Lab 1

nchoosek(n, k) – combinari de n luate cate k

nchoosek([2, 3, 5, 7], 2)

perms(‘word) – permutarile cuvantului ‘word’

Lab 2

randi(N, n, m) – de la 1 la N, dimensiunea n x m

length(unique(randi(zile, 1 nrpers)))

rand – numar random intre 0 si 1

frecventa relativa – probabilitatea

frecventa absoluta – nr cazuri favorabile

Lab 3

Randsample(‘aaaaabbbbcccc’, 3 replacement = false) – fara returnare, 3 litere din acel string

If (ismember(‘a’, extragere)

For j = extragere

frecv\_abs=hist(sume\_sim,sume\_posibile)

sume\_posibile = 4:24

sume\_frecv\_abs=[sume\_posibile;frecv\_abs]'

4 10

5 20 ->>>>pt fiecare suma, numarul de cate ori a aparut

…

24

bar(sume\_posibile,frecv\_abs/m,'hist','FaceColor','b');

axa x = 4: 24, sume posibile

Lab4

m=1000;

hygepdf(3,49,6,6) ---

49 – total de bile in urna

6 – albe, si vrem sa vedem ca am 3 din 6

6 – extrageri

Out: probabilitatea sa prinzi un bilet 3 din 6,,,

daca pui intr o urna bile si albe si negre, ii zici cate extrageri sa fac afara sa pui bila inapoi in urna, iti calculeaza probabilitatea sa ai la sfarsit de exemplu 3 din 6 sa fie albe.

p=sum(hygepdf(3:6,49,6,6))

x=geornd(p,1,m);

simuleaza tentativele pana la primul succes,

p – probabilitatea de success

returneaza cate incercari au fost pana la success

geornd(hygepdf(3:6,49,6,6)) – atatea incercari au fost pana a castigat bilet cu 3

prob\_estim=mean(x>=10)

prob\_teor=1-sum(geopdf(0:9,p))

Lab 5

Lab 6

Normrnd(m, s, l, c) = cand generam pentru distributia normala

M = media (temperaturilor, inaltimilor)

S = sigma(deviatia standar, cart de variate is valorile)

L = nr linii

C = nr coloane

Normpdf(diviziune, m, s) = functia de densitate

Normcdf (170, 165, 10)= functia de repartitie, probabilitatea sa ai o valoare cel mult egala cu 170

Unifrnd(a, b, 1, nr\_punte) putem genera nr in intervalul a – b, uniform distribuite