

Tutorial 2

1. This will get you warmed up with Java programming. Consider $n = p \times q$ where both p, q are (probable) primes. Let

$$p = 470371337651747200580641806655577929345787339815775238995809$$

and let

$$q = 44998447341177314016702369046401726397149954873177618720838261$$

Find $6597986812128043319287253420700005248626359207434742570172557324548952363-8457262351946261146698247560345$ raised to itself (mod n) using (a) modular exponentiation and (b) the Chinese Remainder Theorem. You must implement both of them yourself. Find the time it takes to compute the result by computing the exponentiation 100 times and taking the average. You may use a big integer package and use its multiplication, division, inverse, and other operations except for its modular exponentiation and chinese remainder theorem implementations. (4 marks)

2. Exercise 16 on page 55 of your textbook. (1.5 marks)
3. Exercise 10 on page 58 of your textbook. (1.5 marks)
4. Exercise 1 of your textbook on page 91. (.5 marks)
5. Exercise 3 of your textbook on page 92. (1.5 marks)
6. Exercise 4 of your textbook on page 92. (1.5 marks)
7. Exercise 6 of your textbook on page 92. (1.5 marks)