МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ИНСТИТУТ НЕПРЕРЫВНОГО И ДИСТАНЦИОННОГО ОБРАЗОВАНИЯ

|  |
| --- |
| КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ |

ОЦЕНКА

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| старший преподаватель |  |  |  | Е.О. Шумова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1 |
| Классы, определение методов класса, права доступа |
| по дисциплине: Объектно-ориентированное программирование |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ гр. № | Z1431 |  |  |  | М.Д.Быстров |
|  | номер группы |  | подпись, дата |  | инициалы, фамилия |
| Студенческий билет № | 2021/3572 | |  |  |  |

Санкт-Петербург 2023

**Условие**

Цель работы: изучить принципы создания классов, ограничения прав доступа к полям и методам класса, применив на практике знания базовых синтаксических конструкций языка C++ и объектно-ориентированного программирования.

Закрепить знания по теме: классы, определение методов класса, права доступа.

Описание работы: в работе необходимо реализовать класс в соответствии с вариантом задания и создать объекты (экземпляры класса):

- Поля класса должны иметь спецификатор доступа private;

- Доступ к полям должен осуществляться через public методы;

- В классе должен быть метод вывода (для вывода исходных значений объекта);

- Для присвоения значений полям класса использовать метод(ы)-«сеттер(ы)» (Конструкторы в этой работе не используем);

- В функции main() создается объект (или несколько объектов) на основе созданного класса и осуществляется вызов методов класса.

Индивидуальное задание (вариант 2):

2. Целочисленные поля x и y представляют собой координаты клетки шахматной доски. Учитывая, что левое нижнее поле доски (1, 1) является черным, реализовать метод, проверяющий истинность высказывания: «Данное поле является белым».

**Полный текст (листинг) программы**

1. Файл «main.cpp»
2. #include <iostream>
3. #include <Windows.h>
4. #include "ChessboardSquare.h"
6. **using namespace std;**
8. int main()
9. {
10. SetConsoleCP(1251);
11. **SetConsoleOutputCP(1251);**
13. cout << "ЛР №1 ВАРИАНТ 2" << endl << endl;
15. ChessboardSquare sq1;
16. **ChessboardSquare sq2;**
18. sq1.setX(1);
19. sq1.setY(1);
21. **sq2.setX(1);**
22. sq2.setY(2);
24. if (sq1.isWhite())
25. {
26. **cout << "Белая " ;**
27. }
28. else
29. {
30. cout << "Черная ";
31. **}**
33. sq1.printDescription();
35. if (sq2.isWhite())
36. **{**
37. cout << "Белая " ;
38. }
39. else
40. {
41. **cout << "Черная ";**
42. }
44. sq2.printDescription();
46. **system("pause");**
48. return 0;
49. }
50. Файл «ChessboardSquare.h»
51. class ChessboardSquare
52. {
53. private:
54. int x;
55. **int y;**
57. public:
58. void setX(int x);
59. void setY(int y);
60. **bool isWhite();**
61. void printDescription();
62. };
63. Файл «ChessboardSquare.cpp»
64. #include <iostream>
65. #include "ChessboardSquare.h"
67. using namespace std;
69. void ChessboardSquare::setX(int x)
70. {
71. this->x = x;
72. }
74. void ChessboardSquare::setY(int y)
75. {
76. this->y = y;
77. }
79. bool ChessboardSquare::isWhite()
80. {
81. return (this->x + this->y) % 2 != 0;
82. }
84. void ChessboardSquare::printDescription()
85. {
86. cout << "Клетка с координатами (" << this->x << ", " << this->y << "). ";
87. cout << "Белая: " << (this->isWhite() ? "да" : "нет") << endl;
88. **}**