Rīgas Tehniskā Universitāte

Datorzinātnes un Informācijas Tehnoloģijas fakultāte

Informātikas un programmēšanas katedra

Datoru mācība 1

Laboratorijas darbs Nr. 5
Viendimensiju masīva apstrāde

D I T F
IDBD 1. kurss 14. grupa
Sergejs Terentjevs
Studena apl. 061RDB140

	Darba izpildes grafiks		
	Protokola sagatave	**Darbs ar datoru**	**Ieskaite**
Pēc plāna (nod.)			
Faktiski (nod.)			

1. Darba uzdevums

Izmantojot esošo sagatavi, izstrādāt programmu, kas viendimensiju masīvā nosaka lielākā negatīva elementa vērtību un indeksu, pa pāra indeksiem.

2. Aprēķinu metode

Lai atrastu starp vairākiem viena tipa negatīviem lielumiem lielāko negatīvo lielumu un tā kārtas pāra numuru analizējamajā grupā, rīkosimies pēc sekojoša algoritma:
1) Sākumā uzskatīsim, ka lielāka negatīva vērtība ir vienāda ar 0, tādejādi turpmāk salīdzinot ar masīva elementu vērtībām iegūsim mums vajadzīgo lielāko negatīvo vērtību. Izmantosim apzīmējumus lielneg un Idet. Pāriet uz punktu 2.
2) Piešķirsim elementu skaitītājam vērtību 2 (**i:=2**), lai programma salīdzinātu elementus sākot ar otro pāra elementu. Pāriet uz punktu 3.
3) Turpmāk ja masīva elements nav mazāks par analizējamo lieluma vērtību, tad pāriet uz punktu 4, citādi uz punktu 6.
4) Ja i-tais elements grupā ir mazāks par lielneg vērtību, tad par lielneg vērtību ņemt i-tā elementa vērtību un par Idet vērtību ņemt i vērtību. Pāriet uz punktu 5.
5) Palielināt i par 2, lai turpmāk analizētu nākošos pāra elementu vērtības (**i:=i+2**). Pāriet uz punktu 3.
6) Risinājums ir iegūts un atrodas lielumos lielneg un Idet.

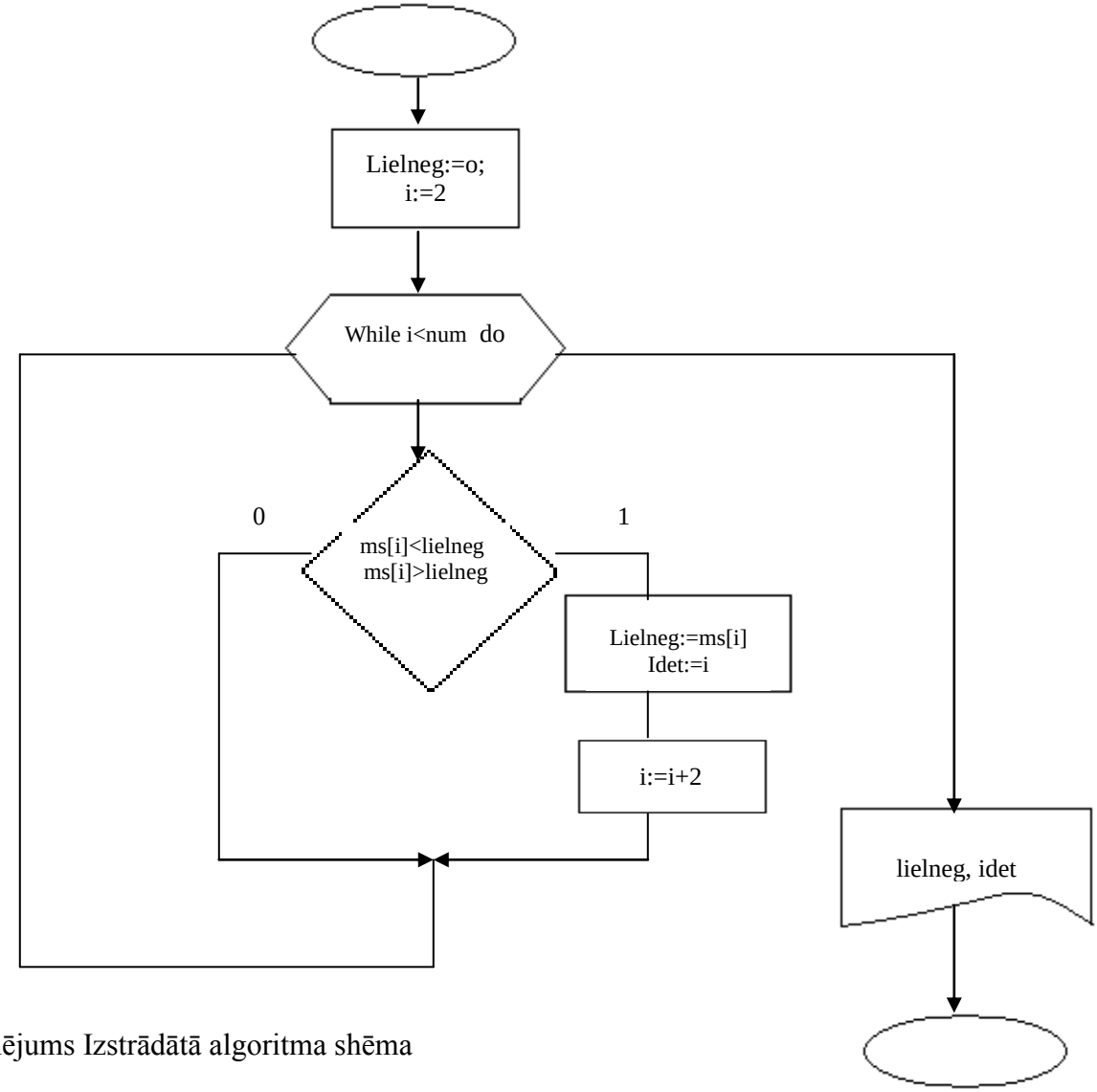
3. Algoritma izstrāde

Sagatave satur mums līdzīga uzdevuma risinājumu un līdzekļus programmas testēšanai ar dažādiem datiem.

Tāpēc ir nepieciešams izstrādāt tikai augstāk apskatītā cikliska algoritma realizāciju.

Izvēlēsimies izstrādātajā fragmentā izmantojamo mainīgo (identifikatoru) vārdus:
lielneg – masīva lielāka negatīva elementa vērtība;
Idet – masīva lielāka negatīva elementa indekss;
i – analizējamā elementa numurs;

Algoritms vienkārši realizējas pielietojot **while** tipa ciklu. Algoritma shēma dota 1. zīmējumā.



1. zīmējums Izstrādātā algoritma shēma

4. Programmas fragmenta pirmteksts

```
{ Meklejam lielako negativa elementu vertibu un indeksu pa para indeksiem}
        i:=2; lielneg:=-MaxLongInt;
 while i<num do begin
        if (ms[i]<0) and (ms[i]>lielneg) then begin
        lielneg:=ms[i];
        idet:=i;
        end;
        i:=i+2;
        end;
  writeln(' Lielaka negativa elementa vertiba ir :', lielneg:10:5);
  writeln(' Lielaka negativa elementa indeks ir :', idet:5);
```

5. Programmas izstrādes un skaņošanas projekts

1. Rediģēt esošo sagatavi.
2. Nokompilēt programmu un likvidēt visas sintaktiskās kļūdas.
3. Pārbaudīt programmas darbību ar kontroldatiem.

6. Kontroldati programmas skaņošanai

Programmas darbību pārbaudīsim vairākas reizes izpildot programmu ar dažādiem datiem. Dažādus datus iegūsim ievadot atšķirīgas vērtības, kuras izmanto masīva elementu vērtību iegūšanai. Svarīgi dotam uzdevumam ir varianti, kur lielākais negatīvais elements ir pirmajā vai trešajā kolonā.

Ievadot vērtības 2 2 iegūsim

” 1” 9.09297	” 2” -7.56802	” 3” - 2.79415	” 4” 9.89358
” 5” -5.44021	” 6” - 5.36573	” 7” 9.90607	” 8” **-2.87903**
” 9” -7.50987	”10” 9.12945	”11” - 0.08851	”12” -9.05578
”13” 7.62558	”14” 2.70906	”15” - 9.88032	”16” 5.51427
”17” 5.29083	”18” - 9.91779	”19” 2.96369	”20” 7.45113

Lielāka negatīva elementa vērtība ir : -2.87903

Lielāka negatīva elementa indekss ir : 8

Ievadot vērtības 5 5 iegūsim

” 1” - 9.58924	” 2” -5.44021	” 3” 6.50288	” 4” 9.12945
” 5” -1.32352	” 6” -9.88032	” 7” - 4.28183	” 8” 7.45113
” 9” 8.50904	”10” **-2.62375**	”11” - 9.99755	”12” -3.04811
”13” 8.26829	”14” 7.73891	”15” - 3.87782	”16” -9.93881
”17” -1.76076	”18” 8.93997	”19” 6.83262	”20” -5.06366

Lielāka negatīva elementa vērtība ir : -2.62375

Lielāka negatīva elementa indekss ir : 10

7. Laboratorijas darba sagatavošanai patērētais laiks

Dotā laboratorijas darba sagatavošanai ir patērēts:
- aprēķinu metodes izstrādei 60 min;
- algoritma izstrādei 30 min;
- programmas pirmteksta fragmenta uzrakstīšanai 30 min;

kopējais laika patēriņš 2 stundas.

8. Laboratorijas darba gaita

1) ir iegūts fails ar programmas pirmtekstu;
2) ir novērstas 7 sintakses kļūdas;
3) veicot skaņošanu pēc kontroldatiem kļūdas nav konstatētas.

9. Rezultāti

Ir apgūta while tipa cikla lietošana, ir praksē realizēta viendimensiju masīva elementu analīze.
Sagatavotajā programmas pirmtekstā ir izlabotas 7 kļūdas.
Programmas darbā kļūdas nav konstatētas.