

CS136 EXERCISE SOLUTIONS: SECTION 3

TRACING C

```
int factorial(int n) {
    int product = 1;
    while (n > 0) {
        product = product * n;
        n = n - 1;
    }
    return product;
}
```

```
int sum_squares(int n) {
    int sum = 0;
    while (n > 0) {
        sum = sum + n * n;
        n = n - 1;
    }
    return sum;
}
```

```
int main(void) {
    int n = 10;
    int total = 0;
    while (n > 0) {
        --n;
        if (!(n % 3)) {
            printf("Skipping %d\n", n);
        } else {
            printf("Including %d\n", n);
        }
    }
    printf("%d\n", total);
}
```

FOR LOOPS

```
int main(void) {
    int i = 10;
    while (i > 0) {
        printf("%d\n", i);
        --i;
    }
}
```

```

int main(void) {
    int i = 10;
    while (i > 0) {
        printf("%d\n", i);
        if (0 == i % 2) {
            --i; // if we simply tranform it, when we use continue skip
                // right over the update_statement.
                // So here we duplicate the update_statement.
            continue;
        }
        i--;
    }
}

```

READING AND WRITING CHAR VALUES

```

int main(void) {
    int count = 0;
    while (read_char(READ_WHITESPACE) != READ_CHAR_FAIL) {
        count++;
    }
    printf("%d\n", count);
}

```

To ignore whitespace, simply replace READ_WHITESPACE with IGNORE_WHITESPACE.

CHAR VALUES

```

int main(void) {
    char c = '\0';
    while (1 == scanf("%c", &c)) {
        printf("%d\n", c);
    }
}

```

STRUCTURES

```

struct posn add_posn(struct posn a, struct posn b) {
    a.x += b.x; // We are passed a struct;
    a.y += b.y; // we modify this local value,
    return a; // then return it.
}

```

```

bool posn_eq(struct posn a, struct posn b) {
    return a.x == b.x && a.y == b.y;
}

```