

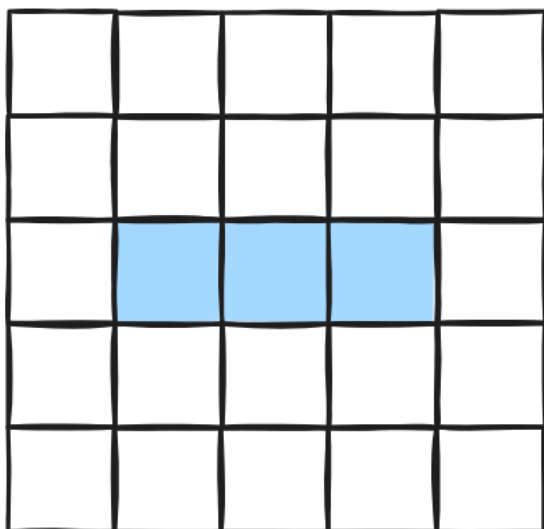
LIFE简介

康威生命游戏(Conway's Game of Life, 简称为 LIFE)是英国数学家康威在1970年代发明的元胞自动机。在 LIFE 中,世界是一个无限大的二维方格矩阵。其中,每个方格中居住着一个存活或者死亡的细胞。世界以离散的时刻进行演化($t=0, 1, 2, 3, \dots$), 每一个细胞在下一个时刻的生死取决于其在当前时刻的8个邻居的状态, 具体规则如下:

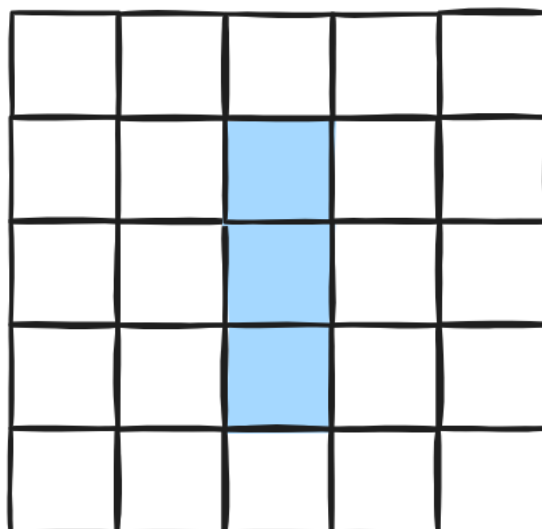
- 当细胞为存活状态时, 若其存活邻居的数量 <2 , 那么它会因孤独而死去;
- 当细胞为存活状态时, 若其存活邻居的数量 >3 , 那么它会因拥挤而死去;
- 当细胞为死亡状态时, 若其存活邻居的数量 $=3$, 那么它会重生;
- 其他任何情况下, 细胞的生存状态保持不变。

给定初始状态, 世界就能在上述规则下自动演化。下图是一个简单而常见的pattern, 其在演化过程中会在两个稳态之间反复切换。

$$t=2n$$



$$t=2n + 1$$



然而并不是所有的pattern都如此易于预测。事实上，由简单的规则与少量的初始细胞演化出的无穷变化正是LIFE的魅力所在。你可以在[这里](#)尝试游玩LIFE，默认的场景是一个不断向外发射滑翔机的"太空船"。

实验描述

在本次实验中，你将使用CUDA编程实现你自己的LIFE程序，但其规则与原版相比做出了以下改动：

1. 世界由2D网格变为3D网格，从而每个细胞有26个邻居。其状态转移规则为：细胞存活时，若存活邻居数 >7 或 <5 ，细胞死亡；细胞死亡时，若存活邻居数 $=6$ ，细胞重生；其他情况下，细胞状态不变。

2. 世界大小由无限变为有限，你需要实现一个长宽高均为 N 的循环宇宙。无限宇宙中的任意一点 (x, y, z) 在循环宇宙中对应的坐标为 $(x \bmod N, y \bmod N, z \bmod N)$ 。也就是说， $(0, 0, 0)$ 与 $(N - 1, N - 1, N - 1)$ 互为邻居。

输入/输出格式

程序接受4个命令行参数。其中，第一个参数为宇宙边长 N ，第二个参数为需要迭代的轮数 T ，第3个参数为输入文件，第四个参数为输出文件。

参数示例

```
./life3d 1024 65536 data.in data.out
```

输入文件与输出文件格式相同，大小均为 N^3 字节，每个字节表示一个细胞的存活状态。其中，坐标为 (x, y, z) 的细胞的状态存储在第 $x \times N^2 + y \times N + z$ 个字节，1表示存活，0表示死亡。

实验环境

实验平台

TODO: 实验平台

git仓库

你可以在 `TODO: git repo` 下找到baseline的git仓库。这个仓库提供：

- `life3d.cu`，包含了一个原始的，未经过任何优化的CPU实现。其中计算部分放在了 `life3d_run` 函数中。你可以基于该实现进行GPU加速；
- `run.sh`，用于随机生成输入文件，并编译运行baseline代码。默认生成的输入规模为 $N = 256, T = 8$ ，你可以在 `run.sh` 中修改这两个参数。

命令参考：

```
cd $YOUR_DIR
git clone $TODO_REPO
cd life3d
./run.sh
```

若运行正常，你将看到如下输出：

```
start population: 3859174
final population: 2634538
time: 20.3569s
cell per sec: 6.59324e+06
```

其中，前两行表示开始/结束状态下，世界中的存活细胞数量；第三行是世界演化计算占用的时间（不包括IO/内存初

始化等)；第四行是平均每秒钟处理的细胞数量 $\frac{TN^3}{time}$ ，用作性能参考。

下面是你要完成的任务顺序：

1. 在 `life3d.cu` 头部添加你的基本信息，git commit；
2. 将 `life3d_run` 函数改写为CUDA核函数，git commit；
3. 若干后续优化/bugfix等内容，git commit。

提交内容

你需要提交的内容包括：

1. 你的git仓库；
2. 你的实验报告。报告中应包括你的并行化思路，采用的优化手段与取得的效果，性能与可扩展性评估等内容。

请将上述内容打包并压缩至zip文件，命名格式为

`SA24011451-张三-CUDA实验.zip`，并在1月5日前发送至邮箱 `sls_ustc@mail.ustc.edu.cn`。

补充说明

1. 程序的**最终运行速度不纳入评分标准**。更重要的是在实验报告中体现出你如何利用学习到的课程内容分析性能瓶颈，做出优化并评估优化效果。

2. 由于最终运行速度不纳入评分标准，使用自己的设备而非我们统一提供的实验平台进行实验也是允许的，但请在报告中说明。
3. 本次实验用于考察CUDA程序设计能力，请勿使用纯CPU算法。
4. **严禁抄袭!** 一经发现，本次实验将作0分处理!