

Pagrindiniai algoritmai dirbant su sveikųjų ir realiųjų skaičių masyvų reikšmėmis

Sumos skaičiavimo algoritmas

Sveikieji skaičiai	Realieji skaičiai
<pre>int Suma (int X[], int n){ int s = 0; for (int i = 0; i < n; i++) s = s + X[i]; return s; }</pre>	<pre>double Suma (double X[], int n){ double s = 0; for (int i = 0; i < n; i++) s = s + X[i]; return s; }</pre>

Kiekio skaičiavimo algoritmas (grąžina, kiek teigiamų reikšmių yra masyve)

Sveikieji skaičiai	Realieji skaičiai
<pre>int Kiekis (int X[], int n){ int k = 0; for (int i = 0; i < n; i++) if (X[i] > 0) k++; return k; }</pre>	<pre>int Kiekis (double X[], int n){ int k = 0; for (int i = 0; i < n; i++) if (X[i] > 0) k++; return k; }</pre>

Didžiausios masyvo reikšmės vieta masyve

Sveikieji skaičiai	Realieji skaičiai
<pre>int Didziausia (int X[], int n){ int m = 0; for (int i = 0; i < n; i++) if (X[i] > X[m]) m = i; return m; }</pre>	<pre>int Didziausia (double X[], int n){ int m = 0; for (int i = 0; i < n; i++) if (X[i] > X[m]) m = i; return m; }</pre>

Mažiausios masyvo reikšmės vieta masyve

Sveikieji skaičiai	Realieji skaičiai
<pre>int Maziausia (int X[], int n){ int m = 0; for (int i = 0; i < n; i++) if (X[i] < X[m]) m = i; return m; }</pre>	<pre>int Maziausia (double X[], int n){ int m = 0; for (int i = 0; i < n; i++) if (X[i] < X[m]) m = i; return m; }</pre>

Reikšmės šalinimo algoritmas

Sveikieji skaičiai	Realieji skaičiai
<pre>void Salinimas (int X[], int & n, int k) { for (int i = k; i < n; i++) X[i] = X[i + 1]; n--; }</pre>	<pre>void Salinimas (double X[], int & n, int k) { for (int i = k; i < n; i++) X[i] = X[i + 1]; n--; }</pre>

Reikšmės įterpimo algoritmas

Sveikieji skaičiai	Realieji skaičiai
<pre>void Iterpimas (int X[], int & n, int k, int koks) { for (int i = n; i > k; i--) X[i] = X[i - 1]; n++; X[k] = koks; }</pre>	<pre>void Iterpimas (double X[], int & n, int k, double koks) { for (int i = n; i > k; i--) X[i] = X[i - 1]; n++; X[k] = koks; }</pre>

Rikiavimo algoritmas

Sveikieji skaičiai	Realieji skaičiai
<pre>void Rikiavimas(int X[], int n){ int m; int sukeitimas; for (int i = 0; i < n - 1; i++) { m = i; for (int j = i + 1; j < n; j++) if (X[j] > X[m]) m = j; sukeitimas = X[i]; X[i] = X[m]; X[m] = sukeitimas; } }</pre>	<pre>void Rikiavimas(double X[], int n){ int m; double sukeitimas; for (int i = 0; i < n - 1; i++) { m = i; for (int j = i + 1; j < n; j++) if (X[j] > X[m]) m = j; sukeitimas = X[i]; X[i] = X[m]; X[m] = sukeitimas; } }</pre>