Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №5**

*по курсу*

*«Паралельні та розподілені обчислення»*

Виконав:

студент III курсу

гр. IO-82

Куцовол В.В.

Київ 2010

1 using System;

2 using System.Collections.Generic;

3 using System.Linq;

4 using System.Text;

5 using System.Threading;

6

7 namespace Lab5\_PRO

8 {

9 class Program

10 {

11 static void Main(string[] args)

12 {

13 int n = 4, value = 1;

14 F1 Task1 = new F1(n, value);

15 F2 Task2 = new F2(n, value);

16 F3 Task3 = new F3(n, value);

17 Thread Thread1 = new Thread(Task1.run);

18 Thread Thread2 = new Thread(Task2.run);

19 Thread Thread3 = new Thread(Task3.run);

20 Thread1.Priority = ThreadPriority.Normal;

21 Thread2.Priority = ThreadPriority.Lowest;

22 Thread3.Priority = ThreadPriority.Highest;

23 Thread1.Start();

24 Thread2.Start();

25 Thread3.Start();

26 }

27 }

28 public class F1

29 {

30 int[] A, B, C;

31 int d;

32 static int N;

33 public F1(int n, int v)

34 {

35 A = new int[n];

36 B = new int[n];

37 C = new int[n];

38 N = n;

39 for (int i = 0; i < n; i++)

40 {

41 A[i] = v;

42 B[i] = v;

43 C[i] = v;

44 }

45 }

46 public static int getn()

47 {

48 return N;

49 }

50 public void run()

51 {

52 Console.WriteLine("F1 thread started.");

53 d = Functions.f1(A, B, C);

54 Console.WriteLine("F1 : d = " + d);

55 Console.WriteLine("F1 thread finished.");

56 }

57 }

58 public class F2

59 {

60 int[,] MA, MB, MC;

61 int val;

62 public F2(int n, int v)

63 {

64 MA = new int[n, n];

65 MB = new int[n, n];

66 MC = new int[n, n];

67 for (int i = 0; i < n; i++)

68 {

69 for (int j = 0; j < n; j++)

70 {

71 MA[i, j] = v;

72 MB[i, j] = v;

73 MC[i, j] = v;

74 }

75 }

76 }

77 public void run()

78 {

79 Console.WriteLine("F2 thread started.");

80 val = Functions.f2(MA, MB, MC);

81 Console.WriteLine("F2 : v = " + val);

82 Console.WriteLine();

83 Console.WriteLine("F2 thread finished.");

84 }

85 }

86 public class F3

87 {

88 int[,] MA, MB;

89 int[] R, B, C;

90

91 public F3(int n, int v)

92 {

93 MA = new int[n, n];

94 MB = new int[n, n];

95 B = new int[n];

96 C = new int[n];

97 for (int i = 0; i < n; i++)

98 {

99 for (int j = 0; j < n; j++)

100 {

101 MA[i, j] = v;

102 MB[i, j] = v;

103 }

104 B[i] = v;

105 C[i] = v;

106 }

107 }

108 public void run()

109 {

110 Console.WriteLine("F3 thread started.");

111 R = Functions.f3(B, C, MA, MB);

112 Console.Write("F3 : R = ");

113 for (int i = 0; i < R.Length; i++)

114 {

115 Console.Write(R[i] + " ");

116 }

117 Console.WriteLine();

118 Console.WriteLine("F3 thread finished.");

119 }

120 }

121 public class Functions

122 {

123 public static int f1(int[] A, int[] B, int[] C)

124 {

125 return mult(A, B) - mult(C, B);

126 }

127

128 public static int f2(int[,] MA, int[,] MB, int[,] MC)

129 {

130 return maxElMatr(add(MA, multMatrix(MB, MC)));

131 }

132

133 public static int[] f3(int[] B, int[] C, int[,] MA, int[,] MB)

134 {

135 int[] arr = sub(B, C);

136 Array.Sort(arr);

137 return multMonV(sortMatrix(multMatrix(MA, MB)), arr);

138 }

139

140 private static int mult(int[] A, int[] B)

141 {

142 int res = 0;

143 for (int i = 0; i < A.Length; i++)

144 res += A[i] \* B[i];

145 return res;

146 }

147

148 private static int[,] add(int[,] MA, int[,] MB)

149 {

150 int n = F1.getn();

151 int[,] res = new int[n, n];

152 for (int i = 0; i < n; i++)

153 for (int j = 0; j < n; j++)

154 {

155 res[i, j] = MA[i, j] + MB[i, j];

156 }

157 return res;

158 }

159

160 private static int[,] multMatrix(int[,] MB, int[,] MC)

161 {

162 int n = F1.getn();

163 int[,] res = new int[n, n];

164 int s;

165 for (int i = 0; i < n; i++)

166 for (int j = 0; j < n; j++)

167 {

168 s = 0;

169 for (int k = 0; k < n; k++)

170 s = s + MB[i, k] \* MC[k, j];

171 res[i, j] = s;

172 }

173 return res;

174 }

175

176 private static int maxElMatr(int[,] MA)

177 {

178 int res = MA[0, 0];

179 int n = F1.getn();

180 for (int i = 0; i < n; i++)

181 for (int j = 0; i < n; i++)

182 if (res < MA[i, j])

183 res = MA[i, j];

184 return res;

185 }

186

187 private static int[] sub(int[] B, int[] C)

188 {

189 int[] res = new int[B.Length];

190 for (int i = 0; i < B.Length; i++)

191 res[i] = B[i] - C[i];

192 return res;

193 }

194

195 private static int[,] sortMatrix(int[,] MA)

196 {

197 int n = F1.getn();

198 int[,] res = new int[n, n];

199 int buf, k;

200 for (int i = 0; i < n; i++)

201 {

202 for (int j = 0; j < n; j++)

203 {

204 k = j;

205 for (int l = 0; l < n; l++)

206 if (MA[i, k] < MA[i, l])

207 k = l;

208 buf = MA[i, k];

209 MA[i, k] = MA[i, j];

210 MA[i, j] = buf;

211 }

212 }

213 return res;

214 }

215

216 private static int[] multMonV(int[,] MA, int[] A)

217 {

218 int[] res = new int[A.Length];

219 int s;

220 for (int i = 0; i < A.Length; i++)

221 {

222 s = 0;

223 for (int j = 0; j < A.Length; j++)

224 s += MA[i, j] \* A[j];

225 res[i] = s;

226 }

227 return res;

228 }

229 }

230}