

Theory of Programming Languages

CMPT 331 • Fall 2019

-Functional Programming - 150 points

Goals	<ul style="list-style-type: none">• To enjoy a simple programming assignment done in a variety of functional programming languages.• To reflect on this experience through a consulting log.• To facilitate discussions about programming and languages.													
Instructions	<p>Develop a set of functions that will allow you to encrypt a string using a Caesar cipher. Develop a set of functions that will allow you to decrypt a string using a Caesar cipher. Develop a set of functions that will help you to solve (break) a Caesar cipher.</p> <p>Implement all of the above functions for all of the following languages:</p> <table><tr><td><i>Encrypt, Decrypt, Solve</i> in LISP</td><td>[15 points]</td></tr><tr><td><i>Encrypt, Decrypt, Solve</i> in ML</td><td>[15 points]</td></tr><tr><td><i>Encrypt, Decrypt, Solve</i> in Erlang</td><td>[15 points]</td></tr><tr><td><i>Encrypt, Decrypt, Solve</i> in Haskell ¹</td><td>[15 points]</td></tr><tr><td><i>Encrypt, Decrypt, Solve</i> in Scala (in a functional manner)</td><td>[15 points]</td></tr></table> <p>¹ If you like, you may, with my approval, substitute another functional programming language for Haskell. Just ask me first. I recommend Prolog.</p> <p><i>Log and Commentary</i> [75 points]</p> <p>Make a prediction about how long you think it will take you to program this assignment. Write it down. Then keep a log of your work, just like you would as a consultant. The format should be similar to the following:</p> <table><thead><tr><th>Date</th><th>Hours Spent</th><th>Tasks / Accomplishments / Issues / Thoughts</th></tr></thead></table> <p>Be thorough and descriptive in your log. Sum the hours spent when you are finished. Note your original prediction on the log. Write a paragraph or two to explain the discrepancy. (It will likely be huge.)</p> <p>Finally, and most importantly, keep a running commentary (“Dear Diary...”) about your thoughts and experience with each language, including how each language is similar or dissimilar to the others. Tell me in great detail about your thoughts on each language regarding its <i>readability</i> and <i>writability</i>, and what you loved and hated about each. Include a list of your Google searches, as I find that fascinating. When you’re done, rank the languages.</p> <p>This is, by far, my favorite part of this assignment. I look forward to reading your thoughts, searches, and comments, so be thoughtful, thorough, amusing, and impress me.</p>	<i>Encrypt, Decrypt, Solve</i> in LISP	[15 points]	<i>Encrypt, Decrypt, Solve</i> in ML	[15 points]	<i>Encrypt, Decrypt, Solve</i> in Erlang	[15 points]	<i>Encrypt, Decrypt, Solve</i> in Haskell ¹	[15 points]	<i>Encrypt, Decrypt, Solve</i> in Scala (in a functional manner)	[15 points]	Date	Hours Spent	Tasks / Accomplishments / Issues / Thoughts
<i>Encrypt, Decrypt, Solve</i> in LISP	[15 points]													
<i>Encrypt, Decrypt, Solve</i> in ML	[15 points]													
<i>Encrypt, Decrypt, Solve</i> in Erlang	[15 points]													
<i>Encrypt, Decrypt, Solve</i> in Haskell ¹	[15 points]													
<i>Encrypt, Decrypt, Solve</i> in Scala (in a functional manner)	[15 points]													
Date	Hours Spent	Tasks / Accomplishments / Issues / Thoughts												
Submitting	<p>Print out your consulting log, commentary, all source code, and output of thorough test runs. Staple it together, and hand it in to me any time before the class in which it is due. Remember to include your name in the write-up and in the comments of source code.</p>													
Examples	<p><i>on the next page</i></p>													

Theory of Programming Languages

CMPT 331 • Fall 2019

-Functional Programming

Examples

The usage for **encrypt** and **decrypt** should be as follows:

```
encrypt(str, shiftAmount)
decrypt(str, shiftAmount)
```

ML example:

```
- val x = encrypt("This is a test string from Alan", 8);
  val x = "BPQA QA I BMAB ABZQVO NZWU ITIV" : string

- decrypt(x, 8);
  val it = "THIS IS A TEST STRING FROM ALAN" : string
```

Things might be easier if you use only capital letters, so consider writing a “toUpper” function so that you can deal with mixed-case input. It’s okay if your output is all caps.

The usage for **solve** should be as follows:

```
solve(str, maxShiftValue);
```

ML example:

```
- solve("HAL", 26);

Caesar 26: HAL
Caesar 25: GZK
Caesar 24: FYJ
Caesar 23: EXI
Caesar 22: DWH
Caesar 21: CVG
Caesar 20: BUF
Caesar 19: ATE
Caesar 18: ZSD
Caesar 17: YRC
Caesar 16: XQB
Caesar 15: WPA
Caesar 14: VOZ
Caesar 13: UNY
Caesar 12: TMX
Caesar 11: SLW
Caesar 10: RKV
Caesar 9: QJU
Caesar 8: PIT
Caesar 7: OHS
Caesar 6: NGR
Caesar 5: MFQ
Caesar 4: LEP
Caesar 3: KDO
Caesar 2: JCN
Caesar 1: IBM
Caesar 0: HAL
val it = "" : string
```