

CSI 201 - Introduction to Computer Science

Sidebar: Top-down development

Professor Neil V. Murray
Instructor

craps.cpp

```
#include <iostream>
#include <cctype>
#include <cstdlib>
#include <ctime>
using namespace std;

void initialize_rng();
// Precond: None
// Postcond: The random number generator has been initialized with a seed value
//           based on the current time

int roll_dice(int& d1, int& d2);
// Precond: The random number generator has been initialized
// Postcond: d1 and d2 will contain the values of the dice rolled (1-6)
//           The function returns the sum of both dice

void playbatch() // This is just a stub -- developed next
{ cout << "I just played N games" << endl; }

void playinteractive() // This is just a stub
{ cout << "I just played N interactive games" << endl; }

////////////////////////////////////////////////////////////////////
// Main Program is on the following page
////////////////////////////////////////////////////////////////////
```

10/27/2006

CSI 201 -- Top-down Design

2

craps.cpp continued

```
int main()
{
    char biq;
    initialize_rng(); // Initialize the random number generator
    cout << "You are going to play craps. You should know the rules.\n\n";
    do
    {
        cout << "Enter b/B (batch) to simulate many games at once\n";
        cout << "Enter i/I (interactive) to play 1 to 9 games at a time\n";
        cout << "Enter q/Q to quit\n" << "What next? ";
        cin >> biq;

        switch (toupper(biq))
        {
            case 'B':
                playbatch(); break; // Let's do this next
            case 'I':
                playinteractive(); break;
            case 'Q':
                cout << "\nQuitting program\n";
                break;
            default:
                cout << "\nUnrecognized command\n";
        }
    }
    while (toupper(biq) != 'Q');
    return(0);
}
```

craps.cpp continued

```
void initialize_rng()
{
    int seed;

    // Get the seed used to initialize the random number generator based
    // on the current time from the system clock
    seed = time(NULL);
    srand(seed);
}

int roll_dice(int& d1, int& d2)
// uses reference parameters to report individual die which is
// not necessary; reporting the total is enough.
{
    d1 = rand() % 6 + 1;
    d2 = rand() % 6 + 1;
    return d1+d2;
}
```

10/27/2006

CSI 201 -- Top-down Design

4

10/27/2006

CSI 201 -- Top-down Design

3

craps.cpp

```
> craps
```

```
You are going to play craps. You should know the rules.
```

```
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? i
I just played N interactive games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? b
I just played N games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? X
```

```
Unrecognized command
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? q
```

```
Quitting program
```

10/27/2006

CSI 201 -- Top-down Design

5

craps.cpp

```
void playbatch() // This is just a stub -- developed next
{ cout << "I just played N games" << endl; }
```

10/27/2006

CSI 201 -- Top-down Design

6

craps.cpp

```
void playbatch(); // Now a declaration, def below
```

```
void playbatch() //
```

10/27/2006

CSI 201 -- Top-down Design

7

craps.cpp

```
void playbatch(); // Now a declaration, def below
```

```
int statsout(int w, int l) // Stub -- used in playbatch
{cout << "Here are the stats" << endl;} // Let's do this next ← ← ←
```

```
bool playgame(char mode) // Stub -- used in playbatch
{ return(true); } // the stub always thinks we won!!
```

```
void playbatch()
{
    int ngames, wins = 0, losses = 0;
    cout << "\nEnter the number of games to simulate: ";
    cin >> ngames;
    if (cin.fail()) exit(1); // non-int impossible to fix
    for (int i = 1; i <= ngames; i++)
        if (playgame('B'))
            wins++;
        else
            losses++;
    statsout(wins, losses);
}
```

10/27/2006

CSI 201 -- Top-down Design

8

craps.cpp

```
> craps
You are going to play craps. You should know the rules.

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? i
I just played N interactive games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? b

Enter the number of games to simulate: 2
Here are the stats
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? j

Unrecognized command
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? q

Quitting program
```

10/27/2006

CSI 201 -- Top-down Design

9

craps.cpp

```
void playbatch(); // Now a declaration, def below

int statsout(int w, int l)
{
    //precond: w and l are win and loss totals
    //postcond: outputs message on wins, losses, w-l percentage
    double floatgames;
    cout << "You have " << w << " wins and " << l << " losses: ";
    cout << " win percentage: ";
    floatgames = w + l; // store sum as float
    cout << 100.0 * (w / floatgames) << "%\n" << endl << endl;
}

bool playgame(char mode) // Stub -- used in playbatch
{ return(true); } // the stub always thinks we won!! ← ← ← Let's do this next

void playbatch()
{
    int ngames, wins = 0, losses = 0;
    cout << "\nEnter the number of games to simulate: ";
    cin >> ngames;
    if (cin.fail()) exit(1); // non-int impossible to fix
    for (int i = 1; i <= ngames; i++)
        if (playgame('B'))
            wins++;
        else
            losses++;
    statsout(wins, losses);
}
```

10/27/2006

CSI 201 -- Top-down Design

10

craps.cpp

```
> craps
You are going to play craps. You should know the rules.

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? I
I just played N interactive games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? B

Enter the number of games to simulate: 200
You have 200 wins and 0 losses; win percentage: 100%

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? {

Unrecognized command
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? Q

Quitting program
```

10/27/2006

CSI 201 -- Top-down Design

11

craps.cpp (making your point always fails)

```
bool playgame(char mode); // Now a declaration, def below

void report_WorL(bool win, char m) // This & report_roll next
{ cout << win; } // Stub, reports a 1 or 0 regardless of the mode

bool roll_for_point(int point, char mode){ return(false); } //stub; always loses

bool playgame(char mode)
{
    int d1, d2, total = roll_dice(d1, d2); bool win;
    report_roll(d1, d2, mode); // Stub, does nothing
    if (mode == 'I')
        cout << "on the come-out roll. ";
    switch (total)
    {
        case 7:
        case 11:
            win = true;
            break;
        case 2:
        case 3:
        case 12:
            if (mode == 'I')
                cout << "Craps. ";
            win = false;
            break;
        default:
            win = roll_for_point(total, mode); // Stub; always loses. ← ← ← Let's do this next
    }
    report_WorL(win, mode); return(win); // report_WorL is a stub
}
```

10/27/2006

CSI 201 -- Top-down Design

12

craps.cpp

```
> craps
You are going to play craps. You should know the rules.

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? i
I just played N interactive games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? b

Enter the number of games to simulate: 20
00000010000100000000You have 2 wins and 18 losses; win percentage: 10%

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? j

Unrecognized command
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? q

Quitting program
```

10/27/2006

CSI 201 -- Top-down Design

13

craps.cpp (playgame is now complete)

```
bool roll_for_point(int point, char mode)
{
    int newroll, d1, d2; bool win;
    do
    {
        newroll = roll_dice(d1, d2); // new roll
        report_roll(d1, d2, mode);
        if (newroll == point)
            return(true);
        else
            if (newroll == 7)
                return(false);
    } while (true);
}

bool playgame(char mode)
{
    int d1, d2, total = roll_dice(d1, d2); bool win;
    report_roll(d1, d2, mode); // Stub, does nothing ← ← ← Let's do this next
    if (mode == 'I')
        cout << "on the come-out roll. ";
    switch (total)
    {
        case 7:
        case 11:
            win = true;
            break;
        case 2:
        case 3:
        case 12:
            if (mode == 'I')
                cout << "Craps. ";
            win = false;
            break;
        default:
            win = roll_for_point(total, mode);
    }
    report_WorL(win, mode); return(win); // report_WorL is a stub ← ← ← Let's do this next
}
```

10/27/2006

CSI 201 -- Top-down Design

14

craps.cpp

```
> craps
You are going to play craps. You should know the rules.

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? i
I just played N interactive games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? b

Enter the number of games to simulate: 8
01001010You have 3 wins and 5 losses; win percentage: 37.5%

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? ?

Unrecognized command
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? q

Quitting program
```

10/27/2006

CSI 201 -- Top-down Design

15

craps.cpp

```
void report_roll(int d1, int d2, char m)
{
    if (m == 'I') // Send to cout stream only when in interactive mode
    {
        cout << "\nYou rolled " << d1 << " and " << d2;
        cout << " for " << d1+d2 << " ";
    }
}

void report_WorL(bool win, char m)
{
    if (m == 'I') // Send to cout stream only when in interactive mode
        if (win)
            cout << " You WIN!\n";
        else
            cout << " You LOSE!\n";
}

report_roll(d1, d2, 'I'); // report_roll(d1, d2, mode); (twice)
report_WorL(win, 'I'); return(win); // report_WorL(win, mode); return(win);
```

The functions above do nothing in batch mode; but we have not yet implemented interactive mode. How to test them?
Make a temporary change wherever we invoke them:

10/27/2006

CSI 201 -- Top-down Design

16

craps.cpp

```
> craps
You are going to play craps. You should know the rules.

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? I
I just played N interactive games
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? B

Enter the number of games to simulate: 2
You rolled 4 and 4 for 8
You rolled 1 and 2 for 3
You rolled 3 and 5 for 8 You WIN!

You rolled 4 and 4 for 8
You rolled 4 and 3 for 7 You LOSE!
You have 1 wins and 1 losses; win percentage: 50%

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? =

Unrecognized command
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? Q

Quitting program
```

10/27/2006

CSI 201 -- Top-down Design

17

craps.cpp (Finally: interactive mode)

```
void playinteractive(); // Was a stub, defined below

void playinteractive()
{
    int ngames, wins = 0, losses = 0; char again;
    do
    {
        ngames = 0;
        cout << "\nEnter y or 1-9 to play; n to leave mode: ";
        cin >> again; // characters '1'-'9' are 49-57 int
        cout << endl;
        // now either leave ngames=0 or set from 1 to 9
        if (toupper(again) == 'Y')
            ngames = 1;
        else
            if ((again > 48) && (again < 58))
                ngames = again - 48;
        for (int i = 1; i <= ngames; i++)
        {
            if (playgame('I'))
                wins++;
            else
                losses++;
            statsout(wins, losses);
        }
    } while (toupper(again) != 'N');
}
```

10/27/2006

CSI 201 -- Top-down Design

18

craps.cpp (Final working version)

```
> craps
You are going to play craps. You should know the rules.

Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? i

Enter y or 1-9 to play; n to leave mode: 2

You rolled 4 and 1 for 5 on the come-out roll.
You rolled 4 and 5 for 9
You rolled 2 and 2 for 4
You rolled 4 and 6 for 10
You rolled 4 and 2 for 6
You rolled 5 and 4 for 9
You rolled 6 and 1 for 7 You LOSE!
You have 0 wins and 1 losses; win percentage: 0%

You rolled 3 and 1 for 4 on the come-out roll.
You rolled 4 and 1 for 5
You rolled 3 and 4 for 7 You LOSE!
You have 0 wins and 2 losses; win percentage: 0%

Enter y or 1-9 to play; n to leave mode: y

You rolled 1 and 3 for 4 on the come-out roll.
You rolled 1 and 2 for 3
You rolled 1 and 3 for 4 You WIN!
You have 1 wins and 2 losses; win percentage: 33.3333%

Enter y or 1-9 to play; n to leave mode: n

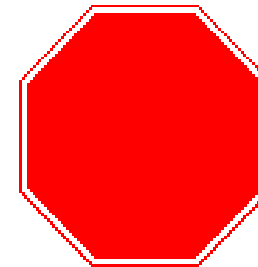
Enter b/B (batch) to simulate many games at once
Enter i/I (interactive) to play 1 to 9 games at a time
Enter q/Q to quit
What next? b

Enter the number of games to simulate: 20000
You have 9821 wins and 10179 losses; win percentage: 49.105%
10/27/2006
```

10/27/2006

CSI 201 -- Top-down Design

19



The End

10/27/2006

CSI 201 -- Top-down Design

20