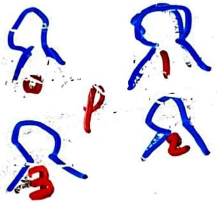

To write:

- ★ Use loops to pass turns
- ★ random(), srandom(SEED): gives unpredictable sequence of nums
- ★ After getting user_input, set the seed in srand()
- ★ use if else to check correct user_input
- ★ Each player rolls till: roll SIDE, Win game = ≥100 points
  - roll SIDE ↓ pass turn
  - Win game ↓ End Game when point[i] ≥ 100 & return names[i]
- ★ need to track each player's points
  - □ use integer array to keep points

TiPs

- ★ %d = int, %c = char, %f = float, %s = char*, % u = unsigned
- ★ use random() % 7 to roll pig
- ★ seed < 0 || seed > (uint32_t) UINT_MAX
- ★ if(pig[random() % 7] == SIDE)
- ★ names[10] = {'wilbur', ...}
  points[10] = { 0 , 0 ... }

Psudo code:

```
  Take user_input
  if (#check for correct input)
  else (default to prompt)
 set player #s, seed;
create int array to track points + designate to players
assign points to each position??? #SIDE=0, JOWLER=5
while (players[i] has points[i] ≤ 100)
 for (player[i] rolling)
   if (pos != SIDE)
     print current player, keep rolling, update scores???    where exactly???
   else
     pass to player [i+1]
```

if end of the game
print winner's name.

Purpose: The idea of this game is to take a user input for number of players as well as a "seed" number to put inside srandom(). If the user inputs fail the criteria/requirement we auto set players to 2, and seed to 2021. Right after, we create a dice using random Generated sequences of number but restrict the boundries to (0-6) = 7 indexes, each for 1/7 positions of the Pig/dice. Then we make an array that keeps scores for the corresponding # of players. We also make an array for the possitions of Pig and names of players. Now for the game, untill there is no winner we keep the game going. start with the 1st player, they keep rolling and adding points until they roll a SIDE = 0 points, or game over. If player rolls a SIDE and not at 100 points, the turn switches to the next player and same story goes on than back to player 1. Once a player reaches 100 game is over.

Description of Program:

1. use i/o to get user-inputs
2. use if else to make sure program can take care of erroneous inputs
3. make 2 arrays [10] for names + points
4. use while loop to keep game going
5. use for loops ~~to~~ cycle through players by rolling
6. use for loops to give + upate players' points
7. end while loop once any players reached 100
8. print winner's name

final Design

Changes and rewrite of the Psudo code:

~~int main() {~~

- Enumeriate Positions ~~and assign struct~~
- make points array for each side
- make array to describe the landing position (string)

- take user inputs for #. Players + Seed
- Check for input errors
- assign user input to both variables

- ~~cr~~ create sudo-random generator
- create int array for scores; keeps track of Players Points

- while loop set to infinite
  - do (roll + add points to players' scores)
    while (player doesnt roll SIDE and scores are < 100)
  - Check if player has reached 100 or more
    - Congratulate
- move to next player; if reached end of Players go back to the first.