Национальный технический университет

«Киевский политехнический институт»

Факультет информатики и вычислительной техники

## Кафедра Вычислительной Техники

#### Лабораторная работа №2

По курсу: Параллельное программирование

студента III-го курса,

группы ИВ-92

Петрука В.О.

## Киев 2011 г.

Source file: ..\main.ada Mon Sep 19 23:46:15 2011

1 -- Laboratorna robota 1

2 -- Petruk Vadim, IO-92

3 -- Variant: 1.13 C = A - B + D

4 -- 2.28 MD = MIN(MA)\*MB\*MC

5 -- 3.14 D = (A + B)\*(MA - MB)

6 ---------------------------------------

7 with Ada.Integer\_Text\_IO, Ada.Text\_iO, Unit;

8 use Ada.Integer\_Text\_IO, Ada.Text\_IO, Unit;

9 ---------------------------------------

10 procedure main is

11

12 task Task1 is

13 pragma Priority(15);

14 end Task1;

15 task body Task1 is

16 A:Vector;

17 B:Vector;

18 C:Vector;

19 D:Vector;

20 begin

21 VectorIn(A);

22 VectorIn(B);

23 VectorIn(D);

24 Func1(A,B,D,C);

25 delay 0.3;

26 put\_Line("Func1: C:=A-B+D ");

27 VectorOut(C);

28 end Task1;

29 --------------------------------------

30 task Task2 is

31 pragma Priority(3);

32 end Task2;

33 task body Task2 is

34 MA:Matrix;

35 MB:Matrix;

36 MC:Matrix;

37 MD:Matrix;

38 begin

39 matrixIn(MA);

40 MatrixIn(MB);

41 MatrixIn(MC);

42 func2(MA,MB,MC,MD);

43 delay 0.1;

44 put\_Line("Func2: MD := MIN(MA)\*MB\*MC");

45 MatrixOut(MD);

46 end Task2;

47 --------------------------------------

48 task Task3 is

49 pragma Priority(10);

50 end Task3;

51 task body Task3 is

52 A:Vector;

53 B:Vector;

54 D:Vector;

55 MA:Matrix;

56 MB:Matrix;

57 MD:Matrix;

58 begin

59 VectorIn(A);

60 VectorIn(B);

61 MatrixIn(MA);

62 MatrixIn(MB);

63 func3(A,B,MA,MB,D);

64 delay 0.2;

65 put\_Line("Func3: D := (A + B)\*(MA - MB)");

66 vectorOut(D);

67 end Task3;

68 --------------------------------------

69 begin

70 null;

71 end Main;

Source file: ..\unit.ads Mon Sep 19 23:46:16 2011

1 with Ada.text\_iO, Ada.Integer\_Text\_IO;

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

3

4 package Unit is

5

6 N:constant := 3;

7

8 type Vector is private;

9 type Matrix is private;

10

11 procedure VectorIn (X : out Vector);

12 procedure VectorOut (X : in Vector );

13 procedure MatrixIn (MX : out Matrix);

14 procedure MatrixOut (MX : in Matrix);

15 procedure Func1 (A, B, D : in Vector; C : out Vector);

16 procedure Func2 (MA, MB, MC: in Matrix; MD : out Matrix);

17 procedure Func3(A, B: in Vector; MA, MB: in Matrix; D: out vector);

18

19 private

20 type Vector is array (1..N) of Integer;

21 type Matrix is array (1..N) of Vector;

22

23 end Unit;

Source file: ..\unit.adb Mon Sep 19 23:46:15 2011

1 package body Unit is

2 ---------------------------------------

3 procedure VectorIn (x : out Vector) is

4 begin

5 for i in 1..N loop

6 x(i):=1;

7 end loop;

8 end VectorIn;

9 ---------------------------------------

10 procedure VectorOut (x : in Vector ) is

11 begin

12 for i in 1..N loop

13 put(x(i));

14 end loop;

15 New\_line;

16 end VectorOut;

17 ---------------------------------------

18 procedure MatrixIn (MX : out Matrix) is

19 begin

20 for i in 1..N loop

21 for j in 1..n loop

22 MX(i)(j):=1;

23 end loop;

24 end loop;

25 end MatrixIn;

26 ---------------------------------------

27 procedure MatrixOut (MX : in Matrix) is

28 begin

29 for i in 1..N loop

30 for j in 1..n loop

31 put(MX(i)(j));

32 if (j=n) then

33 New\_line;

34 end if;

35 end loop;

36 end loop;

37 New\_line;

38 end MatrixOut;

39 ---------------------------------------

40 --Func1: C:=A-B+D

41 procedure Func1 (A : in Vector;

42 b : in Vector;

43 d : in Vector;

44 c : out Vector) is

45 begin

46 for i in 1..n loop

47 c(i) := a(i)-B(i)+D(i);

48 end loop;

49 end Func1;

50 ---------------------------------------

51 --Func2: MD := MIN(MA)\*MB\*MC

52 procedure Func2 (MA : in Matrix;

53 MB : in Matrix;

54 MC : in Matrix;

55 MD : out Matrix) is

56 min:Integer;

57 begin

58 --MD := (0)

59 for i in 1..n loop

60 for j in 1..n loop

61 MD(i)(j) := 0;

62 end loop;

63 end loop;

64 --MD := MB\*MC

65 for i in 1..n loop

66 for j in 1..n loop

67 for k in 1..n loop

68 MD(i)(j) := MD(i)(j)+MB(i)(k)\*MC(k)(j);

69 end loop;

70 end loop;

71 end loop;

72 --serching min(MA)

73 min := Ma(1)(1);

74 for i in 1..N loop

75 for j in 1..n loop

76 if MA(i)(j)<min then

77 min := MA(i)(j);

78 end if;

79 end loop;

80 end loop;

81 -- MD := Min(MA)\*MB\*MC

82 for i in 1..N loop

83 for j in 1..n loop

84 MD(i)(j) := MD(i)(j)\*Min;

85 end loop;

86 end loop;

87 end Func2;

88 ---------------------------------------

89 --Func3: D := (A + B)\*(MA - MB)

90 procedure Func3(A: in Vector;

91 B: in Vector;

92 MA: in Matrix;

93 MB: in Matrix;

94 D: out vector) is

95 C:Vector;

96 MC:Matrix;

97 begin

98 --C:=A + B

99 for i in 1..N loop

100 C(i) := A(i) + B(i);

101 end loop;

102 --MC:=MA - MB

103 for i in 1..N loop

104 for j in 1..n loop

105 MC(i)(j) := MA(i)(j) - MB(i)(j);

106 end loop;

107 end loop;

108 --D := (0)

109 for i in 1..n loop

110 d(i) := 0;

111 end loop;

112 --D := C \* MC

113 for i in 1..n loop

114 for j in 1..n loop

115 D(j):= D(j) + C(j)\*MC(j)(i);

116 end loop;

117 end loop;

118 end func3;

119 ---------------------------------------

120 end Unit;