# Matematika Diskretua
## Laboratorio saioa: RSA gakoak

Laboratorio saio honetan RSA gakoak kalkulatzeko funtzioa inplementatuko dugu R softwarea erabiliz. Bi dira kalkulatu behar diren gakoak: publikoa $(n, r)$ eta pribatua $(s)$.

- Gako publikoa ($n$, $r$). Mezu zifratu bat jaso nahi duen hartzaileak publiko egiten duen gakoa da. Mezuak zifratzeko erabiltzen dira $n$ eta $r$.
- Gako pribatua ($s$). Hartzaileak mezua deszifratzeko behar duen gako pribatua da. Mezua $n$ eta $r$ erabiliz zifratua izan bada, dagokion $s$ gakoarekin deszifratuko da, bestela ez. Horregatik gorde behar da ezkutuan, hartzailearentzat den mezua hartzaileak, eta ez beste inork, deszifra dezan.

RSA gakoen kalkulurako R funtzio hauek bereziki interesatzen zaizkigu: `Primes(a,b)`, `nextPrime(a)`, `previousPrime(a)`, `modinv(a,b)`, `extGCD(a,b)`, `primeFactors(a)`, `format(a, scientific=FALSE)`.

1. **Ariketa**. Idatz ezazu `RSA_gakoak(p,q)` izeneko funtzio bat, $p$ eta $q$ bi zenbaki lehen emanik, $n$, $r$ eta $s$ kalkulatuko dituena. Jarraitu beharreko urratsak honakoak dira:

   1. Bi zenbaki lehen $p$ eta $q$ aukeratu, $p \neq q$.
   2. Kalkulatu $n = p \times q$.
   3. Gako publikoa ($n$ eta $r$ zenbakiak). $m = (p-1) \times (q-1)$ izanik, $m$ zenbakiarekin lehen erlatiboa den $r$ zenbaki bat aurkitu behar da, hau da zkh($m$,$r$)=1 beteko duena.
   4. Gako pribatua ($s$ zenbakia). $r$ gako publikoaren alderantzizkoa modulu $m$ den $s$ balioa aurkitu behar da, hau da, $s = r^{-1} \bmod m$.

   Erabil ezazu funtzioa honako zenbaki lehenetarako gako publikoa eta pribatua kalkulatzeko (aritmetika modularra gaiko ariketen orriko 9. ariketakoak dira). $r$ gakoaren kalkulurako aurreko laboratorio-saioan inplementatutako `lehen_erlatibo_txiki(m)` funtzioa erabil ezazu.

   - RSA_gakoak(5,17): Gako publikoa, n= 85 r= 3, pribatua, s= 43
   - RSA_gakoak(17,23): Gako publikoa, n= 391 r= 3, pribatua, s= 235
     (Eskuzko kalkulua RSA zifratze-algoritmoari buruzko orri teorikoetan dago)
   - RSA_gakoak(97,101): Gako publikoa, n= 9797 r= 7, pribatua, s= 2743
   - RSA_gakoak(307,397): Gako publikoa, n= 121879 r= 5, pribatua, s= 96941

   Tamalez, gako horiek ez dira seguruak. Halakoak erabiliz gero, enkriptatze-sistemak porrot egiten du, gako publikotik pribatua kalkulatzea posible gertatzen delako. Froga ezazu R erabiliz gako horiek ez direla seguruak.

Handixeagoak diren bi zenbaki lehen aukeratzeko, wikipediako zerrenda begira dezakezu. Azken batean, zure gako propioak aukeratu behar dituzu. Seguruak dira?

https://es.wikipedia.org/wiki/Anexo:Números_primos

2. **Ariketa**. `RSA_gako_handiak(a,b)` funtzioa inplementa ezazu, $a, b \in \mathbb{Z}$ bi zenbaki handi emanik gako publikoa eta pribatua kalkulatuko dituena. Aurreko ariketan inplementatu duzun `RSA_gakoak(p,q)` funtzioari honako hobekuntzak egingo dizkiogu:

   - Parametroak: Funtzioak parametro moduan bi zenbaki lehen jaso beharrean bi zenbaki oso eta positibo jasoko ditu. Haietatik abiatuz, funtzioak berak kalkulatuko ditu bi zenbaki lehen `nextPrime` funtzioa erabiliz. Erabiltzailearentzat zaila da handiak diren $p$ eta $q$ bi zenbaki lehen aurkitzea, eta horrela lana errazten zaio.
   - Gakoak: $n$ eta $r$ gako publikoak ere handiak nahi ditugu. Aurreko laboratoriosaioan inplementatutako `lehen_erlatibo(m,atalasea)` funtzioa erabil ezazu $r$ zuk nahi adina handia izango dela ziurtatzeko.
   - Idazkera zientifikoa: Oso handiak diren zenbakiak idazkera zientifikoan erakusten ditu R softwareak. Hala egin ez dezan eta $a$ zenbaki bat bere osotasunean erakuts dezan, honako funtzioa erabil dezakezu: `format(a, scientific=FALSE)`.

   Erabil ezazu funtzioa honako bi zenbaki handietatik abiatuz gako publikoa eta pribatua kalkulatzeko,

   `RSA_gako_handiak(2634758697353,293756536383)`

   Lortu dituzun gakoak seguruak al dira? Zergatik? Non dago seguru izatearen eta ez izatearen muga? Hona hemen beste proba batzuk:

   (a) `RSA_gako_handiak(30000000,300000000)`

   (b) `RSA_gako_handiak(30000000,3000000000)`

   Ondorioren batera iritsi zara?

   Irakur ezazu wiki-artikulu hau: RSA faktorizazio-lehia

   https://eu.wikipedia.org/wiki/RSA_faktorizazio-lehia