

Задаци претраге и сортирања**0265 Секвенцијално претраживање броја у листи**

Написати програм који користи секвенцијално претраживање у несортираној листи [1, 4, 5, 0] да би се пронашао број 0 у листи. Приказати одговарајућу поруку о проналажењу и позицији елемента 0 у листи.

```
def main():
    lista = [1, 4, 5, 0]
    prikaz_liste(lista)
    element = unos_elementa_za_pretragu()
    pronadjen, pozicija = pretraga(lista, element)
    prikaz_rezultata(lista, element, pronadjen, pozicija)

def prikaz_liste(A):
    print(A)

def unos_elementa_za_pretragu():
    a = int(input("Koји element se trazi u listи? "))
    return a

def pretraga(A, a):
    for x in range(len(A)):
        if A[x] == a:
            return True, x
    return False, None

def prikaz_rezultata(A, a, ima, poz):
    print("U listи ", str(A))
    print("trazen je element ", a)
    if ima:
        print("Trazeni element se nalazi u listи.")
        print("Element je pronadjen na poziciji", str(poz))
    else:
        print("Trazeni element se ne nalazi u listи.")

main()
```

0266 Бинарно претраживање броја у листи

Написати програм који користи бинарно претраживање у сортираној листи [0, 1, 4, 15, 23, 99] да би се детектовало да ли постоји елемент 23 у листи. Приказати одговарајућу поруку о проналажењу и позицији елемента 23 у листи.

```
def main():
    lista = [0, 1, 4, 15, 23, 99]
    prikaz_liste(lista)
    pronadjen, pozicija = binarna_pretraga(lista)
    prikaz_rezultata(lista, pronadjen, pozicija)
```

```

def prikaz_liste(A):
    print(A)

def binarna_pretraga(A):
    nizi = 0
    visi = len(A) - 1
    while nizi <= visi:
        sredina = (visi + nizi) // 2
        if A[sredina] == 23:
            return True, sredina
        elif 23 < A[sredina]:
            visi = sredina - 1
        else:
            nizi = sredina + 1
    return False, None

def prikaz_rezultata(A, ima, poz):
    print("U listi ", str(A))
    print("trazen je broj ", 23)
    if ima:
        print("Broj 23 se nalazi u listi.")
        print("Broj 23 se nalazi na poziciji", poz)
    else:
        print("Broj 23 se ne nalazi u listi.")

main()

```

0267 Детекција појаве броја у листи

Написати програм који користи бинарно претраживање у сортираној листи да би се детектовало да ли постоји тражени елемент у листи. Приказати одговарајућу поруку о проналажењу и позицији траженог елемента у листи.

```

def main():
    lista = unos_liste()
    prikaz_liste(lista)
    element = unos_elementa_za_pretragu()
    pronadjen, pozicija = binarna_pretraga(lista, element)
    prikaz_rezultata(lista, element, pronadjen, pozicija)

def unos_liste():
    print("Uneti u listu samo cele brojeve.")
    A = []
    jos = "da"
    while jos == "da":
        x = int(input("Uneti zeljeni broj u listu:"))

```

```

        A.append(x)
        print("Da li treba uneti jos jedan element u listu? ")
        jos = input("Uneti da ili bilo sta drugo za ne: ")
    return A

def prikaz_liste(A):
    print(A)

def unos_elementa_za_pretragu():
    a = int(input("Koji element se trazi u listi? "))
    return a

def binarna_pretraga(A, a):
    nizi = 0
    visi = len(A) - 1
    while nizi <= visi:
        sredina = (visi + nizi) // 2
        if A[sredina] == a:
            return True, sredina
        elif a < A[sredina]:
            visi = sredina - 1
        else:
            nizi = sredina + 1
    return False, None

def prikaz_rezultata(A, a, ima, poz):
    print("U listi ", str(A))
    print("trazen je element ", a)
    if ima:
        print("Trazeni element se nalazi u listi.")
        print("Element se nalazi na poziciji", poz)
    else:
        print("Trazeni element se ne nalazi u listi.")

main()

```

Питања и задаци за самосталан рад

Задаци

092 Написати програм који користи секвенцијално претраживање у несортираној листи [0, 4, 0, 0] да би се детектовало колико има елемената 0 у листи. Приказати одговарајућу поруку после проналажења свих елемената 0 у листи.

0268 Сортирање листе методом избора

Написати програм који користи методу избора да би сортирао од највећег ка најмањем, целе бројеве унутар листе [55, -34, 5, 0, 99, 23] .

```
def prikaz_liste(A):
```

```

print(A)

def sortiranje(A):
    n = len(A)
    for i in range (0, n - 1):
        for j in range (i + 1, n):
            if (A[i] < A[j]):
                b = A[i]
                A[i] = A[j]
                A[j] = b
    return A

def main():
    lista = [55, -34, 5, 0, 99, 23]
    prikaz_liste(lista)
    lista = sortiranje(lista)
    prikaz_liste(lista)

main()

```

0269 Сортирање листе методом bubble-sort

Написати програм који користи методу bubble-sort да би сортирао од најмањег ка највећем, целе бројеве унутар листе [38, 27, 43, 3, 9, 82, 10].

```

def bubble_sort(A):
    for x in range(len(A) - 1, 0, -1):
        for i in range(x):
            if A[i] > A[i + 1]:
                y = A[i]
                A[i] = A[i + 1]
                A[i + 1] = y

```

```

A = [38, 27, 43, 3, 9, 82, 10]
bubble_sort(A)
print(A)

```

Питања и задаци за самосталан рад

Задаци

093 Написати програм који користи методу избора да би сортирао од најмањег ка највећем, целе бројеве унутар листе.

094 Написати програм који користи методу bubble-sort да би сортирао од највећег ка најмањем, целе бројеве унутар листе користећи функције за решавање појединачних корака алгоритма.