

エキスパート活動 課題A 「乱数って何?どのように活用するの?」

太郎：乱数って、何ですか？

花子：乱数とは、簡単に言うと、「**予測不可能な数**」のことだよ。

太郎：うーん、ちょっとイメージがでないなあ。難しい……。もう少し詳しく教えてください。

花子：もう少し詳しく説明すると、例えば、0 から 9 までの数字が、それぞれ同じ確率 1/10 で現れるように並べられた数字の列のことを言います。特定の法則やパターンがなく、次の数が何になるか予測することはできないわ。

太郎：そういうことか。何となく「乱数の発生」の仕組みが分かりました。では、乱数はどのような所で利用されているのですか？

花子：様々な場面があります。例えば、**シミュレーション**、**暗号化**、**ゲーム**など、2学期に学習した統計学の標本の無作為抽出などにも使われています。あとは、**ロボット制御**、**金融工学**など、幅広い分野で活用されていますよ。

太郎：あ、2学期に学習したデータサイエンスにも「無作為抽出」というキーワードがあったのを今思い出しました。このような分野でも使われているのですね。ゲームでは、どのように利用されているのですか？

花子：ゲームでは、乱数は予測不能性と多様性を生み出すために、非常に重要な役割を果たしていますよ。例えば、アイテムドロップ。敵を倒した際に、どのアイテムがドロップするのかをランダムに決定することで、プレイヤーに「何が手に入るのか」という期待感を与えています。他には、ダメージ計算。攻撃力や防御力に加えて、乱数によってダメージのバラつきを作り出すことで、戦闘に緊張感を持たせていますよ。他にもまだまだ利用されていますよ。

花子：あ、太郎さん、ゲームが好きなら代表的な『数当てゲーム～コンピュータが決めた数字を、プレイヤーが当てるミニゲーム～』を紹介しますね。

太郎：はい、ありがとう。

1. 以下は『数当てゲーム』のプログラムです。まずは正しく動作するのか個人でプログラミングしてみよう。

```
import random

# 1から50の間で答えを生成
answer = random.randint(1, 50)
print("1から50までの数字を当ててみてね!")

while True:
    guess = int(input("予想した数字を入力: "))

    if guess < answer:
        print("もっと大きいよ!")
    elif guess > answer:
        print("もっと小さいよ!")
    else:
        print("正解!おめでとう!")
        break
```

2. 質問&ミッション(確認!)

- ① import random は、なぜ必要？
- ② 乱数の発生範囲は？何以上何以下？
- ③ While と For の違いは？
- ④ While の後ろに True があるが、これは何を意味している？
- ⑤ 最後の行 break は何のためにある？

エキスパート活動 課題B 「おみくじプログラム」

1. 乱数とは 「ある一定の範囲内ですべての数が同じ確率で現れるような数のこと。」

2. 3人の対話から『乱数の活用』場面を知ろう

太郎：サイコロみたいな感じかな…？

次郎：たしかに、どの目も同じ確率で出ますよね！

花子：そうですね、サイコロは「1から6の範囲内で、すべての数が同じ確率で」現れます。まさに乱数です。

花子：この乱数を使うと、例えばくじ引きのようなプログラムが作れます。0～10のうち5より小さければアタリ、5より大きければハズレ、5だったら特別賞！みたいなくじ引きですね。

太郎：なるほど～、それなら、出力する内容をアタリ、ハズレじゃなく「大吉」とかにすれば占いゲームも作れそう。

次郎：ゲームでもよく、何が出てくるかわからない、くじ引きみたいなものがあるね！

花子：2人ともその通りですよ。ランダムで出現させたい機能で幅広く使うことができます。ほかにも乱数はシミュレーションをするのに役立ったりします。

花子：例えばサイコロのいずれかの目が出る確率は1/6ですが、それを確認するために100回サイコロを振る、というのはちょっと大変ですよ。

太郎：めんどうだね…

次郎：しかも100回でいいのかっていうと、もっともっとサイコロを振らなきゃいけない気がする…

花子：そういうときにも1～6の範囲の乱数を使うと、1万回でも10万回でも、サイコロを振ったとしたらどうなるか？のシミュレーションができます。

次郎：なるほど！人がやると大変なことをコンピュータ上でシミュレーションできるんですね！

花子：それでは、乱数を使ったプログラムの例をみてみましょう。

3. 乱数を使ったプログラム例

「0～9のうち5より小さければアタリ、5より大きければハズレ、それ以外、つまり5だったら特別賞と表示するくじ引きプログラムをつくと次のようかけます。」 → 【右のプログラム】

```
import random

r = random.randrange(10)

if r < 5:
    print('アタリ')
elif r > 5 :
    print('ハズレ')
else:
    print('特別賞')

print(r)
```

発問① 乱数で生成される数はどんな数？何から何まで？

発問② アタリ、ハズレ、特別賞、それぞれの確率は？

4. 『おみくじプログラム』を作成せよ

上のプログラムを参考にしながら、1/10の確率で大吉、1/10の確率で凶、そのほかは吉と表示される占いプログラムをつくりましょう。乱数で生成させた数も表示させましょう。【Google Colaboratoryで作成しよう】
☆プログラムが完成したら、振り返って改善できることを考えてみよう。また、このプログラムを発展させると、どのようなことができますか。考えてみよう。

【3の発問の解答と4のプログラム例】

3. 乱数を使ったプログラム例

「0～9のうち5より小さければ当たり、
5より大きければハズレ、それ以外、つまり
5だったら特別賞と表示するくじ引きプログラム
をつくと次のようにかけます。」 → 【右のプログラム】

```
import random

r = random.randrange(10)

if r < 5:
    print('当たり')
elif r > 5:
    print('ハズレ')
else:
    print('特別賞')

print(r)
```

発問① 乱数で生成される数はどんな数？何から何まで？

0～9

発問② アタリ、ハズレ、特別賞、それぞれの確率は？

アタリ … 5/10
ハズレ … 4/10
特別賞 … 1/10

4. 『おみくじプログラム』を作成せよ

上のプログラムを参考にしながら、1/10の確率で大吉、1/10の確率で凶、そのほかは吉と表示される占いプログラムをつくりましょう。乱数で生成させた数も表示させましょう。【Google Colaboratoryで作成しよう】
☆プログラムが完成したら、振り返って改善できることを考えてみよう。また、このプログラムを発展させると、どのようなことができますか。考えてみよう。

```
import random

r = random.randrange(10)

if r == 0:
    print('大吉')
elif r == 1:
    print('凶')
else:
    print('吉')

print(r)
```

⇒ 大吉
0

※上記の『大吉』と『凶』の判定で使用する数値は、これと異なる数字でも構わない

エキスパート活動 課題C 「乱数とリストを組み合わせたプログラム」

1. 乱数の種類 ~乱数には、大きく分けて擬似乱数と物理乱数の2種類がある~

○**擬似乱数**：高速に生成でき、多くの用途に適しているが、予測可能性があるため、セキュリティが重視される場面では注意が必要である。

○**物理乱数**：真の乱数に近い性質を持つが、生成速度が遅く、コストが高い。高いセキュリティが求められる場面で利用される。

※「rand関数は一般的に擬似乱数を生成する関数である。」

2. 3人の対話から『乱数の活用』場面を考えよう

花子：ここまで学習したリストと乱数を組み合わせて、ゲームのガチャのような仕組みを作ってみましょう。

次郎：ガチャか…クリックすると、ランダムでキャラクターとかアイテムとかが表示される仕組みということだよな。

太郎：どんなアイテムがゲットできるかわからないから、毎回わくわくするんだよね。

花子：そこがポイント。どんなアイテムが出るかわからない=ランダムで選ばれるということですね！

次郎：そうか、アイテムの種類を**配列(リスト)**の形式で書いて、選択される**添字**をランダムで決まるようにすればいいのかな。

花子：その通り！さっそくやってみましょう。その前に次のプログラムをみて。

3. 乱数とリストを組み合わせたプログラム例

「今日のラッキーカラーを教えてください
占いプログラムをつくと次のように
かけます。」→【右のプログラム】

```
#占い
import random

r = random.randint(0,5)

color = ['赤','青','緑','紫','白','橙']

print('今日のラッキーカラーは',color[r])
```

発問① 乱数で生成される数はどんな数？
何から何まで？

発問② それぞれの色が出現する確率は？

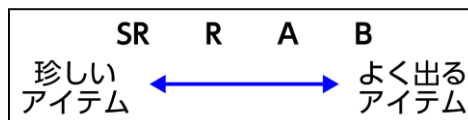
🔄 今日のラッキーカラーは 青

4. 『乱数とリストを組み合わせたプログラム』を作成せよ

上のプログラムをもとに「ゲームでアイテムが入手できるプログラム」をつくりましょう。

アイテムにはレア度が設定されており、

『SR』『R』『A』『B』の4種類があります。



それぞれのアイテムが出る割合は、SR:R:A:B=1:2:3:4となるように、

0~9の数をランダムで出現させ、0の場合は『SR』、1,2の場合は『R』、3~5の場合は『A』、それ以外は『B』のアイテムとなるようにしましょう。

実行例1)

A
3

実行例2)

R
1

【3の発問の解答と4のプログラム例】

3. 乱数とリストを組み合わせたプログラム例

「今日のラッキーカラーを教えてください
占いプログラムをつくと次のように
かけます。」 → 【右のプログラム】

発問① 乱数で生成される数はどんな数？
何から何まで？

0~5

発問② それぞれの色が出現する確率は？

1/6

```
#占い
import random

r = random.randint(0,5)

color = ['赤','青','緑','紫','白','橙']

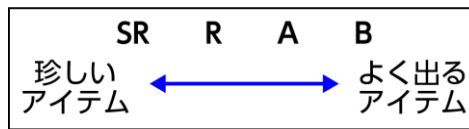
print('今日のラッキーカラーは',color[r])
```

⇒ 今日のラッキーカラーは 青

4. 『乱数とリストを組み合わせたプログラム』を作成せよ

上のプログラムをもとに「ゲームでアイテムが入手できるプログラム」をつくりましょう。

アイテムにはレア度が設定されており、
『SR』『R』『A』『B』の4種類があります。



それぞれのアイテムが出る割合は、SR:R:A:B=1:2:3:4 となるように、

0~9の数をランダムで出現させ、0の場合は『SR』、1,2の場合は『R』、3~5の場合は『A』、それ以外は『B』の
アイテムとなるようにしましょう。

実行例1)
A
3

実行例2)
R
1

```
import random
r = random.randint(0,10)
items=["SR", "R", "A", "B"]
if r == 0:
    print(items[0])
elif r == 1 or r == 2:
    print(items[1])
elif r > 2 and r < 6:
    print(items[2])
else:
    print(items[3])
print(r)
```

⇒ SR
0