Announcements

• New HW assignment due Monday January 23rd at 11:59 pm
  • Has extra credit
  • We will add extra credit in at the end of the semester after assigning grades (so it won’t hurt you not to do it)
• TA oral reviews of programming assignments
  • On half of the program assignments you will have to explain how it works to the TA
  • Worth 50% of the grade on those assignments
  • Will announce which assignments after a given assignment is due, but before the Lab session in which you have to explain how the assignment works
• Midterms are February 6 and March 6
Lecture 4: Overview

- Review of Conditional statements
- Loops
Boolean Expressions

- Actually integers in C
  - 0 = false
  - Other integers = true
- Typically use comparison operations
  - For example:
    - a > b – true if a > b
    - a >= b – true if a >= b
    - a < b – true if a < b
    - a <= b – true if a <= b
    - a == b – true if a equals b
    - a != b – true if a isn’t equal to b
- CAVEAT: BE WARY OF a = b
  - This assigns a to be the value of b and then returns this value
  - Common typo/mistake
More Boolean Expressions

• Negation
  • !a – true if a is false

• OR
  • a||b – true if a or b is true

• AND
  • a&&b – true if a and b is true
Example Conditional Statement

```c
int x;
scanf("%d", &x);
if (x < 0) {
    printf("x is negative\n");
}
```

- If `(x < 0)` is true the `printf` statement is executed
- Otherwise the code isn’t executed
int x;
scanf("%d", &x);
if (x<0) {
    printf("x is negative\n");
} else {
    printf("x is non-negative\n");
}

• If (x<0) is true the printf statement is executed
• Otherwise the else statement is executed
int x;
scanf("%d", &x);
if (x == 1) {
    printf("x is one\n");
} else if (x == 2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
• Can chain together else statements
int x;
scanf("%d", &x);   x=1
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
Else Chains

```c
int x;
scanf("%d", &x);  x = 1
if (x == 1) {
    printf("x is one\n");
} else if (x == 2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
```
int x;
scanf("%d", &x);  x=1
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d",&x);  \textcolor{red}{x=1}
if (x==1) {
  printf("x is one\n");
} else if (x==2) {
  printf("x is two\n");
} else {
  printf("x is not one or two\n");
}
int x;
scanf("%d", &x);  
x=2
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d",&x); x=2
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d", &x);  x=2
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d", &x); x = 2
if (x == 1)
{
    printf("x is one\n");
}
else if (x == 2)
{
    printf("x is two\n");
}
else
{
    printf("x is not one or two\n");
}
int x;
scanf("%d",\&x);
x=2
if (x==1) {
        printf("x is one\n");
} else if (x==2) {
        printf("x is two\n");
} else {
        printf("x is not one or two\n");
}
int x;
scanf("%d",&x); x=3
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d", &x);    // x = 3
if (x == 1) {
    printf("x is one\n");
} else if (x == 2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
Else Chains

int x;
scanf("%d", &x);  
if (x == 1) {
    printf("x is one\n");
} else if (x == 2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d", &x); x = 3
if (x == 1) {
    printf("x is one\n");
} else if (x == 2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d", &x); x = 3
if (x == 1) {
    printf("x is one\n");
} else if (x == 2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
int x;
scanf("%d", &x);
x=3
if (x==1) {
    printf("x is one\n");
} else if (x==2) {
    printf("x is two\n");
} else {
    printf("x is not one or two\n");
}
What if we want to do something multiple times?

• Suppose we want to sum all of the numbers from 1 to 100
• We can use a loop to repeat an operation
• Form:
  
  ```
  while(test condition) {
    statement to repeat;
  }
  ```
Loops

• Example to sum numbers from 1 to 100:

```c
int sum=0;
int count=1;
while(count<=100) {
    /* add the number to the sum */
    sum=sum+count;
    /* increment the number by 1 */
    count=count+1;
}
printf("Sum is %d\n.",sum);
```
For Loops

• Consider the following loop:
  ```
  int x=0; /* initial value for x */
  while(x<100) {
    /* test for x */
    do something;
    x=x+1; /*increment x */
  }
```
For Loops

- Consider the following loop:
  ```
  int x = 0; /* initial value for x */
  while(x < 100) {
      /* test for x */
      do something;
      x = x + 1; /* increment x */
  }
  ```
- Common pattern – there is short hand
  ```
  int x;
  for(x = 0 /* initial value */;
      x < 100 /* test to continue loop */;
      x = x + 1 /* executed after the body */) {
    do something;
  }
  ```
For Loops

- Example:
  ```c
  int x;
  for(x=0; x<100; x=x+1) {
    printf("%d\n",x);
  }
  ```
What if we want to exit early?

- Can use the break statement
- Example:
  ```c
  int x;
  for(x=1;x<=10;x=x+1) {
      if (x==2)
          break;
      printf("%d\n",x);
  }
  ```
What if we skip to the next iteration of the loop?

- Can use the `continue` statement
- Example:
  ```c
  int x;
  for(x=1;x<=10;x=x+1) {
      if (x==2)
          continue;
      printf("%d\n",x);
  }
  ```
Shorthands

- Often write $x = x + 1$ in loops
- Can abbreviate $x = x + 1$ as $x++$
- $x = x - 1$ as $x--$
- $x = x + a$ as $x += a$
- $x = x - a$ as $x -= a$
- $x = x \times a$ as $x *= a$
- $x = x / a$ as $x /= a$
Note about increment/decrement operators

• What does y=x++ do?
• Same as:
  y=x;
  x=x+1;
• We can also write y=++x
• Same as:
  x=x+1;
  y=x;
• Can do the same with --
Problems

• Compute the first 100 digits of the Fibonacci sequence
  1,1,2,3,5,8...
Problems

- Print all the prime numbers between 10 and 100
Problems

• Add a list of user provided numbers where -1 denotes the end of the list