACTTACCTGGGCTGACGTGCGCYAGACCTCCCTAGAGATGGGGTCTCCCTTGTGGTGGTGTACAGGTGTCATTCGTCGTGAGATGTTGGGTTAAGTCCCAGCAGCGCAACCTNATCTCAGTTTGC
8 |CAGCGCTTATGGCGGAGACTCGTGGGAGACTGCCGGGTNAACTCGAGGAAGGTGGGATGACGTCAAGTCAATGTCCTTATGTCAGGGCTTACACATGCTACAATGGCTGATACAGGGGCTCGGATACCGCGAGGTNGAGCGAATCC
9 |TNAAGCGGCTNAGTTCGGATCGAGCTGCAACTGACGTGAGTGGGATCGGTAGTAAATCGCAGATCAGCAACGCTGCGGTAATACGTTCCCGGCTTGTACACACCGGNNCTTACGGCATGAAAGNCGTAACACCGCAACCCATG
10 |GCCCAACCGCTAAGGGGGAGTNGTGAAGTGGGACTTCCAT
11 |>NR_118899.1 Actinomyces bovis strain DSM 43014 16S ribosomal RNA, partial sequence
12 |GGGTGAGTAACACGTGAGTAACCTGCCCNACTTCTGGATAACCGCTTGAAGGGTNGCTAATACGGGATATTTGGCCTGCTCGCATGGGTGGGTTTGAAGGTTCTTTTCTGGTGGGGATGGGTCGCGGCTATCAGCTTGTGGTGGG
13 |TGATGGCTTACCAGGCTTTGACGGGTAGCGGCTGAGAGGGTGGACGGCCACACTGGGACTGAGACAGCCGCGACTCTACGGGAGGACAGCAGTNGGAATATTGCAATAGGGCGCAAGCTGATGACAGCGACCGCGGTGAGGGATGAGG
14 |CCTTCGGGTTGTGAACCTTTTCGCCAGTGAAGCAGCGCCCTCTTGTGGGTTGGTTGACGGTAGCTGGATAAAGAGCGCCGCTAAGTACGTGCCAGCAGCGCGGTAATACGTAGGGCGCNAGCGTTGTCCGGAATTATTGGGCGTAAGAGCT
15 |NCTAGGCGGCTNCTCGGCTCTNTCGTNAATCCTCTNGCTNAACTGGAGGCTTNCGGTGGGTACGGGCGGCTNAGATGCGGTAGGGGAGACTGGAATTCCTGGTGTAGCGGTGGAATGCCAGATATCAGGAGGAACACCGGTGGCGAAGGCGGGT
16 |TCTGGGCGCTAGCTGACGCGTAGGAGCGAAGGCTGGGAGCGAAGGATTAGATACCTGAGTCCAGCGCGTAAGCTTGGGCACTAGGTGGGGGCTCTTCCGGGTTTCCGCGCGCTAGCTAACGCATTAAAGTCCCGCTGGGGAGTAC
17 |GGCCGAAGGCTAAACTCAAAGGAATTGACGGGNNCCGCAACGCGCGGAGCATCGGATTAATTCGATGNAACGCGAAGAACCTTACCAAGGCTTGACATGTGGGTGTGATCAGGAGACGTGGTTCTCTTCGGAGCGCTTACAGGTGG
18 |TGATGTTGTGCTGAGCTGCTGCTGCTGAGATGTTGGGTTNAGTCCCGAACGAGCGCAACCTNCTNCGTGTTGCCAGCAGCTNGTGGTGGGACTCGCGGAGACTGCCGGGTNNACTCGGAGGAAGGTGGGATGACGTCAATCATCATG
19 |CCCTTATGTTGGGCTTACGATGCTACAATGCTGTCAGAGGTTGCGATGTCGTAAGGCGGAGCGAATCCCTTAAAGCCAGTCTCAGTTCCGATCGGTGTCTGCAACTGACACCGGTGAAGTNGGAGTCCGTAGTAAATCGCAGATCAGCNA
20 |CGCTCGGCTGATGAGCTTCTGGGCTTGTACACACCGCTCAGCNNAATNAGCTGACACACCGGCGGCTTATGGGAGCGGCTGAGGTTNGGCTNGT
21 |>NR_094334.1 Aerhaemophilus fulgidus DSM 4304 16S ribosomal RNA, complete sequence >NR_118979.1 Aerhaemophilus fulgidus DSM 4304 strain VC-16 16S ribosomal
22 |RNA, complete sequence >NR_119237.1 Aerhaemophilus fulgidus DSM 4304 strain VC-16 16S ribosomal RNA, complete sequence
23 |ATTCGTTGATGCTGTCAGAGCGCGCTGATCTCCGCTGGGACTGAAGCATCGGCTCAGGGGCTTGTATCCCTTCGGGATGCAAGCACCGCGGAGCGCTCAGTAAACGTGCAACACTCGCTCCGCTGGGCGGATAACCCCGGAACCTCG
24 |GGCTAATCCCCATAGGGATGGGACTAGGATGCTCCATCTCCGAAGCGCTTAGGCGCCGAGGATGGTCTGGCGCGATAGGTTGTGGTGGGTAAAGCGCCACCAAGCGAAGATCGTACGGGCGATGAGAGTGGAGCCCGGAGATGGA
25 |CGCTGAGACAGGGGCTCCAGGCTTACGGGCGCAGCAGCGCGGAACCTCGCAATCGGGGAACCGGACGGGTCAGCGGAGTGTCCGCACTCGCGGCGCTGCGGGGTGCTAAGAGCACTCCACAGCAAGGCGCGGAGGCGCGTGG
26 |CAGCGCGCGGTAAATACCGCGCGCCAGTGGCGGCACTTATTGGGCTTAAGCGCTGAGTACCGGCTGGTAAAGTCTCCGGGAATCTGGGCTTAAACGTCAGACTGCCGGAGGATGCGAGCTAGGAGCCGGAAGGCGGAGGCGGAGG
27 |TATTCGGGAGTGGGGTGAATCTGTGAATCCCGGAGGACACCTGTGGCAGGCGCCGCTGGAGCGGCTGACGGGTGAGGAGCAAGGCGAGGGAGCGAAGCGGATTAGATACCGGGGTAGCTCTGGCTGTAAACGATCGGAGTAGT
28 |GTACCGAGACTAGGCTGCTGGTGTGTCGGGCGGAGCGCTAGTGTGCTGCTGGGCTAGGCGTGAAGGCTGAACCTAAGGGAATGCGGCGGAGCACTACAGCGGCTGGGCTGCGGTTAATGTGATTAAGCTCGGAGCTTA
```

## Test level 2

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level2 16S.fasta
Error, invalid number of arguments input.
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level21 16S.fasta aasd
Error, invalid level input.
```

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

If the file name does not exist in current folder, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level2 16Ss.fasta aasd
Error, cannot find data file 16Ss.fasta
No data inputted.
```

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

Displaying to the console a suitable error message if sequence is not found.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level2 16S.fasta aasd
Error, sequence aasd not found.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level2 16S.fasta NR_1991
Error, sequence NR_1991 not found.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

Display result:

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level2 16S.fasta NR_118899.1
>NR_118899.1 Actinomyces bovis strain DSM 43014 16S ribosomal RNA, partial sequence
GGGTGAGTAACACGTGAGTAACCTGCCCNNACTTCTGGATAACCGCTTGAAAGGGTNGCTAATACGGGATATTTGGCCTGCTCGCATGGGTGGGTTTGGAAAGGTTCTTTTCTGGTT
GGGGATGGGCTCGCGGCTATCAGCTTGTGGTGGGGTGATGGCTTACCAAGGCTTTGACGGGTGAGAGGGTGACGGCCACACTGGGACTGAGACACGGCCACAGACTCCT
ACGGGAGGCAGCAGTGNNGAATATTGCACAATGGGCGCAAGCCTGATGACGACGACGCCGCTGAGGGATGTAGGCCTTCGGGTTGTGAACCTCTTTCGCCAGTGAAGCAGGCCCTCTT
TGTGGGGTGGTTGACGGTAGCTGGATAAGAAGCGCCGCTAAGTACGTCAGCAGCGCGGTAAATACGTAGGCGCGNAGCGTTGTCCGGAATTATTGGGCGTAAAGAGCTNGTAGGCGG
CNTNCGGCTCNTCGTNAAACTCCTTNGCTNAACTGAGGCTTNCGGTGGGTACGGGCGGCTNGAGTGGGTAGGGGAGACTGGAATTCCTGGTGTAGCGGTGGAATGCGCAGATATCA
GGAGGAACACCGGTGGCGAAGCGGGTCTCTGGGCGGTACTGACGCTGAGGAGCGAAAGCGTGGGAGGCGAAGCAGGATTAGATACCTGGTATCCAGCGCTTAAACGTTGGGCACTAG
TGCGGGGCTCTTTCGGGGTTCGCGCCGCTAGCTAACGCAATTAAGTGCCTCGGCTGGGAGTACGGCGCAAGGCTAAACTCAAAGGAATTGACGGGNCCTCGCACAAAGCGCGGAGC
ATGCGGATTAATTCGATGNAACGCGAAGAACCTTACCAAGGCTTGACATGTGGGTGTTGATCACGGAGACGTGGTTTCTCTCGGAGCGCTTACAGGTGGTGCATGGTTGTCTGCTCAGC
TCGTGTCGTGAGATGTTGGGTTNAGTCCCACAGAGCGCAACCTNGTNCNGTGTGGCAGCAGTNGTGGTGGGACTCGCGGAGACTGCGGGGTTNNACTCGGAGGAAGGTGGGGA
TGACGTCAAATCATCATGCCCTTATGTCTTGGGCTTACGATGCTACAATGGCTGGTACAGAGGGTTCGATGTCTNAGGCGGAGCGAATCCCTTAAAGCCAGTCTCAGTTCCGGATC
GGTGTCTGCAACTCGACACCTGAAGTNGAGTGCATAGTAATCGCAGATCAGCNACGCTGCGGTGAATACGTTCTCGGGCTTGTACACACCGCCCTCACGNNATNAAAGCTGGCAAC
ACCGGAAGCCGCTTATGGGAGCGGTGGAAGGTGGGCTNGT

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

Data in 16S.fasta file:

```
1 >NR_118899.1 Actinomyces bovis strain DSM 43014 16S ribosomal RNA, partial sequence
2 GGTCTNATACCGGATATAACAACTCATGGCATGGTTGGTAGTGGAAGCTCCGGCGGTACGGGATAGCCCGCGCTATCAGCTTGTGGTGGGTAATGGCTACCAAGGCGACGCGGTAGCCGCTGAGAGGGTGACCGGCACACTGGGA
3 CTGAGACACGGCCNAGACTCCTACNGGAGNAGCAGTGGGGAATATTGCNCAATGGCGAAAGCCTNATGACGACGACCGCGCTGAGGGATGACGGCTTCGGTTGTAAACCTTTTTCGCCAGGAGCAAGCGCAAGTGACGTTACCTGGAGAAGAA
4 GCACGGCTAATACGTGCCAGCGCCGCTAATACGTAGGGTGCAGCGTGTTCGGGAATACCGGGCTAAAGAGGCTTAGGCGGTTGTCTCGGTTGTTCTGTGAAACCTCACGCTNAACTGTGAGCGTCCGGGCGATACGGGAGACTNGAG
5 TCCGTTAGGGAGAGCTGGAATTCCTGTTAGCGGTGAATGCGCAGATATCAGGAGGAACACCGGTTTGAAGGCGGNNCTTCGGGCGGATACGACGCTGAGGAGCGAAGCGTGGGAGCGAAGAGGATTAGTATCCCTGGTAGTCCAGCGTGT
6 AAACGTTGGGCGCTAGGTGGGCGACATCAGCTGTGCTGGTGTAGCTAACGCATTAAAGCGCCGCTGGGGAGTACGGCGCAAGGCTAAACTCAAAGGAATTGACGGGCGCCGCAAGCGCGGAGCATGTGGAATTAATTCGATGNA
7 CCGGAAGAACCTTACCTGGGCTGACGTGCGYAGACCTCCCTAGAGATGGGCTTCCTTGTGGTTGGTGTACAGTGGTGCATTCGTGCTGAGTCTGTTGTTGAGATGTGGGTAAAGTCCCGCAAGCGCAACCTNATCTCAGTTG
8 CAGCGGTTATGGCGGAGCTCTGGGAGACTGCGGGGTNAACTCGGAGGAAGGTGGGATGACGTCAAGTCAATCATGCCCTTATGTCAGGGCTTCACACATGCTACAATGGCTGGTACAGAGGGCTGCATACCGGAGGTNGAGCAATCC
9 TNAAGCCGCTTNAAGTTCGGATCGGAGCTGCAACTTGAAGTGGGTGAGTGCATAGTAATCGCAGATCAGCAACGCTGCGGTNAACTACGTTCCCGGGCTTGTACACACCGGNCCTTACGGCATGAAGNCGGTAAACCCGGAACCTATG
10 GCCCAACCGTAAAGGGGAGTNGTCAAGGTGGGCTNAGT
11 >NR_118899.1 Actinomyces bovis strain DSM 43014 16S ribosomal RNA, complete sequence
12 GGTGAGTAACACGTGAGTAACCTGCCCNNACTTCTGGATAACCGCTTGAAAGGGTNGCTAATACGGGATATTTGGCCTGCTCGCATGGGTGGGTTTGGAAAGGTTCTTTTCTGGTTGGGATGGGCTCCGGGCTATCAGCTTGTGGTGGG
13 TGATGGCTACCAAGGCTTTGACGGGTAGCGGCTGAGAGGGTGACGGCCACACTGGGACTGAGACAGGCGCGAGCTCTACGGGAGGACAGTNGGAATATTGCACAATGGCGCGAAGCTGATGACGACGCGCGCTGAGGATGTAGG
14 CTTTCGGGTTGTGAACCTCTTTCGCGAGTGAGCAGCGCCCTCTTGTGGGTTGGTGTACGCTGATGAGAGCGCGCGCTAATACGTAGGCGCGNAGCGTTGTCCGGAATTATTGGGCGTAAGAGCT
15 NGTAGGCGGCTNCCGCTNCTGCTNAACTCCTTNGCTNAACTGGAGGCTTNCGGTGGGTACGGGCGGCTNGAGTGGGTAGGGAGACGTGAATTCCTGGTGTAGCGGTGGAATGCGCAGATATCAGGAGGAACACCGGTGGCGAAGCGGGTCT
16 TCTGGGCGGTACTGACGCTGAGGAGCGAAGGCTGGGAGCGCAAGGATTAGATACCTGGTAGTCAAGCGGTAAAGCTTGGGCACTAGTGGGCGCTCTTCGGGGTCTTCGCGCGCTAGCTAACGCATTAAAGTCCCGCTGGGAGTAC
17 GCGCGAAGGCTAAACTCAAAGGAATTGACGGGNCCTCGCAAGCGCGGAGCATCGGATTAATTCGATGNAACCGGAAGAACCTTACCAAGGCTTGACATGTGGGTGTGATACGAGAGAGTGGTTCTCTTCGGAGCGCTTACAGGTGG
18 TGCATGGTGTGCTGACGCTGTTGCTGAGATGTGGGTTNAGTCCCGCAAGCGCGCAACCTNGTNCNGTGTGGCAGCAGTNGTGGTGGGACTCGCGGAGACTGCGGGGTTNNACTCGGAGGAAGGTGGGATGACGTCAAATCATCATGC
19 CCCTTATGCTTGGGCTTCAACGCTCTACACACCGCCCTCACGNNATNAAAGCTGGCAACACCGAAGCGCTGGGCTTATGGGAGCGGTGGAAGTNGGCTNGT
20 >NR_074334.1 Archaeoglobus fulgidus DSM 4304 16S ribosomal RNA, complete sequence >NR_118899.1 Actinomyces bovis strain DSM 43014 16S ribosomal
21 RNA, complete sequence >NR_119237.1 Archaeoglobus fulgidus DSM 4304 strain VC-16 16S ribosomal RNA, complete sequence
22 ATTCGTGTTGATCTGCGCAGAGGCTGCTGCTATCCGCTGGGACTAAGCCATCGGAGTCAAGGGCTTGTATCCTTCGGGATGAGCAAGCAGCGCGGACGCTCAGTAACACGTGGACAACCTGCCTCGGGTGGGAGTAACCCGGGAACCTGG
23 GGCTAATCCCCATAGGGATGGGTACTGGAATGTCCATCTCCGAAGCGCTTAGGCGCCGAGGATGGGTTCGCGCGGATAGGTTGTTGGTGGGTAAACGGCCACCAAGCGAAGATCCGTACGGGCTAGAGAGTGGGAGCCCGGAGATGGA
24 CCCTGAGACACGGGTCCAGGCTTACGGGCGCGCAGCAGCGCGAAGCTCCGAATGCGGGAACCGGAGCGGGTCAAGCGGATGCTCGGATCGCGCGGCTGTCTGGGCTGCCCTAAAGGACCCACAGCAAGGCGCGGCGAGGCGCGTGG
25 CAGCGCGCGGTAATACCGCGCGCGGAGTGGCGGCACCTTTTATGGGCTAAAGCGTCTGAGCGGGCTGGTAAAGTCTCCGGGAAATCTGGCGGCTTAACTGACGCTGCGGAGGATCTGCGAGCTAGGAGCCGGGAGGCGCGGGG
26 TATTCGCGAGTAGGGGTGAATCTCTGTAATCCCGGAGGACACCTGTGGCGAAGCGCGCGCTGGAACGGGTCCGAGGTTGAGGAGCAAGGCGAGGGGAGCGAAGCGGATTAGATACCGGGTATGCTGGCTGTAAACATGCGGACTAGT
27 GTACGCAAGCTAGGAGCTTGGGTTGCTGGGAGGAGGCTTAAAGTCCGCGCTGGGAGTGGGCGAAGGCTGAAGTAAAGGAATTGGGCGGAGCACTAGCAAGCGGTGGGCTTGGGTTTAAATGGATTCAGGCGCGGAGCTTA
```

## Test level 3

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level3 16S.fasta
Arguments, invalid number of arguments input.
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level31 16S.fasta
Error, invalid level input.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

If the query file name does not exist in current folder, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level3 16S.fasta query123.txt result.txt
Cannot find file 'query123.txt'

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>
```

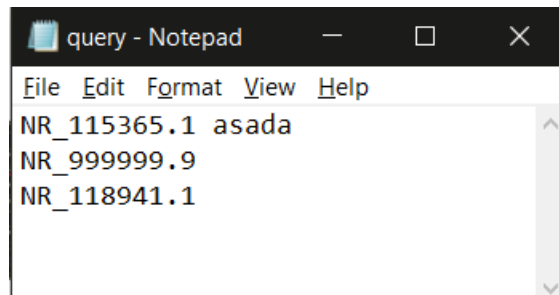
If the input or output file is not correctly formatted, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level3 16S.fasta query.txt result.tx
The output or input file has invalid format.
```

Display result:

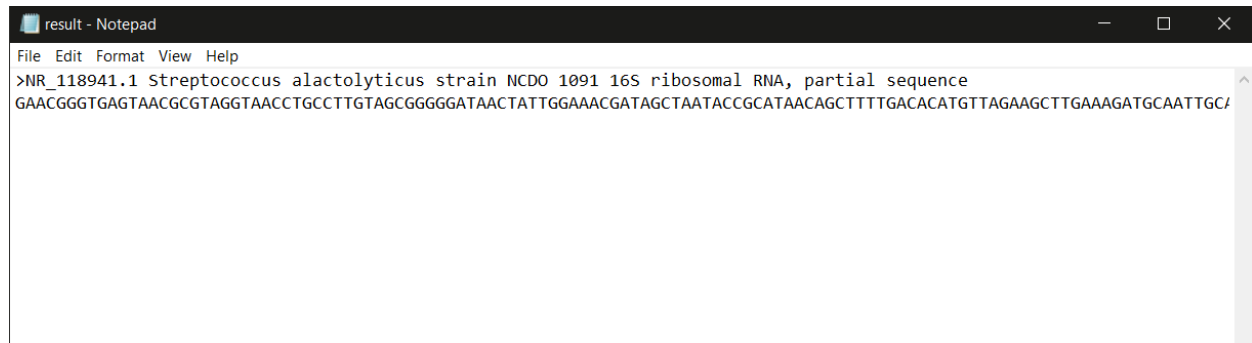
```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level3 16S.fasta query.txt result.txt
Error, sequence 'NR_115365.1 asada ' not found.
Error, sequence 'NR_999999.9' not found.
```

query.text file:



```
query - Notepad
File Edit Format View Help
NR_115365.1 asada
NR_999999.9
NR_118941.1
```

Result.text file:



```
result - Notepad
File Edit Format View Help
>NR_118941.1 Streptococcus alactolyticus strain NCDO 1091 16S ribosomal RNA, partial sequence
GAACGGGTGAGTAACGCGTAGGTAACCTGCCTTGAGCGGGGATAACTATTGGAACGATAGCTAATACCGCATAACAGCTTTGACACATGTTAGAAGCTTGAAAGATGCAATTGC/
```

## Test level 4

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level4 16S.fasta query.txt result.txt
Error, invalid number of arguments input.
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level41 16S.fasta query.txt result.txt
Error, invalid level input.
```

If the query file name does not exist in current folder, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level4 16S.fasta 16Ss.index query.txt result.txt
Error, cannot find file '16Ss.index'

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level4 16S.fasta 16S.index query123.txt result.txt
Error, cannot find file 'query123.txt'
```

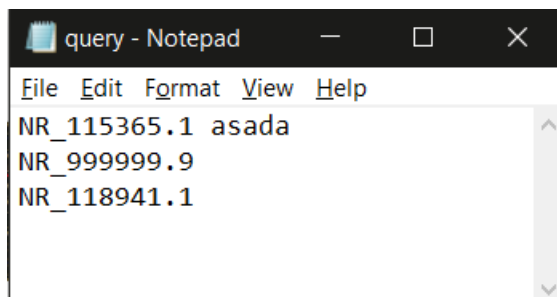
If the input or output file is not correctly formatted, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level4 16S.fasta 16S.index query.txt result.txt
Error, sequence 'NR_115365.1 asada ' not found.
Error, sequence 'NR_999999.9' not found.
```

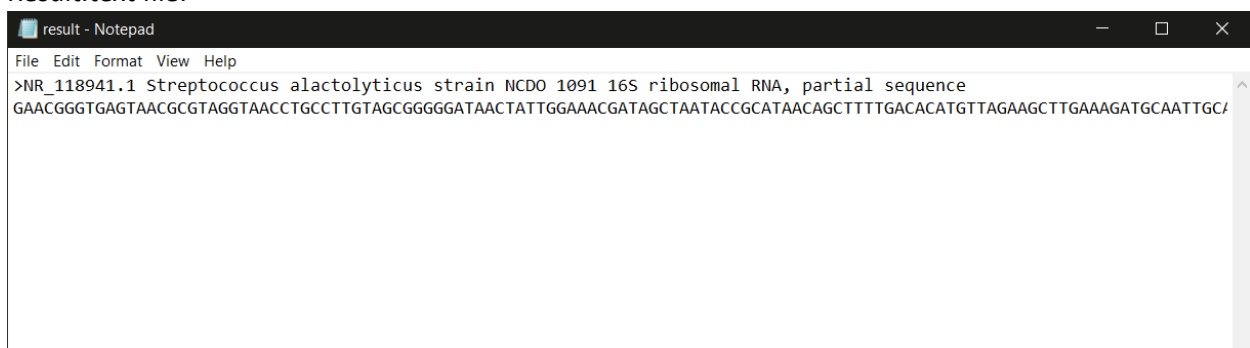
Display result:

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level3 16S.fasta query.txt result.txt
Error, sequence 'NR_115365.1 asada ' not found.
Error, sequence 'NR_999999.9' not found.
```

query.text file:



Result.text file:





## Test level 5

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level5 16S.fasta asd ad
Error, invalid number of arguments input.
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level51 16S.fasta asd
Error, invalid level input.
```

Display result:

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level5 16S.fasta CTGGTACGGTCAACTTGCTCTAAG
No sequence was found.

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level5 16S.fasta GGTTTCGAAGGCGGNNCTCTGGGCCGA
NR_118889.1
```

## Test level 6

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level6 16S.fasta Streome a
Error, invalid number of arguments input.
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level61 16S.fasta Streome a
Error, invalid level input.
```

Display result:

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level6 16S.fasta Archaeoglobus
NR_074334.1
NR_118873.1
NR_119237.1
NR_028831.1
NR_112618.1
NR_028166.1
NR_116822.1
NR_028937.1
NR_118374.1
NR_074522.1
NR_102885.1
NR_103948.1
NR_125486.1

D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level6 16S.fasta Arncackocon
No sequences found.
```

## Test level 7

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level7 16S.fasta ATGX AXGT
Error, invalid number of arguments input.
```

If the level name is invalid, the program will display an error.

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level71 16S.fasta ATGX*GXTTXT
Error, invalid level input.
```

Display result:

```
D:\2.QUT\New folder\Search16\Search16s\bin\Debug>Search16s.exe -level7 16S.fasta GTAAAGAGCTNGTAGCGGTTTGTGCGTNGTTCGTGAA
AACTCCAC*GTGCGGGCG
>NR_118889.1 Amycolatopsis azurea strain NRRL 11412 16S ribosomal RNA, partial sequence
GGTCTNATACCGGATATAACAACCTCATGGCATGTTGGTAGTGGAAGCTCCGGCGGTACGGGATGAGCCCGCGCCTATCAGCTTGTGGTGGGGTAATGGCCTACCAAGGCGACGACG
GGTAGCCGGCCTGAGAGGGTGACCGGCNACACTGGGACTGAGACACGGCCNAGACTCCTACNGGAGGNAGCAGTGGGGAATATTGCNCAATGGGCGAAAGCCTNATGCAGCGACGCCGCG
TGAGGGATGACGGCTTCGGGTTGTAACCTTTTTCGCCAGGGACGAAGCGCAAGTGACGGTACCTGGAGAAGAAGCACC GGCTAACTACGTGCCAGCAGCCGCGGTAACTACGTAGGGTGC
GAGCGTTGTCCGGAATTACCGGGCGTAAAGAGCTNGTAGGCGGTTTGTGCGGTNGTTTCGTGAAAACCTCCACGCTNAACTGAGCGTGCGGGCGATACGGGCAGACTNGAGTTCGGTAGG
GGAGACTGGAATTCTNGTGTAGCGGTGAAATGCGCAGATATCAGGAGGAACACCGGTTTCGAAGGCGGNNCTCTGGGCCGATACGTACGCTGAGGAGCGAAAGCGTGGGGAGCGAACAG
GATTAGATACCTCTGATGTCACGCTGTAACGTTGGGCGCTAGGTGTGGGCGACATCCACGTTTGTCCGTGCCGTAGCTAACGCATTAAGCGCCCCGCTGGGGAGTACGGCCGAAGGC
TAAAACTCAAAGGAATTGACGGGGGCCGCAAGCGGCGGAGCATGTGGATTAATTTCGATGNAACGCGAAGAACCTTACCTGGGGCTGACGTGCGCYAGACRTCCCTAGAGATGGGGCT
TCCCTTGTGGTTGGTGTACAGGTGGTGATTGCTGTCGTGCTGTCGTGAGATGTTGGGTTAAGTCCCGCAACGAGCGCAACCCTNATCCTACGTTGCCAGCGCGTTATGGCGGGG
ACTCGTGGGAGACTGCCGGGTNAACTCGGAGGAAGTGGGGATGACGTCAAGTCATCATGCCCTTATGTCCAGGGCTTACACATGCTACAATGGCTGGTACAGAGGGCTGCGATACC
GCGAGGTNGAGCGAATCCCTNAAAGCCGCTCTNAGTTCGGATCGCAGTCTGCAACTNGACTGCGTGAAGTCGGAGTCGCTAGTAATCGCAGATCAGCAACGCTGCGGTNAATACGTTCCC
GGGCTTGATACACCGGNCGTTACGGCATGAAAGNCGGTAAACCCGAACCATGGCCCAACCGTAAGGGGGAGTNGTCGAAGGTGGGACTNGCGAT
```

## Test IndexSequence16s program

If the number of arguments is invalid, the program will display an error.

```
D:\2.QUT\New folder\IndexSequence16s\IndexSequence16s\bin\Debug>IndexSequence16s.exe 16S.fasta
The number of argument is invalid
```

If the file does not exist, print error message.

```
D:\2.QUT\New folder\IndexSequence16s\IndexSequence16s\bin\Debug>IndexSequence16s.exe 16Sss.fasta 16S.index
Error, cannot find file.
```

If the file has invalid format, print error message

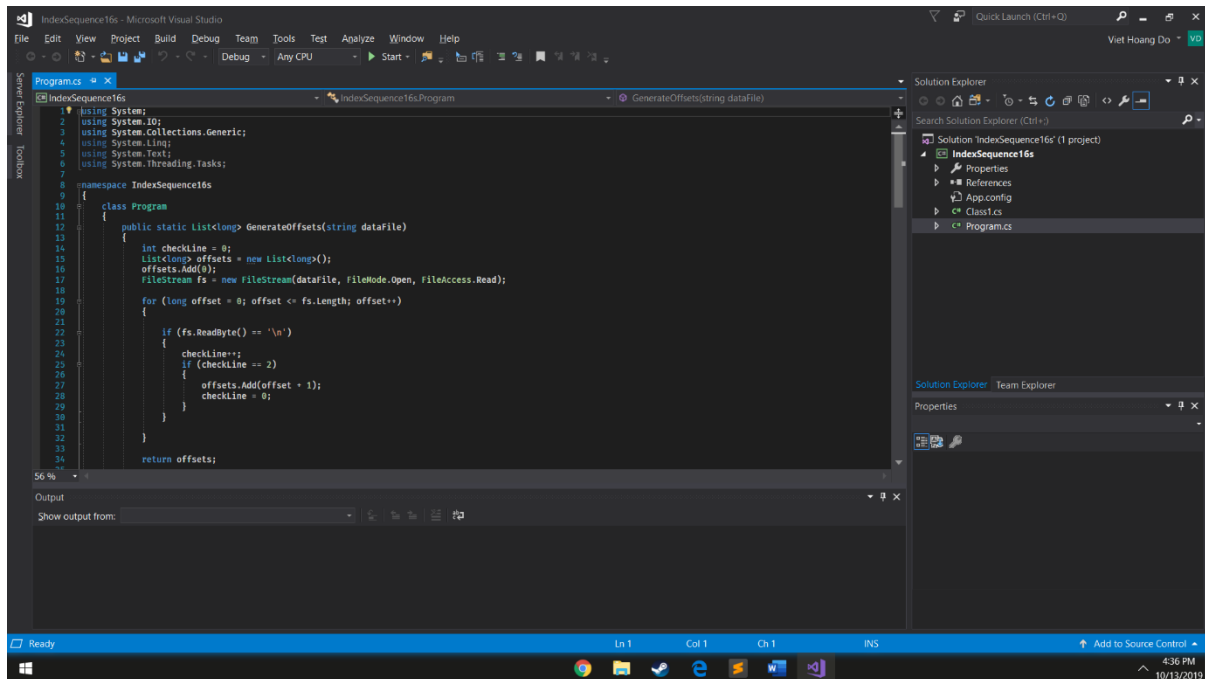
```
D:\2.QUT\New folder\IndexSequence16s\IndexSequence16s\bin\Debug>IndexSequence16s.exe 16S.asta 16S.index
Error, the files have invalid format.
```

```
D:\2.QUT\New folder\IndexSequence16s\IndexSequence16s\bin\Debug>IndexSequence16s.exe 16S.fasta 16S.ndex
Error, the files have invalid format.
```

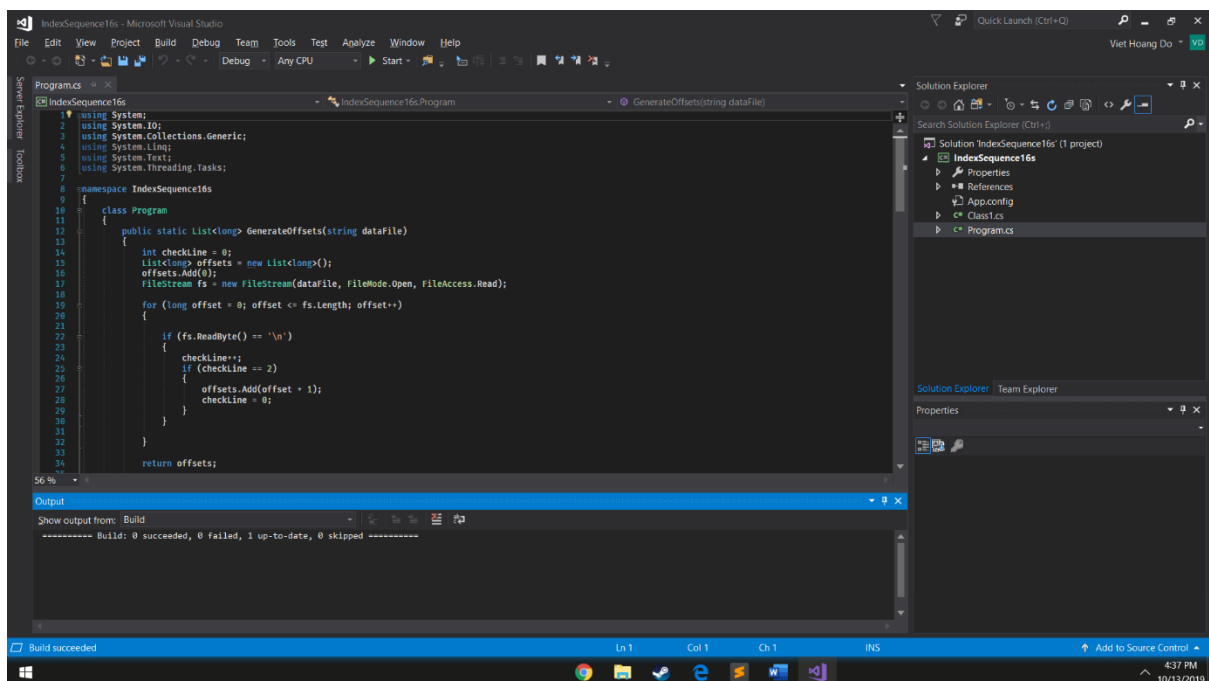
# Program user manual

## Build and run IndexSequence16S

- Open IndexSequence16s.sln file.

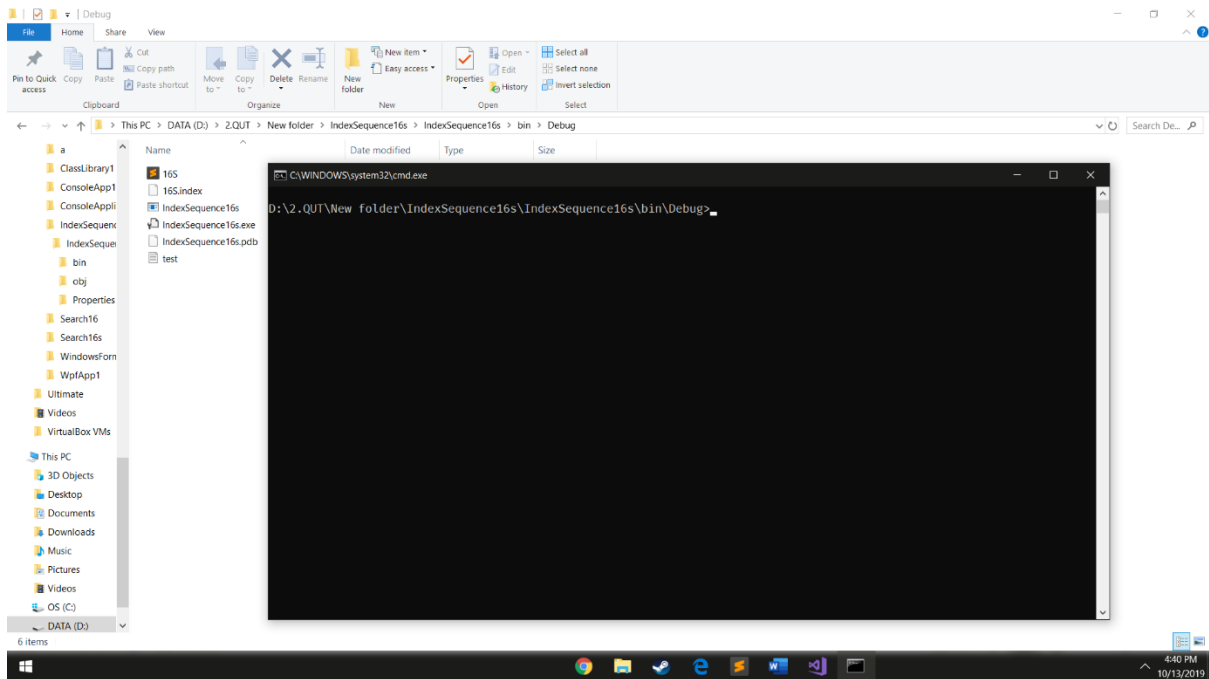


- Press Ctrl + Shift + B to build the program.

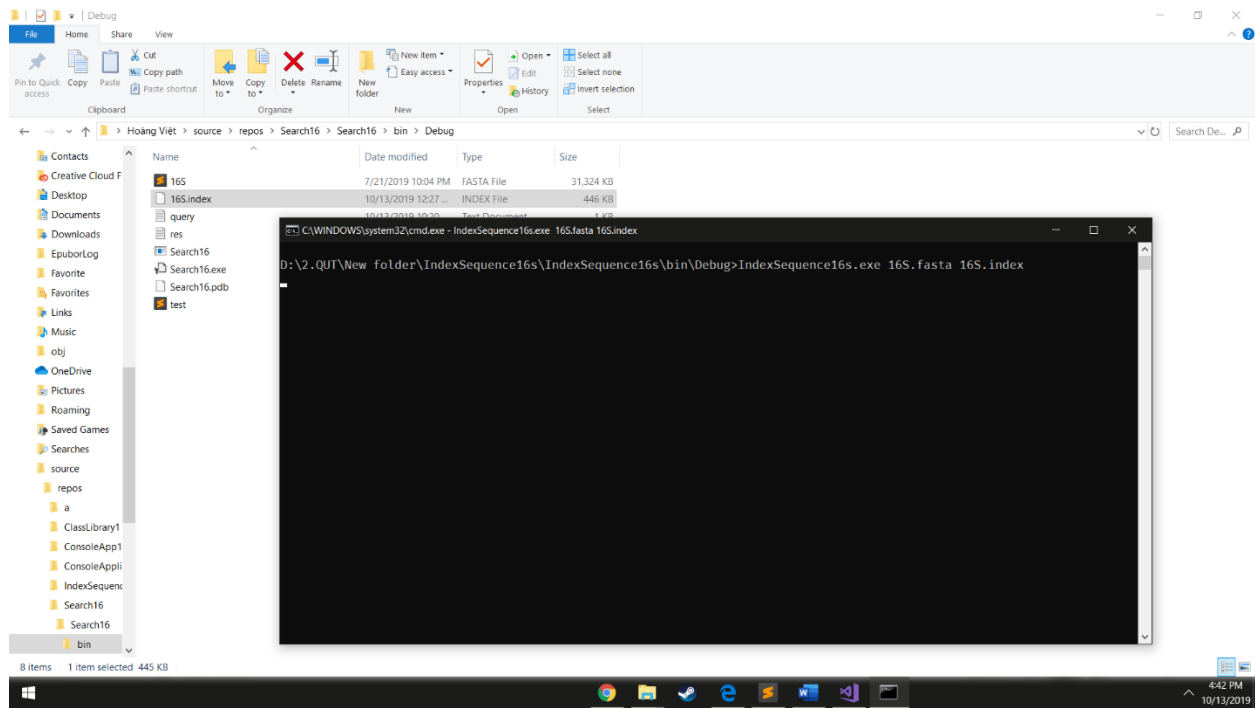




- Open CMD in folder containing .exe file (IndexSequence16s\IndexSequence16s\bin\Debug)

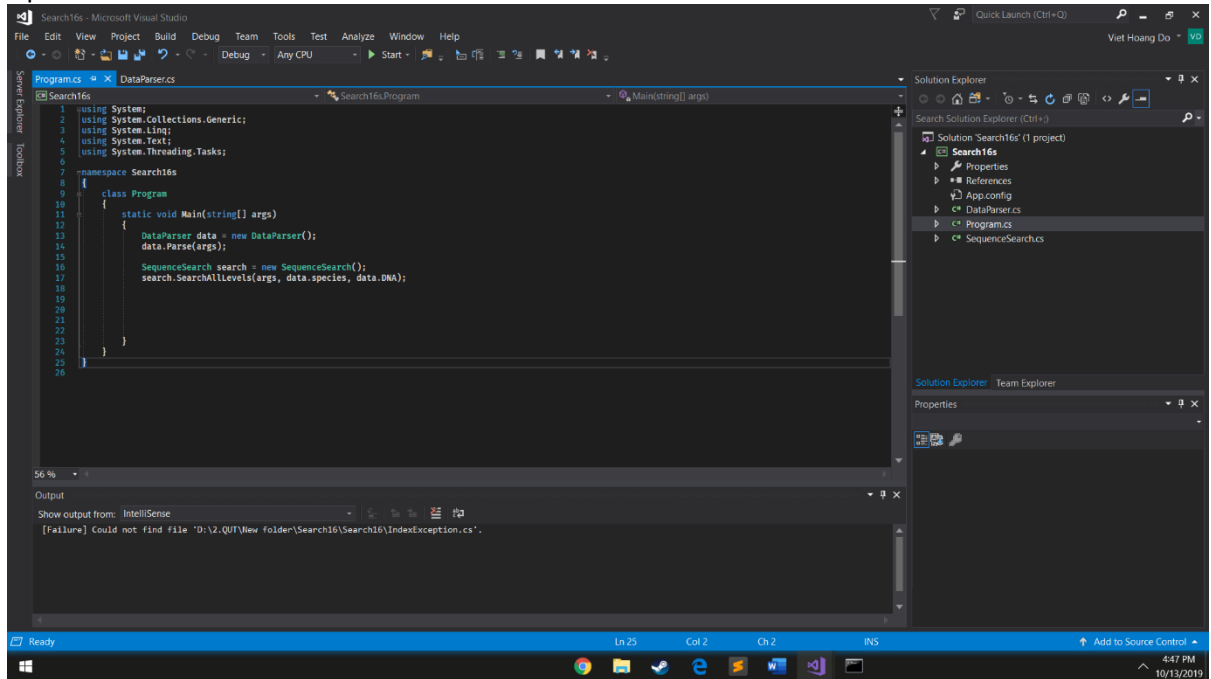


- Type `IndexSequence16s.exe 16S.fasta 16S.index` to run program

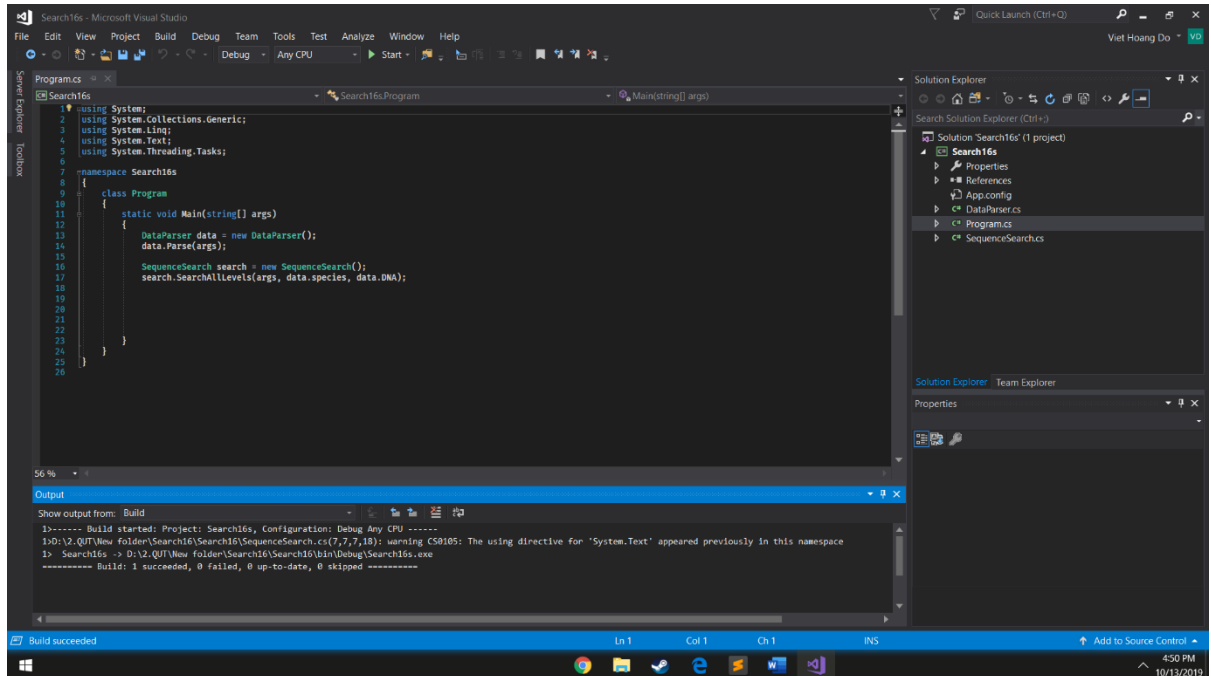


## Build and run Search16s

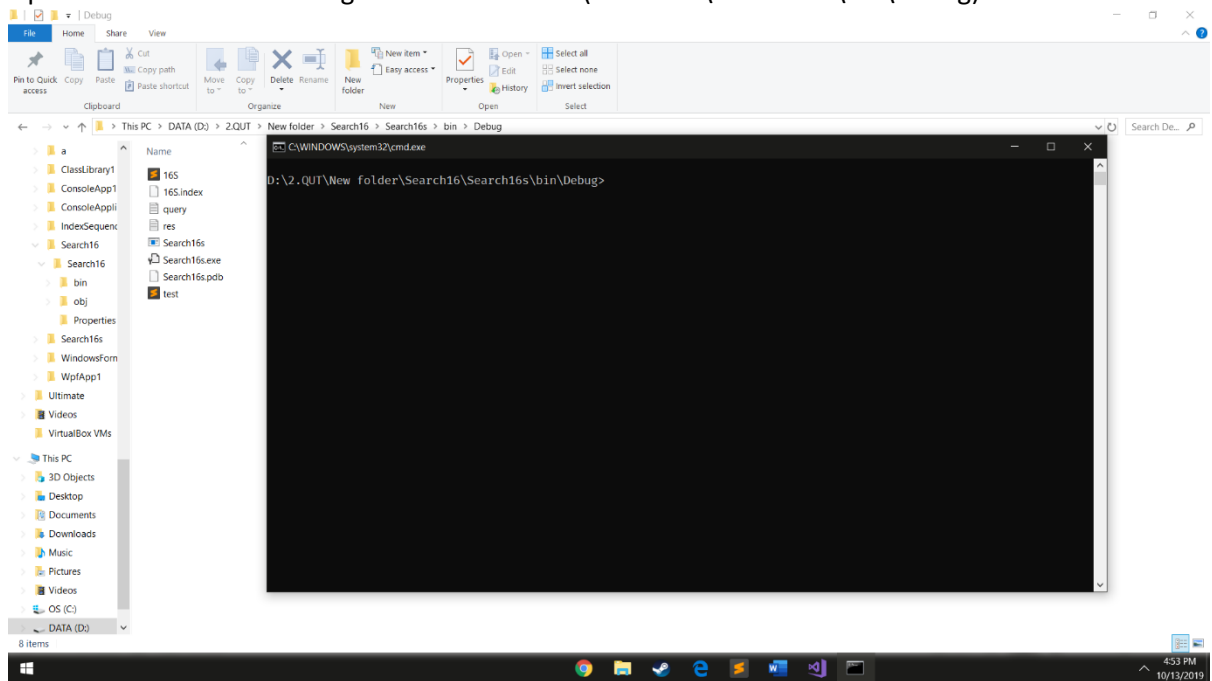
- Open Search16s.sln file



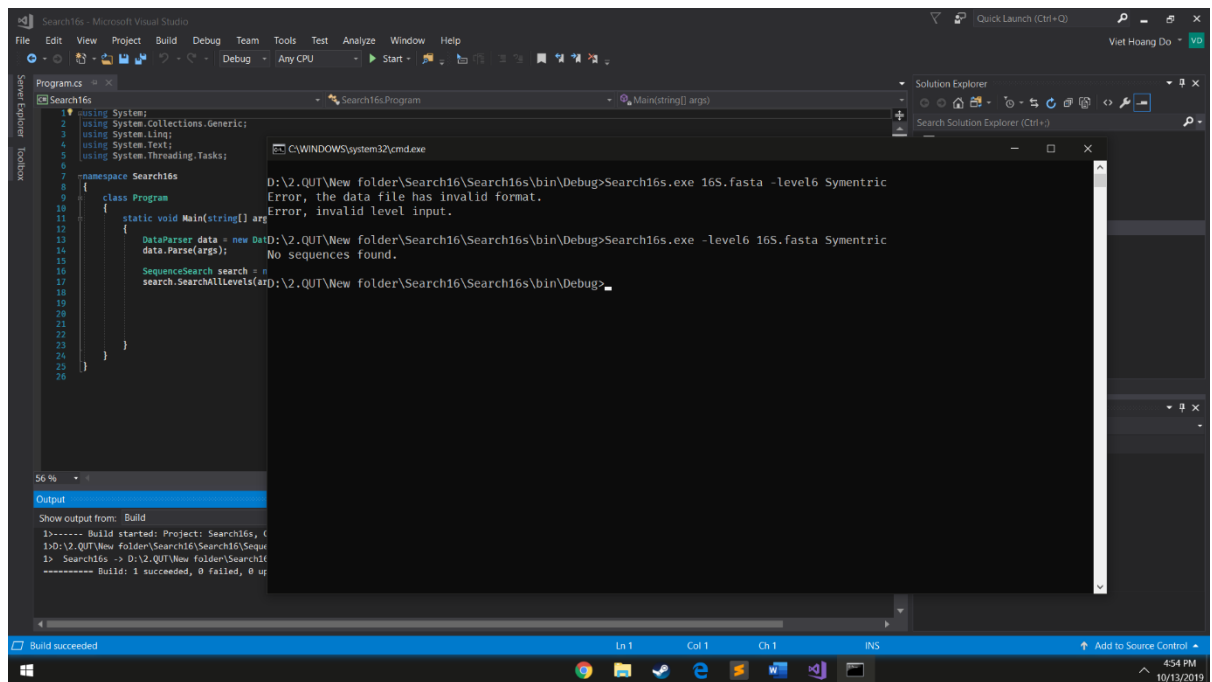
- Press Ctrl + Shift + B to build the program



- Open CMD in folder containing file Search16s.exe (Search16\Search16s\bin\Debug)



- Type correct command to run program



Open Command Prompt in folder "Search16s\Search16s\bin\Debug" to run program.

## 1. Sequential access using a starting position in the file. (level 1)

Given a starting line, the program lists the required number of sequences to the screen, starting from the given line number, and shows both lines that correspond to the sequence.

In this level, the program requires exactly 4 input arguments.

For example: Search16s -level1 16S.fasta 273 10

Search16s is the program name.

The first argument is '-level1' : a flag indicating the type of search required (in this case, level 1 is for finding a sequence by ordinal position)

The second argument is '16S.fasta': a fasta file name to read from. This file should be in same directory as the program.

The third argument is an integer '273': the line number to start the output from. This number must be an odd number because each sequence consists of 2 lines.

The fourth argument is an integer '10': the number of sequences that the program should output

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the file name does not exist in current folder, the program will raise an error.

If the line number is an even number, the program will raise an error.

If the line number or the number of sequences users input exceeds the lines of searching file, the program will raise an error.

If the third or fourth argument is not an integer, the program will raise an error.

## 2. Sequential access to a specific sequence by sequence-id. (level 2)

The program will respond by displaying to the console the respective sequence lines, or a suitable error message if not found.

In this level, the program requires exactly 3 input arguments.

For example: `Search16s -level2 16S.fasta NR_115365.1`

The first argument is '-level2' : a flag indicating the type of search required (in this case, level 2 is for finding a specific sequence by sequence-id)

The second argument is '16S.fasta': a fasta file name to read from. This file should be in same directory as the program.

The third argument is a string 'NR\_115365.1': a sequence name users want to find.

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the file name does not exist in current folder, the program will raise an error.

## 3. Sequential access to find a set of sequence-ids given in a query file and writing the output to a specified result file. (level 3)

The program writes out all matching sequences to the specified results file (not to the console).

Sequence-ids that are not found should however be output to the console with a suitable error message.

In this level, the program requires exactly 4 input arguments.

For example: `Search16s -level3 16S.fasta query.txt results.txt`

The first argument is '-level3' : the flag indicating the type of search required (in this case, level 3 is for finding a set of specific sequences by sequence-id given in a query file)

The second argument is '16S.fasta': the fasta file name to read from. This file should be in same directory as the program.

The third argument is a string 'query.txt ': the name of query file containing a set of sequence-ids.

The fourth argument is a string 'result.txt': the name of output file that contains ONLY the sequences that were found.

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the query file name does not exist in current folder, the program will raise an error.

If the output and input files do not have correct format, the program will raise an error.

#### 4. Indexed file access, implementing direct access to sequences. (level 4)

The program writes out all matching sequences to the specified results file (not to the console), but instead of using a sequential file scan, it uses the index.

Sequence-ids that are not found should however be output to the console with a suitable error message.

In this level, the program requires exactly 5 input arguments.

For example: Search16s -level4 16S.fasta 16S.index query.txt results.txt

The first argument is '-level4': the flag indicating the type of search required (in this case, level 4 is for finding a set of specific sequences by sequence-id given in a query file)

The second argument is '16S.fasta': the fasta file name to read from. This file should be in same directory as the program.

The second argument is '16S.index': the index file name to read from. This file should be in same directory as the program. This file can be generated using IndexSequence16s program

The third argument is a string 'query.txt ': the name of query file containing a set of sequence-ids.

The fourth argument is a string 'result.txt': the name of output file that contains ONLY the sequences that were found.

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the query file name does not exist in current folder, the program will raise an error.



If the output and input files do not have correct format, the program will raise an error.

## 5. Search for an exact match of a DNA query string. (level 5)

The program will display all matching sequence-ids to the console, or a suitable error message if not found.

In this level, the program requires exactly 3 input arguments.

For example: `Search16s -level5 16S.fasta CTGGTACGGTCAACTTGCTCTAAG`

The first argument is '-level5': a flag indicating the type of search required (in this case, level 2 is for finding an exact match of a DNA query string)

The second argument is '16S.fasta': a fasta file name to read from. This file should be in same directory as the program.

The third argument is a string 'CTGGTACGGTCAACTTGCTCTAAG': a DNA string that users want to find.

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the file name does not exist in current folder, the program will raise an error.

## 6. Search for a sequence meta-data containing a given word (level 6)

The program will display all matching sequence-ids to the console, or a suitable error message if not found. A matching sequence must contain the query word in the meta-data.

In this level, the program requires exactly 3 input arguments.

For example: `Search16s -level6 16S.fasta Streptomyces`

The first argument is '-level6': a flag indicating the type of search required (in this case, level 2 is for finding a sequence meta-data containing a given word)

The second argument is '16S.fasta': a fasta file name to read from. This file should be in same directory as the program.

The third argument is a string 'Streptomyces': a given word that users want to find.

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the file name does not exist in current folder, the program will raise an error.

## 7. Search for a sequence containing wild cards. (level 7)

The program will display all matching sequences to the screen, or a suitable error message if not found.

A "\*" stands for any number of characters in the same position.

In this level, the program requires exactly 3 input arguments.

For example: Search16s -level7 16S.fasta ACTG\*GTAC\*CA

The first argument is '-level7': a flag indicating the type of search required (in this case, level 2 is for finding a sequence meta-data containing a given word)

The second argument is '16S.fasta': a fasta file name to read from. This file should be in same directory as the program.

The third argument is a string 'ACTG\*GTAC\*CA': a DNA string containing wildcard that users want to find.

If the number of arguments is invalid, the program will raise an error.

If the level name is invalid, the program will raise an error.

If the file name does not exist in current folder, the program will raise an error.

## Statement of Completeness

This statement of completeness will need to *accurately* state the functionality which has been implemented. There will be a penalty of 3.5 marks (loss of 3.5 marks) for a non-completed or submitted statement of completeness, and a penalty of 1 mark for each inaccurate statement to a maximum of 3 marks.

**In the following section, you are required to mark which functionality you have implemented. In the column on the right please mark 'Y' where you have completed this functionality, and 'N' where you have not. Please fill in any additional text boxes requested, and please note any limitations or bugs in the box at the end of each section. You may expand the table if you need more room for comments.**

Basic Functionality		
<b>Build &amp; Run</b>	When following the Build & Run instructions, the program successfully builds and runs. This was tested in a QUT SEF lab with a new login session, using the same zipped folder that is submitted.	Y
<b>Basic itinerary output</b>	The program displays the data from the file	Y
	The program displays the appropriate line	Y
	The correct amount of information is displayed, e.g. only the relevant entries	Y
	The correct level, provided as a command line flag - <b><i>levelN</i></b> , is executed	Y
	The program <b>does not</b> store the whole file in memory, instead it accesses the file on disk	Y
<b>Error handling</b>	A clear error message is displayed when an incorrect number of arguments is provided	Y
	A clear error message is provided when an incorrect flag is provided (e.g. not -level1, etc.)	Y
	A clear error message is provided when the input file doesn't exist, or is incorrectly formatted	Y
<b>Comments</b>	<i>Please note any limitations, bugs, logical errors, and/or run-time errors here</i>	

### Searching Algorithm – Part I

Please underline, circle or highlight the levels that were completed.

<b>Algorithm</b>	<b>Level:</b> <u>Level 1, level 2, level 3</u>
<b>Comments</b>	<i>Some errors printed out are not correctly.</i>

### Searching Algorithm – Part II

Please underline, circle or highlight the levels that were completed.

<b>Algorithm</b>	<b>Level:</b> <u>Level 4, level 5, level 6, level 7</u>
	<b>Bonus:</b> Sequence matching using wildcards  <i>If completed, please highlight level 7 above and discuss how you approached the problem and why you solved it the way you did.</i> <ul style="list-style-type: none"><li>- <i>I used Regex built-in class to solve this problem.</i></li><li>- <i>The reason is Regex provides a simple way to search a desired string by a pattern.</i></li></ul>
	<b>Level 7 Explanation</b> <i>If you have completed the optional level 7, please include details on how you solved it and any resources you may have used.</i> <ul style="list-style-type: none"><li>- <i>Step 1: I create a Regex pattern based on user input by replace '*' by '.*'</i></li><li>- <i>Step 2: Then I check each DNA line if this line is matched with provided pattern.</i></li><li>- <i>Step 3: If the DNA line is matched, I display the sequence in the console.</i></li></ul>
<b>Comments</b>	<b>Sometimes, my program takes a long time to finish searching sequence in level 7.</b>

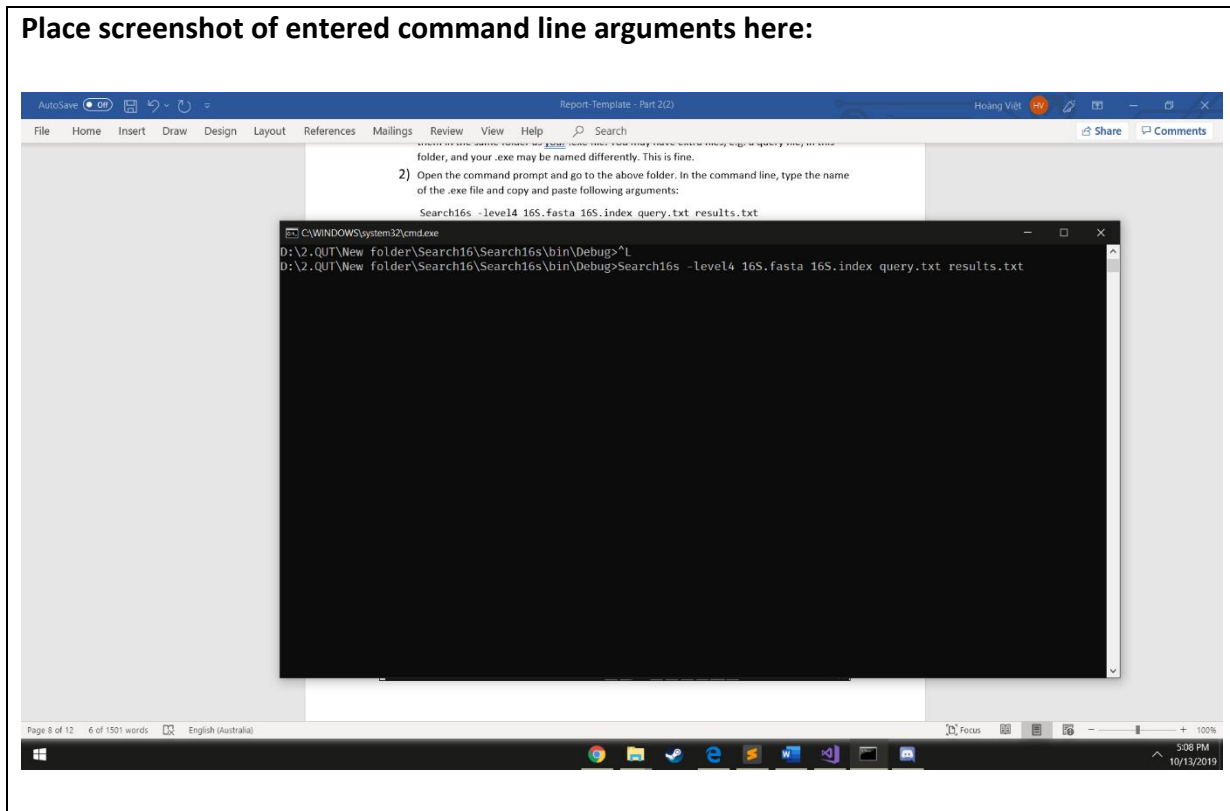
## Screenshots of Functionality

In the following section, you are required to provide screenshots that provide evidence of your program working with provided input. **You must complete this section.**

- 1) The 16S.fasta file has been provided with this template. Download them and place them in the same folder as your .exe file. You may have extra files, e.g. a query file, in this folder, and your .exe may be named differently. This is fine.
- 2) Open the command prompt and go to the above folder. In the command line, type the name of the .exe file and copy and paste following arguments:

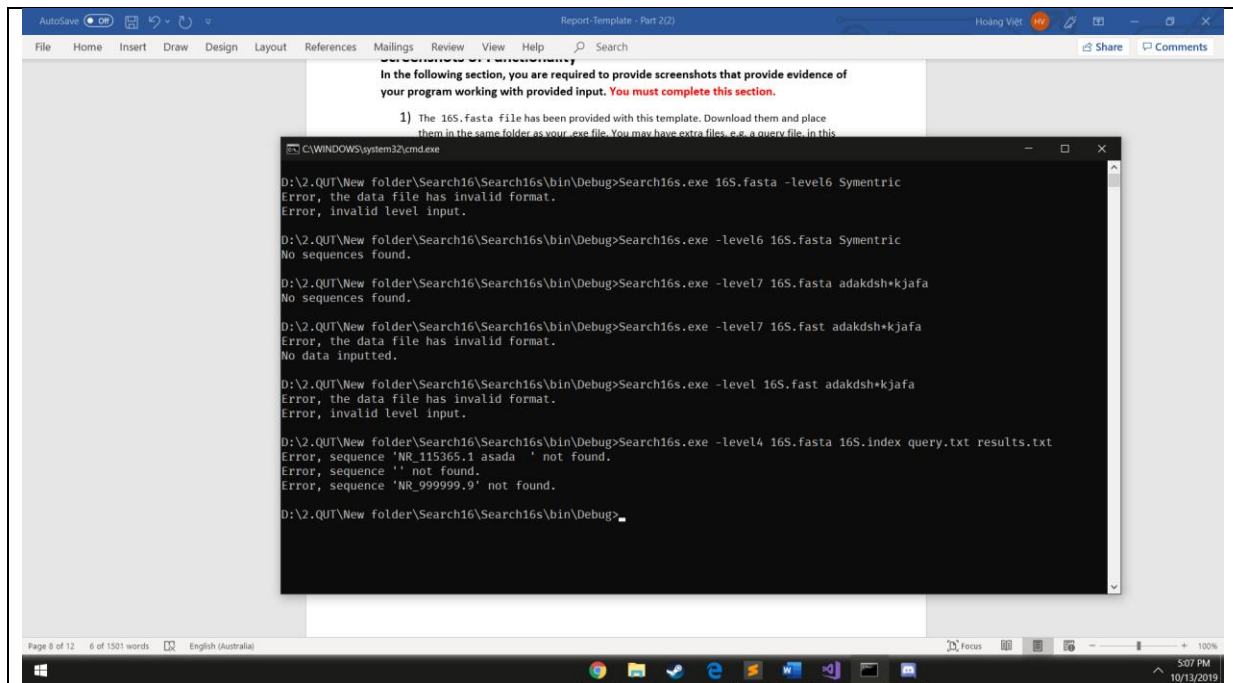
```
Search16s -level4 16S.fasta 16S.index query.txt results.txt
```

**Place screenshot of entered command line arguments here:**



- 3) Hit enter to run your program.

**Place screenshot(s) of the full output to console of your program. You may expand the box as necessary, and use as many screenshots as needed:**





## Self-Assessment:

1) How do I think I went with this assignment?

**This assignment is difficult, especially the code quality part. I have no idea how to get the full mark for this part.**

2) What did I find difficult in this assignment?

**Implementing OOP structure.**

3) What would I do differently next time?

**I will spend more time designing the code structure more carefully.**

4) Were there any bugs in my assignment, if so what were they?

**Sometimes, my program takes a long time to finish searching sequence in level 7.**

**CRA:**

<b><u>Code Quality</u></b>	<b>Part I Marks Available:</b>	<b>Part II Marks Available:</b>
To score points in this section, the student must follow the code quality guidelines as specified in the C# Coding Style Guide on Blackboard	<b>27/30</b>	<b>30/30</b>
Maintained consistent, clear, and meaningful standard in variable and method naming. No magic numbers.	2/3	3/3
Well structured – consistent and appropriate white spacing, line length, indentation, and separation into files within the project (i.e. one class per file)	2/2	2/2
Well commented – class header comment at beginning of each class, comment before every method, and in-line comments to explain complex or not easily discernible code. In-line comments are not excessive.	3/4	4/4
The DRY principle (Don't repeat yourself) is followed where appropriate	3/3	3/3
Methods are single purpose and clear	4/4	4/4
Classes are well designed, with high cohesion and low coupling	7/8	8/8
Classes are separated into reusable modules where appropriate	3/3	3/3
Exceptions are thrown and handled appropriately	3/3	3/3

<b><u>Basic Functionality</u></b>		<b>Part I Marks Available:</b>	<b>Part II Marks Available:</b>
To score marks in this section, your program must be able to be run from the command line with the appropriate arguments.		<b>/15</b>	<b>15/15</b>
<i>Basic Output</i>	The program displays the data from the file	1/1	1/1
	The program displays the appropriate line	1/1	1/1
	The correct amount of information is displayed, e.g. only the relevant entries	1/1	1/1
	The correct level, provided as a command line flag - <b><i>levelN</i></b> , is executed	1/1	1/1
	The program <b>does not</b> store the whole file in memory, instead it accesses the file on disk	5/5	5/5
	<b>Total:</b>	<b>9/9</b>	<b>9/9</b>
<i>Error Handling</i>	A clear error message is displayed when an incorrect number of arguments is provided	2/2	2/2
	A clear error message is provided when an incorrect flag is provided (e.g. not -level1, etc.)	2/2	2/2
	A clear error message is provided when the input file doesn't exist, or is incorrectly formatted	2/2	2/2
	<b>Total:</b>	<b>6/6</b>	<b>6/6</b>

<b><u>Part I</u></b>		<b>Marks Available:</b>
To score marks in this section, your program must be able to run levels 1-3.		<b>55/55</b>
<i>Level 1</i>	The program displays the correct data from the file	4/4
	The program displays the appropriate number of entries	4/4
	A clear error message is provided when a non-existent starting sequence number is requested	1/1
	A clear error message is provided when a problematic number of sequences is provided	1/1
	<b>Total:</b>	<b>10/10</b>
<i>Level 2</i>	The program correctly locates and prints the requested sequence	12/12
	A clear error message is provided when the sequence-id does not exist	3/3
	<b>Total:</b>	<b>15/15</b>
<i>Level 3</i>	The program correctly accepts and interprets a list of instructions as specified by the query file	10/10
	The output is stored in the results file	10/10
	If a sequence is not found, the error message is displayed on the console and not in the results file	5/5
	<b>Total:</b>	<b>25/25</b>
<i>Progress Report</i>	Statement of completeness 2.5	<b>5/5</b>
	Self-reflection 2.5	

Part I final score	100 /100
--------------------	----------

<b><u>Part II</u></b>		<b>Marks Available:</b>
To score marks in this section, your program must be able to run levels 4-7.		<b>55/55</b>
<i>Level 4</i>	The program creates a file as specified by the command line arguments	2/2
	The index file contains a list of all the sequence ids with the appropriate byte-offset	5/5
	The searching program makes use of the created index file to execute a number of queries	5/5
	A clear error message is provided when the index file does not exist	1/1
	Clear error messages are provided when the query file cannot be found, or when a bad query is given, like in Level 3	2/2
	<b>Total:</b>	<b>15/15</b>
<i>Level 5</i>	The program correctly locates and prints the requested sequence ids	10/10
	A clear error message is provided when the sequence does not exist	5/5
	<b>Total:</b>	<b>15/15</b>
<i>Level 6</i>	The program correctly locates and prints the requested sequence ids	15/15
	A clear error message is provided when the keyword does not exist	5/5



	<b>Total:</b>	<b>20/20</b>
<i>Level 7 (Optional)</i>	The program correctly decodes the expression given and identifies any matching sequences	5/5
	A clear error message is provided when the sequence does not exist	5/5
	<b>Total:</b>	<b>10/+10</b>