**1MPR15\_Simona\_Bļinova sb24037**

**1.uzdevums**

Programma, kas veic teksta aizšifrēšanu un atšifrēšanu pēc uzdevuma noteiktā algoritma.

**Kods:**

alfabets = []

for i in range(65, 91):

alfabets.append(chr(i))

alfabets.append('\_')

#print(alfabets)

teksts = input('Ievadiet teksta rindu --> ')

darb = input('Ievadiet vēlamo darbību (c - šifrēt, d - atšifrēt) --> ')

skaitlis = int(input('Ievadiet skaitļi šifrēšanai --> '))

#print(teksts)

#print(len(teksts))

teksts = teksts.upper()

simboli = []

for i in range(len(teksts)):

if teksts[i] == ' ':

simboli.append('\_')

else:

simboli.append(teksts[i])

#print(simboli)

i = 0

soli = []

while len(soli) < len(simboli):

for j in str(skaitlis):

if len(soli) == len(simboli):

break

soli.append(int(j))

if darb == 'd':

virz = -1

else:

virz = 1

sifrs = []

for i in range(len(simboli)):

simbols = simboli[i]

v = alfabets.index(simbols)

solis = soli[i] \* virz

jaun\_v = v + solis

if jaun\_v < 0:

jaun\_v = len(alfabets)+1 + solis

if jaun\_v > len(alfabets)-1:

jaun\_v = jaun\_v - len(alfabets)

sifrs.append(alfabets[jaun\_v])

rinda1 = ''

for i in range(len(simboli)):

rinda1 += str(simboli[i])

rinda2 = ''

for i in range(len(soli)):

rinda2 += str(soli[i])

rinda3 = ''

for i in range(len(sifrs)):

rinda3 += str(sifrs[i])

print('')

print(rinda1)

print(rinda2)

print(rinda3)

Testa piemērs(1)

A screen shot of a computer

AI-generated content may be incorrect.

Testa piemērs(2)

A computer screen shot of white text

AI-generated content may be incorrect.

Testa piemērs(3)

A screen shot of a computer code

AI-generated content may be incorrect.

**2.uzdevums**

Programma, kas veic teksta šifrēšanu un atšifrēšanu izmontojot Morzes kodu

**Kods:**

morse = {'A': '.-',

'B': '-...',

'C': '-.-.',

'D': '-..',

'E': '.',

'F': '..-.',

'G': '--.',

'H': '....',

'I': '..',

'J': '.---',

'K': '-.-',

'L': '.-..',

'M': '--',

'N': '-.',

'O': '---',

'P': '.--.',

'Q': '--.-',

'R': '.-.',

'S': '...',

'T': '-',

'U': '..-',

'V': '...-',

'W': '.--',

'X': '-..-',

'Y': '-.--',

'Z': '--..',

'0': '-----',

'1': '.----',

'2': '..---',

'3': '...--',

'4': '....-',

'5': '.....',

'6': '-....',

'7': '--...',

'8': '---..',

'9': '----.'}

darbiba = input('Ievadiet vēlamo darbību (c - šifrēt, d - atšifrēt) --> ')

print('')

while True:

if darbiba == 'c':

print('Ievadiet tekstu ar latīņu alfabēta burtiem bez punktuacijas zīmem!')

teksts = input('Ievadiet šifrējamo tekstu --> ')

teksts = teksts.upper()

teksta\_vardi = teksts.rsplit(' ')

#print(teksts)

vardi = []

for i in range(len(teksta\_vardi)):

if teksta\_vardi[i] != '':

vardi.append(teksta\_vardi[i])

#print(vardi)

morzes\_kods = ''

for vards in vardi:

for burts in vards:

for key, val in morse.items():

if burts == key:

morzes\_kods += val + ' '

morzes\_kods += ' '

print(f'{teksts} --> {morzes\_kods}')

break

elif darbiba == 'd':

print('Ievadiet tekstu morzes kodā izmantojot tikai "." un "-" ar 1 atstarpi starp katra vārda burtiem un 2 atstarpēm starp vārdiem.')

morzes\_kods = input('Ievadiet tekstu morzes kodā --> ')

morzes\_koda\_simboli = morzes\_kods.rsplit(' ')

#print(teksts)

teksts = ''

for kods in morzes\_koda\_simboli:

#print(kods)

if kods != '':

for key, val in morse.items():

if kods == val:

teksts += key

else:

teksts += ' '

print(f'{morzes\_kods} --> {teksts}')

break

else:

print('Tādas darbības nav!')

print('Programma beidz darbību.')

break

Testa piemērs(1)

A screen shot of a computer screen

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 blue screen with white text

AI-generated content may be incorrect.

**PU1**

Programma, kas veic teksta aizšifrēšanu un atšifrēšanu latviešu alfabēta burtiem

**Kods:**

lat\_burti = ['Ā', 'Č', 'Ē', 'Ģ', 'Ī', 'Ķ', 'Ļ', 'Ņ', 'Š', 'Ū', 'Ž']

alfabets = []

for i in range(65, 81):

alfabets.append(chr(i))

for i in range(82, 87):

alfabets.append(chr(i))

alfabets.append(chr(90))

i = 0

burts = alfabets[i]

while burts != 'Ž':

match burts:

case 'A':

alfabets.insert(i+1, 'Ā')

case 'C':

alfabets.insert(i+1, 'Č')

case 'E':

alfabets.insert(i+1, 'Ē')

case 'G':

alfabets.insert(i+1, 'Ģ')

case 'I':

alfabets.insert(i+1, 'Ī')

case 'K':

alfabets.insert(i+1, 'Ķ')

case 'L':

alfabets.insert(i+1, 'Ļ')

case 'N':

alfabets.insert(i+1, 'Ņ')

case 'S':

alfabets.insert(i+1, 'Š')

case 'U':

alfabets.insert(i+1, 'Ū')

case 'Z':

alfabets.insert(i+1, 'Ž')

i += 1

burts = alfabets[i]

#print(alfabets)

teksts = input('Ievadiet teksta rindu --> ')

darb = input('Ievadiet vēlamo darbību (c - šifrēt, d - atšifrēt) --> ')

skaitlis = int(input('Ievadiet skaitļi šifrēšanai --> '))

#print(teksts)

#print(len(teksts))

teksts = teksts.upper()

simboli = []

for i in range(len(teksts)):

if teksts[i] == ' ':

simboli.append('\_')

else:

simboli.append(teksts[i])

#print(simboli)

i = 0

soli = []

while len(soli) < len(simboli):

for j in str(skaitlis):

if len(soli) == len(simboli):

break

soli.append(int(j))

if darb == 'd':

virz = -1

else:

virz = 1

sifrs = []

for i in range(len(simboli)):

simbols = simboli[i]

v = alfabets.index(simbols)

solis = soli[i] \* virz

jaun\_v = v + solis

if jaun\_v < 0:

jaun\_v = len(alfabets)+1 + solis

if jaun\_v > len(alfabets)-1:

jaun\_v = jaun\_v - len(alfabets)

sifrs.append(alfabets[jaun\_v])

rinda1 = ''

for i in range(len(simboli)):

rinda1 += str(simboli[i])

rinda2 = ''

for i in range(len(soli)):

rinda2 += str(soli[i])

rinda3 = ''

for i in range(len(sifrs)):

rinda3 += str(sifrs[i])

print('')

print(rinda1)

print(rinda2)

print(rinda3)

Testa piemērs(1)

A computer screen shot of white text

AI-generated content may be incorrect.

Testa piemērs(2)

A computer screen shot of white text

AI-generated content may be incorrect.

Testa piemērs(3)

A computer screen with white text

AI-generated content may be incorrect.

**PU2**

Programma, kas veic teksta šifrēšanu izmontojot Morzes kodu ar gaismas signālu

**Grafiska saskarne:**

A screenshot of a computer

AI-generated content may be incorrect.

**Kods:**

import tkinter

import time

def sifrs():

morse = {'A': '.-',

'B': '-...',

'C': '-.-.',

'D': '-..',

'E': '.',

'F': '..-.',

'G': '--.',

'H': '....',

'I': '..',

'J': '.---',

'K': '-.-',

'L': '.-..',

'M': '--',

'N': '-.',

'O': '---',

'P': '.--.',

'Q': '--.-',

'R': '.-.',

'S': '...',

'T': '-',

'U': '..-',

'V': '...-',

'W': '.--',

'X': '-..-',

'Y': '-.--',

'Z': '--..',

'0': '-----',

'1': '.----',

'2': '..---',

'3': '...--',

'4': '....-',

'5': '.....',

'6': '-....',

'7': '--...',

'8': '---..',

'9': '----.'}

teksts = e.get()

#print(teksts)

teksts = teksts.upper()

teksta\_vardi = teksts.rsplit(' ')

#print(teksta\_vardi)

vardi = []

for i in range(len(teksta\_vardi)):

if teksta\_vardi[i] != '':

vardi.append(teksta\_vardi[i])

#print(vardi)

morzes\_kods = ''

for vards in vardi:

#print(vards)

for burts in vards:

#print(burts)

for key, val in morse.items():

if burts == key:

morzes\_kods += val + ' '

morzes\_kods += ' '

#print(morzes\_kods)

# ērtībai visi izpildes laiki tika sadalīti ar 2

for i in range(1, len(morzes\_kods)):

if morzes\_kods[i] == ' ' and morzes\_kods[i-1] == ' ':

time.sleep(3)

elif morzes\_kods[i-1] == ' ':

time.sleep(1)

else:

if morzes\_kods[i-1] == '.':

#print('1')

canva.itemconfig(lamp, fill='red')

canva.update()

time.sleep(0.5)

canva.itemconfig(lamp, fill='grey')

canva.update()

elif morzes\_kods[i-1] == '-':

#print('2')

canva.itemconfig(lamp, fill='red')

canva.update()

time.sleep(1.5)

canva.itemconfig(lamp, fill='grey')

canva.update()

time.sleep(0.5)

logs = tkinter.Tk()

logs.title('Morzes kods')

logs.geometry('300x300')

canva = tkinter.Canvas(logs, height=100, width=100)

canva.place(x=100, y=150)

lamp = canva.create\_oval(10, 10, 90, 90, fill='grey')

l1 = tkinter.Label(logs, text='MORZES KODS')

l1.place(x=100, y=25, height=25, width=100)

l2 = tkinter.Label(logs, text='Ievadiet šifrējamo tekstu:')

l2.place(x=50, y=50, height=25, width=200)

e = tkinter.Entry(logs)

e.place(x=75, y=75, height=25, width=150)

b = tkinter.Button(logs, text='šifrēt', command=sifrs)

b.place(x=125, y=100, height=25, width=50)

logs.mainloop()

Testa piemērs(1) (Testa piemēros ir ēkrānšavieni ar mainīto krāsu)

A screenshot of a computer

AI-generated content may be incorrect.

Testa piemērs(2)

A screenshot of a computer

AI-generated content may be incorrect.

Testa piemērs(3)

A screenshot of a computer

AI-generated content may be incorrect.