Министерство образования Республики Беларусь
Учреждение образования
“Брестский государственный технический университет”
Кафедра интеллектуально-информационных технологий

Лабораторная работа №3

По дисциплине «Современные методы защиты компьютерных систем»

Выполнил:
студент 4 курса
группы ИИ-22
Копанчук Е. Р.
Проверил:
Хацкевич А. С.

Брест-2024

Ход работы:

Задание:

– ознакомиться с теорией, изложенной в п. 1.2 («Атака на основе Китайской теоремы об остатках»);

– получить вариант задания у преподавателя (табл. 7 приложения). Экспонента для всех вариантов $e = 3$);

– используя Китайскую теорему об остатках, получить исходный текст;

– результаты и промежуточные вычисления значений для любых трех блоков шифрованного текста оформить в виде отчета

Код программы:

```
import bigInt from "big-integer"

const C1 =
bigInt("265512042588437453639458173445427122318536978256118193647826665111891811630550740514763292154583777262630200934986793700397088015356648143281626921502104513817517538300069400408279799505580502827141943810340928839392620961410088192924959312117280317857551213307027421288176905372535516652089787251117061504917545510077708069144111210760700205830331284703065293767391237920059172901589128200100098081722345210555092345909209579061775573585406804343848177755390265140102528628628390270738502682247504046344734636187770063578187217247979299235290126005105350956109185866140296796466514653065118123826401469434789946742561");

const C2 =
bigInt("150789556423194257282021584994982118468649308431768978089740558569685295290974246163688312139122910416240489647322461327658569754760526682897159392034138878873822029411640613470696917654375668617536677480752823866366274651649069320108165977229847469211863375782702001298732829200687842015140211094158205289911049370601720816747816085772270710611976925051753780444159151510805605894057760165623962087178769363444374282438079520056836459416340713713883546844994587180692937286009911633650219407097940363872797067260705286816882750375251513484536189771707490334236403137312552623899551780312994011291086179061010076565599");
const C3 =
bigInt("238918479947519996211012950629973800051640209545904234280421715293028320111543177782993819087318711960085807772227491721470068105289062713413081843851201797960276831529319619673823144866131676317886782763872113507762498375793900151657578642083182238165589477981356790380135048551582926851014572799720202971626413155079013060137386358507314473173266788055375194419530328109257040755899391258056779033474904922197435756615347894479405635038561104922933991241906895633829679531336947361996765523586179189476744845086387222306440772299807064795717734858886339206077713648281729706067818969793741874259346808752693813393629");
const N1 =
bigInt("283649083314363981701434876022656953209871800223932329338911826342357153629407804924808727690824143135348814465268739865302606692121141739443738237512765466830200840866340211137901248455444111238930654235216626523719931679467856237760372467713535102589061566556058989685438183586319270554342115533824983306083668341089258087252133951602923759962448114574407107070990442269615000491056777975833976173591744478392255846925468159957034346365935096114478724179347492861489073150752526802666129722114229721673504957874473090464922050830104142146923971346735465998142100886090801102735994876239389964064167495413612759781");
const N2 =
bigInt("159238915104849572266517052476869124423871196463890733294629971472054149753175859607957744817371995715299604744921522183791614704587620431921181254181218758560525786409790115152960991915005891400060948232613084132545438773792581443059383317348611733566190906817218213078533016121879499850936214269043613174338880427891253135025931296493120750372127964056603556694759422225957886565595059562197252921934062758348731511426886027299583496324511022818186240700358609581682327830551634239 2
```

```
1274215081400845008215753168492011304860249761684268851476278657733274269848581844
3218779660839500910540979561358086553820080767991103");
const N3 =
bigInt("2456695808419228670591030430502925158299849746592921789716663970050923757 4
7960435053061971052967858805644744026767525352745008949976512556319021720201058767
7226734370004532227291166285765997052560625559616488795757724056545403130482936207
2545047547083934782926807428798068899311550985949583809250341881624137377601338012
4350901694190597722603038290684658646564006818925333352095300643623593379753875249
1849403436009577974348904400042697540081379707388304834503459306021378223467626283
9682587875821276456132959473217869638242394040244858767462633012020887386942677036
2721167690193330364996358992143010944540151081342891");

const M0 = N1.multiply(N2).multiply(N3)
const m1 = N2.multiply(N3)
const m2 = N1.multiply(N3)
const m3 = N1.multiply(N2)
const n1 = m1.modInv(N1)
const n2 = m2.modInv(N2)
const n3 = m3.modInv(N3)
const S =
C1.multiply(n1).multiply(m1).add(C2.multiply(n2).multiply(m2)).add(C3.multiply(n3)
.multiply(m3))
const M = S.mod(M0)

const nthRoot = (value, n) => {
  let x = bigInt(1), previous;
  do { previous = x; x = previous.multiply(n - 1).add(value.divide(previous.pow(n
- 1))).divide(n); } while (previous.compare(x) !== 0);
  return x;
};

const binaryToTextWin1251 = (binaryStr) => {
  const byteArray = binaryStr.match(/.{1,8}/g).map(byte => parseInt(byte, 2));
  return new TextDecoder('windows-1251').decode(new Uint8Array(new
Uint8Array(byteArray)));
};

console.log(binaryToTextWin1251(nthRoot(M, 3).toString(2)))
```

Пример:

```
PS C:\Users\kopan\OneDrive\Desktop\СМЗКС\lab3> ts-node index.ts
Статья 2. Основные понятия, используемые в настоящем Законе В настоящем Законе и
спользуются следующие основные понятия: государственная тайна - защищаемые госуд
арством сведения в области его военной, внешнеполитической, экономической, разве
дывательной
PS C:\Users\kopan\OneDrive\Desktop\СМЗКС\lab3>
```