Zadanie 2. Uzgadnianie klucza DH pomiędzy partnerami A i B. Wykorzystaj podane na wykładach algorytmy do rozwiązania poniższych zadań[1].

a) Sprawdź, że liczba $p = 4\,194\,329$ jest prawdopodobnie pierwsza z poziomem ufności 95 % (tj. $\Pr(p \notin \mathbb{P}) \leqslant 0{,}05$). (2 p.)

b) Niech parametrami globalnymi protokołu uzgadniania klucza będą p i wybrany losowo[2] generator g grupy $\mathbb{Z}_p^*$, $1 < g \leqslant p-1$, $\operatorname{ord}(g) = p-1 = 4\,194\,328 = 2^3 \cdot 29 \cdot 101 \cdot 179$. (2 p.)

c) Niech parametrami prywatnymi będą wybrane losowo $1 \leqslant a \leqslant p-1$ (znany tylko A) i $1 \leqslant b \leqslant p-1$ (znany tylko B). Oblicz przesyłane publicznie wiadomości:

$$\begin{aligned} \mathsf{A} \to \mathsf{B}\colon &\qquad A \equiv g^a \pmod{p}, \\ \mathsf{B} \to \mathsf{A}\colon &\qquad B \equiv g^b \pmod{p}. \end{aligned}$$

(2 p.)

d) Sprawdź, że

$$K \equiv A^b \equiv B^a \pmod{p}.$$

i wyznacz tajny klucz współdzielony K. (2 p.)

e) Zaatakuj protokół, odtworzając klucz K na podstawie znajomości tylko parametrów globalnych p i g oraz publicznie przesłanych wiadomości A i B, korzystając z algorytmu Shanksa (*baby-step giant-step*). (2 p.)

---

[1] Rozwiąż zadanie za pomocą napisanych samodzielnie programów (języki: C/C++, Java, Python, Fortran, Bash) lub przez obliczenia na kartce. W przesłanym rozwiązaniu zamieść programy (w archiwum `.zip`) lub obliczenia (operacje arytmetyczne na liczbach naturalnych można wykonać za pomocą kalkulatora).

[2] Dla wyboru losowego liczb możesz użyć generatora liczb losowych dostępnego z poziomu bibliotek języka programowania, systemowego generatora liczb losowych (‘polecenie `echo $RANDOM`’ w sysmie Linux, funkcje ‘`Get-Random`’ lub ‘`Get-SecureRandom`’ w PowerShell), funkcji `LOS` arkusza kalkulacyjnego lub przycisku funkcji losowej kalkulatora, o ile taką posiada.