**Лабораторная работа 6**

**(Алгоритм Диффи-Хеллмана)**

Протокол обмена ключами Диффи–Хеллмана В симметричных криптосистемах перед началом работы необходимо передать секретный ключ обеим сторонам. До появления криптосистем с открытым ключом распределение секретных ключей между общающимися сторонами всегда представляло собой сложную задачу, поскольку для этого необходим защищенный канал. Как правило, для обмена ключами использовался специальный курьер. Важным преимуществом криптографии с открытым ключом над симметричными криптосистемами является обмен ключами между удаленными пользователями без применения защищенного канала. Первая практичная схема такого обмена была предложена Диффи и Хеллманом и стала называться протоколом обмена экспоненциальными ключами Диффи–Хеллмана.

Для начала пользователи, Алиса и Боб, договариваются использовать конечное поле Fq и элемент g∈Fq, порождающий группу, имеющую большой порядок. Для простоты будем рассматривать поле Fp, в котором число p является большим простым числом. Стороны могут проверить простоту числа p, используя алгоритм, позволяющий создать число p, для которого известно полное разложение числа p − 1 на множители. Затем, Алиса и Боб могут найти элемент g, порождающий группу F∗ p. Можно представить в виде gx(mod p), где x — некоторое число. Теперь числа p и q можно использовать в качестве общих исходных данных в основном варианте протокола обмена ключами Диффи–Хеллмана.

Цель этой лабораторной работы - узнать об обмене ключами Диффи-Хеллмана. Вы когда-нибудь задумывались, как две стороны могут договориться о криптографическом ключе в присутствии наблюдателя, без того, чтобы наблюдатель не выяснил ключ? Я думаю, нет. Это будет упрощенная версия обмена ключами Диффи-Хеллмана (в реальной жизни следует выбирать лучшие константы и большие переменные). Введите столько раз, сколько хотите.

**Пример:**

Фиксированные числа: g = 10, p = 541.

Шаги участника:

1. Найдите человека, которого вы не знаете, и представьтесь.
2. Один из вас - Алиса (а), а другой - Боб (б). Если пол не совпадает, это нормально, один из вас может быть Аланом, а другой Барб - мне все равно.
3. Вы оба выбираете число от 1 до 100, но не говорите другому человеку этот номер.
4. Алиса, вычислите A = g a mod p = 10 a mod 541.

Боб, вычислите B = g b mod p = 10 b mod 541. Не стесняйтесь вырывать свой калькулятор или смартфон.

1. Алиса и Боб, обменивайтесь А и В устно в присутствии Карла.
2. Алиса, вычислите SecretKeyA = B a mod p = B a mod 541. Обратите внимание, что верхний индекс - это выбранная вами строчная переменная.

Боб, вычислите SecretKeyB = A b mod p = A b mod 541. Обратите внимание, что верхний индекс - это выбранная вами строчная переменная.

1. Если вы все сделали правильно, SecretKeyA должен совпадать с SecretKeyB.

a = 2

A = ga mod p = 102 mod 541 = 100

b = 3

B = gb mod p = 103 mod 541 = 459

Alice and Bob exchange A and B in view of Carl

keya = B a mod p = 4592 mod 541 = 232

keyb = A B mod p = 1003 mod 541 = 232

**Задание 1. Используйте следующие простые числа:**

1. g =4, p = 23.
2. g =5, p = 24.
3. g =7, p = 26.
4. g =8, p = 27.
5. g =9, p = 28.
6. g =3, p = 29.
7. g =5, p = 26.
8. g =3, p = 24.
9. g =7, p = 28.
10. g =8, p = 29.
11. g =6, p = 27.
12. g =6, p = 25.

**Задание 2. Используя алгоритм Диффи-Хеллмана, создать общий секретный ключ для шифрования сообщения.**

**Алгоритм протокола Диффи-Хеллмана**.

Первая часть протокола состоит в генерации ключей – простого числа *p* и нахождения генератора группы по умножению поля *Fp*. Напомним, что элемент *g,* 1 *< g < p,* называется генератором (порождающим элементом), если любой другой ненулевой элемент *b ∈ Fp, b ̸*= 0*,* является степенью элемента *g*. Напомню, g является первообразным корнем по модулю *p*.

Вторая часть состоит в выработке общего секретного ключа и состоит из следующих шагов:

1. Участник *A* выбирает случайное число 1 *< a < p*, а участник *B* – 1 *< b < p*. Потом участники вычисляют *x* = *ga* mod *p* и *y* = *gb* mod *p* и пересылает их открыто другому участнику.

2. После обмена числами участники вычисляют общий ключ *k* по формулам



Используйте следующие простые числа:

1. p = 7 –
2. p = 19 –
3. p = 11 –
4. p = 13 –
5. p = 17 –
6. p = 19 –
7. p = 17 –
8. p = 13 –
9. p = 11 –
10. p = 7 –
11. p = 19 –
12. p = 13 –

**Задача**: найти **g** (первообразный корень – если несколько, то взять один из них), числа **x** и **y** двух участников обмена сообщениями, и общий секретный ключ **k**. **Реализовать программный код нахождения данных чисел и ключа.**