М І Н І С Т Е Р С Т В О О С В І Т И І Н А У К И У К Р А Ї Н И  
Національний технічний університет України

"Київський політехнічний інститут"

Кафедра обчислювальної техніки

# **РОЗРАХУНКОВО-ГРАФІЧНА РОБОТА**

з дисципліни

**"Паралельні та розподілені обчислення"**

Керівник роботи Виконавець роботи

Доц. Корочкін А.В. ст. Куцовол В.В.

Допущена до захисту група № ІО-82

\_\_\_\_\_\_\_\_\_\_»\_\_» 2010 р.

Захищена

\_\_\_\_\_\_\_\_\_\_\_»\_\_» 2010 р.

Київ - 2010 р.

М I Н I С Т Е Р С Т В О О С В I Т И I Н А У К И У К Р А I Н И   
Національний технічний університет України

"Київський політехнічний інститут"

Кафедра обчислювальної техніки

Факультет інформатики та обчислювальної техніки.

Кафедра обчислювальної техніки.

# ЗАВДАННЯ НА РОЗРАХУНКОВО-ГРАФІЧНУ РОБОТУ

з дисципліни паралельні та розподілені обчислення.

Група ІО-82 студент Куцовол В.В. Термін отримання\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Термін виконання\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Керівник роботи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

НAЙМЕНУВАННЯ ЗАВДАННЯ

Розробити програму, що містить набір паралельни*х* задач (T1 - T6) для реалізації функцій F1, F2, F3 з лабораторної роботи номер 1, а також задач для введення і виведення даних і результатів для цих функцій. Організувати запуск вказаних задач, а також передачу необхідних даних і результатів між відповідними задачами за допомогою механізму рандеву.

ПЕРЕЛІК ГРАФІЧНОЇ ДОКУМЕНТАЦIЇ

Структурна схема взаємодії задач.

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Бар Р. Язык Ада в проектировании систем. - М.; Мир, 1988.- 320 с.

2. Пайл Я. Ада - язык встроенных систем. -М.; Финансы и статистика, 1984. - 120 с.

3. Джехани Н. Язык Ада. - М.; Мир, 1988.- 552 с.

4. Василеску Ю. Прикладное программирование на языке Ада.- М.:Мир,1990. - 332 с.

5. Вегнер П. Программирование на языке Ада. - М.; Мир, 1983.- 78 с.

6. Органик Э. Организация системы ИНТЕЛ - 432. - М.; Мир, 1987.-224 с.

7. Корочкин А.В. Ада 95: Введение в программирование. - Киев; Свит, 1998.- 260 с.

8. Богачев К.Ю. Основы параллельного программирования. – М.: БИНОМ. Лаб зна-

ний, 2003. – 342 с .

9. Корочкин А., Мустафа Акрам Параллельные вычисления: Ада и Java. – Вісн. НТУУ “КПІ”, Інформатика, управління та обчис лювальна техніка, 1999, К.: – № 32, С. 13–17.

ПІДПИС КЕРІВНИКА РОБОТИ\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Розширене технічне завдання**

**Варіант 17**

Вид функцій F1, F2, F3 відповідає варіантам відповідно 1.18, 2.19 та 3.20:

F1: d = (A\*B) - (C\*B)

F2: v = MAX(MA + MB\*MC)

F3: R = SORT(B - C)\*SORT(MA\*MB)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Варіант РГР** | **Задача Т1** | **Задача Т2** | **Задача Т3** | **Задача Т4** | **Задача Т5** | **Задача Т6** |
| 17 | Вивід результатів | F2 | F3  F1 | Введення даних | - | - |

**Виконання розрахунково-графічної роботи**

**1. Розробка алгоритмів задач**

**Задача Т1**

1. Прийняти від задачі Т2 результат: скаляр v.
2. Виведення результату: скаляр v.
3. Прийняти від задачі Т3 результат: вектор R.
4. Виведення результату: вектор R .
5. Прийняти від задачі Т3 результат: скаляр d.
6. Виведення результату: скаляр d .

**Задача Т2**

1. Прийняти від задачі Т4 матриць MA, MB, MС.
2. Обчислення функції F2: v = MAX(MA + MB\*MC).
3. Передати в задачу Т1 результат : скаляр v.

**Задача Т3**

1. Прийняти від задачі Т4 векторів B, C, матриць MA, MB.
2. Обчислення функції:

F3: R = SORT(B - C)\*SORT(MA\*MB)

1. Передати в задачу Т1 результат : вектор R.
2. Прийняти від задачі Т4 векторів A, B, C.
3. Обчислення функцій:

F1: d = (A\*B) - (C\*B)

1. Передати в задачу Т1 результат : скаляр d.

**Задача Т4**

1. Введення матриць MA, MB, MC.
2. Передати задачі Т2 матриці MA, MB, MC.
3. Введення векторів B, C та матриць MA, MB.
4. Передати задачі Т3 векторів B, C та матриць MA, MB.
5. Введення векторів A, B, C.
6. Передати задачі Т3 векторів A, B, C.

**2. Розробка схеми взаємодії задач**



*Рис. 2.1. Структурна схема взаємодії задач*

**3. Розробка програми**

Програма складається з трьох частин:

1. Специфікація пакету Pack, що містить опис приватних типів Vector та Matrix, та специфікацію процедур, що реалізують ввід, виведення даних і виконання функцій F1, F2 та F3.
2. Тіло пакету Pack, що містить реалізацію процедур вводу, виведення даних і виконання функцій F1, F2 та F3. Ці процедури використовують для своїх обчислень приватні типи Vector та Matrix.
3. Основний модуль (randevu.ada), що містить головну процедуру Randevu, в декларативній частині якій розміщуються специфікації та тіла задач Т1-Т4. Задачі виконують дії відповідно до алгоритмів представлених в пункті 1. Крім того, в усіх описаних задачах реалізовано вивід повідомлень про запуск та закінчення задачі. В специфікаціях задач Т1 та Т4 описані входи, що використовуються для реалізації взаємодії задач. В тілах задач Т1 та Т4 розміщені оператори ассеpt, що реалізують відповідні входи.

Лістинг приведено у Додатку А.

**Висновки**

1. Механізм рандеву дозволяє не тільки передавати дані між задачами, а й синхронізувати їх.
2. Для ефективнішої роботи програми реалізовані такі тіла операторів accept, що виконують лише операції присвоювання.

**Список використаної літератури**

1. Жуков І.А., Корочкін О.В. Паралельні та розподілені обчислення. - К.:Корнійчук, 2005. -224 с.
2. Жуков И.А., Корочкин А.В. Параллельные и распределенныевычисления. Лабораторный практикум. – К.:Корнейчук, 2008.- 226 с.
3. Корочкин А.В. Ада 95: Введение в программирование. - Киев; Свит, 1998.- 260 с.

**Додаток А. Лістинг програми**

Source file: ..\randevu.ada Thu Dec 01 02:59:59 2010

1 --- РГР ------------

2 -- F1: 1.18 d=(A\*B)-(C\*B) ------------

3 -- F2: 2.19 v=MAX(MA+MB\*MC) ----------

4 -- F3: 3.20 R=SORT(B-C)\*SORT(MA\*MB) --

5 -- Виконав студент групи ІО-82, ФІОТ -

6 -- Куцовол Віктор Вікторович ---------

7

8 with Ada.Text\_IO, Ada.Integer\_Text\_IO, Pack;

9 use Ada.Text\_IO, Ada.Integer\_Text\_IO, Pack;

10

11 procedure Randevu is

12

13 task T4 is

14 entry InputDataF1(A,B,C: out Vector);

15 entry InputDataF2(MA,MB,MC: out Matrix);

16 entry InputDataF3(B,C: out Vector; MA,MB: out Matrix);

17 end T4;

18

19 task body T4 is

20 A\_buf,B\_buf,C\_buf:Vector;

21 MA\_buf,MB\_buf,MC\_buf:Matrix;

22 begin

23 Put\_Line("T4 started");

24 -- Put\_Line("Input Vector A(F1): ");

25 Vector\_Input(A\_buf,1);

26 -- Put\_Line("Input Vector B(F1): ");

27 Vector\_Input(B\_buf,1);

28 -- Put\_Line("Input Vector C(F1): ");

29 Vector\_Input(C\_buf,1);

30 -- Put\_Line("Input Matrix MA(F2): ");

31 MAtrix\_Input(MA\_buf,1);

32 -- Put\_Line("Input Matrix MB(F2): ");

33 Matrix\_Input(MB\_buf,1);

34 -- Put\_Line("Input Matrix MC(F2): ");

35 Matrix\_Input(MC\_buf,1);

36 for i in 1..3 loop

37 select

38 accept InputDataF1(A,B,C: out Vector) do

39 A:=A\_buf;

40 B:=B\_buf;

41 C:=C\_buf;

42 end InputDataF1;

43 or

44 accept InputDataF2(MA,MB,MC: out Matrix) do

45 MA:=MA\_buf;

46 MB:=MB\_buf;

47 MC:=MC\_buf;

48 end InputDataF2;

49 or

50 accept InputDataF3(B,C: out Vector; MA,MB: out Matrix) do

51 B:=B\_buf;

52 C:=C\_buf;

53 MA:=MA\_buf;

54 MB:=MB\_buf;

55 end InputDataF3;

56 end select;

57 end loop;

58 Put\_Line("T4 finished");

59 end T4;

60

61 task T1 is

62 entry ResultOutF1(d: in Integer);

63 entry ResultOutF2(v: in Integer);

64 entry ResultOutF3(R: in Vector);

65 end T1;

66

67 task T2;

68 task body T2 is

69 MA,MB,MC:Matrix;

70 v:Integer;

71 begin

72 Put\_Line("T2 started");

73 T4.InputDataF2(MA,MB,MC);

74 Func2(MA,MB,MC,v);

75 T1.ResultOutF2(v);

76 Put\_Line("T2 finished");

77 end T2;

78

79 task T3;

80 task body T3 is

81 A,B,C:Vector;

82 MA,MB:Matrix;

83 R:Vector;

84 d:Integer;

85 begin

86 Put\_Line("T3 started");

87 T4.InputDataF3(B,C,MA,MB);

88 Func3(B,C,MA,MB,R);

89 T1.ResultOutF3(R);

90 T4.InputDataF1(A,B,C);

91 Func1(A,B,C,d);

92 T1.ResultOutF1(d);

93 Put\_Line("T3 finished");

94 end T3;

95 task body T1 is

96 s:Integer;

97 Vec:Vector;

98 begin

99 Put\_Line("T1 started");

100 for i in 1..3 loop

101 select

102 accept ResultOutF1(d: in Integer) do

103 s:=d;

104 end ResultOutF1;

105 Put("F1: d=");

106 Put(s);

107 New\_Line;

108 or

109 accept ResultOutF2(v: in Integer) do

110 s:=v;

111 end ResultOutF2;

112 Put\_Line("F2: v=");

113 Put(s);

114 New\_Line;

115 or

116 accept ResultOutF3(R: in Vector) do

117 Vec:=R;

118 end ResultOutF3;

119 Put("F3: R=");

120 Vector\_Output(Vec);

121 end select;

122 end loop;

123 Put\_Line("T1 finished");

124 end T1;

125 begin

126 null;

127 end Randevu;

Source file: ..\pack.ads Thu Dec 01 02:59:55 2010

1

2 --- РГР ------------

3 -- Специфікація пакету ---------------

4 -- F1: 1.18 d=(A\*B)-(C\*B) ------------

5 -- F2: 2.19 v=MAX(MA+MB\*MC) ----------

6 -- F3: 3.20 R=SORT(B-C)\*SORT(MA\*MB) --

7 -- Виконав студент групи ІО-82, ФІОТ -

8 -- Куцовол Віктор Вікторович ---------

9

10 package Pack is

11

12 n:integer:=500;

13 type Vector is private;

14 type Matrix is private;

15

16 procedure Vector\_Input(V:out Vector);

17 procedure Vector\_Input(V:out Vector; Value:in Integer);

18 procedure Vector\_Output(V:in Vector);

19 procedure Matrix\_Input(MA:out Matrix);

20 procedure Matrix\_Input(MA:out Matrix; Value:in Integer);

21 procedure Matrix\_Output(MA:in Matrix);

22 procedure Func1(A,B,C:in Vector; d:out Integer);

23 procedure Func2(MA,MB,MC:in Matrix; v:out Integer);

24 procedure Func3(B,C: in Vector; MA,MB: in Matrix; R: out Vector);

25

26 private

27 type Vector is array(1..n) of Integer;

28 type Matrix is array(1..n) of Vector;

29 end Pack;

Source file: ..\pack.adb Thu Dec 01 02:59:52 2010

1

2 --- РГР ------------

3 -- Тіло пакету -----------------------

4 -- F1: 1.18 d=(A\*B)-(C\*B) ------------

5 -- F2: 2.19 v=MAX(MA+MB\*MC) ----------

6 -- F3: 3.20 R=SORT(B-C)\*SORT(MA\*MB) --

7 -- Виконав студент групи ІО-82, ФІОТ -

8 -- Куцовол Віктор Вікторович ---------

9

10 with Ada.Text\_IO, Ada.Integer\_Text\_IO;

11 use Ada.Text\_IO, Ada.Integer\_Text\_IO;

12

13 package body Pack is

14

15 -- Процедури введення вектора

16 procedure Vector\_Input(V:out Vector) is

17 flag: Boolean;

18 begin

19 for i in 1..n loop

20 flag := false;

21 loop

22 begin

23 Get(V(i));

24 flag := true;

25 exception

26 when Data\_Error =>

27 Put("Data error! reInput.");

28 New\_Line;

29 Skip\_Line;

30 end;

31 exit when flag;

32 end loop;

33 end loop;

34 end Vector\_Input;

35

36 procedure Vector\_Input(V:out Vector; Value:in Integer) is

37 begin

38 for i in 1..n loop

39 V(i) := Value;

40 end loop;

41 end Vector\_Input;

42

43 -- Процедура виведення вектора

44 procedure Vector\_Output (V:in Vector) is

45 begin

46 for i in 1..n loop

47 Put(V(i));

48 end loop;

49 New\_Line;

50 end Vector\_Output;

51

52 -- Процедури введеня матриці

53 procedure Matrix\_Input(MA:out Matrix) is

54 flag: Boolean;

55 begin

56 for i in 1..n loop

57 for j in 1..n loop

58 flag := false;

59 loop

60 begin

61 Get(MA(i)(j));

62 flag := true;

63 exception

64 when Data\_Error =>

65 Put("Data error! Input again.");

66 New\_line;

67 Skip\_line;

68 end;

69 exit when flag;

70 end loop;

71 end loop;

72 end loop;

73 end Matrix\_Input;

74

75 procedure Matrix\_Input(MA:out Matrix; Value:in Integer) is

76 begin

77 for i in 1..n loop

78 for j in 1..n loop

79 MA(i)(j) := Value;

80 end loop;

81 end loop;

82 end Matrix\_Input;

83

84 -- Процедура виведення матриці

85 procedure Matrix\_Output (MA:in Matrix) is

86 begin

87 for i in 1..n loop

88 for j in 1..n loop

89 Put(MA(i)(j));

90 end loop;

91 end loop;

92 end Matrix\_Output;

93

94 -- Процедура "функція 1": операція з векорами

95 procedure Func1(A,B,C:in Vector; d:out Integer) is

96 -- Функція скалярного добутку векторів

97 function MulVec (A,B:Vector) return Integer is

98 s:Integer:=0;

99 begin

100 for i in 1..n loop

101 s:=s+A(i)\*B(i);

102 end loop;

103 return s;

104 end MulVec;

105 begin

106 for i in 1..n loop

107 d:=MulVec(a,b)-MulVec(c,b);

108 end loop;

109 end Func1;

110

111 -- Процедура "функція 2": операція з матрицями

112 procedure Func2(MA,MB,MC:in Matrix; v:out Integer) is

113 -- Функція знаходження максимуму серед елементів матриці

114 function MaxElMatr (MD:Matrix) return Integer is

115 m:Integer:=MD(1)(1);

116 begin

117 for i in 1..n loop

118 for j in 1..n loop

119 if m<MD(i)(j) then

120 m:=MD(i)(j);

121 end if;

122 end loop;

123 end loop;

124 return m;

125 end MaxElMatr;

126 -- Функція добутку двох матриць

127 function MultMatr (mb,mc:Matrix) return Matrix is

128 s:integer;

129 MT:Matrix;

130 begin

131 for i in 1..n loop

132 for j in 1..n loop

133 s:=0;

134 for k in 1..n loop

135 s:=s+MB(i)(k)\*MC(k)(j);

136 MT(i)(j):=s;

137 end loop;

138 end loop;

139 end loop;

140 return MT;

141 end MultMatr;

142

143 -- Функція суми двох матриць

144 function AddMatr (ma,mb:Matrix) return Matrix is

145 m:Matrix;

146 begin

147 for i in 1..n loop

148 for j in 1..n loop

149 m(i)(j):=ma(i)(j)+mb(i)(j);

150 end loop;

151 end loop;

152 return m;

153 end AddMatr;

154 begin

155 v:=MaxElMatr(AddMatr(MA,MultMatr(MB,MC)));

156 end Func2;

157

158 -- Процедура "функція 3": операції з векторами та матрицями

159 procedure Func3(B,C: in Vector; MA,MB: in Matrix; R: out Vector) is

160 -- Функція сортування елементів вектора за зростанням

161 function SortVector(A:Vector) return Vector is

162 M:Vector:=A;

163 buf:Integer;

164 min,k:Integer;

165 begin

166 for i in 1..(n-1) loop

167 k:=i;

168 for j in (i+1)..n loop

169 if M(k)>M(j) then

170 k:=j;

171 end if;

172 end loop;

173 buf:=M(k);

174 M(k):=M(i);

175 M(i):=buf;

176 end loop;

177 return M;

178 end SortVector;

179

180 -- Функція сортування елементів вектора за спаданням

181 function SortVector2(A:Vector) return Vector is

182 M:Vector:=A;

183 buf:Integer;

184 min,k:Integer;

185 begin

186 for i in 1..(n-1) loop

187 k:=i;

188 for j in (i+1)..n loop

189 if M(k)<M(j) then

190 k:=j;

191 end if;

192 end loop;

193 buf:=M(k);

194 M(k):=M(i);

195 M(i):=buf;

196 end loop;

197 return M;

198 end sortVector2;

199

200 -- Функція сортування рядків матриці за спаданням

201 function SortMatrix(MA:Matrix) return Matrix is

202 MT:Matrix;

203 begin

204 for i in 1..n loop

205 MT(i):=sortVector2(MA(i));

206 end loop;

207 return MT;

208 end SortMatrix;

209

210 -- Функція різниці двох векторів

211 function SubVec(B,C:Vector) return Vector is

212 K:Vector;

213 begin

214 for i in 1..n loop

215 K(i):=B(i)-C(i);

216 end loop;

217 return K;

218 end SubVec;

219

220 -- Функція добутку двох матриць

221 function MultMatr (mb,mc:Matrix) return Matrix is

222 s:integer;

223 MT:Matrix;

224 begin

225 for i in 1..n loop

226 for j in 1..n loop

227 s:=0;

228 for k in 1..n loop

229 s:=s+MB(i)(k)\*MC(k)(j);

230 MT(i)(j):=s;

231 end loop;

232 end loop;

233 end loop;

234 return MT;

235 end MultMatr;

236

237 -- Функція множення матриці на вектор

238 function multMatrixOnVector(MA:Matrix; A:Vector) return Vector is

239 s:Integer;

240 begin

241 for i in 1..n loop

242 s:=0;

243 for j in 1..n loop

244 s:=s+MA(i)(j)\*A(j);

245 end loop;

246 R(i):=s;

247 end loop;

248 return R;

249 end multMatrixOnVector;

250

251 begin

252 R:=multMatrixOnVector(SortMatrix(MultMatr(MA,MB)),SortVector(SubVec(B,C)));

253 end Func3;

254

255 -- Головна програма

256 begin

257 null;

258 end Pack;