How Many Heads is Enough?
Analyzing Runs with the TI-83+

Professor Karen Benbury, Ph. D.
Mathematics Department
Bowie State University
Bowie, MD 20715

When looking for projects in elementary probability, I was interested in some that are easy to understand, but are nevertheless unusual projects. One such, that has been mentioned to me in passing, is the longest run of either heads or tails when tossing a fair coin. By a "run" is meant a sequence of tosses that result in all "heads" of all "tails". I give an assignment in which each student is asked to toss a coin 200 times. However, not everyone actually has to toss the coin. The students choose "heads" or "tails", and record their choice; we seal the record in an envelop, and then toss a coin. Those who "win" make up the data, while those who "lose" are supposed to actually toss the coin. A copy of a work sheet that I use for this purpose is included.

Copies of the results are made and distributed to each student, and each is to analyze the sequence of H's and T's to decide whether the tosses were real or faked. Each student is asked to analyze five or six data sets, so that we have five or six opinions on the authenticity of each data set. The professor also does this. While there is no way to know in advance, about half the data should be real and about half, faked. The origin of the data on each sheet is later revealed, so that we can all check our classifications. I have done this by "trusting" the students, and opening the records, and also by asking the student how he or she obtained the data (without asking how her or she was supposed to obtain the data). Usually, we expect that students who make up data will put down about half heads and half tails. They also usually know that exactly half heads and half tails is not all that likely, so they arrange to be off by a little. However, most do not have a good sense of how long the longest run of either heads or tails is likely to be.

The theory is not so easy to develop. In "The Longest Run of Heads" [The College Math Journal, Vol. 21, No. 3, May 1990, p. 196-206], Mark Schilling discusses the distributions of both the longest run of heads for a given number of tosses and the longest run for a given number of tosses of a fair coin. He also discusses biased coins. For 200 tosses, we might expect a longest run of 7, plus or minus 3 heads or tails in a row. Schilling suggests a rule of thumb of the integer nearest $\log_2(n/2) \pm 3$. In many of the courses I teach, the ideas needed to comprehend this paper are well beyond the scope of the course, so I wondered whether a long run relative frequency approach would help. Now, I can't have students perform many replications of tossing a coin 200 times, but calculators do not get so tired and bored. Moreover, the calculator can find the length of each run much more easily than people can. (This is a surprisingly difficult task.)

The program ONCE tosses a coin a number of times (to be selected), and stores the tosses in List 6 (under STAT edit) of the TI-83+, and the lengths of the runs in List 5. This program can be used to give the "losers" of the coin toss an easier time. The program MANY does a selected number of replications of tossing the coin 200 times. As currently written, it produces result of the last tossing of 200 coins in List 6, the lengths of the runs in the last tossing in List 5, and for each replication, the length of the longest run in List 4. Both call a third program, RUNCT, which "tosses" the coin and counts the lengths of the runs. It is not a good idea to ask for too many replications –the calculator is slow, and its memory can be overrun. However, in a class of 25, if each student asks for 5 or 6 replications, a lot of data can be gathered in a short period of time. Furthermore, one has an interesting lead-in to confidence intervals and hypothesis testing. A recent (small) class found a 95% confidence interval of (7.6, 8.4) for the average length of the longest run in 200 tosses of a fair coin. Both programs also allow one to elect to toss a fair die instead. Some sample results of using the programs are attached.

Another simple, interesting problem that is not usually in the elementary books is the question: how many times must one toss a fair die in order to have each of the numbers come up at least once? The program ALLNOS will do a requested number of replications of this experiment. Once again, it is a good idea not to ask for too many. Experimentally, we have found that about 15 are needed, but once again, the class was small.

Math Project Name_______________________

1. Toss a coin 200 times, and record the results (H or T for each toss) in the table below. If you are “faking” the data, try to make it seem real. Fill the table by rows.

2. How many heads did you get? _______

What was the proportion of heads? _________

3. Is this a surprise? What would you expect? Explain.

4. Make a frequency table of the length of the runs. A run is a sequence of symbols that are all the same. So if we toss a coin 20 times and get HHHHTTHHHTTTTTTHTHTT, the runs in this sequence are HHHH, TT, HHH, TTTTTT, H, T, H, TT. The lengths of these runs are 4, 2, 3, 6, 1, 1, 1, and 2, respectively. Thus, a frequency table would be

Length	Frequency
1	3
2	2
3	1
4	1
5	0
6	1

Make such a frequency table for the runs in your 200 tosses.

5. What is length of the longest run? ________

6. What is the average length of a run? ________

7. Does this surprise you? Explain.

Math Analysis Sheet Name_____________

Each of the work sheets that you have received has been given a number. Each person's work sheet will be analyzed by six other people. For each of the worksheets, examine the data, and decide whether the sequence of heads and tails is more likely to be real (or calculator simulated) data or faked data. Bear in mind that, while we do not know how many of each type there were in the class, there should be about half of each type. Give reasons for your classifications; it may be hard to express a "gut feeling", but try.

Sheet Number	Real or Faked	Reason(s) for Choice

Math Work Sheet Name___________________

We have seen that, for a fair die, the probability that each of the values comes up on a single toss is 1/6 theoretically, and about 1/6 using long range relative frequencies. Now we ask a slightly different question. In the average, how many tosses of a die are needed to have each number come up at least once?

Run your calculator program 20 times and bring your data to class.

Make a frequency table and a histogram.

Now, using the total amount of simulated data from the class, find a 95% confidence interval for the average number of tosses required to have each number appear at least once.

Many

```
ClrHome
ClrList L₄,L₅,L₆
Menu("TYPE","COIN",A,"DIE",B)
Lbl A
2 → F
0 → G
Goto D
Lbl B
6 → F
1 →G
Lbl D
Disp "NUM TOSSES"
Input N
Disp "REPLICATIONS"
Input R
1 → L
For(J,1,R)
prgmRUNCT
max(L₅) → L₄(L)
L+1 → L
End
Stop
```

ONCE

```
ClrHome
ClrList L₅,L₆
Menu("TYPE","COIN",A,"DIE",B)
Lbl A
2 → F
0 → G
Goto D
Lbl B
6 → F
1 →G
Lbl D
Disp "NUM TOSSES"
Input N
prgmRUNCT
Disp "MAX RUN",max(L₅)
Disp "MEAN RUN",mean(L₅)
Stop
```

RUNCT

```
1→C
int(F*rand)+G → S
S → T
1 → K
S → L6(1)
For(I,2,N)
int(F*rand)+G → S
S → L6(I)
If (S=T)
Then
C+1 → C
Else
C → L5(K)
1 → C
K+1 → K
S → T
End
End
C → L5(K)
Return
```

ALLNOS

```
ClrHome
ClrList L4
Disp "REPLICATIONS"
Input R
For(M,1,R)
ClrList L5,L6
1 → N
1 → I
int(6*rand)+1 → S
S → L6(I)
S → L5(N)
Repeat (dim(L6)=6)
int(6*rand)+1 → S
N+1 → N
S → L5(N)
0 → C
For(K,1,I)
C+(L6(K)=S) → C
End
If C=0
Then
I+1 → I
S → L6(I)
End
End
N → L4(M)
End
Stop
```