Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Московский государственный технический университет имени Н. Э. Баумана (национальный исследовательский университет)

Факультет «Робототехника и комплексная автоматизация»

Кафедра «Системы автоматизированного проектирования»

**Отчет по практической работе**

По курсу «Проектно-технологическая практика»

Выполнил: Студент Сергеева Д.К.

Группа РК6-36Б

Проверил: \_\_\_Козов\_А.В.\_\_\_\_\_\_\_\_

Дата \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва, 2020 г.

**Вариант P13 Последовательность Хэмминга**

**Задание:**

Требуется составить ООП для получения любого начального фрагмента последовательности Хэмминга из целых чисел, которые не имеют других простых делителей кроме 2,3 и 5. При этом последовательность должна рождаться в порядке роста значений её элементов без тотальной проверки делимости всех чисел или вычисления степеней делителей. Количество элементов должен задавать аргумент командной строки вызова программы. Результат вычислений должен отображаться через поток стандартного вывода, где для каждого элемента указан элемент, из которого он получен.

**Алгоритм:**

Создадим класс, в котором будут хранится: последовательность Хэмминга из целых чисел, которые не имеют других простых делителей кроме 2,3 и 5, в виде массива типа int и число, равное количеству элементов в последовательности. В классе буду хранится методы: вывода последовательности в поток стандартного вывода, метод, который формирует последовательность Хэмминга и конструктор этого класса. Для формирования последовательности в соответствующем методе создаётся приоритетная очередь, в которой изначально лежат только простые числа: 2, 3 и 5. Далее записываем в массив нашей результирующей последовательности самое маленькое число n, так как нам необходимо, чтобы последовательность была в порядке роста значений её элементов, предварительно удаляя его из очереди, и отправляем в очередь числа: n\*2, n\*3, n\*5. И так повторяем пока у нас в результирующей последовательности не будет необходимое количество элементов.

**Входные данные:**

Целое число – количество элементов в последовательности Хэмминга.

**Выходные данные:**

Начальный фрагмент последовательности Хэмминга из целых чисел, которые не имеют других простых делителей кроме 2,3 и 5.

**Текст программы:**

#include <iostream>

#include <queue>

#include <vector>

using namespace std;

class Hamming

{

private:

vector<int> sequence;

int amount;

public:

Hamming(int n = 0) : amount(n) {};

void Ham();

void print\_seq(void);

};

void Hamming::print\_seq(void)

{

for (int i = 0; i < sequence.size(); ++i)

cout << sequence[i] << ' ';

}

void Hamming::Ham()

{

priority\_queue<int, vector<int>, greater<int>> priorQueue;

priorQueue.push(2);

priorQueue.push(3);

priorQueue.push(5);

int n = amount;

while (n != 0) {

int elem = priorQueue.top();

while (priorQueue.top() == elem)

priorQueue.pop();

sequence.push\_back(elem);

priorQueue.push(elem \* 2);

priorQueue.push(elem \* 3);

priorQueue.push(elem \* 5);

--n;

}

}

int main()

{

cout << "Amount of elements = ";

int n;

cin >> n;

Hamming hamming(n);

hamming.Ham();

hamming.print\_seq();

return 0;

}

**Тесты:**

|  |  |
| --- | --- |
| **Вход** | **Выход** |
| 3 | 2 3 4 |
| 10 | 2 3 4 5 6 8 9 10 12 15 |
| 0 |  |

**Список использованной литературы:**

* Волосатова Т.М., Родионов С.В. Лекции по курсу «Объектно-ориентированное программирование»
* bigor.bmstu.ru