**Итоговая работа по программированию (1ПМ+1МО)**

**Минимальные критерии**

1. (ПМ и МО) Разработать программу: анимацию процесса, либо игру с использованием библиотеки SFML:
   1. Столкновение шариков
   2. Полет шарика под углом к горизонту
   3. И т.п.
2. (ПМ и МО) Использовать объектно-ориентированный подход.
3. (ПМ и МО) Оформить код с использованием пространства имен и с разбиением на файлы.
4. (ПМ) Подготовить CMake файл для автоматизации развертывания программы.
5. (ПМ) Подготовить релизную версию.
6. (ПМ) Подготовить небольшую (пару строчек) документацию по программе. Желательно оформить в формате Markdown:
   1. Системные требования (какая ОС)
   2. Как запустить для использования
   3. Как запустить для разработки (какой компилятор использовался, среда разработки)
7. (ПМ) При разработке использовать систему контроля версий Git.

Минимальные критерии должны быть соблюдены, но вы не ограничены, если хотите сделать нечто большее. Творчество приветствуется. Только будьте готовы объяснить то, что вы сделаете.

**Как будет происходить оценивание работы?**

Учитывается:

1. Соответствие работы критериям 1-7 для ПМ и 1-3 для МО.
2. Ответы студента на возможные вопросы (см. список).
3. Оценка за лабораторные работы.

Поскольку творчество приветствуется, хорошая работа может послужить поводом добавить балл к оценке.

**Допуск к экзамену (для 1ПМ, Тарачков М.В.)\***

1. Сдано 11 л/р – допуск, максимальная оценка за экзамен 5.
2. Сдано 9 л/р – допуск, максимальная оценка за экзамен 4.
3. Сдано 7 л/р – допуск, максимальная оценка за экзамен 3.
4. Сдано <7 л/р – допуска нет.

**Учебная практика (для 1ПМ, Тарачков М.В.)\***

1. Сдано 5 задач – оценка 5.
2. Сдано 3 задачи – оценка 4.
3. Сдано 1 задача – оценка 3.
4. Сдано 0 задач – зачета нет.

\*Условия получения допуска и оценки уточняйте у своих преподавателей по практике.

\*\*Преподаватели могут поставить «автомат», я это учту, и сдавать итоговую работу не придется.

**Список вопросов по основам программирования**

1. Что такое стандарт языка?
2. Сколько места в оперативной памяти занимает тип bool и почему?
3. Как работает тернарный оператор? Привести пример.
4. Что означает создать переменную «на стеке»? Какие плюсы и минусы такого подхода?
5. Какие преимущества дает хип и какие проблемы порождает?
6. Сколько места в оперативной памяти занимают указатели unsigned int\* и char\*? Почему?
7. Что хранит в себе тип char\*\*?
8. Сколько места в оперативной памяти занимает ссылка?
9. В чем разница передачи аргумента по ссылке и по значению?
10. Как вы понимаете «абстракцию»?
11. Как вы понимаете «инкапсуляцию»?
12. В чем основное отличие класса от структуры?
13. Почему многие библиотеки на С++ используют пространство имен? Привести пример потенциальной проблемы, которая может быть без использования пространства имен.
14. Объясните, что означает \*& в следующем фрагменте кода

void update(list\*& root)

{

…

}

1. Что такое компиляция и линковка?
2. В чем отличие статической и динамической библиотек?
3. Чему равен sizeof от std::vector? Почему? (для ПМ)
4. Чему равен sizeof от std::string? Почему? (для ПМ)