

# February 6

---

- **Test next time!**
  - Everything we've studied so far
  - Easy to grade
  - Bring paper, pencils
  - Closed book, notes, etc.
  - No synthesis of code.
  - Vocabulary
  - Performance
  - Analysis of code
  - Something you haven't thought about...
  - Be here **ON TIME!**
- Questions?

# Making code work!

---

- Today's example is available online.
- No, you can't use a C compiler to write your code for you...
- Yes, you should write algorithms in C first
  - Then HAND translate...

# C to Assembler

---

```
char S1[] = "This is a string."  
int Foo[16];  
char S2[100];
```

```
void main()  
{  
    strcpy(S2, S1);  
}
```

```
void strcpy(char* dst, char* src)  
{  
    int i = 0;  
    while((dst[i] = src[i]) != 0)  
        i = i+1;  
}
```

# Data Segment

---

```
#char S1[] = "This is a string."
```

```
#int Foo[16];
```

```
#char S2[100];
```

```
.data
```

```
S1: .ascii "This is a string."
```

```
.space 1 # NULL at the end
```

```
.align 8
```

```
Foo: .space 16 # some junk
```

```
S2: .space 100 # desitination string
```

# Text Segment (main)

---

```
#void main() { strcpy(S2, S1); }
```

```
.text
```

```
.globl main
```

```
main:
```

```
subu $sp, $sp, 4    # get space on the stack
sw   $ra, 0($sp)    # save main's return address
la   $a0, S2        # address of S2 in the first argument
la   $a1, S1        # address of S1 in the second argument
jal  strcpy         # call strcpy
lw   $ra, 0($sp)    # restore main's return address
addi $sp, $sp, 4    # restore the stack pointer
jr   $ra            # exit main
```

# strcpy

---

```
#void strcpy(char* dst, char* src) {  
#   int i = 0;  
#   while((dst[i] = src[i]) != 0)  
#       i = i+1;  
# }
```

strcpy:

```
    move $t0,$zero        # i in $t0 = 0
```

L1:

```
    add  $t1,$a1,$t0      # address of src[i] in $t1  
    lb   $t2, 0($t1)      # t2 = src[i]  
    add  $t1,$a0,$t0      # address of dst[i] in $t1  
    sb   $t2, 0($t1)      # dst[i] = $t2  
    addi $t0, $t0, 1      # i = i+1  
    bne  $t2,$zero,L1     # if dst[i] != 0 repeat  
  
    jr   $ra              # return
```