2. d) – Pseudocot

Function citeste\_date1(numar1)

<Se introduc de la tastatura elementele listei numar1>

returneaza numar1

Function citeste\_date2(numar2)

< Se introduc de la tastatura elementele listei numar2>

returneaza numar2

Function adunare(numar1, numar2)

Numar3 <- [] // lista ce stocheaza adunarea numar1 cu numar2

Transport = 0

Pentru I <- lungime\_lista – 1, I <= -1, pas -1 // indexare de la 0

Numar3[i] <- ((numar1[i] + numar2[i]) + transport) % 2

daca numar1[i] + numar2[i] + transport >= 2:

transport <- (numar1[i] + numar2[i] + transport) / 2 (impartire intreaga)

else:

transport = 0

rez <- <inverseaza numar3>

returneaza rez

Citeste k (numar natural nenul ce reprezinta numarul de biti pe care se face adunarea)

numar1 <- [] // lista de cifrea binare a primului numar

numar1 <- citeste\_date1(numar1)

numar2 <- [] // lista de cifre binare celui de-al doilea numar

numar2 <- citeste\_date2(numar2)

verificare <- adunare(numar1, numar2)

Scrie verificare

daca lungime\_verificare >= k // indexare de la 0

Scrie "Adunarea nu este valida, se produce depasirea domeniului de", k, "biti"