

プログラミング演習I (第6回) 課題

• 基本① basic_Nabeatsu

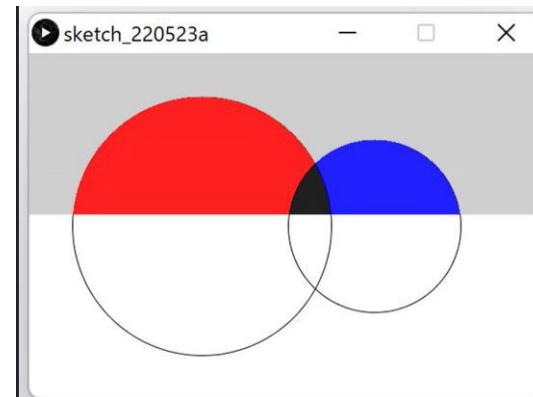
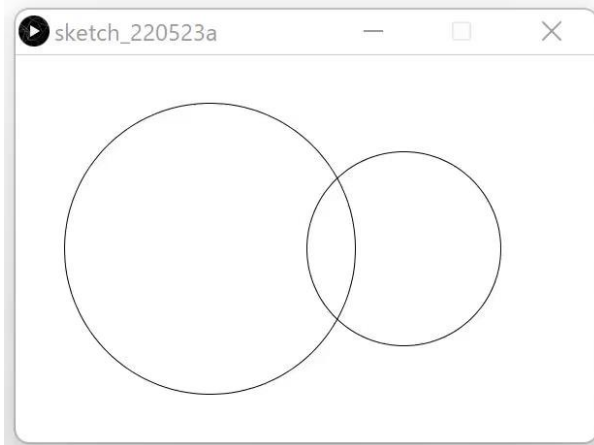
- プログラムを起動（実行）すると、99から1まで1ずつ標準出力に改行しながらカウントダウン表示するようにせよ
- なお、「数字が3の倍数」または「数字に3が含まれるとき」には、数字の代わりにahoと出力するようにせよ。
- whileまたはforを利用してプログラムを記述すること
- 右図は、出力例の一部である

```
aho
98
97
aho
95
94
aho
aho
91
aho
89
88
aho
:
```

プログラミング演習I (第6回) 課題

• 基本② basic_Pixel

- 600x400のウィンドウを作成し、下記のような(200, 200)を中心とした直径300の円、(400, 200)を中心とした直径200の円を描画せよ。
- 次に、1フレームごとに (drawごとに) y座標を0から1ずつ進め、そのy座標についてx座標を0から599まで変化させ、すべての点について、**点の場所**が「A only」なら赤色(255,0,0)、「B only」なら青色(0,0,255)、「A and B」なら黒色(0,0,0)、「not A or B」なら灰色(100,100,100)の点を描画していくことにより、下記のように徐々に色がついていくプログラムを作成せよ。例えば、yが0の時、1回のdrawで(0,0)(1,0)(2,0)...(599,0)を調べていきそのそれぞれに色を付け、横一列は一瞬で描画される。
- pointの色はstrokeで指定する。最初の円はsetupで描画しましょう



<https://gyazo.com/0505e0bef2cf753f5c9da5ecdb70ac23>

プログラミング演習I (第6回) 課題

• 基本③ スケッチ名 : basic_Dice4

「サイコロを4個振り、1つでも6がでる確率を求めよ」

- 1から6までの目が等しい割合で出るサイコロを4個振ったとき、1つでも6が出る確率をプログラムでシミュレーションすることで求めたい。
- randomを用いて4つのサイコロを振り、1つでも6があるかどうかをチェックするという試行を1億回繰り返すことによって求めた結果を標準出力せよ。
- なお、確率は0から1までの値での表記とする。

ヒント

- 基本課題①

- 99から1まで改行しつつ標準出力して、割り切れる条件と3が含まれる条件（10の位と1の位に3が含まれるかを判定する条件）で挙動を変えよう
 - 1の位の値はどうやって求める？（10で割った余り）
 - 10の位の値はどうやって求める？（10で割った値）

- 基本課題②

- drawのたびにxを0から599まで繰り返して変化させてそのすべての点について判定を行おう！
- 判定を行ったらその条件に応じて色を塗ろう！

- 基本課題③

- サイコロを4個振るのは4回randomをやるだけ
- 確率は何を何で割れば求まる？

プログラミング演習I (第6回) 課題

• 発展①スケッチ名 : advanced_Calendar

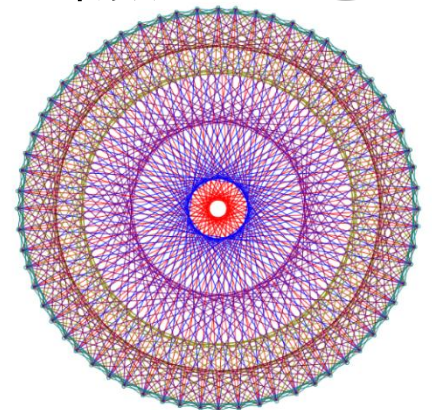
- プログラムを起動 (実行) するたびに、その月の1日目が日曜～土曜のどの曜日から始まるか、28～31日の何日で終わるかをランダムに決定し、下記のように1日から最終日 (28、29、30、31のいずれかになる) までをカレンダーとして曜日と日にちを標準出力するプログラムを作成せよ。
- なお、表示においてはスペースを使い、右揃えになるように工夫せよ。
- ただしプログラムはコメント行や空行を除き30行以内とせよ
 - 20行程度を目指せ
 - できるひとは文字数を削りまくってみましょう！
 - 私がやってみて139文字でした
 - 三項演算子を使わないと175文字

Sun	Mon	Tue	Wed	Thu	Fri	Sat
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

プログラミング演習I (第6回) 課題

• 発展課題② スケッチ名 : advanced_Mandala

- 800x800のウィンドウ上に, 中心から300ピクセル離れたところに, 等間隔に64個の小さな円を描け (直径が5ピクセルとする)
 - 半径300ピクセルの円周上に等間隔に小さな円を描く
- 64個のすべての点から, 素数個分だけ時計回り or 反時計回りに離れた点に線を描け (例, 5, 13, 19, 31など)
 - 例えば素数として「5」を選択した場合は, 1個目の点から6個目の点, 2個目の点から7個目の点, 2個目の点から7個目の点, ..., 63個目の点から68個目の点 (実際は $68\%64=4$ 個目の点), 64個目の点から69個目の点 (実際は $69\%64=5$ 個目の点) をそれぞれつなぐような線を描く
- 上記の処理について, 5つ以上の素数を使って線のパターンを描き 下記のような糸掛け曼荼羅を作成せよ. なお設定する素数ごとに色を変更することで見え方を工夫せよ



ヒント

- 64個の点の座標を0から63までの変数を使って表現しよう！
- 360度の64等分なので、5.625度ずつ離れている計算になる
 - 例えば素数として31を選定した場合は、 i 番目と $i+31$ 番目が対象となる
 - つまり $5.625 * i$ と $5.625 * (i+31)$ になる
- 度を使う場合は、radiansを使うとよいよ！