Міністерство освіти і науки, молоді та спорту України

Національний технічний університет України

«Київський політехнічний інститут»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

**Лабораторна робота №1**

З дисципліни «ПРО»

Виконав: Перевірив:

Студент групи ІО-21 доц. Корочкін О.В.

Коноз А.О.

Дата здачі\_\_\_\_\_\_\_\_\_\_\_\_\_

Захищено з балом\_\_\_\_\_

Київ 2014

**Завдання**

1.23 2.13 3.17

1. F1: E:=A+B+C+D(MA\*MZ).

2. F2: MZ:=Min(MA)\*MB+Max(MT)\*(MX\*MZ).

3. F3: d:=Min(A\*Trans(MB\*MM)+B\*Sort(MC))

**Лістинг коду**

GNAT GPL 2014 (20140331)

Copyright 1992-2014, Free Software Foundation, Inc.

Compiling: lab1.adb (source file time stamp: 2014-09-16 18:47:14)

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

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

3. -------------Основна програма---------------------------------

4. --ПРО Lab1

5. --Коноз Андрій Олександрович

6. --ІО-21

7. --13.09.14

8. --F1: E:=A+B+C+D(MA\*MZ)

9. --F2: MZ:=Min(MA)\*MB+Max(MT)\*(MX\*MZ)

10. --F3: d:=Min(A\*Trans(MB\*MM)+B\*Sort(MC))

11.

12. procedure Lab1 is

13. --Розмірність векторів і матриць

14. n : Integer :=3;

15. --Ініціалізація пакету

16. package Lab1Data is new Data(n);

17. use Lab1Data;

18. --Оголошення змінних, які будуть використовуватися в програмі

19. A,B,C,D,E: Vect;

20. MA,MB,MT,MX,MZ,MM,ME,MC: Matrix;

21. y: Integer;

22.

23. begin

24. --Введення даних для F1

25. Put("All vectors had ");

26. Put(n);

27. Put(" items");

28. New\_Line;

29.

30. Put("Arguments for F1");

31. Put("Enter vector A");

32. New\_Line;

33. readVect(A);

34.

35. New\_Line;

36. Put("Enter vector B");

37. New\_Line;

38. readVect(B);

39.

40. New\_Line;

41. Put("Enter vector C");

42. New\_Line;

43. readVect(C);

44.

45. New\_Line;

46. Put("Enter vector D");

47. New\_Line;

48. readVect(D);

49.

50. New\_Line;

51. Put("Enter MA");

52. New\_Line;

53. readMatr(MA);

54.

55. New\_Line;

56. Put("Enter MZ");

57. New\_Line;

58. readMatr(MZ);

59. --Виконання функції F1

60. F1(A,B,C,D,MA, MZ,E);

61. --Введення даних для F2

62. Put("Arguments for F2");

63. New\_Line;

64. Put("Enter MA");

65. New\_Line;

66. readMatr(MA);

67.

68. New\_Line;

69. Put("Enter MB");

70. New\_Line;

71. readMatr(MB);

72.

73. New\_Line;

74. Put("Enter MX");

75. New\_Line;

76. readMatr(MX);

77.

78. New\_Line;

79. Put("Enter MZ");

80. New\_Line;

81. readMatr(MZ);

82.

83. New\_Line;

84. Put("Enter MT");

85. New\_Line;

86. readMatr(MT);

87. --Виконання функції F2

88. F2(MA,MB,MT,MX,MZ,ME);

89. --Введення даних для F3

90. Put("Arguments for F3");

91. Put("Enter vector A");

92. New\_Line;

93. readVect(A);

94.

95. New\_Line;

96. Put("Enter vector B");

97. New\_Line;

98. readVect(B);

99.

100. New\_Line;

101. Put("Enter MB");

102. New\_Line;

103. readMatr(MB);

104.

105. New\_Line;

106. Put("Enter MM");

107. New\_Line;

108. readMatr(MM);

109.

110. New\_Line;

111. Put("Enter MC");

112. New\_Line;

113. readMatr(MC);

114. --Виконання функції F3

115. F3(A,B,MB, MM,MC,y);

116. --Виведення результатів функцій

117. Put("F1 = ");

118. showVect(E);

119. Put("F2 = ");

120. showMatr(ME);

121. Put("F3 = ");

122. Put(y);

123.

124. end Lab1;

124 lines: No errors

GNAT GPL 2014 (20140331)

Copyright 1992-2014, Free Software Foundation, Inc.

Compiling: data.adb (source file time stamp: 2014-09-17 04:46:16)

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

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

3. with Ada.Streams; use Ada.Streams;

4. ---------Тіло пакету Data--------------

5. package body Data is

6. --Функція виконує додавання двох векторів

7. function AddVect (A, B :Vect) return Vect is

8. g : Vect;

9. begin

10. for index in 1..N loop

11. g(index):=A(index)+B(index);

12. end loop;

13. return g;

14. end AddVect;

15. --Функція виконує множення двох матриць

16. function MultMatr (A, B : Matrix) return Matrix is

17. Cell : Integer;

18. Result : Matrix;

19. begin

20. for i in 1..N loop

21. for j in 1..N loop

22. Cell := 0;

23. for k in 1..N loop

24. Cell := Cell + A(i,k)\*B(k,j);

25. end loop;

26. Result(i,j):= Cell;

27. end loop;

28. end loop;

29. return Result;

30. end MultMatr;

31. --Функція транспонує матрицю

32. function Transp (A : Matrix) return Matrix is

33. Result : Matrix;

34. begin

35. for i in 1..N loop

36. for j in 1..N loop

37. Result(i,j) := A(j,i);

38. end loop;

39. end loop;

40. return Result;

41. end Transp;

42. --Функція виконує пошук максимального елементу матриці

43. function SearchMax(A : Matrix) return Integer is

44. Var : Integer := A(1,1);

45. begin

46. for i in 1..N loop

47. for j in 1..N loop

48. if A(i,j) > Var then

49. Var := A(i,j);

50. end if;

51. end loop;

52. end loop;

53. return Var;

54. end SearchMax;

55. --Функція виконує пошук мінімального елемента матриці

56. function SearchMin(A : Matrix) return Integer is

57. Var : Integer := A(1,1);

58. begin

59. for i in 1..N loop

60. for j in 1..N loop

61. if A(i,j) < Var then

62. Var := A(i,j);

63. end if;

64. end loop;

65. end loop;

66. return Var;

67. end SearchMin;

68. --Функція виконує пошук мінімального елемента в векторі

69. function SearchVectMin(A: Vect) return Integer is

70. min : Integer;

71. begin

72. min := A(1);

73. for i in 1..N loop

74. if A(i) < min then

75. min := A(i);

76. end if;

77. end loop;

78. return min;

79. end SearchVectMin;

80. --Функція виконує додавання двох матриць

81. function AddMatr(A, B : Matrix) return Matrix is

82. Result : Matrix;

83. begin

84. for i in 1..N loop

85. for j in 1..N loop

86. Result(i,j) := A(i,j) + B(i,j);

87. end loop;

88. end loop;

89. return Result;

90. end AddMatr;

91. --Функція виконує множення вектора на матрицю

92. function MultVectMatr(A : Vect; B : Matrix) return Vect is

93. Cell : Integer;

94. Result : Vect;

95. begin

96. for j in 1..N loop

97. Cell := 0;

98. for k in 1..N loop

99. Cell := Cell + A(k)\*B(k,j);

100. end loop;

101. Result(j):= Cell;

102. end loop;

103. return Result;

104. end MultVectMatr;

105. --Функція виконує сортування вектора за зростанням

106. function SortVect(A : Vect) return Vect is

107. m : Integer;

108. Result: Vect;

109. begin

110. for i in 1..N loop

111. Result(i):= A(i);

112. end loop;

113. for i in reverse 1..N loop

114. for j in 1..(i-1) loop

115. if Result(j) > Result(j+1) then

116. m := Result(j);

117. Result(j):=Result(j+1);

118. Result(j+1):=m;

119. end if;

120. end loop;

121. end loop;

122. return Result;

123. end SortVect;

124. --Функція виконує сортування матриці по спаданню

125. function sortMatr(MA: Matrix) return Matrix is

126. Result : Matrix;

127. m: Integer;

128. begin

129. for i in 1..N loop

130. for j in 1..N loop

131. Result(i,j):=MA(i,j);

132. end loop;

133. end loop;

134.

135. for k in 1..N loop

136. for i in reverse 1..N loop

137. for j in 1..(i-1) loop

138. if Result(k,j) < Result(k,j+1) then

139. m := Result(k,j);

140. Result(k,j):=Result(k,j+1);

141. Result(k,j+1):=m;

142. end if;

143. end loop;

144. end loop;

145. end loop;

146. return Result;

147. end sortMatr;

148. --Функція виконує множення числа на матрицю

149. function MultNumb(a: Integer; MA: Matrix) return Matrix is

150. Result : Matrix;

151. begin

152. for i in 1..N loop

153. for j in 1..N loop

154. Result(i,j):= a\*MA(i,j);

155. end loop;

156. end loop;

157. return Result;

158. end MultNumb;

159. --F1: E:=A+B+C+D(MA\*MZ)

160. procedure F1(A,B,C,D: in Vect; MA, MZ: in Matrix; E : out Vect) is

161. begin

162. E := AddVect(AddVect(A,B),AddVect(C,MultVectMatr(D,MultMatr(MA,MZ))));

163. end F1;

164. --F2: MZ:=Min(MA)\*MB+Max(MT)\*(MX\*MZ)

165. procedure F2(MA,MB,MT,MX,MZ: in Matrix; ME: out Matrix) is

166. begin

167. ME := AddMatr(MultNumb(SearchMin(MA),MB),MultNumb(SearchMax(MT),MultMatr(MX,MZ)));

168. end F2;

169. --F3: d:=Min(A\*Trans(MB\*MM)+B\*Sort(MC))

170. procedure F3(A,B: in Vect; MB, MM,MC: in Matrix; d: out Integer) is

171. begin

172. d:= SearchVectMin(AddVect(MultVectMatr(A,Transp(MultMatr(MB,MM))),MultVectMatr(B,sortMatr(MC))));

173. end F3;

174. --Процедура зчитування вектора з консолі

175. procedure readVect(A : out Vect) is

176. begin

177. for i in 1..N loop

178. Get(A(i));

179. end loop;

180. end readVect;

181. --Процедура відображення вектора в консолі

182. procedure showVect(A : in Vect) is

183. begin

184. for i in 1..N loop

185. Put(A(i));

186. end loop;

187. end showVect;

188. --Процедура зчитування матриці з консолі

189. procedure readMatr(MA: out Matrix) is

190. begin

191. for i in 1..N loop

192. Put("New row");

193. New\_Line;

194. for j in 1..N loop

195. Get(MA(i,j));

196. end loop;

197. end loop;

198. end readMatr;

199. --Процедура відображення матриці в консоль

200. procedure showMatr(MA: in Matrix) is

201. begin

202. for i in 1..N loop

203. New\_Line;

204. for j in 1..N loop

205. Put(MA(i,j));

206. end loop;

207. end loop;

208. end showMatr;

209.

210.

211.

212. end Data;

Compiling: data.ads (source file time stamp: 2014-09-17 04:46:16)

1. -----Специфікація пакету Data----------

2. generic

3. N : in Integer;

4. package Data is

5. --Оголошення приватних типів

6. type Vect is private;

7. type Matrix is private;

8. --F1: E:=A+B+C+D(MA\*MZ)

9. procedure F1(A,B,C,D: in Vect; MA, MZ: in Matrix; E : out Vect);

10. --F2: MZ:=Min(MA)\*MB+Max(MT)\*(MX\*MZ)

11. procedure F2(MA,MB,MT,MX,MZ: in Matrix; ME: out Matrix);

12. --F3: d:=Min(A\*Trans(MB\*MM)+B\*Sort(MC))

13. procedure F3(A,B: in Vect; MB, MM,MC: in Matrix; d: out Integer);

14. --Процедура зчитування вектора з консолі

15. procedure readVect(A : out Vect);

16. --Процедура відображення вектора в консолі

17. procedure showVect(A : in Vect);

18. --Процедура зчитування матриці з консолі

19. procedure readMatr(MA: out Matrix);

20. --Процедура відображення матриці в консоль

21. procedure showMatr(MA: in Matrix);

22. --Визначення приватних типів

23. private

24. type Vect is array(1..N) of Integer;

25. type Matrix is array(1..N, 1..N) of Integer;

26. end Data;

212 lines: No errors