МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ ИМЕНИ

ПАТРИСА ЛУМУМБЫ»

(РУДН)

Факультет Искусственного Интеллекта

факультет

**ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №**

Практикум по программированию

наименование дисциплины

Вариант №8

тема (вариант)

Преподаватель \_\_\_\_\_\_\_\_\_\_ В.С. Тынченко

подпись, дата инициалы, фамилия

Студент ЗФИмд01-24 1132249437 \_\_\_\_\_\_\_\_\_\_ А.В. Клюкин

номер группы номер студ. билета подпись, дата инициалы, фамилия

Москва 2025

**Цель работы**

Приобретение навыков применения указателей в программе на языке С

Изучение языковых средств и способов динамического выделения

памяти под одномерные и двумерные массивы

Изучение особенностей обработки динамических массивов

**Задачи работы**

Дан вещественный массив А[N][М]. Определить индекс столбца с максимальным количеством нулевых элементов. Переставить элементы данного

столбца в обратном порядке.

**Основные этапы**

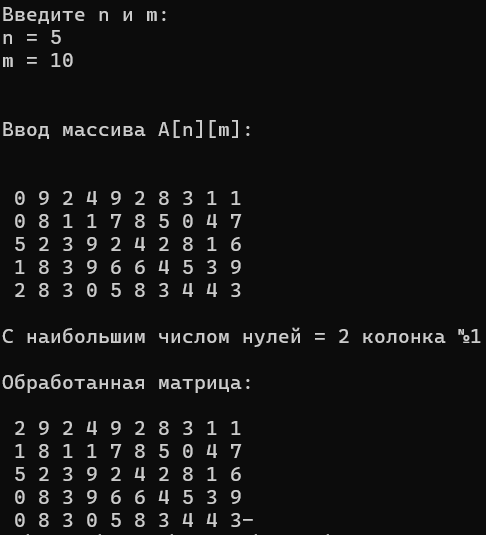


Рисунок 1 Скриншот Результата работы программы с использованием индексов

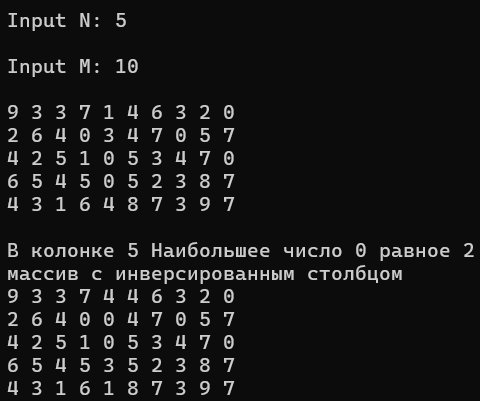


Рисунок 2 Скриншот Результата работы программы без использования индексов

**Листинг**

Реализация программы с индексами

1. #include <iostream>

2. #include <stdlib.h>

3. #include <time.h>

4. using namespace std;

5.

6. int main()

7. {

8. setlocale(LC\_ALL, "RU");

9. int n, m, i, j, \* A, \* p, iter, counter = 0, column=0, c;

10. cout << "Введите n и m:" << endl << "n = ";

11. cin >> n;

12. cout << "m = ";

13. cin >> m;

14. cout << endl;

15.

16. A = new int[n \* m];

17. srand((unsigned)time(0));

18.

19. for (p = A; p - A < n \* m; p++)

20. {

21. \*p = rand() % 10; // 10 было взято для того, чтобы вероятность получить 0 в столбце была выше

22. }

23.

24. cout << endl << "Вывод массива A[n][m]:" << endl << endl;

25. for (i = 0, p = A; i < n \* m; i++, p++)

26. {

27. if (i % m == 0)

28. {

29. cout << endl;

30. }

31. cout.width(2);

32. cout << \*p;

33. }

34. cout << endl;

35.

36.

37.

38. for (i = 0; i < m; i++)

39. {

40. c = 0;

41. for(j = 0; j < n; j++, p++)

42. {

43. if (A[j \* m + i] == 0)

44. {

45. c++;

46. }

47.

48. }

49. if (counter < c)

50. {

51. counter = c;

52. column = i;

53. }

54. }

55.

56. cout << endl << "С наибольшим числом нулей = "<< counter << " колонка №" << column + 1 << endl;

57.

58. for (i = 0; i < n / 2; i++)

59. {

60. //cout << endl << A[i \* m + column] << A[(n - 1 - i) \* m + column];

61. j = A[i \* m + column];

62. A[i \* m + column] = A[(n - 1 - i) \* m + column];

63. A[(n - 1 - i) \* m + column] = j;

64.

65. //cout << endl << A[i \* m + column] << A[(n - 1 - i) \* m + column];

66.

67. }

68.

69.

70.

71. cout << endl << "Обработанная матрица: " << endl;

72. for (i = 0, p = A; i < n \* m; i++, p++)

73. {

74. if (i % m == 0)

75. {

76. cout << endl;

77. }

78. cout.width(2);

79. cout << \*p;

80. }

81.

82.

83. delete[] A;

84.

85. }

86.

1. #include <iostream>

2. #include <stdlib.h>

3. #include <time.h>

4. using namespace std;

5. int main()

6. {

7. setlocale(LC\_ALL, "RU");

8.

9. int n, m, c, max = 0, column = 0;

10. cout << "Input N: ";

11. cin >> n;

12. cout << "\nInput M: ";

13. cin >> m;

14. cout << endl;

15.

16. srand((unsigned)time(0));

17. //создаём массив из n указателей по m

18. int\*\* A = new int\* [n];

19. int\*\* prow = A;

20. while (prow - A < n)

21. {

22. \*prow = new int[m];

23. int\* pi = \*prow;

24. int\* pend = pi + m;

25. while (pi < pend) {

26. \*pi = rand() % 10;

27. pi++;

28. }

29. prow++;

30. }

31.

32. // выводим массив в консоль

33. prow = A;

34. while (prow - A < n)

35. {

36. int\* pi = \*prow;

37. int\* pend = pi + m;

38. while (pi < pend)

39. {

40. cout << \*pi << " ";

41. pi++;

42. }

43. cout << endl;

44. prow++;

45. }

46.

47.

48. prow = A;

49. int\* pi = \*prow;

50. int i = 0;

51. while (i < m)

52. {

53. c = 0;

54. for (prow = A; prow - A < n; prow++)

55. {

56. //int\* pi = \*prow;

57. pi = \*prow;

58.

59. pi += i;

60. if (\*pi == 0)

61. {

62. c++;

63. }

64. }

65. if (c > max)

66. {

67. max = c;

68. column = i;

69. }

70. i++;

71. }

72.

73.

74. cout << endl << "В колонке " << column + 1;

75. cout << " Наибольшее число 0 равное " << max << endl;

76.

77. int a, b, index;

78. for (prow = A; prow - A < n / 2; prow++)

79. {

80. index = prow - A;

81. int\* pi1 = A[index] + column;

82. int\* pi2 = A[n - 1 - index] + column;

83. int temp = \*pi1;

84. \*pi1 = \*pi2;

85. \*pi2 = temp;

86.

87.

88.

89.

90. }

91.

92. cout << "массив с инверсированным столбцом" << endl;

93.

94. prow = A;

95. while (prow - A < n)

96. {

97. int\* pi = \*prow;

98. int\* pend = pi + m;

99. while (pi < pend)

100. {

101. cout << \*pi << " ";

102. pi++;

103. }

104. cout << endl;

105. prow++;

106. }

107.

108. }

109.