**C #**

**Blok 2**

**Nizoviiliste**

**Zadatak1 (I verzija – koristitiniz, II verzija – koristi List)**

Za uneti niz brojeva

1. ispisati sumu elemenata koji prethode prvoj nuli u nizu

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

List<int> lista=new List<int>();

for (int i = 0; i < n; i++)

lista.Add(Convert.ToInt32(Console.ReadLine()));

int k = lista.Contains(0)?lista.IndexOf(0):n;

List<int> sublista = lista.GetRange(0, k);

Console.WriteLine(sublista.Sum());

// Prethodne dve linije su mogle da budu i ovako resene

// Console.WriteLine(lista.GetRange(0, k).Sum());

Console.ReadKey();

}

1. ispisati sumu elemenata koji se nalaze iz poslednje nule u nizu

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

List<int> lista = new List<int>();

for (int i = 0; i < n; i++)

lista.Add(Convert.ToInt32(Console.ReadLine()));

int k = lista.Contains(0) ? lista.LastIndexOf(0) : n;

List<int> sublista = lista.GetRange(k,n-k);

Console.WriteLine(sublista.Sum());

}

**Zadatak 2 (I verzija – koristitiniz, II verzija – koristi List)**

Ispisati sve proste brojeve od 2 do n. Za određivanje primeniti Eratostenovo sito.

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

List<int> lista = new List<int>();

for (int i = 2; i < n; i++)

lista.Add(i);

int tekuci = 0;

while (tekuci < lista.Count())

{

int x = lista[tekuci];

lista.RemoveAll(element => element % x == 0 && element > x);

tekuci++;

}

foreach (int x in lista)

Console.Write(x);

Console.ReadKey();

}

**Zadatak 3 (I verzija – koristitiniz, II verzija – koristi List)**

Za uneti niz brojeva odrediti indeks maksimalnog i indeks minimalnog elementa niza koji pripadaju intervalu (a,b).

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

int a = Convert.ToInt32(Console.ReadLine());

int b = Convert.ToInt32(Console.ReadLine());

List<int> lista = new List<int>();

for (int i = 0; i < n; i++)

{

lista.Add(Convert.ToInt32(Console.ReadLine()));

}

/\* lista.RemoveAll(element => element <= a || element >= b);

Console.WriteLine("==> {0}", lista.Max());

Console.WriteLine("==> {0}", lista.Min());

\*/

Console.WriteLine("==> {0}", lista.FindAll(element => element > a || element <b).Max());

Console.WriteLine("==> {0}", lista.FindAll(element => element > a || element <b).Min());

Console.ReadKey();

}

**Zadatak 4 (I verzija – koristitiniz, II verzija – koristi List)**

Neka su dati nizovi a i b od n i m elemenata. Elementi nizova su celi brojevi i unutar nizova nema ponavljanja elemenata.

1. Odrediti niz koji predstavlja presek nizova a i b.

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

int m = Convert.ToInt32(Console.ReadLine());

List<int> a = new List<int>();

List<int> b = new List<int>();

List<int> c = new List<int>();

for (int i = 0; i < n; i++)

{

a.Add(Convert.ToInt32(Console.ReadLine()));

}

for (int i = 0; i < m; i++)

{

b.Add(Convert.ToInt32(Console.ReadLine()));

}

foreach (var x in a) {

if (b.Contains(x))

{

c.Add(x);

}

}

foreach (var x in c)

{

Console.Write(x + " ");

}

Console.ReadKey();

}

1. Odrediti niz koji predstavlja uniju nizova a i b.

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

int m = Convert.ToInt32(Console.ReadLine());

List<int> a = new List<int>();

List<int> b = new List<int>();

for (int i = 0; i < n; i++)

{

a.Add(Convert.ToInt32(Console.ReadLine()));

}

for (int i = 0; i < m; i++)

{

b.Add(Convert.ToInt32(Console.ReadLine()));

}

foreach (var x in b) {

if (!a.Contains(x))

{

a.Add(x);

}

}

foreach (var x in a)

{

Console.Write(x + " ");

}

Console.ReadKey();

}

c) Odrediti niz onih elemenata niza a koji nisu elementi niza b.

d) Ispitati da li se svi elementi niza b nalaze u nizu a.

e) Ispitati da li je niz b podniz niza a (mora se poklapati redosled elemenata)

**Zadatak 5 (I verzija – koristitiniz, II verzija – koristi List)**

U dati niz celih brojeva od n elemenata uređen u neopadajućem poretku umetnuti broj K tako da niz ostane uređen. (Nije dozvoljeno koristiti metod sort())

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

List<int> lista = new List<int>();

for (int i = 0; i < n; i++)

{

lista.Add(Convert.ToInt32(Console.ReadLine()));

}

int k = int.Parse(Console.ReadLine());

int prviveci = lista.Find(x => x > k);

Console.WriteLine(prviveci);

int pozicija = lista.IndexOf(prviveci);

lista.Insert(pozicija, k);

foreach(int x in lista)

Console.WriteLine(x);

Console.ReadLine();

}

**Matrice I testerasti nizovi**

**Zadatak 1 (I verzija – matrica, II verzija – niznizova / testarastamatrica, III verzija – koristi List)**

Za datu matricu realnih brojeva:

1. ispisati sumu pozitivnih elemenata iznad glavne dijagonale

class Program

{

static void Main(string[] args)

{

int n = Convert.ToInt32(Console.ReadLine());

int m = Convert.ToInt32(Console.ReadLine());

/\* Matrica

int[,] a=new int[n,n];

for (int i = 0; i < n; i++)

for (int j = 0; j < n; j++)

a[i,j] = Convert.ToInt32(Console.ReadLine());

int suma = 0;

for (int i = 0; i < n; i++)

for (int j = 0; j < n; j++)

if ((i < j) && (a[i, j] > 0))

suma += a[i, j];

Console.WriteLine("Suma iznad glavne dijagonale: {0}", suma);

\*/

// Niz nizova

int[][] a = new int[n][];

for (int i = 0; i < n; i++)

{

a[i] = new int[m];

for (int j = 0; j < m; j++ )

a[i][j] = Convert.ToInt32(Console.ReadLine());

}

int suma = 0;

for (int i = 0; i < n; i++)

for (int j = 0; j < n; j++)

if ((i < j) && (a[i][j] > 0))

suma += a[i][j];

Console.ReadKey();

}

}

1. odrediti sume po vrstama
2. utvrditi da li je simetrična u odnosu na vertikalnu osu simetrije matrice
3. ciklično pomeriti elemente prve vrste za 1 u desno, elemente druge vrste za 2 u desno itd.

**Zadatak 2 (I verzija – niznizova / testarastamatrica, III verzija – koristi List)**

Neka je za N đaka u odeljenju dat o po nekoliko ocena iz matematike. Napisati program koji će ispisati:

redni broj učenika sa najboljom prosečnom ocenom

static void Main(string[] args)

{

int n = int.Parse(Console.ReadLine());

List<List<int>> ocena = new List<List<int>>();

Random R = new Random();

for (int i = 0; i < n; i++)

{

List<int> l = new List<int>();

int m = R.Next(1,10);

for (int j = 0; j < m; j++)

{

l.Add(R.Next(1,6));

}

ocena.Add(l);

}

foreach (var x in ocena)

{

foreach (var y in x)

{

Console.Write(" " + y);

}

Console.WriteLine();

}

float prosekMx = 0;

int rbr = 0, rbrmax=0;

foreach (var x in ocena)

{

if((float)x.Sum()/x.Count > prosekMx){

prosekMx = (float)x.Sum() / x.Count;

rbrmax = rbr;

}

rbr++;

}

Console.WriteLine("redni broj {0}:" , rbrmax);

Console.ReadKey();

}

1. ordrediti najzastupljeniju ocenu

**Stringovi**

**Zadatak 1**

Napisati program koji učitava nekoliko linija teksta I utvrđuje:

1. da li iza svake tačke postoji bar jedan blanko karakter

static void Main(string[] args)

{

int n = int.Parse(Console.ReadLine());

bool jeste = true;

for (int h = 0; h < n; h++)

{

string s = Console.ReadLine();

int poc = 0;

while (s.IndexOf(".", poc) != -1)

{

int poz = s.IndexOf(".", poc);

if (s[poz + 1] != ' ') jeste = false;

poc = poz + 1;

}

}

if (jeste)

Console.WriteLine("Iza svake tacke je blanko");

else

Console.WriteLine("Nije iza svake tacke blanko");

Console.ReadKey();

}

1. koliko se reči pojavaljuje u tekstu
2. koja je druga najčešće upotrebljena reč
3. koliko se puta u tekstu javio veznik ***sa***

**Zadatak 2**

Napisati program koji simulira rad čekera koji se koristi u jednom on-line testu. Odgovori na pitanja su dati u abc formi, a onaj koradi test unosi oznaku/e stavki koje smatra tačnim. Neka se na početku programa unosi broj pitanja, a zatim u svakoj liniji redni brojpitanja I oznaka tačnog odgovora. Nakon toga se unose odgovori testiranog. Program na kraju ispisuje broj pitanjana koje je testirani dao tačne odgovore. Među ponuđenimo dgovorima na jedno pitanje:

1. samo jedan može da bude tačan
2. može biti više ispravnih, od kojihsvaki mora da budenaveden od strane testiranog da bi se odgovor smatrao tačnim.

**Zadatak 3**

Napisati program koji utvrđuje da li je jedan string sastavljen od karaktera drugog stringa.

1. Karakteri se ne ponavljaju
2. Karakteri se ponavljaju

**Bitovske operacije**

**Zadatak 1**

Napisati program koji ispisuje binarni zapis unetog neoznacenog broja.

**Zadatak 2**

Napisati program koji ucitava nisku nulai jedinica koja predstavlja binarni zapis nekog broja. Program ispisuje taj broj.

**Zadatak 3**

Napisati program koji za unetu nisku brojeva ispisuje da li svi parn ili ne.

BONUS ZADACI

**Zadatak 1.**

Igra "školičica" sastoji se od pravougaone table dimenzije m x n (1 <= m, n <= 20). Igračnapočetkuima 10 zvezdicainalazi se u gornjemlevomuglu table. Bacanjemkockice, igrač se pomerapotabli. Pre negoštoigračstanena novo polje, na to polje se upisujetrenutnibrojzvezdicakojeigračima. Igra se igranasledećinačin: bacanjemkockice se određujepravacismerkretanjaigračai to nasledećinačin:

2 - jednopolje u levo

4 - jednopolje u desno

1 - jednopolje gore

3 - jednopolje dole

5,6 - igrač ne menjastanje

Svakopomeranjelevooduzima 3 zvezdiceigraču

Svakopomeranjedesnododajejednuzvezdicuigraču

Svakovertikalnopomeranje ne utičenabrojzvezdicaigrača

Igra se zaustavljakadatakmičarizgubisvezvezdiceili ne može da odigrazadatipotez. Napisati program kojisimuliraopisanuigru. Sa standardnogulazaunetidimenzije table a zatim, dokle god je to moguće, unositivrednostikojesu pale nakockici. Kada se igrazavrši, odštampati u matričnomoblikuizgled table zaigranje. Poljakojanemajuvrednost( igračnikadanijestaonanjih) imajuoznaku