НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

КАФЕДРА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

Лабораторна робота №1

з дисципліни **«**Програмування паралельних комп’ютерних систем**»**

Виконав:

студент 3 курсу

ФІОТ гр. ІО-34

Кривоносов Олексій

Перевірив:

Корочкін О. В.

Київ – 2016 р.

**Тема:** Програмування для комп’ютерних систем зі спільною пам’яттю. Ада. Семафори

**Технічне завдання:** Розробити програму для розв’язання ПКС із СП (структура на рис. 1) математичної задачі: МА = МВ\*МС+(МО\*MR)\*α

Мова програмування: Ада.

Засоби взаємодії: семафори.



Рис. 1 Структурна схема ПКС

**Виконання роботи:**

**Етап 1. Побудова паралельного алгоритму**

MAH = MBH\*MC+(MOH\*MR)\*α

Спільний ресурс: MC, MR, α

**Етап 2. Розроблення алгоритмів роботи кожного процесу**

|  |  |  |  |
| --- | --- | --- | --- |
| Задача Т1 |  | Задача Т2 |  |
| 1. Введення MB, MC, MO, MR, α |  | 1. Чекати введення MB, MC, MO, MR, α у задачі T1. | W1,1 |
| 2. Сигнал задачі T2 про введення MB, MC, MO, MR, α | S2,1 | 2. Копіювати MC2 := MC | КУ |
| 3. Копіювати MC1 := MC | КУ | 3. Копіювати MR2 := MR | КУ |
| 4. Копіювати MR1 := MR | КУ | 4. Копіювати α2 := α | КУ |
| 5. Копіювати α1 := α | КУ | 5. Обчислення MAH = MBH\*MC2+(MOH\*MR2)\*α2 |  |
| 6. Обчислення MAH = MBH\*MC1+(MOH\*MR1)\*α1 |  | 6. Сигнал Т1 про завершення  обчислень | S1,1 |
| 7. Чекати сигнал від Т2 про завершення  обчислень | W2,1 |  |  |
| 8. Вивід MA |  |  |  |

**Етап 3. Розроблення структурної схеми взаємодії задач**

На структурній схемі взаємодії задач уведено такі семафори

* Skd1 – для керування доступом до спільних ресурсів MC
* Skd2 – для керування доступом до спільних ресурсів MR
* Skd3 – для керування доступом до спільних ресурсів α
* S1 – для синхронізації із завершенням введення в T1
* S2 – для синхронізації із завершенням обчислень в T2

Рис. 2 Структурна схема взаємодії задач

**Етап 4. Розроблення програми**

GNAT GPL 2015 (20150428-49)

Copyright 1992-2015, Free Software Foundation, Inc.

Compiling: lab1.adb

Source file time stamp: 2016-02-21 15:18:06

Compiled at: 2016-02-23 18:31:32

1. -----------------------------------------------------------------

2. -- --

3. -- Programming for parallel computer systems --

4. -- Laboratory work #1. Ada. Semaphore --

5. -- --

6. -- --

7. -- Task: MA = MB\*MC + (MO\*MR)\*a --

8. -- --

9. -- Author: Krivonosov Olexiy, group IO-34 --

10. -- Date: 21.02.2016 --

11. -- --

12. ------------------------------------------------------------------

13.

14. with Ada.Text\_IO, Ada.Integer\_Text\_IO, Ada.Synchronous\_Task\_Control;

15. use Ada.Text\_IO, Ada.Integer\_Text\_IO, Ada.Synchronous\_Task\_Control;

16. procedure Lab1 is

17. N : Natural :=2;

18. begin

19. Put\_Line ("Input N: ");

20. Get (Item => N);

21.

22. declare

23. P : Integer:=2;

24. H : Integer:= N/P;

25.

26. subtype Index is Positive range 1..N;

27. type Vector is array (Index) of Integer ;

28. type Matrix is array (Index) of Vector;

29.

30.

31.

32. MA, MB, MC, MO, MR : Matrix;

33. alpha, tempV : Integer;

34. S1,S2, Skd1 : Suspension\_Object; --semaphores

35.

36. procedure Input\_Matrix(MA : out Matrix);

37. procedure Output\_Matrix(MA : in Matrix);

38.

39.

40.

41. procedure Start\_tasks is

42. task T1;

43. task body T1 is

44. MC1,MR1, Buf11, Buf12 : Matrix;

45.

46. alpha1:integer;

47. begin

48. Put\_Line("T1 started");

49. -- Entering data (1)

50. Input\_Matrix(MB);

51. Input\_Matrix(MC);

52. Input\_Matrix(MO);

53. Input\_Matrix(MR);

54. alpha := 1;

55.

56. --Signal to T2 about end of input (2)

57. Set\_True(S1);

58. -- copying shared resources (3,4,5)

59. Suspend\_Until\_True(Skd1);

60. MC1 := MC; --Critical sections

61. MR1 := MR;

62. alpha1 := alpha;

63. Set\_True(Skd1);

64. -- Calculation MAH = MBH\*MC1 + (MOH\*MR1)\*a1 (6)

65. for i in 1..H loop

66. for j in Index loop

67. tempV:=0;

68. for k in Index loop

69. tempV:=tempV+MO(i)(k)\*MR1(k)(j);

70. Buf11(i)(j):=tempV;

71. end loop;

72. end loop;

73. end loop;

74.

75. for i in 1..H loop

76. for j in Index loop

77. Buf11(i)(j) := Buf11(i)(j) \* alpha1;

78. end loop;

79. end loop;

80.

81. for i in 1..H loop

82. for j in Index loop

83. tempV:=0;

84. for k in Index loop

85. tempV:=tempV+MB(i)(k)\*MC1(k)(j);

86. Buf12(i)(j):=tempV;

87. end loop;

88. end loop;

89. end loop;

90.

91. for i in 1..H loop

92. for j in Index loop

93. MA(i)(j) := Buf11(i)(j) + Buf12(i)(j);

94. end loop;

95. end loop;

96.

97.

98.

99. --Waiting signal from Task2 (7)

100.

101. Suspend\_Until\_True(S2);

102. -- Output result MA (8)

103. Output\_Matrix(MA);

104. Put\_Line("T1 finished");

105. end T1;

106.

107. task T2;

108. task body T2 is

109. MC2,MR2, Buf21, Buf22 : Matrix;

110.

111. tempV, alpha2:integer;

112.

113.

114. begin

115. Put\_Line("T2 started");

116. -- Waiting to the end of the input data in Task1 (1)

117. Suspend\_Until\_True(S1);

118. -- Copying shared resources (2,3,4)

119. Suspend\_Until\_True(Skd1);

120. MC2 := MC;

121. MR2 := MR;

122. alpha2 := alpha;

123. Set\_True(Skd1);

124.

125. -- Calculation MAH = MBH\*MC2 + (MOH\*MR2)\*a2 (5)

126.

127.

128. for i in H+1..N loop

129. for j in Index loop

130. tempV:=0;

131. for k in Index loop

132. tempV:=tempV+MO(i)(k)\*MR2(k)(j);

133. Buf21(i)(j):=tempV;

134. end loop;

135. end loop;

136. end loop;

137.

138. for i in H+1..N loop

139. for j in Index loop

140. Buf21(i)(j) := Buf21(i)(j) \* alpha2;

141. end loop;

142. end loop;

143.

144. for i in H+1..N loop

145. for j in Index loop

146. tempV:=0;

147. for k in Index loop

148. tempV:=tempV+MB(i)(k)\*MC2(k)(j);

149. Buf22(i)(j):=tempV;

150. end loop;

151. end loop;

152. end loop;

153.

154. for i in H+1..N loop

155. for j in Index loop

156. MA(i)(j) := Buf21(i)(j) + Buf22(i)(j);

157. end loop;

158. end loop;

159.

160.

161.

162.

163. Set\_True(S2);

164. -- Signal to Task1 about end of calculation (6)

165. Put\_Line("T2 finished");

166. end T2;

167. begin

168. null;

169. end;

170.

171. procedure Input\_Vector (M : out Vector)is

172.

173. begin

174. for J in Index loop

175. M(J):=1;

176. end loop;

177. end Input\_Vector;

178.

179. procedure Input\_Matrix (MA : out Matrix)is

180. begin

181. for I in Index loop

182. for J in Index loop

183. MA(I)(J):=1;

184. end loop;

185. end loop;

186. end Input\_Matrix;

187. procedure Output\_Matrix (MA:in Matrix) is

188. begin

189. if N < 6 then

190. for i in Index loop

191. for j in Index loop

192. Put(MA(i)(j));

193. end loop;

194. Put\_line("");

195. end loop;

196. end if;

197.

198. end Output\_Matrix;

199. procedure Vector\_Output (A:in Vector) is

200. begin

201. if N<6 then

202. Put\_Line("");

203. for i in Index loop

204. Put(A(i));

205. end loop;

206. Put\_Line("");

207. end if;

208. end Vector\_Output;

209.

210.

211. begin

212. Put\_Line("Main procedure start");

213. Set\_True(Skd1); -- initial value

214. Start\_tasks;

215. null;

216. end;

217. end Lab1;

218.

218 lines: No errors