**Додаток Є. Лістинг коду програми ПРГ1**

GNAT 4.6.4

Copyright 1992-2010, Free Software Foundation, Inc.

Compiling: matrixoperations.adb (source file time stamp: 2016-03-19 18:55:56)

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

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

3. ------------------------------------------------------------------

4. -- File: matrixOperations.adb --

5. -- --

6. -- Автор: Олег Кахерський, група ІП-31 --

7. -- Дата: 12.03.2016 --

8. ------------------------------------------------------------------

9. package body MatrixOperations is

10.

11. procedure Output\_Matrix (MA : Matrix) is

12. begin

13. New\_Line;

14. for I in Index loop

15. for J in Index loop

16. Put(Item => MA(i)(j), Width => 5);

17. end loop;

18. New\_line;

19. end loop;

20. end Output\_Matrix;

21.

22. procedure Matrix\_Matrix\_Multiply

23. (Left : Matrix;

24. Right : Matrix;

25. Start\_Index: Integer;

26. End\_Index: Integer;

27. const: Integer;

28. Result\_Matrix: out Matrix) is

29. begin

30. for i in Start\_Index..End\_Index loop

31. for J in Index loop

32. Result\_Matrix(I)(J) := 0;

33. for K in Index loop

34. Result\_Matrix(I)(J) := Result\_Matrix(I)(J) + Left(I)(K) \* Right(K)(J);

35. end loop;

36. Result\_Matrix(I)(J) := Result\_Matrix(I)(J) \* const;

37. end loop;

38. end loop;

39. end Matrix\_Matrix\_Multiply;

40.

41. procedure Matrix\_Matrix\_Add

42. (Left: Matrix;

43. Right: Matrix;

44. Start\_Index: Integer;

45. End\_Index: Integer;

46. const: Integer;

47. Result\_Matrix: out Matrix) is

48. begin

49. for I in Start\_Index .. End\_Index loop

50. for J in Index loop

51. Result\_Matrix(I)(J) := Left(I)(J) + Right(I)(J)\*const;

52. end loop;

53. end loop;

54. end Matrix\_Matrix\_Add;

55.

56. function Min

57. (Vect: Vector;

58. Start\_Index: Integer;

59. End\_Index: Integer) return Integer is

60. min: Integer;

61. begin

62. min := Vect(Start\_Index);

63. for I in Start\_Index .. End\_Index loop

64. if min > Vect(I) then

65. min := Vect(I);

66. end if;

67. end loop;

68. return min;

69. end Min;

70.

71. function Input\_Vector return Vector is

72. V: Vector;

73. begin

74. for I in Index loop

75. V(I) := 1;

76. end loop;

77. return V;

78. end Input\_Vector;

79.

80. function Input\_Constant return Integer is

81. begin

82. return 1;

83. end Input\_Constant;

84.

85. function Input\_Matrix return Matrix is

86. Result\_Matrix: Matrix;

87. begin

88. for I in Index loop

89. for J in Index loop

90. Result\_Matrix(I)(J) := 1;

91. end loop;

92. end loop;

93. return Result\_Matrix;

94. end Input\_Matrix;

95.

96. end MatrixOperations;

Compiling: matrixoperations.ads (source file time stamp: 2016-03-15 20:00:35)

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

2. -- File: matrixOperations.ads --

3. -- --

4. -- Автор: Олег Кахерський, група ІП-31 --

5. -- Дата: 12.03.2016 --

6. ------------------------------------------------------------------

7. generic

8. N: in Integer; --розмірність масивів та матриць

9. package MatrixOperations is

10. type Vector is private;

11. type Matrix is private;

12.

13. --Вивід матриці на екран

14. procedure Output\_Matrix (MA : Matrix);

15.

16. --Множення матриці на матрицю(діапазон рядків StartIndex..EndIndex),

17. --результат множення множиться на константу

18. procedure Matrix\_Matrix\_Multiply

19. (Left : Matrix;

20. Right : Matrix;

21. Start\_Index: Integer;

22. End\_Index: Integer;

23. const: Integer; --константа, на яку множиться результат множення

24. Result\_Matrix: out Matrix);

25.

26. --Додавання матриць, Right(I)(J) перед додаванням множиться на константу

27. procedure Matrix\_Matrix\_Add

28. (Left: Matrix;

29. Right: Matrix;

30. Start\_Index: Integer;

31. End\_Index: Integer;

32. const: Integer;

33. Result\_Matrix: out Matrix);

34.

35. --Знаходження мінімального значення в діапазоні StartIndex, EndIndex

36. function Min

37. (Vect: Vector;

38. Start\_Index: Integer;

39. End\_Index: Integer) return Integer;

40.

41. --Ввід вектора

42. function Input\_Vector return Vector;

43.

44. --Ввід константи

45. function Input\_Constant return Integer;

46.

47. --Ввід матриці

48. function Input\_Matrix return Matrix;

49.

50. private

51. subtype Index is Integer range 1..N;

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

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

54. end MatrixOperations;

96 lines: No errors