

t4-curs 3-curs 4- Program-Coduri numerice

1.Coduri numerice ponderate (8421, 2421, 84-2-1, bichinar)

Realizati programul pt codificarea cifrelor de la 0-9 in unul din codurile de mai sus.

2.Coduri numerice neponderate (Exces3, Gray, 2 din 5)

Realizati programul pt codificarea cifrelor de la 0-9 in unul din codurile de mai sus.

Programul este realizat in orice limbaj de programare doriti.

Fiecare punct trebuie sa contina urmatoarele elemente:

- concept teoretic
- formula de calcul
- program sursa + EXPLICATII
- print screen cu rezultatul rularii programului
- bibliografie

[Link Github pentru codul sursa:](#)

<https://github.com/masterofcsharp/8421-and-exces3>

```
'''
Acesta este programul pentru convertirea oricarei cifre de la 0 la 9 in cod 8421
Conceptul teoretic este simplu, variabila "cat" ia valoarea cifrei, care este citita de la tastatura, restul ia valoarea catului
modulo 2, iar catul se actualizeaza prin impartirea intreaga la 2 de fiecare data. Dupa acesti 2 pasi, restul se aduna la inceputul
variabilei de tip string cod8421. Acest procedeu se executa pana cand catul ia valoarea 0. In final, se printeaza cod8421.
'''

'''
Mai jos este codul pentru convertirea unei cifre citite de la tastatura in cod 8421.
digit = int(input("Please enter a digit: (from 0 to 9) "))
cod8421 = ""
cat = digit
while (True):
    rest = cat % 2
    cat = int(cat/2)
    cod8421 = str(rest) + cod8421
    if (cat == 0):
        break
print(cod8421)

'''

# Urmatoarele linii pun toate cifrele de la 0 la 9 intr-un vector de dimensiune 10 si le convertesc pe rand in cod 8421.
# Codul folosit pentru transformare este cel explicat intre ''' ''' mai sus
# Apoi, fiecare cifra si codul corespunzator sunt afisate pe ecran

cod8421 = ''
digits = []
digitsin8421 = []
for i in range(10):
    digits.append(i)
    # print(digits[i])

for i in range(10):
    cat = digits[i]
    while (True):
        rest = cat % 2
        cat = int(cat/2)
        cod8421 = str(rest) + cod8421
        if (cat == 0):
            break
    k = len(cod8421)
    # adauga zerouri in fata codului pana se ajunge la o lungime de 4, exemplu 1 in 8421 este 0001, in loc de 1
    cod8421 = '0' * (4-k) + cod8421
    digitsin8421.append(cod8421)
    cod8421 = ''

for i in range(10):
    print(f"Cifra {i} este {digitsin8421[i]} in cod 8421.")
```

```
PS C:\Users\User> python -u "c:\Users\User\Desktop\tema BTI\Cifre in cod 8421.py"
Cifra 0 este 0000 in cod 8421.
Cifra 1 este 0001 in cod 8421.
Cifra 2 este 0010 in cod 8421.
Cifra 3 este 0011 in cod 8421.
Cifra 4 este 0100 in cod 8421.
Cifra 5 este 0101 in cod 8421.
Cifra 6 este 0110 in cod 8421.
Cifra 7 este 0111 in cod 8421.
Cifra 8 este 1000 in cod 8421.
Cifra 9 este 1001 in cod 8421.
PS C:\Users\User>
```

```

'''
Acesta este programul pentru convertirea oricarei cifre de la 0 la 9 in cod EXCES-3.
Conceptul teoretic este simplu, algoritmul incepe prin a aduna 3 la fiecare cifra. Cifrele sunt salvate intr-un vector. Apoi,
valoarea "cat" ia valoarea cifrei+3, restul ia valoarea catului modulo 2, iar catul se actualizeaza prin impartirea intreaga la 2
de fiecare data. Dupa acesti 2 pasi, restul se aduna la inceputul variabilei de tip string exces3, care contine codul cifrei in EXCES-3.
Acest procedeu se executa pana cand catul ia valoarea 0. In final, se printeaza cifra si valoarea corespunzatoare in EXCES-3.
'''

# Urmatoarele linii pun toate cifrele de la 0 la 9 marite cu 3 intr-un vector de dimensiune 10 si le convertesc pe rand in cod EXCES-3.
# Apoi, fiecare cifra si codul corespunzator sunt afisate pe ecran

exces3 = '' # Aceasta variabila este utilizata pentru memorarea codului exces-3 al fiecarei cifre
digits = []
digitsinexces3 = []
for i in range(10):
    digits.append(i+3)
    # print(digits[i])
    # se creeaza vectorul cu cifrele de la 0 la 9 convertite in 3-12, pentru a incepe convertirea in codul exces-3

for x in digits:
    cat = x
    while (True):
        rest = cat % 2
        cat = int(cat/2)
        exces3 = str(rest) + exces3
        if (cat == 0):
            break
    k = len(exces3)
    # adauga zerouri in fata codului pana se ajunge la o lungime de 4, exemplu 1 in exces-3 este 0011, in loc de 11
    exces3 = '0' * (4-k) + exces3
    digitsinexces3.append(exces3)
    exces3 = ''
    # Variabila exces3 se reseteaza la '' pentru a putea fi reinitializata cu codul unei noi cifre

for i in range(10):
    print(f"Cifra {i} este {digitsinexces3[i]} in cod EXCES-3.")

```

```
PS C:\Users\User> python -u "c:\Users\User\Desktop\tema BTI\Cifre in cod exces3.py"
Cifra 0 este 0011 in cod EXCES-3.
Cifra 1 este 0100 in cod EXCES-3.
Cifra 2 este 0101 in cod EXCES-3.
Cifra 3 este 0110 in cod EXCES-3.
Cifra 4 este 0111 in cod EXCES-3.
Cifra 5 este 1000 in cod EXCES-3.
Cifra 6 este 1001 in cod EXCES-3.
Cifra 7 este 1010 in cod EXCES-3.
Cifra 8 este 1011 in cod EXCES-3.
Cifra 9 este 1100 in cod EXCES-3.
PS C:\Users\User>
```