Lecture 5
CS 137
Fall 2014
by Chantelle Gellert

Announcments

- Quiz Thursday September 25th

Function

Define a function called GCD

```
int gcd(int a, int b){
 int temp = 0;
 while(b != 0){
  temp = b;
  b = a % b;
  a = temp;
 }
 return a;
}

#include <stdio.h>

int main(void){
 printf("%d\n, gcd(338,806));
 printf(%d\n", gcd(10,25)+3);
 return 0;
}
```

Function

```
return-type - name (parameter list){
 body; //sequence of statements
return somewhere in here
}
```

```
#include(stdio.h){
function here
}
int main(void){
testing
return 0;
}
```

Leap year

Is 2100 a leap year? int leap(int years) return true or false // return 0 or 1

```
int leap(int year){
 if(year %400 == 0){
        return 1;
 }else if(year %100 == 0){
        return 0;
 }else if(year %4 == 0){
        return 1;
 }else{
        return 0;
 }
}
```

For which years does this work?
-not negative numbers
-not before 1753

Testing

```
void testLeap(int year){
 if(year < 1753){
       printf("%d is before 1753\n", year);
 }else if(leap(year)){
       printf("%d is a leap year\n", year);
 }else{
       printf("%d is not a leap year\n", year);
 }
}
```

```
#include <stdio.h>
//functions here

int main(void){
       testleap(2100);
       testleap(1996);
       testleap(2000);
       testleap(2014);
       testleap(0);
       teatleap(-1);
return 0;
}
```

Is 271 a prime number: A prime number: natural number with exactly two divisors (1 and itself)

```c
#include <stiod.h>


int isPrime(int n)

int main(voi){
	int	number = 0;

	while(1){
		scanf("%d", number);
		if(number <= 0){
			return 0;
		}
		if(isPrime(number)){
			printf("%d is prime\n", number);
		}else{
			printf("%d is not prime\n", number);
		}
	}
 return 0;
}
```

```c
int isPrime(int n){
	int divisor = 2;

	int(n <= 1){
		return 0;
	}

	while(n/divisor >= divisor){
	 if(n%divisor == 0){
		return 0;
	 }
	 divisor ++
	}
 return 0;
}
```

Assertions

```
#include <assert.h>

int leap(int year){
        assert(year > 1753);
        if(year ...)

        }
}

#include <assert.h>
assert(expression);

if True => does nothing;
if False => it aborts the program with an error message
```