

EE231002 Introduction to Programming

Lab05. Blackjack and Expected Values

Due: Oct. 21, 2014

Blackjack is a popular card game. In this game, a player is dealt two cards initially. The number of points can then be calculated as follows. The cards **2** to **10** have the face values, face cards (**J**, **Q**, and **K**) have 10 points, and an **A** can have either 1 or 11 points. Depending on the total points, the player can make a decision to ask for more card (hit) or not (stand). As long as the number of points does not exceed 21, the play can continue to request for more cards. The objective of the game is to get more points than the dealer without going over 21 points.

In this lab, you will practice using the `rand()` function to find the some probabilities of playing Blackjack game. In particular, you need to create a table that lists the expected value of the total number of points after the first hit, and the probability of busting.

Example program output is given below.

```
$ ./a.out
```

Points	E(hit)	% Busted
2	9.29	0
3	xx.xx	xx.xx
4	xx.xx	xx.xx
5	xx.xx	xx.xx
6	xx.xx	xx.xx
7	xx.xx	xx.xx
8	xx.xx	xx.xx
9	xx.xx	xx.xx
10	xx.xx	xx.xx
11	xx.xx	xx.xx
12	xx.xx	xx.xx
13	xx.xx	xx.xx
14	xx.xx	xx.xx
15	xx.xx	xx.xx
16	xx.xx	xx.xx
17	xx.xx	xx.xx
18	xx.xx	xx.xx
19	xx.xx	xx.xx
20	26.48	92.33

The program output is essentially a table. The first column of the table is the number of points at hand. The second column is the expected value of points after one hit (adding one card). And the third column is the percentage of busting (number of points going over 21).

The Blackjack game is usually played with 2 or 4 decks of cards. A deck of cards has 52 cards with 4 different suits and each suit has 13 cards. In this lab, we assume the game is played using

a large number of decks and thus, the cards **A** to **K** all have equal probability whenever is dealt. You can use the following expression to draw a card:

```
k = rand() % 13 + 1;
```

The variable **k** has the value of the drawn card.

In practice, one needs to play the game a large number of times to get an accurate expected value. Thus, each row of the table should be the results of at least **10000** experiments. For example, for the first row we assume there are two points at hand and request a new card. The total number of points can then be calculated. This is one experiment. Repeat this experiment 10000 times, the sum of all the points of each experiment divides 10000 is then the expected value. The percentage of busted is simply the number of experiments that the total number of points going over 21 points. In your program please define the number of experiment as a macro **N**, $N \geq 10000$.

In this game, if a card **A** is drawn then it is taken as an 11 if it will not result in busting, otherwise it is treated as a 1.

Notes.

1. Create a directory **lab05** and use it as the working directory.
2. Name your program source file as **lab05.c**.
3. The first few lines of your program should be comments as the following.

```
/* EE231002 Lab05. Blackjack and Expected Values
   ID, Name
   Date:
*/
```

4. To use the **rand()** function, you need to include the **stdlib.h** header file in your program as

```
#include <stdlib.h>
```

5. After you finish verifying your program, you can submit your source code by

```
$ ~ee231002/bin/submit lab05 lab05.c
```

If you see a "submitted successfully" message, then you are done. In case you want to check which file and at what time you submitted your labs, you can type in the following command:

```
$ ~ee231002/bin/subrec lab05
```

It will show your submission records for lab05.

6. You should try to write the program as efficient as possible. The format of your program should be compact and easy to understand. These are part of the grading criteria.