

# Computer Systems Fundamentals

[odd\\_even.c](#)

read a number and print whther its odd or even

```
#include <stdio.h>

int main(void)_{
    int x;

    printf("Enter a number: ");
    scanf("%d", &x);

    if ((x & 1) == 0){
        printf("Even\n");
    } else {
        printf("Odd\n");
    }

    return 0;
}
```

[odd\\_even.simple.c](#)

```
#include <stdio.h>

int main(void)_{
    int x, v0;

    printf("Enter a number: ");
    scanf("%d", &x);

    v0 = x & 1;
    if (v0 == 1) goto odd;
    printf("Even\n");
    goto end;
odd:
    printf("Odd\n");
end:
    return 0;
}
```

[odd\\_even.s](#)

read a number and print whther its odd or even

```

main:
    la    $a0, string0    # printf("Enter a number: ");
    li    $v0, 4
    syscall

    li    $v0, 5           # scanf("%d", &x);
    syscall

    and    $t0, $v0, 1     # if (x & 1 == 0) {
    beq    $t0, 1, odd

    la    $a0, string1    # printf("Even\n");
    li    $v0, 4
    syscall

    b     end

odd:
    # else
    la    $a0, string2    # printf("Odd\n");
    li    $v0, 4
    syscall

end:
    li    $v0, 0           # return 0
    jr    $ra

.data
string0:
    .asciiz "Enter a number: "
string1:
    .asciiz "Even\n"
string2:
    .asciiz "Odd\n"

```

### [print10.c](#)

print integers 1..10 one per line

```

#include <stdio.h>

int main(void) {
    for (int i = 1; i <= 10; i++) {
        printf("%d\n", i);
    }
    return 0;
}

```

### [print10.simple.c](#)

```

#include <stdio.h>

int main(void) {
    int i;
    i = 1;
loop:
    if (i > 10) goto end;
    i++;
    printf("%d", i);
    printf("\n");
    goto loop;
end:
    return 0;
}

```

### [print10.s](#)

print integers 1..10 one per line

```

main:                # int main(void)_{
                    # int i; // in register $t0

    li    $t0, 1      # i = 1;

loop:                # loop:
    bgt   $t0, 10 end  # if (i > 10) goto end;

    move  $a0, $t0     # printf("%d" i);
    li    $v0, 1
    syscall

    li    $a0, '\n'    # printf("%c", '\n');
    li    $v0, 11
    syscall

    add   $t0, $t0 1   # i++;

    b     loop         # goto loop;

end:
    li    $v0, 0       # return 0
    jr    $ra

```

### [sum 100 squares.c](#)

calculate  $1*1 + 2*2 + \dots + 99*99 + 100*100$

```

#include <stdio.h>

int main(void){
    int sum = 0;

    for (int i = 0; i <= 100; i++){
        sum += i * i;
    }

    printf("%d\n", sum);

    return 0;
}

```

### [sum 100 squares.simple.c](#)

```

#include <stdio.h>

// sum of first 100 squares.

int main(void){
    int i, sum, t3;

    sum = 0;
    i = 0;
loop:
    if (i > 100) goto end;
    t3 = i * i;
    sum = sum + t3;
    i = i + 1;
    goto loop;

end:
    printf("%d", sum);
    printf("\n");

    return 0;
}

```

### [sum 100 squares.s](#)

calculate  $1*1 + 2*2 + \dots + 99*99 + 100*100$

sum in \$t0, i in \$t1

```
main:
    li $t0, 0          # sum = 0;
    li $t1, 0          # i = 0

loop:
    bgt $t1, 100 end    # if (i > 100) goto end;
    mul $t3, $t1, $t1    # t3 = i * i;
    add $t0, $t0, $t3    # sum = sum + t3;

    add $t1, $t1, 1      # i = i + 1;
    b loop

end:
    move $a0, $t0        # printf("%d", sum);
    li $v0, 1
    syscall

    li $a0, '\n'        # printf("%c", '\n');
    li $v0, 11
    syscall

    li $v0, 0            # return 0
    jr $ra
```

**COMP1521 20T2: Computer Systems Fundamentals** is brought to you by  
the [School of Computer Science and Engineering](#)  
at the [University of New South Wales](#), Sydney.  
For all enquiries, please email the class account at [cs1521@cse.unsw.edu.au](mailto:cs1521@cse.unsw.edu.au)

CRICOS Provider 00098G