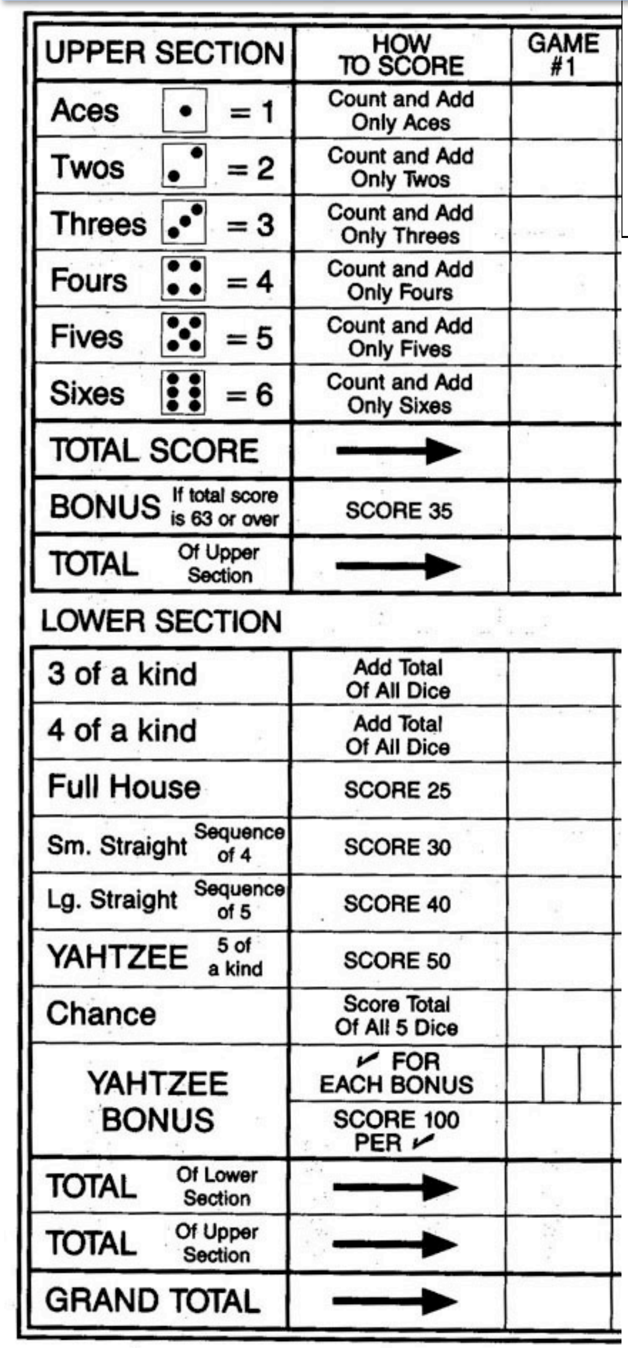
How to play Yahtzee, for robots

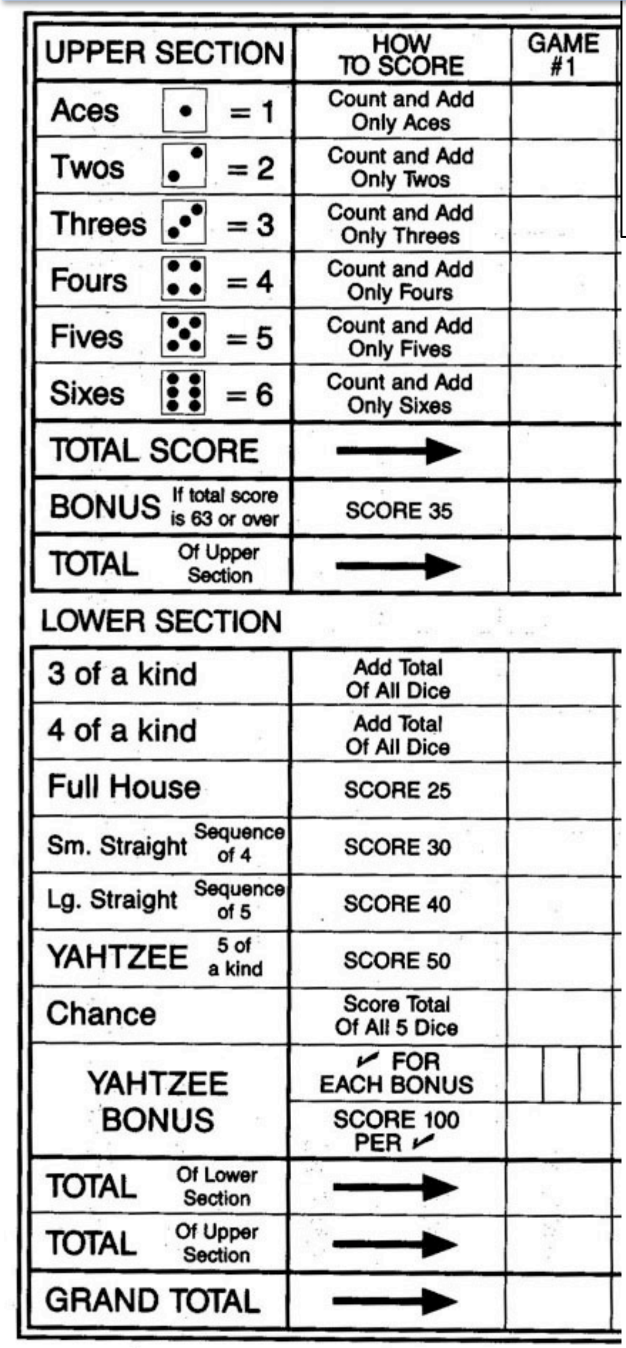
- Rules of the game
- Article on strategy
- Translating strategy into code
- Mechanical prototypes
- Computer/electronic version
- Future ideas



UPPER SECTION	HOW TO SCORE	GAME #1
Aces ⚀ = 1	Count and Add Only Aces	
Twos ⚁ = 2	Count and Add Only Twos	
Threes ⚂ = 3	Count and Add Only Threes	
Fours ⚃ = 4	Count and Add Only Fours	
Fives ⚄ = 5	Count and Add Only Fives	
Sixes ⚅ = 6	Count and Add Only Sixes	
TOTAL SCORE	→	
BONUS If total score is 63 or over	SCORE 35	
TOTAL Of Upper Section	→	
LOWER SECTION		
3 of a kind	Add Total Of All Dice	
4 of a kind	Add Total Of All Dice	
Full House	SCORE 25	
Sm. Straight Sequence of 4	SCORE 30	
Lg. Straight Sequence of 5	SCORE 40	
YAHTZEE 5 of a kind	SCORE 50	
Chance	Score Total Of All 5 Dice	
YAHTZEE BONUS	✓ FOR EACH BONUS	□ □ □
	SCORE 100 PER ✓	
TOTAL Of Lower Section	→	
TOTAL Of Upper Section	→	
GRAND TOTAL	→	

Rules

- Roll 5 dice in effort to fill the 13 categories
- Up to 3 rolls per turn
- 13 turns
- Scoring indicated on score sheet
- Taking 0 pts for a category is always ok
- Multiple Yahtzees:
  - Add 100 pts for Yahtzee Bonus
  - Score the roll in the upper section if available
  - Else fill any of the lower section – Full house, Small straight, and Large straight all get full points



UPPER SECTION	HOW TO SCORE	GAME #1
Aces ⚀ = 1	Count and Add Only Aces	
Twos ⚁ = 2	Count and Add Only Twos	
Threes ⚂ = 3	Count and Add Only Threes	
Fours ⚃ = 4	Count and Add Only Fours	
Fives ⚄ = 5	Count and Add Only Fives	
Sixes ⚅ = 6	Count and Add Only Sixes	
TOTAL SCORE	→	
BONUS If total score is 63 or over	SCORE 35	
TOTAL Of Upper Section	→	
LOWER SECTION		
3 of a kind	Add Total Of All Dice	
4 of a kind	Add Total Of All Dice	
Full House	SCORE 25	
Sm. Straight Sequence of 4	SCORE 30	
Lg. Straight Sequence of 5	SCORE 40	
YAHTZEE 5 of a kind	SCORE 50	
Chance	Score Total Of All 5 Dice	
YAHTZEE BONUS	✓ FOR EACH BONUS	□ □
	SCORE 100 PER ✓	
TOTAL Of Lower Section	→	
TOTAL Of Upper Section	→	
GRAND TOTAL	→	

Keys to success

- Extra 35 points if top section ≥ 63
  - 3 in each category → Σ = 63
- 50 points for Yahtzee
  - 100 points for additional!
- 40 for large straight
- 30 for small straight

Scores

- Maximum is 1575
- Maximum with 1 Yahtzee is 375
- Typically get a Yahtzee every 2-3 games
- 200 – 250 is a good average

Glenn 2006, An optimal strategy for Yahtzee

- Yahtzee is much more complicated that tic-tac-toe
- Use of “elementary” combinatorics and graph theory
- Bottom line:

Strategy	Expected Score	Std. Deviation
Yahtzees Only	171.52	68.17
Yahtzees and Straights	202.51	65.90
Greedy	218.05	46.87
Rational Yahtzees	219.86	65.99
Heuristic	240.67	60.90
Better Heuristic	244.87	57.39
Optimal	254.59	59.61

From Glenn, J. (2006). An optimal strategy for Yahtzee. Technical Report CS-TR-0002, Loyola College in Maryland, 4501 N. Charles St, Baltimore MD 21210, USA, May 2006.

Strategies

Strategy	Average	Rerolling	Scoring
Yahtzees Only	171.52	Keeps dice with greatest tally. Ties broken by highest open number, then highest rank	Score in whatever category gives greatest score on current turn
Yahtzees and Straights	202.51	Same as above, but keep a straight if Small Straight and Large Straight are both open	Same as above
Rational Yahtzees	219.86	More refined. Among other things, keep 2x over a 3x if the 2x category is open on top	Same as above

Glenn, J. (2006). An optimal strategy for Yahtzee. Technical Report CS-TR-0002, Loyola College in Maryland, 4501 N. Charles St, Baltimore MD 21210, USA, May 2006.

My own algorithms

	YAHTZEE BOTZEE SCORING COMPARISONS			RUN	MEAN	SD
VERSION	STRATEGIES		LINES	TIME	score	score
	rerolling	scoring		(s)	Yahtzees	Yahtzees
					Bonus	Bonus
4.02	no reroll	scores in order	349	1	**54.3**	**20.9**
4.03	reroll # with highest tally prefer higher #s	scores in order	353	1.5	**47.1**	**16.2**
4.07	reroll # with highest tally prefer higher #s	Precedence: Y,K4,K3,FH,K3,LS,SS Num 2x+,Chance 15+ Num (lo to hi), Chance Zero (Y,LS,SS,FH,K4,K3)	622	1.5	**162.6** 0.55 0.05	**59.8** 0.76 0.22
4.08	if SS is rolled, & SS is open, & LS is filled, hold all if LS is rolled, & (SS is open OR LS is open), hold all if FH is rolled, & rollcount is 2, & FH is open, hold all else reroll greatest # with max tally *fixed SS scoring glitch	Precedence: Y,K4,K3,FH,K3,LS,SS Num 2x+,Chance 15+ Num (lo to hi), Chance Zero (Y,LS,SS,FH,K4,K3)	672	2	**195.9** 0.60 0.15	**60.7** 0.68 0.37

More of my own

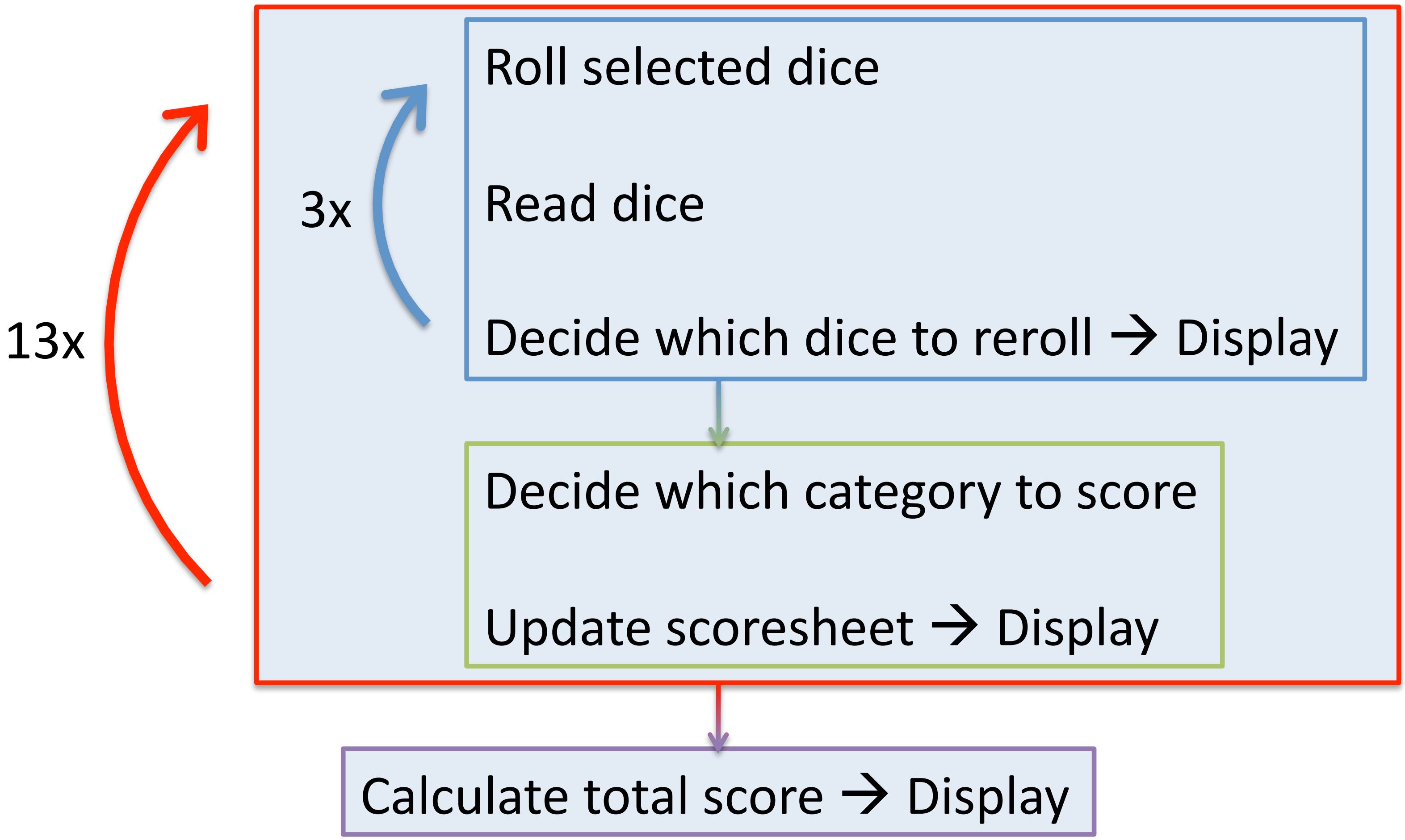
4.09	if SS is rolled, & SS is open, & LS is filled, hold all if LS is rolled, & (SS is open OR LS is open), hold all if FH is rolled, & rollcount is 2, & FH is open, hold all ELSE If possible, keep greatest max tally if num OR K3 open Else, If possible, keep greatest (max-1) tally still open Else, keep greatest max tally	Precedence: Y,K4,K3,FH,K3,LS,SS Num 2x+,Chance 15+ Num (lo to hi), Chance Zero (Y,LS,SS,FH,K4,K3)	757	2	**204.6** 0.63 0.15	**68.7** 0.74 0.36

4.11	if LS is rolled, & (SS is open OR LS is open), hold all
	if FH is rolled, & rollcount is 2, & FH is open, & turn >=5 hold all
	if FH is rolled, & FH is open, & turn >=11 hold all
	if SS is rolled, & (SS is open OR LS is open), hold SS and reroll other
	ELSE
	If possible, keep greatest max tally if num OR K3 open
	Else, If possible, keep greatest (max-1) tally still open
	Else, keep greatest max tally
		Precedence:
		Y,K4,K3>=15, Num 3x+,FH,K3,LS,SS
		Num 2x+,Chance 15+
		Num (lo to hi), Chance
		Zero (Y,LS,SS,FH,K4,K3)
			876	2	**226.3**
					0.46
					0.18
						66.3
						0.73
						0.39

More of my own

4.12	if LS is rolled, & (SS is open OR LS is open), hold all if FH is rolled, & rollcount is 2, & FH is open, & turn >=5 hold all if FH is rolled, & FH is open, & turn >=11 hold all if SS is rolled, & (SS is open OR LS is open), hold SS and reroll other ELSE If possible, keep greatest max tally if num OR K3 open Else, If possible, keep greatest (max-1) tally still open PLAN C. Else, If possible, keep greatest (max-2) tally still open Else, keep greatest max tally AND TARGET WHATEVER IS LEFT ON LAST TURN	Precedence: Y,K4,K3>=15, Num 3x +,FH,K3,LS,SS Num 2x+,Chance 15+ Num (lo to hi), Chance Zero (Y,LS,SS,FH,K4,K3)	1102	5	**226.2** 0.52 0.20	**66.5** 0.74 0.40
4.13	just like 4.12, minus PLAN C		1078	5	**240.3** 0.68 0.32	**59.9** 0.65 0.47
7.20	similar to above in terms of robot rerolling and scoring decision added human player via 3 buttons on NXT Brick added multiplayer with user selected # of players, up to 8, and user selection of which players are humans and which robots	added GL animations of dice spinning in this case, 8 robots playing together	1654	600+	**251.4** 0.75 0.13	**70.7** 0.89 0.35

Program architecture



13x

3x

Roll selected dice

Read dice

Decide which dice to reroll → Display

Decide which category to score

Update scoresheet → Display

Calculate total score → Display

Variables

```
//  Dice variables
bool die[]={0,1,1,1,1,1};
int dice[]={0,0,0,0,0,0};
int Tally[]={0,0,0,0,0,0,0};

//  Score variables
int nums[];int Bonus;int K3;int K4;int FH;
int SS;int LS;int Y;int BY; int C;int top;
int bot;int tot;

//  Flow variables
int i;int j;int k;int rollcount=0;int turn=1;
int rerollplan = 0;
```

Example scoring function

```
sub K3score()
{
if (Tally[1]>=3 || Tally[2]>=3 || Tally[3]>=3 ||
Tally[4]>=3 || Tally[5]>=3 || Tally[6]>=3)
{
K3[cp]= dice[1] + dice[2] + dice[3] + dice[4] +
dice[5];
}
else K3[cp]=0;
}
```

Example rerolling function

```
//                           PLAN A
//           Keep the dice that you have the most of

for(i=6;i>0;i--)
{
if( (Tally[i]>=Tally[1] && Tally[i]>=Tally[2] &&
Tally[i]>=Tally[3] && Tally[i]>=Tally[4] &&
Tally[i]>=Tally[5] && Tally[i]>=Tally[6]) && (nums[cp]
[i]==-1 || K3[cp]==-1) )
{
for(j=1;j<6;j++)
{
if(dice[j]==i) die[j]=0;   else die[j]=1;
}
return;
}
}
```

Example scoring function

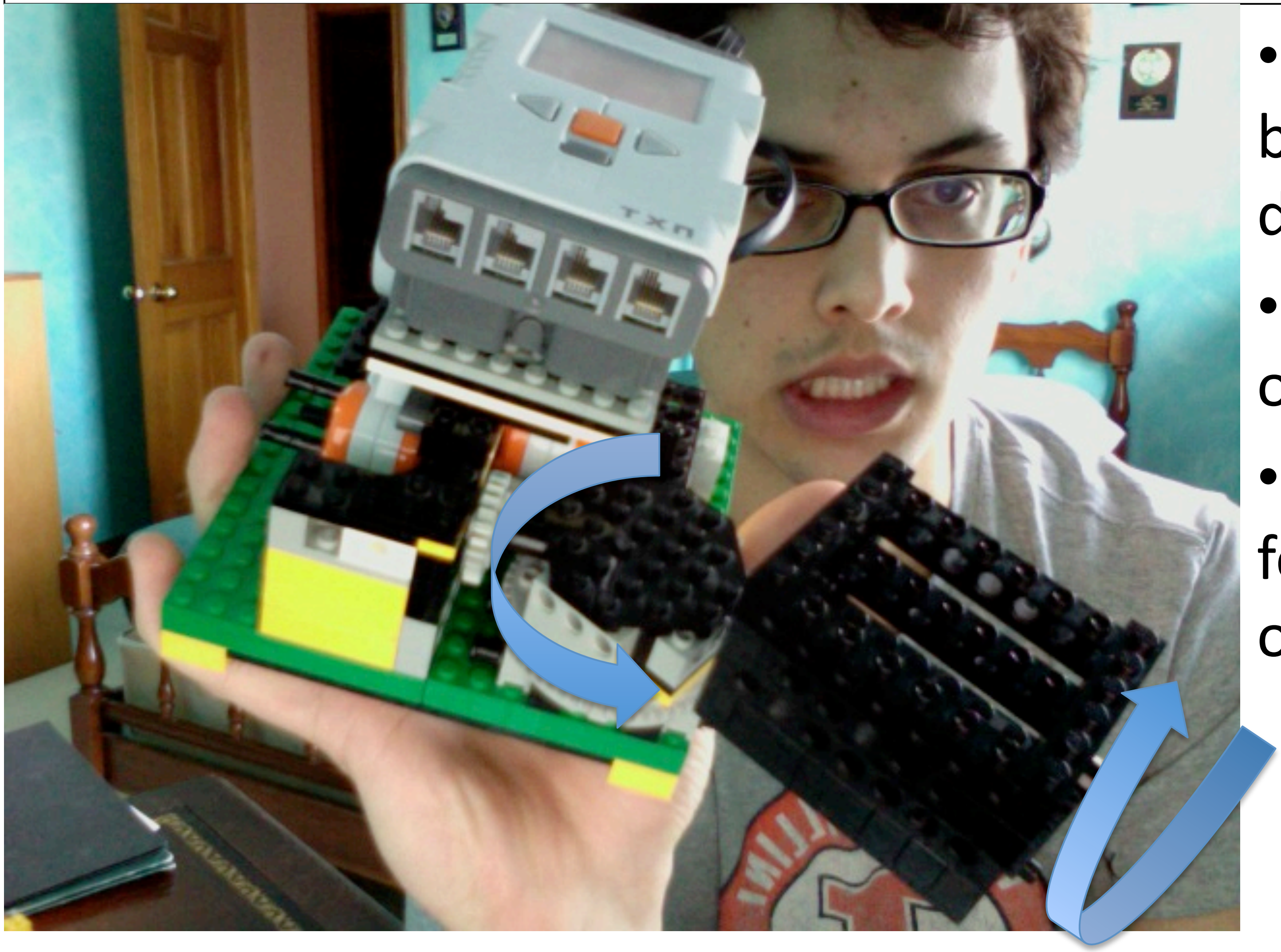
```
//                          LARGE STRAIGHT

if ( (Tally[1]==1 && Tally[2]==1 && Tally[3]==1 &&
Tally[4]==1 && Tally[5]==1) || (Tally[2]==1 && Tally[3]==1
&& Tally[4]==1 && Tally[5]==1 && Tally[6]==1) )
{
if(LS[cp]==-1)
{
LSscore();
return;
}
}

//  The magic is in the order in which you call these
kinds of functions.  i.e. Yahtzee, 4 of a kind, full
house, 3 of a kind >=15, 3 of a kind, large straight, …
```

Mechanical prototype

- Initial goal of this whole project was to make a Yahtzee-playing robot out of Legos
- Hardware: 1 Lego NXT Kit
    - 1 Microprocessor
    - 3 Motors with built-in optical encoders
    - 4 Sensors
        - Photocell, push button, mic, ultrasonic

Lego Kit

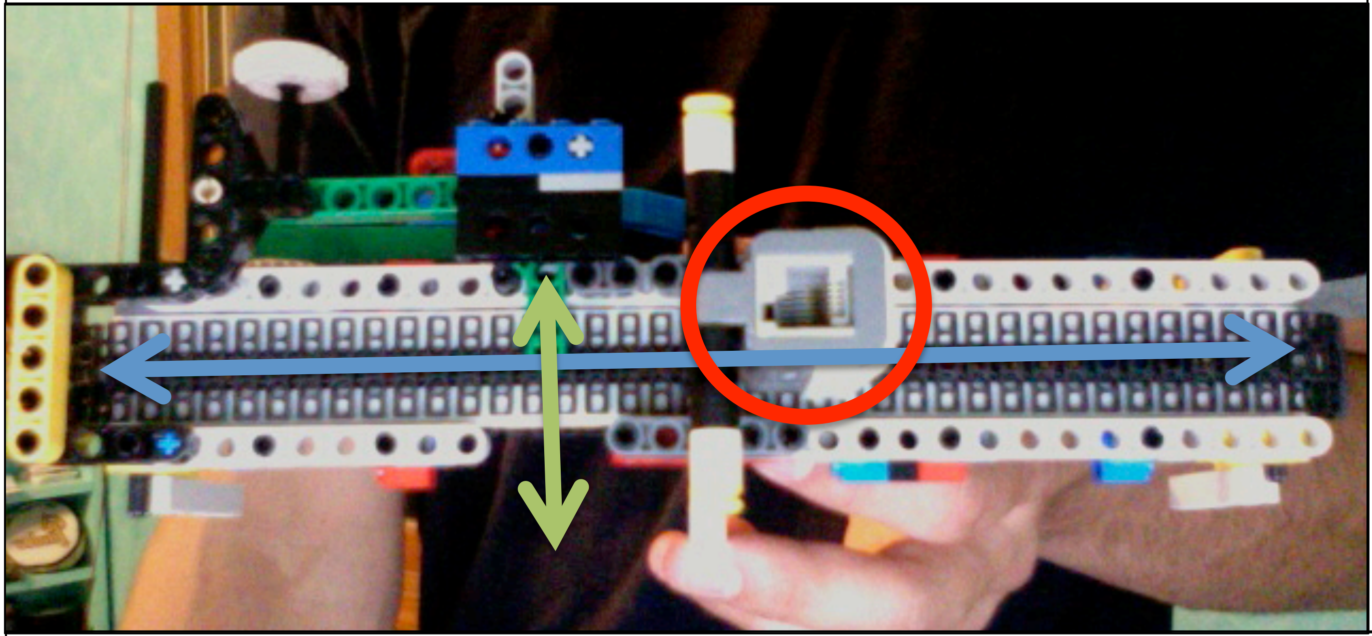
Design “2”



- Turntable and box shake the dice
- Box turns over to roll
- No clear plan for reading dice or sorting them

Design “3”

Top View



Blue Arrow:
Conveyor belt slides dice back & forth

Green Arrow:
Arm pops out to reject dice

Red Circle:
Light sensor reads dice (difficult for a simple photocell)

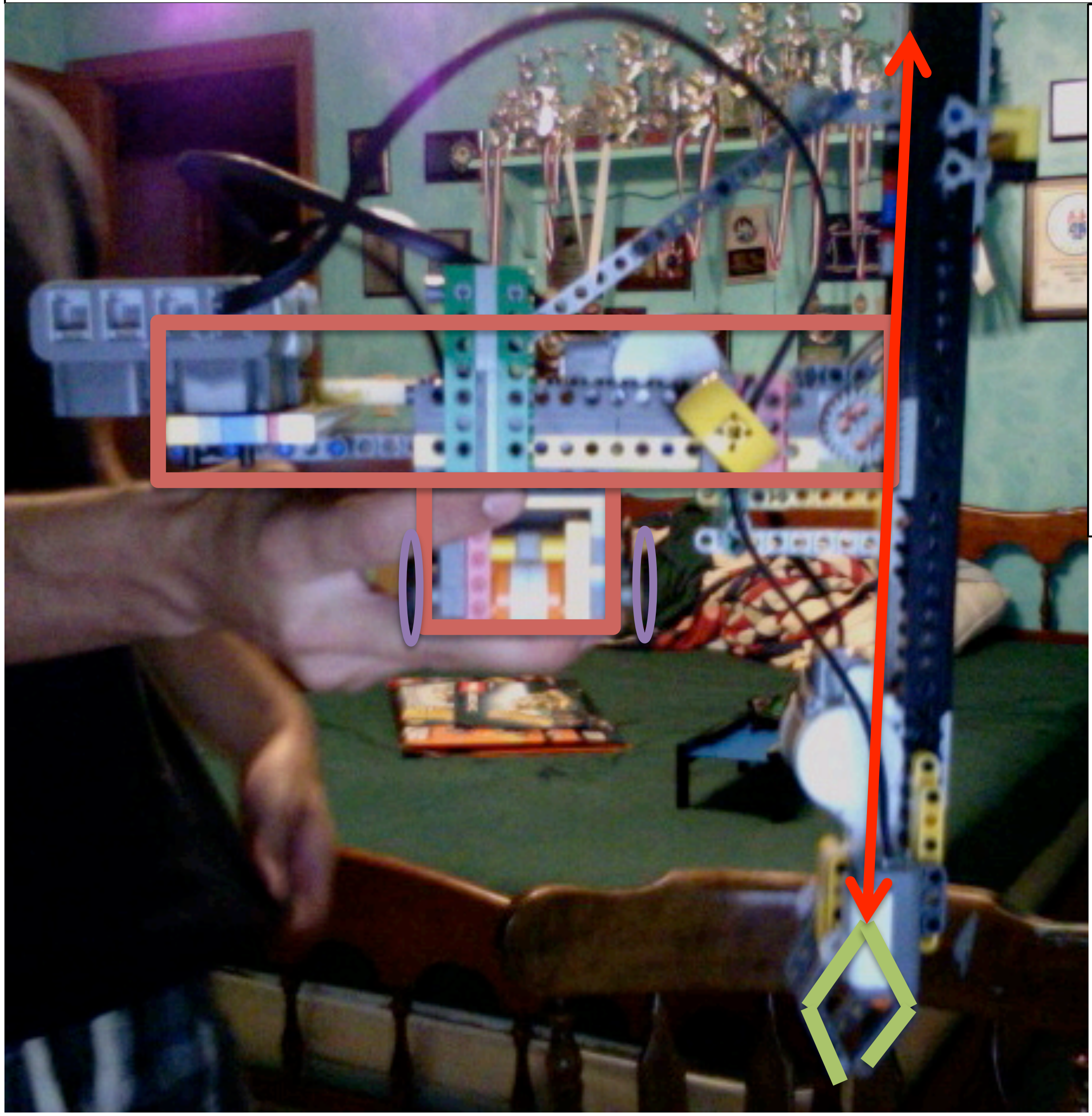
Operation

Rolling: Dice dumped onto the conveyor belt by an elevator (not shown)

Reading: By light sensor

Reroll decision: Unwanted dice are rejected by the arm, roll down a slide, and are rerolled by the elevator

Design "4"



Green Arrow:

Claw to grab die

Red Arrow:

Gear rack lifts and lowers die

Brownish:

Body of vehicle

Purple:

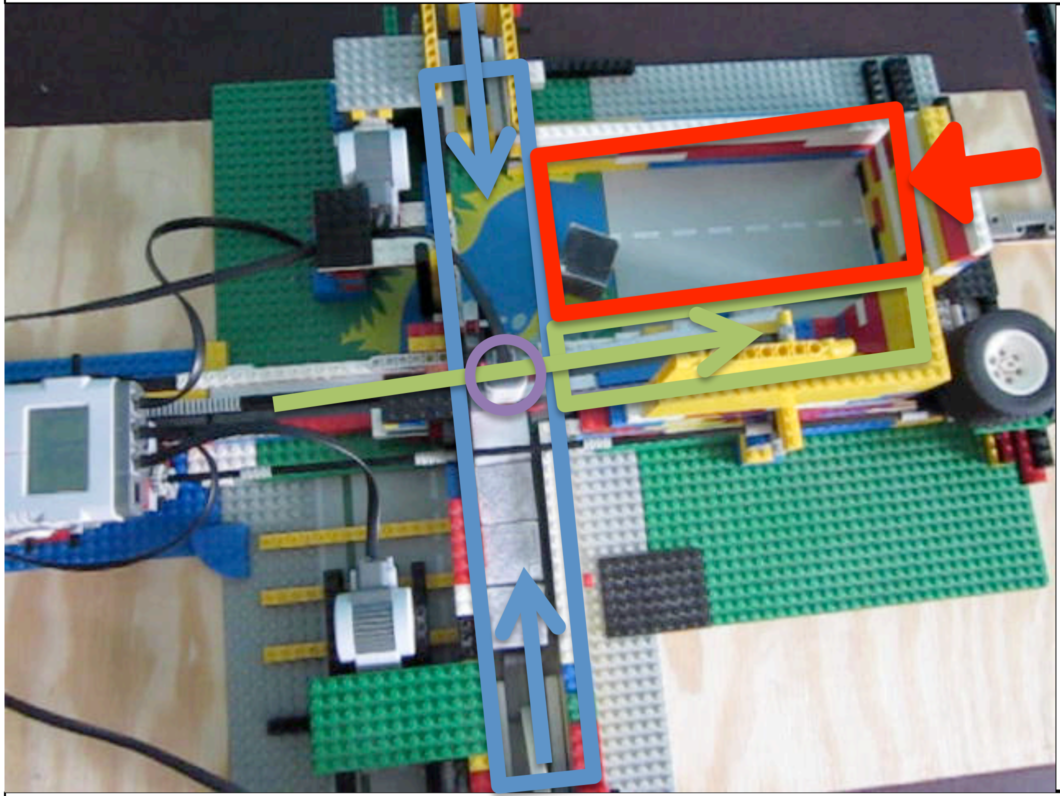
Wheels drive back and forth along catwalk

Operation

Dice are placed along ground below catwalk (not shown)

Robot drives to die, picks it up, drives it over to dice tower (not shown), rolls it, and brings it back

Design "5 / 6"



Blue:

Hallway and sliders to move and sort dice

Green:

Slide and slider to roll dice

Red:

Room and moving wall to straighten dice and return them to hallway

Purple:

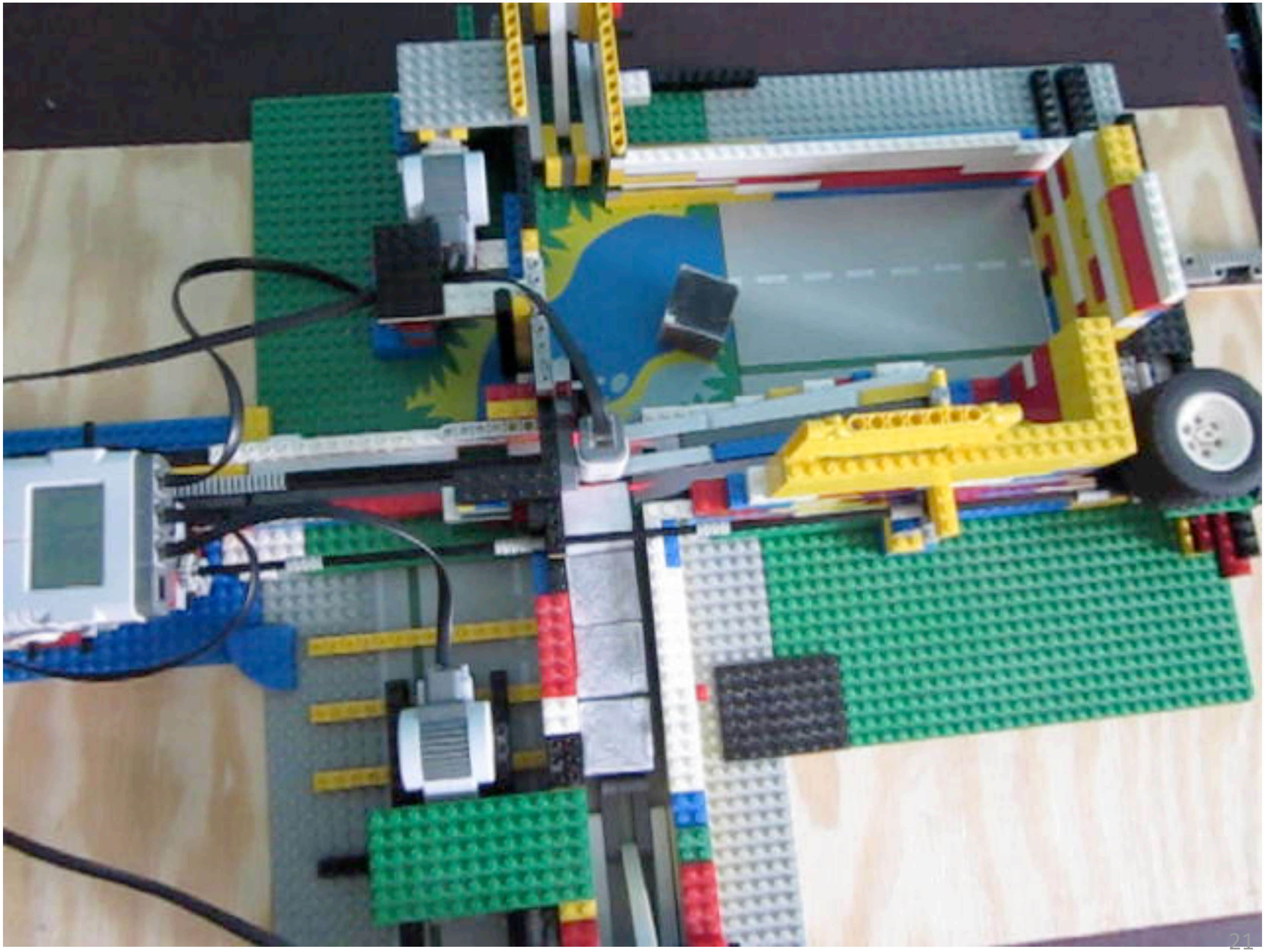
Light sensor to read custom grayscale dice

Dice are aligned in hallway using blue slider

1 by 1, dice are reroll by getting pushed up the slide, across the room, and back into the hallway

At this point, the die is read by the light sensor

Now the next die can be rolled



See video on Youtube http://www.youtube.com/user/PantsFaceHead#p/a/u/1/5VDzcsqOyhc

Mechanical summary

- Played 12 out of 13 turns
- System timed out after playing for 1 hour
- Jammed 3 times, requiring human intervention

→ Slow and inevitably suffers mechanical failures

→ Motivates an all electronic solution

“Computer”

- Still want to only use Legos
- Kit includes brick with microprocessor, tiny LCD screen, and 3 buttons

Implementation

- Most of the code can be copied
- Random number generator instead of dice
- Multiplayer
- Allow humans to play alongside the “robot(s)”

See video on Youtube http://www.youtube.com/user/PantsFaceHead#p/a/u/0/uY0eePAkkPI and show the real thing

Future ideas

- Develop an input device (i.e. mouse) to make it easier to select dice and categories
- Adapt the whole thing to run on an actual computer and add better graphics
- Keep track of all-time high scores