

Randomization in C++ (p. 128)

- C++ provides a library function `rand()` that generates random numbers
- You need to include the `cstdlib` library
- Example:
 - `y = rand();`
- In reality it's a pseudorandom number!!!

2/27/09

CS250 Introduction to Computer Science II

1

Example

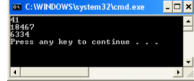
```
#include <iostream>
#include <cstdlib>

using namespace std;

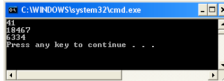
int main()
{
    cout << rand() << endl;
    cout << rand() << endl;
    cout << rand() << endl;

    return 0;
}
```

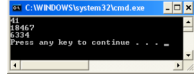
Run 1



Run 2



Run 3



2/27/09

CS250 Introduction to Computer Science II

2

Seeding the Generator

- We need to randomize the results of `rand()`
- To do that, we use `srand()` to seed the random number generator
- Different seed values will result in different random numbers

2/27/09

CS250 Introduction to Computer Science II

3

Example

```
#include <iostream>
#include <cstdlib>

using namespace std;

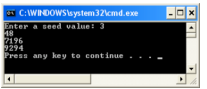
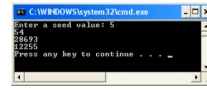
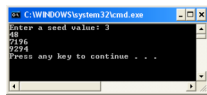
int main()
{
    unsigned seed;

    cout << "Enter a seed value: ";
    cin >> seed;

    srand(seed);

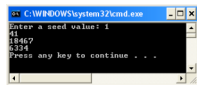
    cout << rand() << endl;
    cout << rand() << endl;
    cout << rand() << endl;

    return 0;
}
```



Example

- What happens if we use a seed value of 1?



- Note that the random numbers generated may be different on your computer

Better Seeding

- What would be a better way of seeding the random number generator?
 - Use the time!
- You must include `ctime`
- Set the seed to `time(0)`
 - `seed = time(0);`
- `time(0)` returns the number of seconds that have elapsed since January 1, 1970

Example

```
#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int main()
{
    unsigned seed;

    seed = time(0);
    srand(seed);

    cout << rand() << endl;
    cout << rand() << endl;
    cout << rand() << endl;

    return 0;
}
```

Specifying a Range

- A lot of the times we would like to specify a range for the random numbers being generated
 - Between 1 and 6 inclusive for faces of a dice for example
- To do that we add one to the random number and % it by the maximum
 - `diceFace = 1 + rand() % 6;`

Random Number Class

- We are now going to write a class whose functions generate and return rand numbers of various types

Random Class

```
#ifndef RANDOM_H
#define RANDOM_H

class Random
{
private:
    int seed;
    int multiplier;
    int addOn;
    bool pseudorandom;
    void reseed();
public:
    Random(bool pseudo = true);
    // returns a random floating point number in the range [0,1)
    double randomReal();
    // returns a random whole number in the range [0,max)
    double randomReal(int max);
    // returns a random whole number in the range [1,max+1)
    int randomInt(int max);
    // returns a random whole number in the range [1,max+1)
    int randomIntZero(int max);

    void setRandom(bool pseudo, int mul, int add)
};

#endif
```

2/27/09

CS250 Introduction to Computer Science II

10

Snake Eyes

- Snake eyes is the outcome of rolling a dice and getting only one pip on each die
- To simulate this game, we will use the Random class that we just created
- We'll also need a Dice class to simulate each die

2/27/09

CS250 Introduction to Computer Science II

11

Dice Class

```
#include "Random.h"

#ifndef DICE_H
#define DICE_H

const int MAX = 6;

class Dice
{
private:
    int face;
    Random random;
public:
    Dice(bool p, int m, int a)
    {
        random.setRandom(p, m, a);
    }
    void rollDice();
    int getFace();
    void displayFace();
};

#endif
```

2/27/09

CS250 Introduction to Computer Science II

12

Snake Eyes

```
int main()
{
    int count = 0;
    Dice firstDice(false, 34645, 234);
    Dice secondDice(false, 35465, 865);

    firstDice.rollDice();
    secondDice.rollDice();

    while(firstDice.getFace() != 1 || secondDice.getFace() != 1)
    {
        firstDice.displayFace();
        cout << " ";
        secondDice.displayFace();
        cout << endl;
        firstDice.rollDice();
        secondDice.rollDice();
        count++;
    }

    firstDice.displayFace();
    cout << " ";
    secondDice.displayFace();
    cout << endl;
    cout << "SNAKE EYES!!!" << endl;
    cout << "It took " << count << " turns" << endl;

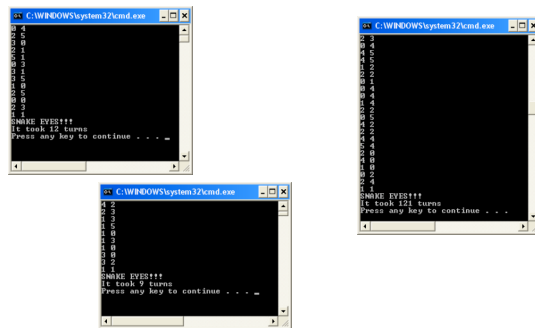
    return 0;
}
```

2/27/09

CS250 Introduction to Computer Science II

13

Snake Eyes Output



2/27/09

CS250 Introduction to Computer Science II

14
