

## Описание задания:

Необходимо реализовать программу с использованием объектно-ориентированного подхода и статической типизацией. В рамках конкретного задания предполагается использование следующих структур данных:

## Обобщённый артефакт:

Фильм. Каждая из описанных ниже альтернатив (жанров) содержит название (строка) и год выхода (целочисленная переменная).

Общей функцией для всех альтернатив является нахождение частного от деления года выхода фильма на количество символов в его названии (числовая переменная действительного типа).

Базовые альтернативы (жанры):

- Документальный фильм. Характеризуется длительностью (целочисленная переменная).
- Игровой фильм. Содержит имя режиссёра (строка).
- Мультфильм. Характеризуется одним из следующих способов создания:

- 1) Рисованный
- 2) Кукольный
- 3) Пластилиновый

Способ создания хранится в переменной перечислимого типа.

Также необходимо обеспечить хранение и обработку данных с помощью спроектированного для этих целей контейнера. Помимо базового функционала необходимо реализовать упорядочивание посредством перемещения в конец контейнера тех элементов, для которых значение, полученное с использованием функции, общей для всех альтернатив, больше, чем среднее арифметическое для всех элементов контейнера, полученное с использованием этой же

функции. Остальные элементы сдвинуть к началу без изменения их порядка.

Основные характеристики программы:

- Число интерфейсных модулей:
- Число модулей реализации:
- Общий размер исходных текстов:
- Размер исполняемого кода:
- Время работы программы на различных тестах:

Test 01: 0.002 с

Test 02: 0.004 с

Test 03: 0.005 с

Test 04: 0.398 с

Test 05: 0.002 с

Film (абстрактный класс)			4
Поля	static	RandomInt rnd3 RandomInt rnd2000 RandomInt rnd 180 RandomString rnd_film_name RandomString rnd_director_name	8 8 8 8 8
	private	string name int year	8 4
Методы	virtual	void In(ifstream& ifst) void InRnd() void Out(ofstream& ofst)	—
	static	Film* StaticIn(ifstream& ifst) Film* StaticInRnd()	
	private	double Quotient()	

Cartoon (производный класс)			12
Поля	local	Type type	4[0]
Методы	virtual	void In(ifstream& ifst) void InRnd() void Out(ofstream& ofst)	–
Enum		Types {DRAWN = 1, STOP_MOTION, PLASTICINE}	4[4]
Documentary (производный класс)			8
Поля	private	int length	4
Методы	virtual	void In(ifstream& ifst) void InRnd() void Out(ofstream& ofst)	–
Gaming (производный класс)			12
Поля	private	string director	8[0]
Методы	virtual	void In(ifstream& ifst) void InRnd() void Out(ofstream& ofst)	–

## Описание работы main

Вызов функций	Используемая память:
Init(c)	argc: int 4[0]
In(c) or InRnd(c)	argv: char* 8[4]
Out(c)	c: container 10008[12]
Clear(c)	

### **Сравнительный анализ:**

Выделим преимущества ООП подхода по сравнению с процедурным:

- Читаемость кода
- Прозрачность структур библиотек
- Цельная иерархия классов
- Уменьшенный риск возникновения ошибок в больших проектах

Недостатки ООП подхода:

- Тратится большой объём памяти
- Огромное количество методов, достигаемое за счёт декомпозиции