

冬のレビュー2022キャンペーン

ターム3D

概要 : 3D レンダリングプログラム...ターミナルで！
？

コンテンツ

I	序文	2
II	共通事項	3
III	件名	4
IV	ボーナス	6
V	リソース	7

第1章 序 文

あのかわいいうアスキーのグラフィックをどうやって作っているのか、不思議に思ったことはありませんか？
このプロジェクトでは、ターミナルで3Dオブジェクトをレンダリングするプログラムを作ります！

第二章

共通事項

- プロジェクトはC言語で書かれている必要があります。
- あなたのプロジェクトは**Norm**に沿って書かれていなければなりません。ボーナスファイル/関数がある場合は、それらも規範チェックに含まれ、内部に規範エラーがある場合は0が表示されます。
- 未定義の動作とは別に、関数が予期せず終了する（セグメンテーション・フォールト、バス・エラー、ダブル・フリーなど）ことがあってはなりません。このような現象が発生した場合、プロジェクトは非機能的であるとみなされ、評価の際に0点が付けられます。
- ヒープに割り当てられたメモリ空間は、必要に応じて適切に解放されなければなりません。リークは許されません。
- 必須部分とボーナス部分の評価は別々に行われます。
- ボーナスポイントを獲得するには、必須項目を完了している必要があります。

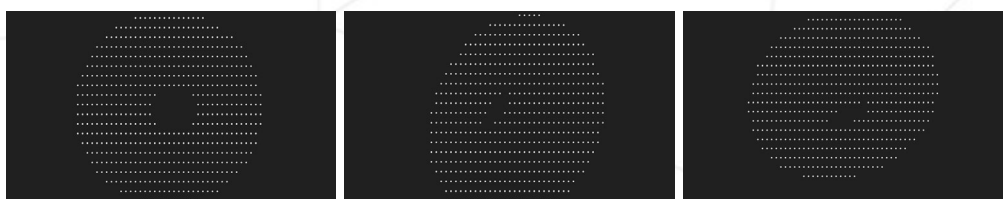
第三章 主題

プログラム名	ターム3D
ファイルの提出	すべての必要なファイル、makefileがあればそれも含めて、atレンダリングするオブジェクトファイルを1つ以上
Makefile	オプション
議論	任意の3Dファイルフォーマット
エクスターナル・ファンクティス	すべての標準Cライブラリ関数
リプト公認	はい。
説明	このプロジェクトの目標は、3Dオブジェクトをレンダリングすることです。 をASCIIテキストで表示します。

制約条件は以下の通りです。

- プログラムは、3Dオブジェクトの表現を含むファイルを引数として受け取る必要があります。
- オブジェクトファイルは、オブジェクトを表す3D座標が含まれていれば、どのような形式でも構いません。(.ply, .obj, .stl, または独自のフォーマット)
- ファイルを引数として与えられたプログラムは、表現された3Dオブジェクトをターミナルに描画する必要があります。
- プログラムは、無効なファイルやファイルが全くない場合にエラーを処理する必要があります。審査のために、少なくとも1つの有効なファイルを提出してください。
- レンダリングに必要な条件
 - 対象は次元3でなければならない。線でもなく、面でもない。
 - オブジェクトは軸に沿って回転するようにアニメーションする必要があります。
 - アニメーションはインプレースで表示され、各フレームが最後のフレームの上にプリントされる必要があります。

- サンプルファイルとしてtorus.3dが添付されています。このファイルは、トーラスをその表面に沿った3Dポイントのセットとして表現したものです。
- ここでは、簡単な透視投影を始めるための基本的な流れをご紹介します。
 - 1 ステップ1：物体の表面を表す3Dポイントのセットを読み込みます（サンプルファイルは添付ファイルにあります）。
 - 1 Step2：各ポイントをループさせ、透視投影によってスクリーン上の位置にマッピングします。
 - Step3：端末の投影位置に文字を印刷する。
 - Step4：軸に沿って点に回転を加える。
 - ステップ5：ループしてステップ-を繰り返す2 4.
- これは、物体をレンダリングするための数多くの方法の一つです。欲を言えば、線画、ラスターライズ、レイトレーシングなども試してみてください。



図III.1：上の写真は、サンプルファイルを適切にレンダリングした場合のイメージです。



図III.2：そして、これはボーナスの1つ（ノーマルシェーディング）である。

第4章 ボーナス

- ボーナスセクションは、楽しみの始まりです。ボーナスポイントには、クリエイティブな自由度があります。
- 記載されているボーナスは、それぞれ以下に記載されている量のポイントを提供します。最大のボーナスはポイント25です。
 - 提供されたサンプル以外の3Dオブジェクトをレンダリングすることができます。(+5)
 - 複数のオブジェクトを一度にレンダリングする。(+5)
 - 複数の異なるファイル形式を扱う。(+5)
 - 🔧 フルカメラコントロール：ユーザーのインタラクションにより、カメラの移動や回転をリアルタイムに行うことができます。(+10)
 - シェーディング：異なる文字を使って面の角度を表現します。(+15)
 - 線画：線画アルゴリズムを使用してオブジェクトのエッジを描画します。(+15)
 - 🔧 ラスタライズ：エッジや頂点だけのファイルをレンダリングし、点の代わりにポリゴンを描画すること。(+25)
 - 🔧 レイトレーシング：エッジや頂点だけのファイルをレンダリングし、各カーソル位置を通る光線を計算します。(+25)
- さらにクリエイティブなボーナスがあれば、それぞれ5ポイントが加算されます。ボーナスの対象となるためには、変革をもたらすものでなければなりません。言い換えれば、ユーザーエクスペリエンスに意味のある影響を与えるものでなければなりません。最終的な判断は審査員に委ねられていますので、積極的にアピールしてください。
- ボーナスの対象にならないものの例
 - ASCII文字の色を変更する（シェーディングと組み合わせた場合を除く）。
 - カスタムファイルフォーマットの使用

第5章 リソース

- 必須部分のリソース
 - [3Dポイントのピクセル座標の計算](#)
 - [遠近法と正射影のマトリックス](#)
 - [ビデオレンスンパースペクティブプロジェクション](#)
 - [コンピュータグラフィックスにおける3次元回転](#)
- ボーナスのためのリソース。
 - [シェーディングの紹介](#)
 - [ラスタライズ：実践編](#)
 - [レンダリング手法「レイトレーシング」の概要](#)
 - [レイトレーシング \(Ray-Tracing\) 。カメラレイの生成](#)