

本日の予定

1. レポート課題1の復習
 - 森林火災モデル
 - 基本実験
2. スクリプト入門(ごく入口)

1. レポート課題の復習

提出物と採点基準(満点 20)

1. 森林火災モデルの説明 (5)
2. 実験の内容と実験方法 (5, 工夫加点 +5 まで)
3. 実験結果の解析 (5, 考察加点 +5 まで)

読み手は何も知らないという想定で書くこと

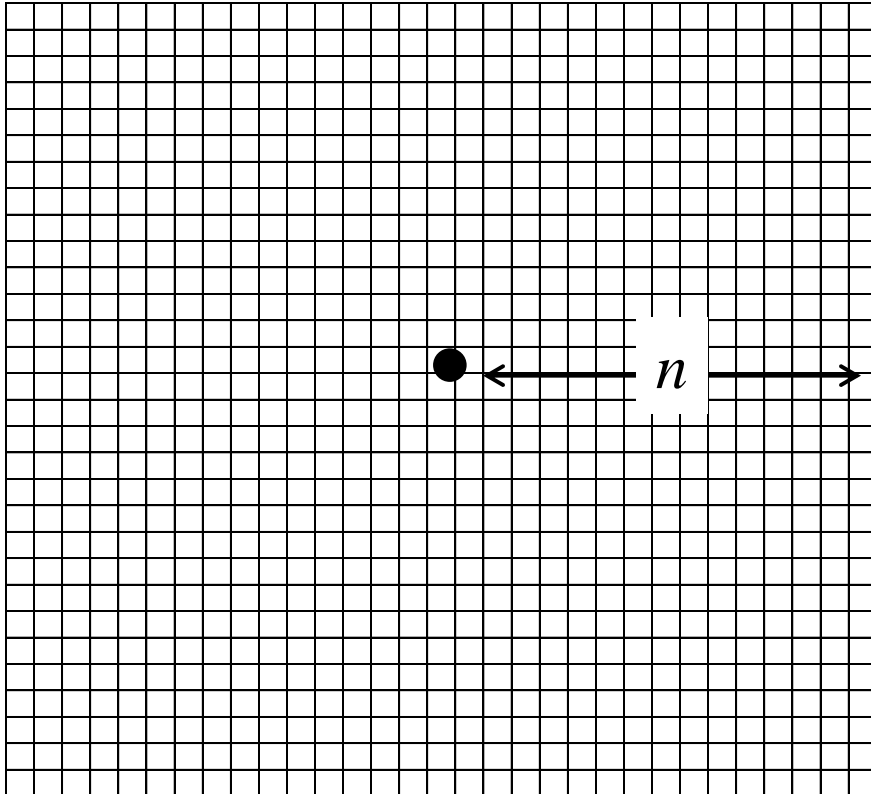
以下はオプション(加点 ≤ 5)

4. 自分なりの実験

復習: 森林火災の超簡略モデル

青字がパラメータ

火災のモデル化



・森の形

- $(2n+1) \times (2n+1)$ の格子状
- 各格子点に木が 1 本

・燃え方

- 森の中心の木 1 本が燃え始める
- 毎時, 隣に確率 p で類焼
- 隣 = 周囲 8 箇所
- 木は発火から b 時間で燃え尽きる

・プログラム用 (プログラムの都合)

- シミュレーション打ち切り時間 t
- 乱数の種 $seed$

復習: 基本実験

臨界全焼率 p_0 と燃焼時間 b との関係

臨界類焼率 p_0 = 森が全焼する可能性が急速に高くなる類焼率
← 50本以下では? (木の総数の5%)

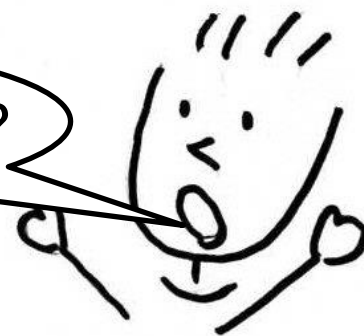
(1) 各種パラメータを適宜定める: $n = 50$ $t = 150$

(2) いろいろな b に対して, 類焼率 p を変えて実験し, p_0 を求める

類焼率 p %	10	11	12	13	14
試行 1	204	133	85	68	36
試行 2	231	153	89	39	27
試行 3	205	131	73	56	29
試行 4	235	128	83	73	32
試行 5	232	118	97	63	38
試行 6	216				
試行 7	261				
試行 8	253	155	92	59	38
試行 9	223	130	82	49	40
試行 10	225	177	79	43	29
全焼率%	0	0	0	30	100

p_0

ええ, これやるのお!??



2. スクリプト入門

スクリプト = 台本

Ruby はスクリプト言語とされています

実験そのものをプログラムにしよう！

実験のスクリプトの概要

```
# 実験の設定: 固定パラメータ  
n = 50, nt = (2*n+1)**2, limit = nt * 0.005 # 全焼の基準(0.5%)
```

```
# 実験用パラメータ  
ex = 30, b = 5, p = 10, time = 150, seed = 1
```

```
# 実験  
bcount = 0  
以下を ex 回繰り返す
```

```
fire をパラメータ n p b time seed で実行し答えを得る  
t ← 時間, nb ← 燃焼木の本数, nu ← 生存木の本数  
if nu <= limit  
  bcount = bcount + 1  
end  
画面に今回の結果を表示  
seed = seed + 1
```

```
R = bcount / ex # 割合を求める  
画面に p と全焼率 R を表示
```

では, 実際に使ってみよう

1. ログインする.
2. **Terminal** を動かす (TSUBAME と直接対話する窓口).
 - 2.1. **cd** cs2kadai1
 - 2.2. **ls** ← directory simex があることを確認
 - 2.3. **cd** simex ← simex の部屋に行く(ここで実験しよう)
ls ← ファイルを確認しよう
fire.exe ex.rb ← これらを使う
fire0.exe ← これは燃える様子を出す方
fire.c, fire0.c, mt.h ← C 言語のプログラム

実は速い実行ができるように
C言語で書いたプログラムを**コンパイル**
したものを使っている

覚えてます?!
教科書 2 章を

実際のスクリプト例

ex.rb

```
# 実験の設定: 固定パラメータ
n = 50; nt = (2*n+1)**2; limit = nt * 0.005 # 全焼の基準 (0.5%)

# 実験用パラメータ
ex = 30; b = 5; p = 10; time = 150; seed = 1

# 実験
bcount = 0
for i in (1..ex)
  result = `./fire.exe #{n} #{p} #{b} #{time} #{seed}`
  r1, r2, r3 = result.chomp.split(/¥s*,¥s*/)
  t = r1.to_i
  nb = r2.to_i
  nu = r3.to_i
  if nu <= limit
    bcount = bcount + 1
  end
  print(t, ", ", nu, ": ", bcount, "¥n")
  seed = seed + 1; i = i + 1
end
R = bcount.to_f / ex.to_f
print(p, ", ", R, "¥n")
```

まとめ

Terminal 上のコマンド

命令	使用例	意味
mkdir	mkdir kadai2	kadai2 というフォルダ(部屋)を作る
cd	cd kadai2	kadai2 というお部屋に入る
	cd ..	上の(大きな)部屋に戻る
	cd ../..	上の上の部屋に戻る
ls	ls	その部屋にあるファイルを表示する
rm	rm foo.rb	foo.rb を消す(戻らないので注意)
cat	cat foo.txt	foo.txt の中身を画面に表示する
機械語実行	./○○.exe	機械語プログラムを実行する