第 3 回春輪講（プログラム）

2015 年 3 月 17 日

1　輪講課題

以下 RSA 暗号に関すること.
公開鍵 $\{e, n\} = \{3, 493\}$, 秘密鍵 $\{d, n\} = \{299, 493\}$ とする. この時 cipher.txt に書かれている暗号文を解読せよ. なお e, d は RSA 暗号の諸条件は満たしているので細かいことは気にしなくて良い. また, 明らかに $n = 29 \times 17$ であるが気にしなくて良い. （イメージとしては cipher.txt は自分が公開した鍵によって暗号化されているため, cipher.txt を受信した自分が秘密鍵でそれを復号しようとするような状況）

2　RSA 暗号を文字列に適用する

例として公開鍵 $\{e, n\} = \{3, 55\}$, 秘密鍵 $\{d, n\} = \{27, 55\}$ の場合に, メッセージ文字列 “Hello” を RSA により暗号化, 復号化を行う手順を以下に示す.
まずメッセージ文字列のそれぞれの文字を文字コードに従って変換し, 順に連結した 2 進系列を得る. 半角英語の場合, “H” のアスキー符号による 8bit 表現は “01001000”, “e” は “01100101”, “l” は “01101100”, “o” は “01101111” なので以下の系列を得る.

“Hello” → “01001000011001010110110001101100011011111”

次にこの系列を 5bit ずつ区切る（今回の課題ではこの 5bit はお約束とする）. さらにこれを 10 進数に変換して次の系列を得る.

“01001000011001010110110001101100011011111” → “9, 1, 18, 22, 24, 27, 3, 15”

この系列の要素それぞれに RSA の暗号化関数を施して文字系列を得る. 今回の例ではいずれも mod 55 の演算.

$$9^3 \equiv 14, 1^3 \equiv 1, 18^3 \equiv 2, 22^3 \equiv 33, 24^3 \equiv 19, 27^3 \equiv 48, 3^3 \equiv 27, 15^3 \equiv 20$$

この

“14, 1, 2, 33, 19, 48, 27, 20”

が, 暗号文であり, cipher.txt に書かれている内容にあたる.
暗号を解読する側は次のような手順で解読する.

1. 整数系列である暗号文の要素をそれぞれ RSA の復号関数を施して, 整数系列

$$14^{27} \equiv 9, 1^{27} \equiv 1, 2^{27} \equiv 18, 33^{27} \equiv 22, 19^{27} \equiv 24, 48^{27} \equiv 27, 27^{27} \equiv 3, 20^{27} \equiv 15$$

   を得る.
2. 上記系列を約束事に従ってそれぞれ 5bit ずつの系列に変換し, 2 進数列

   “9, 1, 18, 22, 24, 27, 3, 15” → “010010000110010101101100011011000110111”

   を得る.
3. これを 8bit ごとに ASCII コードとして解釈し, 文字列

   “Hello”

   を得る.

なお上記の方法では, 0, 1 を暗号化しても 0, 1 のままであるがそんなことはもちろん気にしてはいけない.

3 注意

やってほしいことと注意等は端的に言って以下の通り

- 入力を cipher.txt として出力を復号したもの（＝平文）としたプログラムを作成する.
- 言語はなんでも良い. なんなら暗号化ライブラリ的なものを探してもよい.
- 冪乗を計算するときはうまくやらないと数値がでかすぎてエラーがでるかも. 注意.
- RSA についてよくわからなくても暗号復号の式に忠実に計算すればよい.

専門的なライブラリを使ってもむしろめんどいかどうかは検証しておりませんので自己責任でお願いします. 自分で書く場合, 冪乗自体を求める必要はないので合同算術をうまく使いましょう. まずは hello の例を実装しましょう.hello の例が実装できれば一瞬で終わります. 解読できたら先輩に「I am ほにゃらら（解読した文章)」とお伝えください. また, 書いたプログラムは一応見るので残しておいてください. 以上頑張ってください.

期限: 4/7 （水）