

第 12 回 宿題

「第 1 2 回」の宿題の雛形プログラムは、「第 1 1 回」の宿題プログラムにテクスチャマッピングの機能を追加しようとするものです。そのために、関数 `hline3D()` に配列の引数 `t[3][2]` を追加して、これに描画中の三角形の三つの頂点に対応するテクスチャ座標を与えています。

```
static void hline3D(int x0, int z0, int x1, int z1, int y, double (*a)[2],  
double (*c)[3], double (*t)[2]);
```

この関数に対して、以下の手順でテクスチャマッピングを実装してください。

- (1) 処理対象の画素 (x, y) の三角形内での座標値 (u, v) は「第 1 1 回」の宿題プログラムのスムーズシェーディングのために既に求めていますから、これを使って t を補間して、 (x, y) に対応するテクスチャ座標 (s, t) を求めてください。
- (2) テクスチャ座標 (s, t) の位置にあるテクスチャの画素値を取り出してください。テクスチャの画素値を取り出すために、関数 `texture()` を用意しています。

```
extern void texture(double tcol[3], double s, double t)
```

- (3) この画素における反射光強度 `col` の各要素に `tcol` の対応する要素を乗じて、反射光強度にテクスチャの色が付くようにしてください。なお、これだと鏡面反射光にもテクスチャの色がついてしまいます。鏡面反射光がテクスチャの影響を受けないようにするには、関数 `shade()` を修正する必要があります（余力があればやってみてください）。
- (4) テクスチャに使う画像 `image.tga` を、自分で作ったものに変更してください。この画像は 256×256 画素、RGB 各 8 ビットの TGA (Truevision Advanced Raster Graphics Adapter, TARGA) 形式である必要があります。
 - この形式の画像を作成するには、Photoshop や GIMP など 256×256 画素の RGB 形式の画像を作成し、ファイルの保存の際に Photoshop なら Targa を指定してください。Targa オプションでは色数は 24bits/Pixel (アルファチャンネルを付けない) を指定し、RLE 圧縮はしてもしなくても構いません。
 - GIMP の場合には「ファイル」メニューの「エクスポート」でファイル名に `image.tga` を指定してください。RLE 圧縮はしてもしなくても構いませんが、ビットマップの起点は「左上」にしてください。
 - あるいは、<http://www.wakayama-u.ac.jp/~tokoi/lecture/cg/tga/> で JPEG 画像などから変換できます（ここは学内からしか使用できません）。
 - もしかしたら 256×256 画素以外のサイズの画像も使用できるかもしれませんが、確認していません。
- (5) 「第 1 1 回」の宿題プログラムと同様に、図形を作る関数 `wave()` を書き換えて、図形を元のものとは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、ソースファイル `cgsample12.c` と画像データ `image.tga`、および、もし変更していれば `main.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 19 日 (木) 中です。