Національний технічний університет України

«Київський Політехнічний Інститут»

Факультет інформатики і обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №5**

**З предмету «Паралельні та розподілені обчислення»**

**Ада. Захищені модулі**

Виконав:

Студент  
IІІ курсу ФІОТ  
групи ІО-12  
Бута С. О.

Залікова книжка №1205

Київ-2014

**Техническое задание**

1. Структура ПКС с ОП: 
2. Задача: .
3. Язык программирования: Ада.
4. Средства взаимодействия задач: защищенные модули.

**Выполнение работы**

1. Разработка параллельного математического алгоритма.
2. ei = min(ZH);
3. e = min(e, ei);
4. AH = e\*BH – l\*C\*(MO\*MKH); OP: e, l, C, MO;
5. Разработка алгоритмов процессов.

**Задача Т1:**

1. Ввод Z, l
2. Сигнал Т2, Т3, Т4 о завершении ввода S2,3,4;1
3. Ждать завершение ввода в Т2, Т4 W2,4;1
4. Счёт1
   1. e1 = min(ZH)
5. Счет2 КУ
   1. e = min(e, e1);
6. Сигнал Т2, Т3, Т4 о завершении Cчёт2 S2,3,4;2
7. Ждать завершение Счёт2 в Т2, Т3, Т4 W2, 3, 4;1
8. Копии КУ
   1. l1 = l;
   2. e1 = e;
   3. C1 = C;
   4. MO1 = MO;
9. Счёт3
   1. AH = e\*BH – l\*C\*(MO\*MKH)
10. Ждать завершение Счёт3 в Т2, Т3, Т4 W2,3,4;2
11. Вывод А

**Задача Т2:**

1. Ввод MO, MK
2. Сигнал Т1, Т3, Т4 о завершении ввода S1,3,4;1
3. Ждать завершение ввода в Т1, Т4 W1,4;1
4. Счёт1
   1. e1 = min(ZH)
5. Счет2 КУ
   1. e = min(e, e1);
6. Сигнал Т1, Т3, Т4 о завершении Cчёт2 S1,3,4;2
7. Ждать завершение Счёт2 в Т1, Т3, Т4 W1, 3, 4;1
8. Копии КУ
   1. l2 = l;
   2. e2 = e;
   3. C2 = C;
   4. MO2 = MO;
9. Счёт3
   1. AH = e\*BH – l\*C\*(MO\*MKH)
10. Сигнал Т1 о завершении Cчёт3 S1;2

**Задача Т3:**

1. Ждать завершение ввода в Т1, Т2, Т4 W1,2,4;1
2. Счёт1
   1. e1 = min(ZH)
3. Счет2 КУ
   1. e = min(e, e1);
4. Сигнал Т1, Т2, Т4 о завершении Cчёт2 S1,2,4;2
5. Ждать завершение Счёт2 в Т1, Т2, Т4 W1, 2, 4;1
6. Копии КУ
   1. l3 = l;
   2. e3 = e;
   3. C3 = C;
   4. MO3 = MO;
7. Счёт3
   1. AH = e\*BH – l\*C\*(MO\*MKH)
8. Сигнал Т1 о завершении Cчёт3 S1;3

**Задача Т4:**

1. Ввод В, С
2. Сигнал Т1, Т2, Т3 о завершении ввода S1,2,3;1
3. Ждать завершение ввода в Т1, Т2 W1,2;1
4. Счёт1
   1. e1 = min(ZH)
5. Счет2 КУ
   1. e = min(e, e1);
6. Сигнал Т1, Т2, Т3 о завершении Cчёт2 S1,2,3;2
7. Ждать завершение Счёт2 в Т1, Т2, Т3 W1, 2, 3;1
8. Копии КУ
   1. l4 = l;
   2. e4 = e;
   3. C4 = C;
   4. MO4 = MO;
9. Счёт3
   1. AH = e\*BH – l\*C\*(MO\*MKH)
10. Сигнал Т1 о завершении Cчёт3 S1;4

**Разработка схемы взаимодействия процессов:**



**Разработка программы.**

**Листинг**:

gnatmake -q -c -gnatc -u -PC:UsersSerhiyDocumentsadaprjsLab5lab5.gpr main.adb  
  
GNAT GPL 2014 (20140331)  
Copyright 1992-2014, Free Software Foundation, Inc.  
  
  
Checking: C:UsersSerhiyDocumentsadaprjsLab5main.adb (source file time stamp: 2014-05-20 12:41:40)  
  
1. --------------------------------------------------------------------  
2. -- Лабораторна робота №5. Ада. Захищені модулі  
3. -- Бута С. О.  
4. -- Математична операція: A = min(Z)\*B - l\*C\*(MO\*MK).  
5. -- Дата: 20.05.14  
6. --------------------------------------------------------------------  
7.  
8. with Ada.Text\_IO, Ada.Integer\_Text\_IO, Ada.Calendar;  
9. use Ada.Text\_IO, Ada.Integer\_Text\_IO, Ada.Calendar;  
10.  
11. procedure Main is  
12. N: Integer := 4;  
13. P: Integer := 4;  
14. H: Integer := N / P;  
15. FILLER: Integer := 1;  
16.  
17. type Vector is array(1..N) of Integer;  
18. type Matrix is array(1..N) of Vector;  
19.  
20. MO, MK: Matrix;  
21. A, B, C, Z: Vector;  
22. l: Integer;  
23.  
24. procedure VectorInput(V: out Vector) is  
25. begin  
26. for i in 1..N loop  
27. V(i) := FILLER;  
28. end loop;  
29. end VectorInput;  
30.  
31. procedure MatrixInput(M: out Matrix) is  
32. begin  
33. for i in 1..N loop  
34. for j in 1..N loop  
35. M(i)(j) := 1;  
36. end loop;  
37. end loop;  
38. end MatrixInput;  
39.  
40. procedure VectorOutput(V: in Vector) is  
41. begin  
42. for i in 1..N loop  
43. put(V(i));  
44. put(" ");  
45. end loop;  
46. Put\_Line("");  
47. end VectorOutput;  
48.  
49. protected ResourceMonitor is  
50. procedure WriteC(V: in Vector);  
51. procedure WriteMO(M: in Matrix);  
52. procedure WriteL(lInp: in Integer);  
53. procedure CalculateE(ei: in Integer);  
54. function CopyE return Integer;  
55. function CopyL return Integer;  
56. function CopyC return Vector;  
57. function CopyMO return Matrix;  
58. private  
59. l: Integer;  
60. e: Integer := Integer'Last;  
61. C: Vector;  
62. MO: Matrix;  
63. end ResourceMonitor;  
64.  
65. protected SynchronizeMonitor is  
66. procedure SignalInput;  
67. procedure SignalCalculate2;  
68. procedure SignalCalculate3;  
69. entry WaitInput;  
70. entry WaitCalculate2;  
71. entry WaitCalculate3;  
72. private  
73. flagInput: Integer := 0;  
74. flagCalculate2: Integer := 0;  
75. flagCalculate3: Integer := 0;  
76. end SynchronizeMonitor;  
77.  
78. protected body ResourceMonitor is  
79. procedure WriteC(V: in Vector) is  
80. begin  
81. C := V;  
82. end WriteC;  
83.  
84. procedure WriteMO(M: in Matrix) is  
85. begin  
86. MO := M;  
87. end WriteMO;  
88.  
89. procedure WriteL(lInp: in Integer) is  
90. begin  
91. l := lInp;  
92. end WriteL;  
93.  
94. procedure CalculateE(ei: in Integer) is  
95. begin  
96. if ( e > ei ) then  
97. e := ei;  
98. end if;  
99. end CalculateE;  
100.  
101. function CopyE return Integer is  
102. begin  
103. return e;  
104. end CopyE;  
105.  
106. function CopyL return Integer is  
107. begin  
108. return l;  
109. end CopyL;  
110.  
111. function CopyC return Vector is  
112. begin  
113. return C;  
114. end CopyC;  
115.  
116. function CopyMO return Matrix is  
117. begin  
118. return MO;  
119. end CopyMO;  
120. end ResourceMonitor;  
121.  
122. protected body SynchronizeMonitor is  
123. procedure SignalInput is  
124. begin  
125. flagInput := flagInput + 1;  
126. end SignalInput;  
127.  
128. procedure SignalCalculate2 is  
129. begin  
130. flagCalculate2 := flagCalculate2 + 1;  
131. end SignalCalculate2;  
132.  
133. procedure SignalCalculate3 is  
134. begin  
135. flagCalculate3 := flagCalculate3 + 1;  
136. end SignalCalculate3;  
137.  
138. entry WaitInput  
139. when flagInput = 3 is  
140. begin  
141. null;  
142. end WaitInput;  
143.  
144. entry WaitCalculate2  
145. when flagCalculate2 = 4 is  
146. begin  
147. null;  
148. end WaitCalculate2;  
149.  
150. entry WaitCalculate3  
151. when flagCalculate3 = 3 is  
152. begin  
153. null;  
154. end WaitCalculate3;  
155.  
156. end SynchronizeMonitor;  
157.  
158. procedure tasksProcedure is  
159. task T1 is  
160. end T1;  
161.  
162. task body T1 is  
163. ei, li, sum: Integer;  
164. Ci: Vector;  
165. MOi, Res: Matrix;  
166. begin  
167. Put\_Line("Task1 started.");  
168. --Ввод Z, l  
169. VectorInput(Z);  
170. l := FILLER;  
171. ResourceMonitor.WriteL(l);  
172.  
173. --Сигнал Т2, Т3, Т4 о Еавершении ввода  
174. SynchronizeMonitor.SignalInput;  
175.  
176. --Ждать Еавершение ввода в Т2, Т4  
177. SynchronizeMonitor.WaitInput;  
178.  
179. --Счёт1  
180. ei := Integer'Last;  
181. for i in 1..H loop  
182. if (Z(i) < ei) then  
183. ei := Z(i);  
184. end if;  
185. end loop;  
186.  
187. --Счёт2  
188. ResourceMonitor.CalculateE(ei);  
189.  
190. --Сигнал Т2, Т3, Т4 о Еавершении Cчёт2  
191. SynchronizeMonitor.SignalCalculate2;  
192.  
193. --Ждать Еавершение Счёт2 в Т2, Т3, Т4  
194. SynchronizeMonitor.WaitCalculate2;  
195.  
196. --Копии  
197. li := ResourceMonitor.CopyL;  
198. ei := ResourceMonitor.CopyE;  
199. Ci := ResourceMonitor.CopyC;  
200. Moi := ResourceMonitor.CopyMO;  
201.  
202. --Счёт3  
203. for i in 1..H loop  
204. for j in 1..N loop  
205. sum := 0;  
206. for k in 1..N loop  
207. sum := sum + MOi(j)(k)\*MK(k)(i);  
208. end loop;  
209. Res(j)(i) := sum;  
210. end loop;  
211. end loop;  
212.  
213. for i in 1..H loop  
214. sum := 0;  
215. for j in 1..N loop  
216. sum := sum + Res(j)(i)\*Ci(j);  
217. end loop;  
218. A(i) := ei\*B(i) - li\*sum;  
219. end loop;  
220.  
221. --Ждать Еавершение Счёт3 в Т2, Т3, Т4  
222. SynchronizeMonitor.WaitCalculate3;  
223. --Вывод А  
224. if (N < 17) then  
225. Put\_Line("Vector A:");  
226. VectorOutput(A);  
227. end if;  
228. Put\_Line("Task1 finished.");  
229. end T1;  
230.  
231.  
232.  
233. task T2 is  
234. end T2;  
235.  
236. task body T2 is  
237. ei, li, sum: Integer;  
238. Ci: Vector;  
239. MOi, Res: Matrix;  
240. begin  
241. Put\_Line("Task2 started.");  
242. --Ввод MO, MK  
243. MatrixInput(MO);  
244. MatrixInput(MK);  
245. ResourceMonitor.WriteMO(MO);  
246.  
247. --Сигнал Т1, Т3, Т4 о Еавершении ввода  
248. SynchronizeMonitor.SignalInput;  
249.  
250. --Ждать Еавершение ввода в Т1, Т4  
251. SynchronizeMonitor.WaitInput;  
252.  
253. --Счёт1  
254. ei := Integer'Last;  
255. for i in H..2\*H loop  
256. if (Z(i) < ei) then  
257. ei := Z(i);  
258. end if;  
259. end loop;  
260.  
261. --Счёт2  
262. ResourceMonitor.CalculateE(ei);  
263.  
264. --Сигнал Т1, Т3, Т4 о Еавершении Cчёт2  
265. SynchronizeMonitor.SignalCalculate2;  
266.  
267. --Ждать Еавершение Счёт2 в Т1, Т3, Т4  
268. SynchronizeMonitor.WaitCalculate2;  
269.  
270. --Копии  
271. li := ResourceMonitor.CopyL;  
272. ei := ResourceMonitor.CopyE;  
273. Ci := ResourceMonitor.CopyC;  
274. Moi := ResourceMonitor.CopyMO;  
275.  
276. --Счёт3  
277. for i in H..2\*H loop  
278. for j in 1..N loop  
279. sum := 0;  
280. for k in 1..N loop  
281. sum := sum + MOi(j)(k)\*MK(k)(i);  
282. end loop;  
283. Res(j)(i) := sum;  
284. end loop;  
285. end loop;  
286.  
287. for i in H..2\*H loop  
288. sum := 0;  
289. for j in 1..N loop  
290. sum := sum + Res(j)(i)\*Ci(j);  
291. end loop;  
292. A(i) := ei\*B(i) - li\*sum;  
293. end loop;  
294.  
295. --Сигнал Т1 о Еавершении Cчёт3  
296. SynchronizeMonitor.SignalCalculate3;  
297. Put\_Line("Task2 finished.");  
298. end T2;  
299.  
300.  
301.  
302. task T3 is  
303. end T3;  
304.  
305. task body T3 is  
306. ei, li, sum: Integer;  
307. Ci: Vector;  
308. MOi, Res: Matrix;  
309. begin  
310. Put\_Line("Task3 started.");  
311. --Ждать Еавершение ввода в Т1, Т2, Т4  
312. SynchronizeMonitor.WaitInput;  
313.  
314. --Счёт1  
315. ei := Integer'Last;  
316. for i in 2\*H..3\*H loop  
317. if (Z(i) < ei) then  
318. ei := Z(i);  
319. end if;  
320. end loop;  
321.  
322. --Счёт2  
323. ResourceMonitor.CalculateE(ei);  
324.  
325. --Сигнал Т1, Т2, Т4 о Еавершении Cчёт2  
326. SynchronizeMonitor.SignalCalculate2;  
327.  
328. --Ждать Еавершение Счёт2 в Т1, Т2, Т4  
329. SynchronizeMonitor.WaitCalculate2;  
330.  
331. --Копии  
332. li := ResourceMonitor.CopyL;  
333. ei := ResourceMonitor.CopyE;  
334. Ci := ResourceMonitor.CopyC;  
335. Moi := ResourceMonitor.CopyMO;  
336.  
337. --Счёт3  
338. for i in 2\*H..3\*H loop  
339. for j in 1..N loop  
340. sum := 0;  
341. for k in 1..N loop  
342. sum := sum + MOi(j)(k)\*MK(k)(i);  
343. end loop;  
344. Res(j)(i) := sum;  
345. end loop;  
346. end loop;  
347.  
348. for i in 2\*H..3\*H loop  
349. sum := 0;  
350. for j in 1..N loop  
351. sum := sum + Res(j)(i)\*Ci(j);  
352. end loop;  
353. A(i) := ei\*B(i) - li\*sum;  
354. end loop;  
355.  
356. --Сигнал Т1 о Еавершении Cчёт3  
357. SynchronizeMonitor.SignalCalculate3;  
358. Put\_Line("Task3 finished.");  
359. end T3;  
360.  
361.  
362.  
363. task T4 is  
364. end T4;  
365.  
366. task body T4 is  
367. ei, li, sum: Integer;  
368. Ci: Vector;  
369. MOi, Res: Matrix;  
370. begin  
371. Put\_Line("Task4 started.");  
372. --Ввод В, С  
373. VectorInput(B);  
374. VectorInput(C);  
375. ResourceMonitor.WriteC(C);  
376.  
377. --Сигнал Т1, Т2, Т3 о Еавершении ввода  
378. SynchronizeMonitor.SignalInput;  
379.  
380. --Ждать Еавершение ввода в Т1, Т2  
381. SynchronizeMonitor.WaitInput;  
382.  
383. --Счёт1  
384. ei := Integer'Last;  
385. for i in 3\*H..N loop  
386. if (Z(i) < ei) then  
387. ei := Z(i);  
388. end if;  
389. end loop;  
390.  
391. --Счёт2  
392. ResourceMonitor.CalculateE(ei);  
393.  
394. --Сигнал Т1, Т2, Т3 о Еавершении Cчёт2  
395. SynchronizeMonitor.SignalCalculate2;  
396.  
397. --Ждать Еавершение Счёт2 в Т1, Т2, Т3  
398. SynchronizeMonitor.WaitCalculate2;  
399.  
400. --Копии  
401. li := ResourceMonitor.CopyL;  
402. ei := ResourceMonitor.CopyE;  
403. Ci := ResourceMonitor.CopyC;  
404. Moi := ResourceMonitor.CopyMO;  
405.  
406. --Счёт3  
407. for i in 3\*H..N loop  
408. for j in 1..N loop  
409. sum := 0;  
410. for k in 1..N loop  
411. sum := sum + MOi(j)(k)\*MK(k)(i);  
412. end loop;  
413. Res(j)(i) := sum;  
414. end loop;  
415. end loop;  
416.  
417. for i in 3\*H..N loop  
418. sum := 0;  
419. for j in 1..N loop  
420. sum := sum + Res(j)(i)\*Ci(j);  
421. end loop;  
422. A(i) := ei\*B(i) - li\*sum;  
423. end loop;  
424.  
425. --Сигнал Т1 о Еавершении Cчёт3  
426. SynchronizeMonitor.SignalCalculate3;  
427. Put\_Line("Task4 finished.");  
428. end T4;  
429. begin  
430. put("");  
431. end tasksProcedure;  
432. begin  
433. put\_line("Main procedure started");  
434.  
435. tasksProcedure;  
436.  
437. put\_line("Main procedure finished");  
438. end Main;  
  
438 lines: No errors  
gcc.exe: warning: C:UsersSerhiyDocumentslisting.lst: linker input file unused because linking not done  
  
[2014-05-20 15:41:41] process terminated successfully, elapsed time: 00.30s