

## 2017年度メディア情報処理(CG)課題

課題	自分で定義した立体図形の透視投影像を描き、これを印刷する。
目的	齊次座標による座標変換、ビューポート変換、透視投影を理解するとともに、C言語によるプログラミングにも習熟する。
提出期限	2017年5月30日(火)
提出場所	学びステーション

### 課題の説明

- 立体図形は頂点および稜線を定義し、線図形として描く(ワイヤーフレームモデル)。

—頂点の定義

1  $(x_1, Y_1, Z_1)$

2  $(x_2, Y_2, Z_2)$

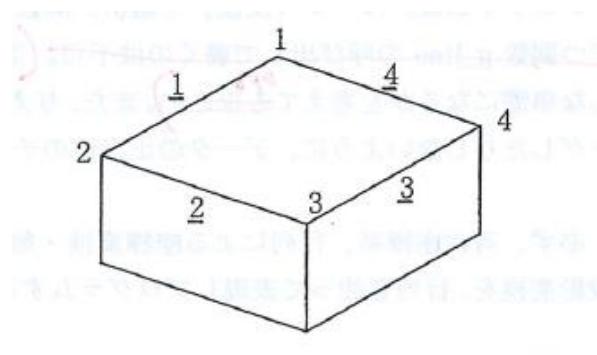
:

—稜線の定義

1 (1,2)

2 (2,3)

:



- 立体図形は世界座標系の原点の近傍で定義する。

- 視点を適当な世界座標  $(x_e, Y_e, Z_e)$  に置き、ここから世界座標の原点を見る。

- 投影面の始点からの距離  $D$  および投影面のサイズ  $S$  は適当に決める。

- 世界座標系による立体のデータを視点座標系に変換し、これを透視投影し投影図を求める。

- 隠線処理、ビューポリューム、3次元クリッピングの処理は行わなくてよい。ただし、データが画像からはみださないように注意する。ビューポートのサイズと位置は任意に決める。

- プログラムはBKC内のPC(Linux)でLinuxシステムを利用してC言語で作成し、実行する。

- レポートとして提出するもの:

1. 作成したプログラムのソースを印刷したもの。
2. 実行結果のImage Magicの画像のプリントアウト。実行時のパラメータ(視点位置など)を明記する。実行結果は、パラメータ(視点位置など)を変えて複数実行したもの添付すること。
3. レポートには表紙をつけ、氏名、学籍番号を明記すること。また、
4. 本レポート課題に関する簡単な感想をつけること。

以上

## 2017年度メディア情報処理(CG)レポート課題についての補足

担当教員 田中弘美

1. 対象 3 次元図形の定義データ (頂点表、稜線表) をプログラム中に埋め込みます、別のファイルからデータとして読み込むようにすること。対象の図形を取り替えるたびに、毎回プログラムをコンパイルしなくてはいけないようなプログラムは最低のものである。
2. プログラムは、ループ(反復)や選択、関数などを使って、きれいに書くこと。(頂点や稜線の数がたとえば数百、数千あるような場合、どんな事態になるかを考えてみる)また、データの正当性のチェックも必ず入れること。
3. 必ず、齊次座標系、行列による座標変換・幾何変換の考え方を使うこと。(ビューイング変換と透視投影変換を、行列を使って表現しプログラムすること。そうしないと一般的なプログラムにならない)
4. 座標変換・幾何変換の処理は「関数」としてプログラムする。変換の合成は、それぞれの変換行列の積」としてプログラムする。
5. プログラムは理解しやすいように書く。  
(コメントを入れる、変数の定義をわかりやすくするなど)
6. 表示結果の正しさだけでなく、以上のような「プログラミングスタイル」についても評価します。友達と相談をして作成するのはもちろん構いませんが、全く同じプログラム、あるいは同じ結果になっているものは、プログラムを単にコピーして実行してものと見分けがつかないので、このようなレポートは両方とも不合格にします。