**1MPR07\_Simona\_Bļinova sb24037**

**1.uzdevums**

Programma, kas meklē un nosaka ievadīta vārda atrašanas vietu nejauši ģenerēta vārdu masīvā.

**Kods:**

import numpy

import random

def kartosana(a, sv, bv):

if sv < bv:

i = sv

j = bv

solis = -1

lv = True

while i != j:

g1 = len(a[i])

g2 = len(a[j])

if g1 < g2:

mg = g1

else:

mg = g2

b = 0

for l in range(mg):

if a[i][l] != a[j][l]:

b = l

break

if lv == (ord(a[i][b]) > ord(a[j][b])):

x = a[i]

a[i] = a[j]

a[j] = x

x = i

i = j

j = x

lv = not lv

solis = -solis

j = j + solis

kartosana(a, sv, i-1)

kartosana(a, i+1, bv)

def meklet(a, b):

l = 0

r = len(a) - 1

while (l <= r):

i = (l+r) // 2

#print(l)

#print(r)

#print(a[i])

#print(b)

paz = burts\_indeks(a[i], b)

#print(paz)

g1 = len(a)

g2 = len(b)

if g1 < g2:

mg = g1

else:

mg = g2

for n in range(mg):

if a[i][n] != b[n]:

paz = n

break

else:

if len(a[i]) == mg:

l = i + 1

else:

r = i - 1

if a[i] == b:

break

elif ord(a[i][n]) < ord(b[n]):

l = i + 1

else:

r = i - 1

if a[i] == b:

return i

else:

return -1

def burts\_indeks(a, b):

g1 = len(a)

g2 = len(b)

if g1 < g2:

mg = g1

else:

mg = g2

for i in range(mg):

if a[i] != b[i]:

paz = i

break

else:

if mg == len(a):

paz = True

else:

paz = False

return paz

vardi = numpy.arange(10000)

vardi = numpy.array(vardi, dtype='U')

n = 0

while n < 10000:

garums = random.randint(3, 8)

vards = ''

for i in range(garums):

burts = random.randint(65, 90)

vards += chr(burts)

vardi[n] = vards

n += 1

print(vardi)

kartosana(vardi, 0, len(vardi)-1)

print(vardi)

#print(vardi[2005])

v = input('Ievadiet burtu virkni (3-8 gara) --> ')

if v.isalpha() == True:

v = v.upper()

#print(v)

if len(v) < 3 or len(v) > 8:

print('Burtu virknē ir nepareizs burtu skaits!')

else:

vieta = meklet(vardi, v)

if vieta == -1:

print(f'{v} netika atrasts masīvā.')

else:

print(f'{v} tika atrasts masīvā {vieta}.vietā.')

Testa piemērs(1)

A screenshot of a computer code

AI-generated content may be incorrect.

Testa piemērs(2)

A black background with white text

AI-generated content may be incorrect.

Testa piemērs(3)

A screenshot of a computer

AI-generated content may be incorrect.

**2.uzdevums**

Programma, kas veic divu naturālo skaitļu saskaitīšanu.

**Kods:**

import numpy

def parbaude(a):

skaititajs = 1

while skaititajs < 3:

for i in range(len(a)):

if ord(a[i]) < 48 or ord(a[i]) > 57:

break

else:

return a

skaititajs += 1

a = input('Ievadiet skaitli vēlreiz --> ')

else:

print('Skaitlis netika ierakstīts, programma beidz darbību.')

exit()

def masivs(a):

g = len(a)

b = numpy.arange(g)

b = numpy.array(b, dtype='i')

return b

def virkne\_masivs(a, b):

for i in range(len(b)):

a[i] = b[len(b)-1-i]

return a

def masivs\_virkne(a):

c = ''

for i in range(len(a)):

c += str(a[len(a)-1-i])

return c

# Pārbaude ka sakumā nav nulle

sk1 = input('Ievadiet pirmo skaitli --> ')

sk1 = parbaude(sk1)

sk2 = input('Ievadiet otro skaitli --> ')

sk2 = parbaude(sk2)

m1 = masivs(sk1)

m2 = masivs(sk2)

m1 = virkne\_masivs(m1, sk1)

m2 = virkne\_masivs(m2, sk2)

#print(m1)

#print(m2)

g1 = len(m1)

g2 = len(m2)

if g1 <= g2:

maz\_sk = m1

liel\_sk = m2

else:

maz\_sk = m2

liel\_sk = m1

m3 = masivs(liel\_sk)

for i in range(len(maz\_sk)):

cip = maz\_sk[i] + liel\_sk[i]

atl = cip // 10

if atl != 0:

cip -= atl\*10

if i+1 < len(liel\_sk):

liel\_sk[i+1] += atl

else:

liel\_sk = numpy.append(liel\_sk, atl)

m3 = numpy.append(m3, 0)

m3[i] = cip

else:

for i in range(len(maz\_sk), len(liel\_sk)):

m3[i] = liel\_sk[i]

sk3 = masivs\_virkne(m3)

print(f'{sk1} + {sk2} = {sk3}')

Testa piemērs(1)

A blue background with white text

AI-generated content may be incorrect.

Testa piemērs(2)

A blue background with white text

AI-generated content may be incorrect.

Testa piemērs(3)

A blue background with white text

AI-generated content may be incorrect.

**3.uzdevums**

Programma, kas veic divu naturālo skaitļu atņemšanu.

**Kods:**

import numpy

def parbaude(a):

skaititajs = 1

while skaititajs < 3:

for i in range(len(a)):

if ord(a[i]) < 48 or ord(a[i]) > 57:

break

else:

return a

skaititajs += 1

a = input('Ievadiet skaitli vēlreiz --> ')

else:

print('Skaitlis netika ierakstīts, programma beidz darbību.')

exit()

def masivs(a):

g = len(a)

b = numpy.arange(g)

b = numpy.array(b, dtype='i')

return b

def virkne\_masivs(a, b):

for i in range(len(b)):

a[i] = b[len(b)-1-i]

return a

def masivs\_virkne(a):

c = ''

for i in range(len(a)):

c += str(a[len(a)-1-i])

return c

# Pārbaude ka sakumā nav nulle

sk1 = input('Ievadiet pirmo skaitli --> ')

sk1 = parbaude(sk1)

sk2 = input('Ievadiet otro skaitli --> ')

sk2 = parbaude(sk2)

m1 = masivs(sk1)

m2 = masivs(sk2)

m1 = virkne\_masivs(m1, sk1)

m2 = virkne\_masivs(m2, sk2)

#print(m1)

#print(m2)

g1 = len(m1)

g2 = len(m2)

if g1 == g2:

for i in range(g1):

if m1[g1-1-i] == m2[g2-1-i]:

maz\_sk = m2

liel\_sk = m1

continue

elif m1[g1-1-i] < m2[g2-1-i]:

maz\_sk = m1

liel\_sk = m2

else:

maz\_sk = m2

liel\_sk = m1

elif g1 < g2:

maz\_sk = m1

liel\_sk = m2

else:

maz\_sk = m2

liel\_sk = m1

m3 = masivs(liel\_sk)

for i in range(len(maz\_sk)):

if liel\_sk[i] < maz\_sk[i]:

for j in range(i+1, len(liel\_sk)):

if liel\_sk[j] != 0:

n = j

break

for k in range(n, i, -1):

liel\_sk[k] -= 1

liel\_sk[k-1] += 10

#print(liel\_sk)

cip = liel\_sk[i] - maz\_sk[i]

m3[i] = cip

#print(cip)

else:

for i in range(len(maz\_sk), len(liel\_sk)):

m3[i] = liel\_sk[i]

indeksi = []

for i in range(len(m3)-1, 0, -1):

if m3[i] != 0:

bv = i

break

else:

indeksi.append(i)

#print(m3)

m3 = numpy.delete(m3, indeksi)

#print(m3)

sk3 = masivs\_virkne(m3)

if int(sk1) < int(sk2):

a = sk1

b = sk2

else:

a = sk2

b = sk1

print(f'{b} - {a} = {sk3}')

Testa piemērs(1)

A screenshot of a computer

AI-generated content may be incorrect.

Testa piemērs(2)

A blue screen with white text

AI-generated content may be incorrect.

Testa piemērs(3)

A blue screen with white text

AI-generated content may be incorrect.

**4.uzdevums**

Programma, kas veic divu naturālo skaitļu reizināšanu.

**Kods:**

import numpy

def parbaude(a):

skaititajs = 1

while skaititajs < 3:

for i in range(len(a)):

if ord(a[i]) < 48 or ord(a[i]) > 57:

break

else:

return a

skaititajs += 1

a = input('Ievadiet skaitli vēlreiz --> ')

else:

print('Skaitlis netika ierakstīts, programma beidz darbību.')

exit()

def masivs(a):

g = len(a)

b = numpy.arange(g)

b = numpy.array(b, dtype='i')

return b

def virkne\_masivs(a, b):

for i in range(len(b)):

a[i] = b[len(b)-1-i]

return a

def masivs\_virkne(a):

c = ''

for i in range(len(a)):

c += str(a[len(a)-1-i])

return c

def summa(a, b):

g1 = len(a)

g2 = len(b)

if g1 <= g2:

maz\_sk = a

liel\_sk = b

else:

maz\_sk = b

liel\_sk = a

m3 = masivs(liel\_sk)

for i in range(len(maz\_sk)):

cip = maz\_sk[i] + liel\_sk[i]

atl = cip // 10

if atl != 0:

cip -= atl\*10

if i+1 < len(liel\_sk):

liel\_sk[i+1] += atl

else:

liel\_sk = numpy.append(liel\_sk, atl)

m3 = numpy.append(m3, 0)

m3[i] = cip

else:

for i in range(len(maz\_sk), len(liel\_sk)):

m3[i] = liel\_sk[i]

return m3

# Pārbaude ka sakumā nav nulle

sk1 = input('Ievadiet pirmo skaitli --> ')

sk1 = parbaude(sk1)

sk2 = input('Ievadiet otro skaitli --> ')

sk2 = parbaude(sk2)

m1 = masivs(sk1)

m2 = masivs(sk2)

m1 = virkne\_masivs(m1, sk1)

m2 = virkne\_masivs(m2, sk2)

#print(m1)

#print(m2)

g1 = len(m1)

g2 = len(m2)

if g1 <= g2:

maz\_sk = m1

liel\_sk = m2

else:

maz\_sk = m2

liel\_sk = m1

saskaitamie = []

for i in range(len(maz\_sk)):

pagaidu\_masivs = numpy.zeros(len(liel\_sk)+i, dtype=int)

for j in range(len(liel\_sk)):

sk = maz\_sk[i] \* liel\_sk[j]

atl = sk // 10

pagaidu\_masivs[j+i] += sk - 10\*atl

#print(pagaidu\_masivs)

if j+i+1 >= len(pagaidu\_masivs):

if atl != 0:

pagaidu\_masivs = numpy.append(pagaidu\_masivs, atl)

else:

pagaidu\_masivs[j+i+1] = atl

saskaitamie.append(pagaidu\_masivs)

#print(saskaitamie)

rez = None

if len(saskaitamie) == 1:

rez = masivs\_virkne(saskaitamie[0])

else:

for i in range(1, len(saskaitamie)):

if i < 2:

rez = summa(saskaitamie[i-1], saskaitamie[i])

else:

rez = summa(saskaitamie[i], rez)

sk3 = masivs\_virkne(rez)

print(f'{sk1} \* {sk2} = {sk3}')

Testa piemērs(1)

A screenshot of a computer

AI-generated content may be incorrect.

Testa piemērs(2)

A screenshot of a computer screen

AI-generated content may be incorrect.

Testa piemērs(3)

A screenshot of a computer

AI-generated content may be incorrect.