

Домашно 1 (почти OBJ)

Тримерен модел ще наричаме съвкупност от точки в тримерното пространство и съвкупност от многоъгълници с върхове измежду дадените върхове.

Често използван формат за съхраняване на такива обекти във файл е Wavefront (.obj).

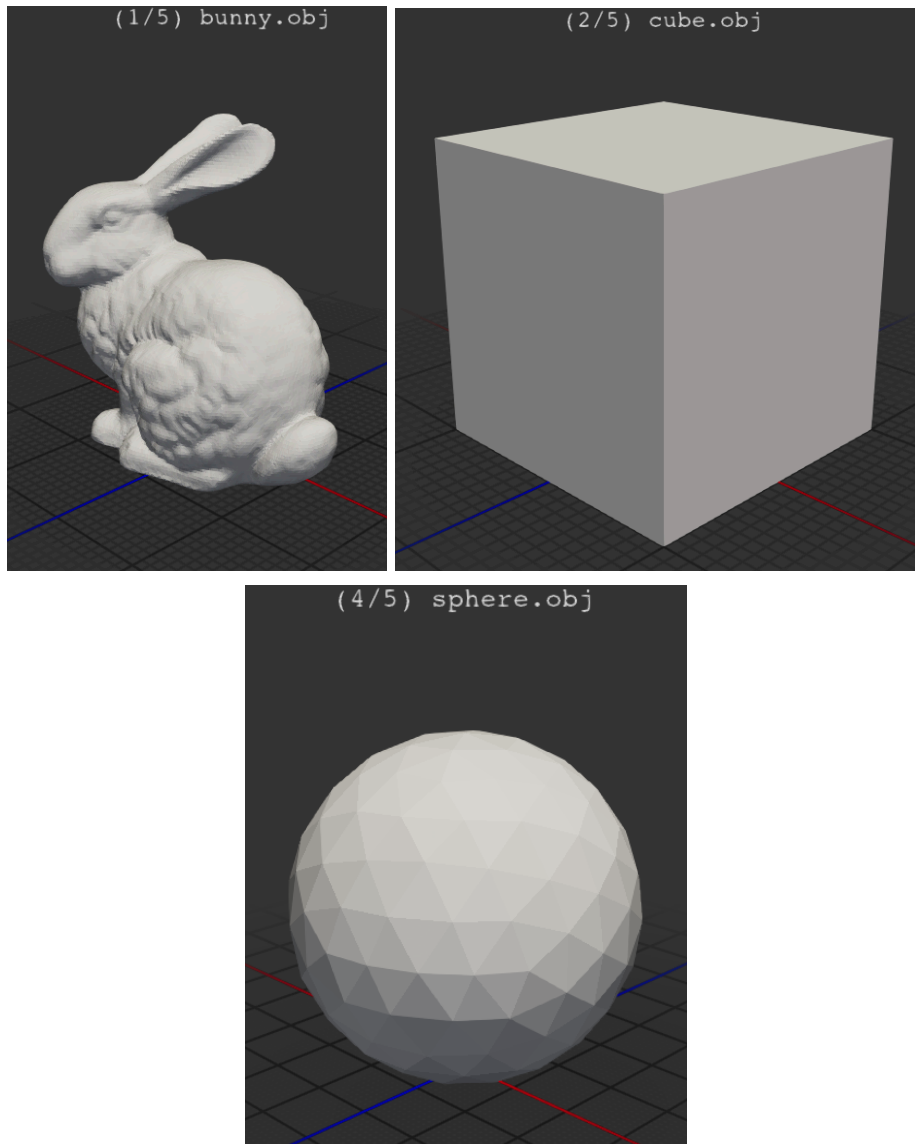
Файл от този тип е текстов и се състои три типа редове:

- 'v <x> <y> <z>' - описва връх в модела. <x>, <y> и <z> са рационални числа, записани в текстов вид (10.001, 123.045, 2, -5.000).
- 'f <v1> <v2> ... <vn>' - описва многоъгълник от повърхността на модела. <v1>, <v2>, ..., <vn> са индексите на върховете, които участват в многоъгълника (цели положителни числа).
- '# ...' - коментар; Всеки ред, започващ с '#' се игнорира при четене от файл.

Такива файлове може да визуализирате чрез приложението 3D Viewer в Windows 10.

Друга добра опция е [f3d](#).

Пример:



Гореположения модел на куб, записан във файл, изглежда така:

```
v -1 -1 -1
v 1 -1 -1
v 1 -1 1
v -1 -1 1
v -1 1 -1
v 1 1 -1
v 1 1 1
v -1 1 1
f 1 2 3 4
f 8 7 6 5
f 1 5 6 2
```

```
f 2 6 7 3
f 3 7 8 4
f 4 8 5 1
```

Задание:

Реализирайте клас `Object3d`, който представя тримерен модел в езика C++.

- Реализирайте конструктор, който приема име на файл и зарежда модел от него.
- Реализирайте конструктор, който приема поток за вход и зарежда модел от него.
- Реализирайте методи `int getVertexCount()` и `int getFaceCount()`, които връщат съответно броя върхове и броя многоъгълници в модел.
- Реализирайте метод `void save(const string& file)`, който приема име на файл и съхранява модела във файла.
- Реализирайте метод `void print(ostream &)`, който приема изходен поток, в който да се запишат данните на обекта в указания формат.
- Реализирайте метод `Object3d cut(function<bool>(float x, float y, float z) f)`, който връща модел съставен само от тези многоъгълници в модела, координатите на чиито върхове отговарят на предиката 'f'.
- Реализирайте метод `void flip()`, който обръща всички многоъгълници в модела наобратно. (Четириъгълникът 1 2 3 4 става 4 3 2 1)
- Реализирайте статични методи за създаване на куб и сфера с параметри съответно дължина на ръб и радиус. (Не е задължително сферата да изглежда точно като на картинката по-горе)

При опит за прочитане на невалиден файл изведете подходящи съобщения.

Ресурси:

Файловете от по-горе:

- [cube.obj](#)
- [sphere.obj](#)
- [bunny.obj](#)

Пример:

```
Object3d b("./bunny.obj");
Object3d b_cut = b.cut([](float x, float y, float z) -> bool {
    return y - x < 2.5;
});
b_cut.save("./bunny_cut.obj");
```

Резултат след изпълнение на този код е:



Краен срок: 03.05.2024

Инструкции за предаване

Предайте всички '.cpp', '.hpp', '.h' файлове в заданието в мудъл. Ако използвате Make или CMake, качете и съответните файлове, описващи компилацията. Не предавайте '.obj', '.zip' и други бинарни файлове.

Разрешено е използването само и единствено на библиотеките `vector`, `string`, `cstring`, `iostream`, `fstream`, `functional`, `cstdint`, `cmath`, `limits` от стандарта на езика.