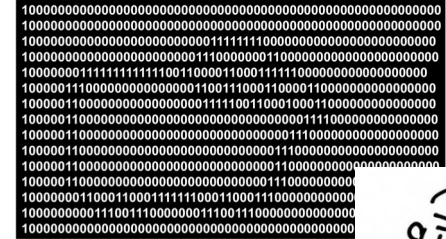


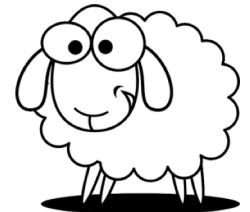
CS第1 レポート課題1

課題 四則演算でアニメーション

- 四則演算でアニメーションを行う Ruby プログラムを作成
- 作成したプログラムに関するレポートを作成



ひつじさん



CS第1 レポート課題1

提出方法

- OCWi から提出: 10月10日 レポート課題1
- 提出期限: 10月17日 午前9:00
 - 10月17日 午後9:00までのレポートは, 採点しますが, 大きく減点します.
- 提出物
 - レポート(PDFファイル): kadai1.pdf
 - プログラム: anime.rb

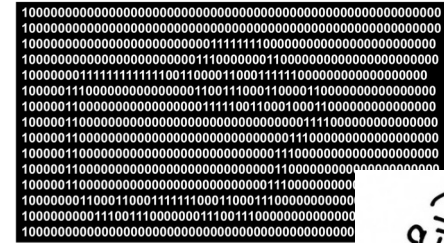
CS第1 レポート課題1

課題 四則演算でアニメーション

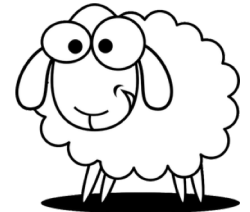
- ・ 四則演算でアニメーションを行う Ruby プログラムを作成

アニメーションプログラムの要求仕様

1. ファイル名は anime.rb とすること.
2. 動きがあること.
2. 「計算」を使うこと.
 - 演算は四則演算程度
 - 繰り返し文, 条件分岐文もOK
3. おもしろいこと. ← これはオマケ



ひつじさん

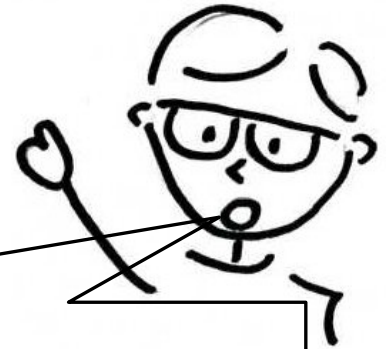


要求仕様とは
プログラムに要求される条件のこと

CS第1 レポート課題1

課題 四則演算でアニメーション

レポート内容



採点者は**プログラムも読むかも**！
レポートに書かれたことのみで採点します！

1. 作成したアニメーションプログラム anime.rb の使い方、
ならびにアニメ効果の説明
 2. 作成したアニメーションプログラムの計算の仕組みの説明
 3. 工夫した点の説明(これは別途**加点項目**となる可能性有り)
- 付録: アニメーションプログラムを印刷したもの (配点5点)

採点基準(満点 15) レポート点数 (15) + 工夫 (~ 10 ← 加点)

アニメーション・コンテスト

なんだってえ！
俺の力作を見てくれ～！！

よろしい，見てあげましょう



採点者は**プログラムも読むかも**

CS第1 レポート課題1

アニメーションプログラムの例

ステップ1: 絵を作る

```
# smile.rb
# 出力: スマイルマーク

d1 = 10000000000000000000000000000000
d2 = 10000000000110000110000000000000
d3 = 10000000000110000110000000000000
d4 = 10000000000000000000000000000000
d5 = 10000011000000000000000110000000
d6 = 1000000110000000000001100000000
d7 = 10000000110000000001100000000
d8 = 10000000001111111100000000000
d9 = 10000000000000000000000000000000
d10 = 10000000000000000000000000000000
```

各変数には 28 桁の数が入っている
その各々が絵の1行分を表している。

```
t = 0
while t < 29
  puts(d1)
  puts(d2)
  puts(d3)
  puts(d4)
  puts(d5)
  puts(d6)
  puts(d7)
  puts(d8)
  puts(d9)
  puts(d10)
  puts()
  sleep(0.1)
  t = t + 1
end
```

t = 0 ~ 28 まで
29 回繰り返す。

空行を出力

「0.1秒休メ」という命令

CS第1 レポート課題1

アニメーションプログラムの例

```
# smile.rb
# 出力: スマイルマーク

d1 = 10000000000000000000000000000000
d2 = 1000000000011000011000000000000
. . . .
t = 0
while t < 29
  puts(d1)
  puts(d2)
  . . . .
  puts()
  sleep(0.1)

  d1 = d1 / 10
  d2 = d2 / 10
```

} 画面に出す部分

ステップ2: 動かす

```
d3 = d3 / 10
d4 = d4 / 10
d5 = d5 / 10
d6 = d6 / 10
d7 = d7 / 10
d8 = d8 / 10
d9 = d9 / 10
d10 = d10 / 10

t = t + 1
end
```

あとは各自で工夫
して下さい