

C Course :: Fall 2008, Lab Session - II

13 September 2008

Problem 1

Problem

Calculate the approximate value of π (pi) using a simple c program.

Strategy

Generate n random points inside a square of dimension 2×2 centered at $(1, 1)$. Now, among the n points, count the number of points which lie inside the circle centered at $(1, 1)$ and having radius 1.

Now the ratio of $\frac{\text{points inside the circle}}{\text{points inside the square}}$ should be roughly equal to the ratio of their area i.e. $\frac{\pi r^2}{x^2}$. Where, r = radius of the circle and x = side of the square.

Use the function: `drand48()` to generate a random number in $[0.0, 1.0)$. You may have to multiply it by 2 to get a number in $[0.0, 2.0)$. There are a number of other functions for generating random numbers, each having a different functionality, e.g, `int rand()`, `long int lrand48()`.

For more on `drand48()`, see the man page (`$ man drand48`).

Take n (no. of points) to be large for better accuracy.

Structure:

```
int main() {  
  
}  
  
int isInsideCircle(double x, double y) {  
  
    // this functions returns 1 if the point (x, y) lie inside  
    // the circle centered at (1, 1), returns 0 otherwise.  
  
}
```

Problem 2

Problem

Write a function in C that concatenates two strings. Let the program have the following structure:

```
int main() {
    char *str1 = "some string"; // say length of str1 is len1
    char *str2 = "other string"; // length of str2 be len2
    char str3[len]; // len = len1 + len2 + 1 (why + 1?)

    // len can also be computed as len = sizeof(str1) + sizeof(str2) + 1

    concatenate(str1, str2, str3);
    printf("concatenated string is %s", str3);
}

void concatenate(char *str1, char* str2, char *result) {

    // this function uses a for/while loop to copy str1 and str2 into

    // result, one after the other.

}
```

Things to note: `sizeof(str1)` returns the number of bytes in the `str1` array. And `sizeof()` doesn't count the `'\0'` at the end of the string while returning its length. So, if `char *str1 = "hello"`, `sizeof(str1)` returns 5