

ЗАДАЧИ СО ЕДНОДИМЕНЗИОНАЛНИ НИЗИ

1. Да се напише програма која ќе ги испечати елементите на низа кои се иницијализирани при декларацијата

```
#include <iostream>
using namespace std;

int main ()
{
    int b, n[5] = {10,20,100,350,1010};

    for(b=0;b<5;b++)
        cout <<n[b]<<endl;

    return 0;
}
```

2. Да се напише програма која ќе прочита низа од 10 елементи а потоа ќе ги испечати по обратен редослед

```
#include <iostream>
using namespace std;

int main ()
{
    int b, n[10];
    cout <<" Vnesi 10 broja\n";
    for(b=0;b<10;b++)
        cin >> n[b];

    for(b=9;b>=0;b--)
        cout << n[b]<<" ";

    return 0;
}
```

3. Да се напише програма која ќе го пресмета збирот на елементите од низата и нивната аритметичка средина

```
#include <iostream>
using namespace std;

int main ()
{
    int b, n[5];
    float zbir = 0, sr;
    cout <<"vnesi 5 elementi na niza\n";
    for(b=0;b<5;b++)
    {
        cin>>n[b];
        zbir=zbir+n[b];
    }
    sr=zbir/5;
    cout<<"Zbirot na elementite od nizata iznesuva "<<zbir<<endl;
    cout<<"Aritmetichkata sredina na broevite od nizata iznesuva "<<sr;

    return 0;
}
```

4. Да се напише програма која ќе пресмета колку елементи на низата се парни а колку непарни броеви

```
#include <iostream>
using namespace std;

int main ()
{
    int parni=0, neparni=0, b, n[10];
    cout << "Vnesi 10 elementi na niza\n";
    for(b=0;b<10;b++)
    {
        cin >> n[b];
        if(n[b]%2==0)
            parni++;
        else
            neparni++;
    }
    cout<<"Nizata ima "<<parni<<" parni broevi"<<endl;
    cout<<"i "<<neparni<<" neparni broevi";

    return 0;
}
```

5. Да се напише програма која ќе прочита 10 цели броја, а потоа ќе ги пресмета нивните квадратни корени и трети степени

```
#include <iostream>
#include <cmath>
using namespace std;

int main ()
{
    int i, koren, stepen, n[10];
    cout<<"vnesi elementi na nizata\n";
    for(i=0;i<10;i++)
        cin>>n[i];
    for(i=0;i<10;i++)
    {
        koren=sqrt(n[i]);
        stepen=pow(n[i],3);
        cout<<"Kv.koren od "<<n[i]<<" iznesuva "<<koren<<endl;
        cout<<n[i]<<" na tret stepen iznesuva "<<stepen<<endl;
    }
    return 0;
}
```

6. Да се напише програма која ќе ги подреди елементите на низата по големина

```
#include <iostream> //podreduvanje na elementi od niza
using namespace std;

int main ()
{
    int i, j, priv, n[5];
    cout <<"Vnesi 5 broja\n";

    for(i=0;i<5;i++) //vnesuvanje na elementi
        cin >> n[i];

    for(i=0;i<5;i++) //element koj se sporeduva
    {
        for(j=i+1;j<5;j++) //ostanati elementi
        {
            if(n[i]>n[j]) //sporedba
            {
                priv=n[i]; //zamena na mesta
                n[i]=n[j];
                n[j]=priv;
            }
        }
    }

    for(i=0;i<5;i++)
        cout <<n[i]<<" "; //pechatenje na podredena niza

    return 0;
}
```

7. Да се напише програма која ќе ги определи најголемиот и најмалиот елемент од низа од 10 елем.

```
#include <iostream>
using namespace std;
int main()
{
    int n[10], najgolem, najmal;
    cout << "Vnesi 10 broevi" << endl;
    cin>>n[0];
    najgolem=najmal=n[0]; /* najgolemiot i najmaliot ja zemaat
                           vrednosta od prviot element */

    for(int i=1;i<=9;i++)
    {
        cin>>n[i];
        if(n[i]>najgolem) //ako elementot e pogolem od najgolemiot
            najgolem=n[i]; //togash toj stanuva najgolem
        if(n[i]<najmal) //ako elementot e pomal od najmailiot
            najmal=n[i]; //togash toj stanuva najmal
    }
    cout<<"najgolem element vo nizata e "<<najgolem<<endl;
    cout<<"najmal element vo nizata e "<<najmal<<endl;
    return 0;
}
```

8. Да се напише програма која ќе ги прикаже елементите на Фибоначиева низа до одредена позиција

```
#include <iostream>
using namespace std;
int main()
{
    int a;
vnes: cout << "Vnesi dolzhina na niza:" << endl;
    cin>>a;
    if(a<2)
    {
        cout<<"Nema dovolno elementi"<<endl;
        goto vnes;
    }
    int fib[a];
    fib[0]=0;
    fib[1]=1;
    cout<<0<<"", "<<1<<"", " ";
    for(int i=2;i<a;i++)
    {
        fib[i]=fib[i-1]+fib[i-2];
        cout<<fib[i]<<"", " ";
    }
    return 0;
}
```