**Stergere in vector**

1. **Stergerea elementului de pe pozitia x din vectorul a cu n elemente intregi**

**Ex: a=(2, 9, 100, 45, 6), n=5, k=3**

**Afiseaza: 2 9 45 6**

// deplasez elementele ak+1…an cu o pozitie la stanga

for(i=k+1; i<=n; i++) a[i-1]=a[i];

//micsorez numarul de elemente cu 1

n--;

1. **Stergere repetata/multipla – Sa se stearga toate elementele pare din vector**

**Ex: 2 4 6 7 9 86 52 34 29**

i=1;

while (i<=n)

if (a[i]%2==0)

{ for(j=i+1; j<=n; j++) a[j-1]=a[j];

n--;

}

else i++;

**Tema: 1452, 163, 1453, 2734**

**Inserare in vector**

1. **Sa se insereze valoarea x pe pozitia k in vectorul a cu n el. intregi**

**Ex: a=(2, 9, 10, 45, 6), n=5, k=3, x=100**

**Afiseaza: 2 9 100 10 45 6**

// deplasez el. ak…an cu o pozitie in dreapta

for(i=n; i>=k; i--) a[i+1]=a[i];

//plasez val. x pe pozitia k

a[k]=x;

// creste n cu 1

n++;

**Obs. Deplasarea la dreapta incepe de la dreapta catre stanga.**

1. **Inserare repetata/multipla. Sa se insereze dupa fiecare numar par dublul sau.**

**Ex: 2 10 6 7 9 6 52 4 29**

**2 4 10 20 6 12 7 9 6 12 52 104 4 8 29**

i=1;

while (i<=n)

if (a[i]%2==0)

{ inserez 2\*a[i] pe pozitia i+1

for (j=n; j>=i+1; j--) a[j+1]=a[j];

a[i+1]=2\*a[i];

n++;

**i+=2; // sare peste el. inserat**

}

else i++;

**Permutari circulare pe vector**

1. **Sa se afiseze toate permutarile circulare la stanga ale unui vector a cu n elemente.**

**Ex.**

**A=(10, 20, 30, 40), n=4**

**20 30 40 10**

**30 40 10 20**

**40 10 20 30**

**a1, a2, a3, …, an**

**a2, a3, a4, …., a1**

for (k=1; k<n; k++)

{ // genaram urmatoarea permutare circulara la stanga

aux=a[1];

// deplasez a2….an cu o pozitie la stanga

for(i=2; i<=n; i++) a[i-1]=a[i];

// mutam a[1] la sfarsitul vectorului

a[n]=aux;

// afisam pemutarea curenta

for(i=1; i<=n; i++) cout<<a[i]<<” ”;

}

**Aplicatie: Sa se afiseze a k-a permutare circulara la stanga**

**Ex.**

**a=(10, 20, 30, 40, 50), n=5, k=3**

**Afiseaza: 40, 50, 10, 20, 30**

**Sol.1. Ineficienta ca timp (O(n\*k)), dar eficienta ca memorie**

**for (j=1; j<=k; j++)**

{ // genaram urmatoarea permutare circulara la stanga

aux=a[1];

// deplasez a2….an cu o pozitie la stanga

for(i=2; i<=n; i++) a[i-1]=a[i];

// mutam a[1] la sfarsitul vectorului

a[n]=aux;

}

// afisam pemutarea

for(i=1; i<=n; i++) cout<<a[i]<<” ”;

**Sol.2. Ineficienta ca memorie (O(n+k)) , dar eficienta ca timp O(n+k)**

**Ex.**

**a=(10, 20, 30, 40, 50), n=5, k=3 -> 40 50 10 20 30**

**Pas 1: copiez primele k el. la sf. vecorului -> a=(10, 20, 30, 40, 50, 10, 20, 30)**

for (i=1; i<=k; i++) a[n+i]=a[i];

**Pas 2: stergem primele k elemente -> deplasam el. a[k+1] la a[k+n] cu k pozitii in stanga**

for (i=k+1; i<=k+n; i++) a[i-k]=a[i];

**Pas3. // afisam pemutarea**

for(i=1; i<=n; i++) cout<<a[i]<<” ”;

**Tema: 158, 159, 160, 162, 1365, 1366**

1. **Sa se afiseze toate permutarile circulare la dreapta ale vectorului a cu n elemente.**

**Ex. A=(10, 20, 30, 40), n=4**

**40 10 20 30**

**30 40 10 20**

**20 30 40 10**

int n, a[100], i, k, aux;

int main()

{

cin>>n;

for(i=1;i<=n;i++)

cin>>a[i];

for(i=1;i<=n;i++)

cout<<a[i]<<" ";

cout<<endl;

for(k=1;k<n;k++)

{

aux=a[n];

for(i=n-1;i>=1;i--) // deplasez a1…a­­­n-1cu o pozitie la dreapta

a[i+1]=a[i];

a[1]=aux;

for(i=1;i<=n;i++)

cout<<a[i]<<" ";

cout<<endl;

}

return 0;

}

**1365)**

int n,i,a[200],j;

int main()

{

cin>>n;

for(i=1;i<=n;i++)

cin>>a[i];

i=1;

while(i<n)

if((a[i] + a[i+1]) % 2 == 0)

{

for(j=n;j>=i+1;j--) a[j+1]=a[j];

a[i+1]=(a[i]+a[i+1])/2;

n++;

i=i+2;

}

else i++;

for(i=1;i<=n;i++) cout<<a[i]<<" ";

return 0;

}

**1366)**

int main()

{

int n , **ok = 1** , a[100001];

cin >> n;

for(int i = 1 ; i <= n ; ++i) cin >> a[i];

for(int i = 1 ; i <= n ; ++i) cout << a[i] << " ";

cout << endl;

while(ok)

{

**ok = 0;** // pp ca in parcurgerea curenta nu se va mai face nicio inserare

for(int i = 1 ; i <= n - 1; ++i)

if(abs (a[i] % 2) == abs(a[i+1] % 2 ) && a[i] != a[i+1])

{

for(int j = n ; j >= i+1 ; j--) a[j + 1] = a[j];

a[i + 1] = (a[i] + a[i+1]) / 2;

n ++;

**ok = 1;**

i++; // sare peste elementul inserat

}

if(ok)

{

for(int j = 1 ; j <= n ;++j)

cout << a[j]<< " ";

cout << endl;

}

}

return 0;

}