Министерство образования и науки Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Алтайский государственный технический университет им. И.И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчёт защищён с оценкой\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Преподаватель Егорова Е.В.

«\_\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г.

Отчёт

по лабораторной работе № 8

«Введение в объектно-ориентированное программирование»

по дисциплине «Программирование - 2»

Студент группы ПИ 92 В.М. Шульпов

Преподаватель доцент, к.т. н. Егорова Е.В.

Барнаул 2020

Задания

Построить класс, представляющий заданный абстрактный объект.

Обязательно включить в класс функции:

* + **set** (установка значений данных-элементов объекта)
  + **get** (получение значений данных-элементов объекта)
  + **prin**t (печать объекта)
  + конструктор без параметров
  + конструктор с параметрами
  + конструктор копирования (не обязательно)
  + деструктор

Написать программу main, которая демонстрирует включенные в класс функции.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Общежитие | Число комнат, число проживающих мальчиков, число проживающих девочек | * добавление нового жильца в общежитие * определение среднего числа жильцов, проживающих в комнате * выселение жильца из общежития * определение общего числа жильцов общежития |

Код

#include <stdio.h>

#include <conio.h>

#include <windows.h>

#include <locale.h>

class hostel {

int numbersOfRooms; //кол-во комнат

int numbersOfBoys; //число проживающих мальчиков

int numberOfGirls; //число проживающих девочек

public:

hostel(); //КОНСТРУКТОР БЕЗ ПАРАМЕТРОВ

hostel(int \_numbersOfRooms, int \_numbersOfBoys, int \_numberOfGirls); //КОНСТРУКТОР C ПАРАМЕТРАМИ

~hostel(); //ДЕСТРУКТОР

int set(int \_numbersOfRooms, int \_numbersOfBoys, int \_numberOfGirls); //установка значений переменных-членов класса

int get(); //получение значений класса (и вывод)

int addRoomer(); //добавление жильца

int evictRoomer(); //выселить жильца

int getAverageRoomersInRoom(); //получение среднего значения жильцов в комнате (и вывод)

int getTotalNumberOfRoomers(); //получение общего числа жильцов

};

// Определение контруктора без параметров класса Hostel

hostel::hostel()

{

numbersOfRooms = numbersOfBoys = numberOfGirls = 0;

printf("Hostel is created\n"); // общежитие создано

}

// Определение контруктора с параметрами класса Hostel

hostel::hostel(int \_numbersOfRooms, int \_numbersOfBoys, int \_numberOfGirls)

{

// Указатель this ссылается на объект, вызывающий ф-цию Hostel

this->numbersOfRooms = \_numbersOfRooms;

this->numbersOfBoys = \_numbersOfBoys;

this->numberOfGirls = \_numberOfGirls;

printf("Hostel with parameters is created\n"); // общежитие создано

}

// Определение деструктора класса Hostel

hostel::~hostel()

{

printf("Hostel is deleted\n"); // общежитие удалено

}

int hostel::addRoomer() //добавление жильца

{

int choice; //выбор

printf("Введите 1, если хотите добавить мальчика\nвведите 2, если - девочку\nВаш выбор:\t");

scanf\_s("%d", &choice);

if (choice == 1)

{

this->numbersOfBoys++;

return 0;

}

else if (choice == 2)

{

this->numberOfGirls++;

return 0;

}

else

{

printf("Вы не можете сделать такой выбор. Добавление жильца не состоялось.\n");

return 1;

}

}

int hostel::evictRoomer() //выселение жильца

{

int choice; //выбор

printf("Введите 1, если хотите выселить мальчика\nвведите 2, если - девочку\nВаш выбор:\t");

scanf\_s("%d", &choice);

if (choice == 1)

{

if (this->numbersOfBoys > 0)

{

this->numbersOfBoys--;

return 0;

}

else {

printf("мальчиков в общежитие нет, нельзя выселить мальчика\n");

return 1;

}

}

else if (choice == 2)

{

if (this->numberOfGirls > 0)

{

this->numberOfGirls--;

return 0;

}

else

{

printf("девочек в общежитие нет, нельзя выселить девочку\n");

return 1;

}

}

else

{

printf("Вы не можете сделать такой выбор. Добавление жильца не состоялось.\n");

return 2;

}

}

int hostel::set(int \_numbersOfRooms, int \_numbersOfBoys, int \_numberOfGirls)

{

if (\_numbersOfRooms < 0 || \_numbersOfBoys < 0 || \_numberOfGirls < 0)

{

printf("есть отрицательный параметр, установка прервана\n");

return 1;

}

else

{

this->numbersOfRooms = \_numbersOfRooms;

this->numbersOfBoys = \_numbersOfBoys;

this->numberOfGirls = \_numberOfGirls;

printf("Hostel parameters are changed\n"); // параметры общежития изменены

}

return 0;

};

int hostel::get()

{

printf("\tколичество комнат:\t%d\n", this->numbersOfRooms);

printf("\tколичество мальчиков:\t%d\n", this->numbersOfBoys);

printf("\tколичество девочек:\t%d\n", this->numberOfGirls);

return 0;

};

int hostel::getAverageRoomersInRoom()

{

if (numbersOfRooms != 0)

{

float average = float(this->numbersOfBoys + this->numberOfGirls) / this->numbersOfRooms;

printf("среднее значение жильцов в комнате:\t%.2f", average);

return 0;

}

else

{

printf("В общежитии нет комнат\n");

return 1;

}

}

int hostel::getTotalNumberOfRoomers()

{

int total = this->numbersOfBoys + this->numberOfGirls;

printf("общее число жильцов:\t%d\n", total);

return 0;

}

int main() { /\* Основная программа \*/

setlocale(LC\_ALL, "Rus");

system("color F0");

printf("Class hostel\n\n");

puts("\n\_\_OBJECT HOSTEL CREATED WITHOUT PARAMETERS\n");

hostel hostel1; // Создаем обЪект hostel1

puts("\n\_\_GET()\n");

hostel1.get(); //получаем выводим все параметры

puts("\n\_\_SET PARAMETERS (100, 110, 120)\n");

hostel1.set(100, 110, 120); //устанавливаем все параметры

puts("\n\_\_GET()\n");

hostel1.get(); //получаем выводим все параметры

puts("\n\_\_ADD\_ROOMER()\n");

hostel1.addRoomer(); //добавляем жильца

puts("\n\_\_GET()\n");

hostel1.get(); //получаем выводим все параметры

puts("\n\_\_EVICT\_ROOMER()\n");

hostel1.evictRoomer(); //выселяем жильца

puts("\n\_\_GET()\n");

hostel1.get(); //получаем выводим все параметры

puts("\n\_\_GET\_AVERAGE\_ROOMERS\_IN\_ROOM()\n");

hostel1.getAverageRoomersInRoom();

puts("\n\_\_GET\_TOTAL\_BUMBER\_OF\_ROOMERS()\n");

hostel1.getTotalNumberOfRoomers();

}

Тест

 