

プログラミング基礎

乱数とシミュレーション

今日の内容

- 乱数
- 待ち行列シミュレーション

乱数

【使い方】

Aの乱数

- 0～A-1までの乱数を生成

(例)

10の乱数

- 0～9までの乱数を生成

確認(1) 乱数

- 以下のプログラムを実行してみよう

```
100回  
  10の乱数を表示
```

※「設定」で「実行方式」を「コンソール」に変更しておくといい

練習(1) 丁半ゲーム

- 次のようなプログラムを作成する
 - ▣ まずユーザは丁(偶数)か半(奇数)を選択する
 - 尋ねるを使う
 - ▣ コンピュータは1～6までの乱数を1個生成する
 - その数がユーザの選択と同じならばユーザの勝ち
 - そうでないならばコンピュータの勝ち
 - ※ 普通はサイコロを2個使いますがここでは1個だけの想定
 - ▣ 次ページにサンプルプログラム
 - ▣ できた人はサイコロを2個使うように変更してみよう

練習(1) 丁半ゲーム

選択肢は「丁半」を文字列分解したもの
選択肢からリスト選択して、ユーザ選択に代入
コンピュータ選択はサイ振り

「あなたの選択は {ユーザ選択} 」と表示
「サイ振りの結果は {コンピュータ選択} 」と表示

もしユーザ選択=コンピュータ選択ならば
「あなたの勝ち」と表示
違うなら
「あなたの負け」と表示

●サイ振り

サイコロ =

もし (サイコロ%2=0) ならば「丁」を戻す
「半」を戻す

乱数初期化

【使い方1】

Aで乱数初期化

- 乱数の種をAとする

【使い方2】

乱数初期化

- 適当な数を乱数の種とする

注意) 乱数初期化

- コンピュータの「乱数」は計算して生成している
 - 完全にランダムな訳ではない
 - 同じ種からは同じ乱数が生成される

10で乱数初期化
100回
10の乱数を表示

待ち行列シミュレーション

- 窓口には何人ぐらいの人が待つか？
 - 客は概ね間隔 L (分)で到着する
 - 窓口での処理は概ね M (分/人)かかる



- 毎分の客の到着 = $1/L$ の確率
 - L 面のサイコロを振って特定の面(1とか)が出たら客到着
 - 毎分の窓口処理 = $1/M$ の確率
 - M 面のサイコロを振って1特定の面が出たら1人処理終了
- ※もちろん客がいるときのみ処理する

確認(2) 待ち行列シミュレーション

```
// パラメータ
来客間隔 = 10
処理時間 = 8
実験時間 = 300

// プログラム本体
待客数 = 0
処理数 = 0

乱数初期化
実験時間回
    来客チェック = 来客間隔の乱数
    もし来客チェック = 0ならば
        待客数 = 待客数 + 1
    処理チェック = 処理時間の乱数
    もし (処理チェック = 0) かつ (待客数が1以上) ならば
        待客数 = 待客数 - 1
        処理数 = 処理数 + 1
    「 {回数} : 待客数 = {待客数}   処理数 = {処理数} 」 と表示
```

練習(2) 待ち行列シミュレーション

以下の変更を行ってみよう

- 最後に平均待ち行列長を表示
 - ▣ 各時刻の待ち行列長の合計/実験時間
- 各時刻の待ち行列長をファイルに出力
 - ▣ Excelでグラフを書ける(時間があればやります)

待ち行列理論

- 単位時間あたりの来客数 λ
 - $\lambda = 1/L$ ($L=10$ なら $\lambda = 0.1$)
 - 単位時間あたりの窓口処理数 μ
 - $\mu = 1/M$ ($M=8$ なら $\mu = 0.125$)
 - 単位時間あたりの混み具合 ρ
 - $\rho = \lambda / \mu$ ($L=10, M=8$ なら $\rho = 0.8$)
 - 平均待ち行列長さ(理論値)
 - $W = \rho / (1 - \rho)$ ($\rho = 0.8$ なら $W = 4$)
- ※ 理論値と実験値がどれぐらい合うか調べてみよう

第8回 課題

- 窓口が2つの場合の実験を行うプログラムを作成
 - ▣ 待ち行列は1つとする
 - ユニクロとかロフトのレジ方式
- 余力のある人は以下にもチャレンジしてみよう
 - ▣ 窓口ごとに待ち行列ができる場合
 - スーパーのレジ方式