

第 11 回 宿題

「第 11 回」の宿題の雛形プログラムをコンパイルして実行すると、右図のような図形が表示されます (ESC, q, Q 以外のキーをタイプすると変形します)。

このソースプログラムの `cgsample11.c` の関数 `gouraudshading()` は Gouraud Shading によるスムーズシェーディングのために頂点ごとに陰影付けを行うものですが、関数の内部が実装されていません。

また、`hline3D()` では三角形の内部のパラメータ座標 (u, v) は求めているものの、これを用いた頂点の色の補間を行っていません。

そこで、ソースプログラム中の (1) ~ (7) の部分を実装して、Gouraud Shading によるスムーズシェーディングが行われるようにしてください。

- (1) 関数 `gouraudshading()` の引数に与えられた nv 個の頂点データ v と nf 個の三角形の面データ f をもとにすべての三角形の法線単位ベクトルを求め、それらを引数に与えられた配列 vf に格納しなさい。法線ベクトルの求め方は `flatshading()` と同じです。
- (2) 引数に与えられた nv 個の頂点の法線ベクトル vn をすべて 0 にしなさい。
- (3) すべての三角形について、その法線ベクトル vf をその三つの頂点の法線ベクトル vn のそれぞれに加算しなさい。
- (4) すべての頂点の法線ベクトルを正規化しなさい。
- (5) すべての頂点の陰影を求めて引数 `color` に設定しなさい。これは (4) と同時に行うことができます。
- (6) 関数 `hline3D()` において画素の色 `col` を三角形の三つの頂点の色 c をパラメータ u, v で線形補間して求めなさい。
- (7) 図形を作る関数 `wave()` を書き換えて、図形を上図と異なるものにしてください。

プログラムが期待通り動作したら、作成したソースファイル (`cgsample11.c`) を授業のホームページのアップローダからアップロードしてください。期限は 1 月 12 日 (木) 中です。

