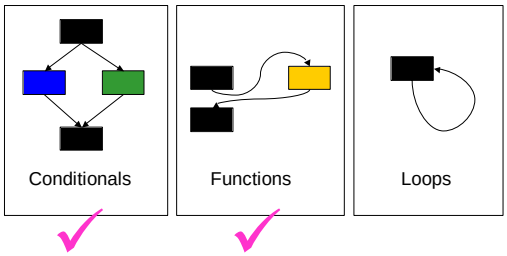
CSE 142
Programming I

Iteration

Control Structures



Why a New Control Structure?

- So far our programs are very limited
  - ➤They can only run "straight through"
  - ➤What if the user enters something invalid?
- Most programs repeat actions
  - ➤Repeating actions is called iteration

Loops

- The control structures we'll see to perform iteration are called loops
- Loops are not the only way to do this—we'll see another way at the end of the course

Where We're Going

- `while` loops
- `for` loops
- `do…while` loops
- Nested loops—what happens when you put one loop inside another

The while loop

```
while (condition) { statements }
```

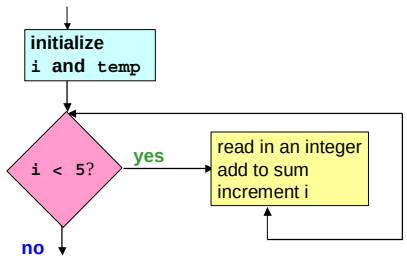
- The while loop will repeatedly execute the statements until the condition is false
  - ➤Something in the body of the while loop must change the condition or the loop will never end!

While Loop Example

- Add five integers:

```
int i = 0;
int temp, sum = 0;
while (i < 5) {
  scanf("%d", &temp);
  sum = sum + temp;
  i++;
}
```


While Loop Control Flow



Computing a Factorial

- 9! = 1 * 2 * 3 * 4 * 5 * 6 * 7 * 8 * 9
- We could write this without loops (yuck!)
- How could we write it with loops?

Computing a Factorial

```
int i = 1;
int product = 1;
while(i <= 9){
   product = product * i;
   i++;
}
```


Factorial Function

```
int fact(int number){
   int i = 1;
   int product = 1;
   while(i <= number){
        product = product * i;
        i++;
   }
   return product;
}
```


Interest Calculation

- How long will it take before your money doubles at 10% interest?

```
money = 1.00;
i = 0;
while (money < 2.00){
     money = money * 1.1;
     i++;
}
```

Handling User Input

- Let's write a loop to compute the average of floating point numbers
- We want the user to be able to enter as many numbers as they wish

Averaging Numbers

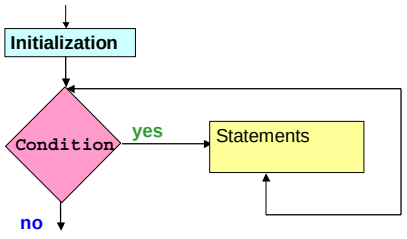
New Loop Syntax

- The for loop is different syntax for the same thing: iteration
- for and while loops can always be interchanged…
  - ➤...but some cases are much better suited to for loops than while loops

For Loop Syntax

```
for ( initialization ;
      condition;
      update ) { statements }
```


For Loop Flowchart



For Loop vs. While Loop

```
i = 0;
while (i<10) {
  printf("%d\n", i);
  i++;
}
```

```
for
(i=0; i<10; i++) {
  printf("%d\n", i);
}
```

Which Loop Should You Choose?

- They're interchangeable, so to some extent it's personal preference
- Use the for loop if you know how many times you're going to do something
- Think which construct will yield the simplest, clearest code

Primality Testing

- A number is prime if it is only divisible by 1 and itself
- How do we tell if one number divides another evenly?
- How would we use a loop to test if a number were prime?

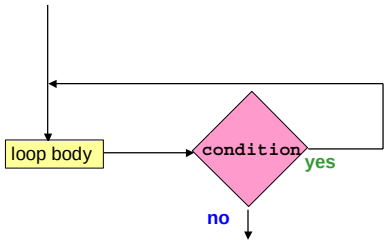
Primality Testing

```
int isPrime(int number){




}
```


One More Bit 'O Syntax

- The do…while loop is like a while loop, but it does the test at the end
- This can be very useful for doing things like checking user input

Do...While Loop Control Flow



Do…While Example

- Checking user input:

```
do {
   printf("Enter your age: ");
   scanf("%d", &age);
} while (age < 0);
```

Constructing Single Loops

- Should the program loop…
  - ➤…a specified number of times
  - ➤…until something happens
- What is my
  - ➤initialization?
  - ➤test?
  - ➤update?

Nested Loops

- When you put one loop inside another then you have nested loops
- Nested loops can be much more confusing than single loops
- Some loops are more “tightly coupled” than others—this makes them trickier!

“Loosely Coupled” Loop

```
do {
  printf("Enter a number: ");
  scanf("%d", &foo);
  for (i=0;i<foo;i++)
     printf("%d\n", i);
  printf("Play again? ");
  scanf("%c%c", &trash, &cont);
} while (cont == 'y');
```


“Tightly Coupled” Loop

```
for (i=1; i<=10; i++){
   for (j=1; j<=i; j++){
       printf("*");
   }
   printf("\n");
}
```


Multiplication Table

- How would we write a program to print a multiplication table:

```
 1   2   3   4   5   6   7   8   9  10
 2   4   6   8  10  12  14  16  18  20
                . . .
 9  18  27  36  45  54  63  72  81  90
10  20  30  40  50  60  70  80  90 100
```


Multiplication Table

"Horribly Intertwined Loop of Death" Loop

- What if the inner loop affects the outer loop?
  - ➤Yuck

A Contrived BAD Example

```
for (cat=0; cat<15; cat=cat+5){
  for (dog=0; dog<cat; dog++){
      cat = cat - dog/5;
      printf("meow!\n");
  }
}
```


2-D Figure Number 1

- How would we print the following:

```
*
**
***
****
*****
```


2-D Figure Number 1

2-D Figure Number 2

- How would we print the following:

```
*****
****
***
**
*
```


2-D Figure Number 2

How Does One Develop a Loop?

- Look for familiar patterns
- Look for familiar patterns
- Look for familiar patterns