

Nicole Luong

# Week 3

While loops, Structs, Enums, Variable Names

The background features several thick, hand-drawn teal lines. One line starts from the left edge, curves downwards, and then turns right. Another line starts from the top left, goes right, then down, then right again, and finally down. A third line starts from the top left, goes right, then down, then right again, and finally down. These lines intersect to form a series of open, irregular shapes.

**Before we  
begin**

Help Sessions

Lab check-ins

Nicole Luong

# While Loops

**A**

```
#include <stdio.h>

int main(void) {
    int i = 0;
    while (i < 32) {
        printf("%d\n", i);
        i = i + 2;
    }
    return 0;
}
```

**B**

```
#include <stdio.h>

int main(void) {
    int i = 5;
    while (i >= 0) {
        printf("%d\n", i);
        i--;
    }
    return 0;
}
```

**C**

```
#include <stdio.h>

int main(void) {
    int i = 0;
    int keep_going = 1;
    while (keep_going == 1) {
        if (i > 3) {
            keep_going = 0;
        }
        i++;
    }
    printf("%d\n", i);
    return 0;
}
```

**D**

```
#include <stdio.h>

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

**E**

```
#include <stdio.h>

int main(void) {
    int i = 0;
    int max = 32;
    while (i < max) {
        printf("%d\n", i);
        max = max + 2;
    }
    return 0;
}
```

**F**

```
#include <stdio.h>

int main(void) {
    int i = 0;
    int keep_going = 0;
    while (keep_going == 1) {
        if (i > 3) {
            keep_going = 0;
        }
        i++;
    }
    printf("%d\n", i);
    return 0;
}
```

Nicole Luong

# 2D While loops

# 2D While loops

Match the patterns to the code snippets

A

```
OXXX
XOXX
XXOX
XXXO
```

B

```
OXOX
OXOX
OXOX
OXOX
```

C

```
OXOO
XXXX
OXOO
OXOO
```

D

```
XXXX
XOOX
XOOX
XXXX
```

1

```
#include <stdio.h>

#define SIZE 4

int main(void) {
    int row = 0;
    while (row < SIZE) {
        int col = 0;
        while (col < SIZE) {
            if (col != 1 && row != 1) {
                printf("O");
            } else {
                printf("X");
            }

            col++;
        }
        row++;
        printf("\n");
    }
    return 0;
}
```

2

```
#include <stdio.h>

#define SIZE 4

int main(void) {
    int row = 0;
    while (row < SIZE) {
        int col = 0;
        while (col < SIZE) {
            if (row == col) {
                printf("O");
            } else {
                printf("X");
            }

            col++;
        }
        row++;
        printf("\n");
    }
    return 0;
}
```

3

```
#include <stdio.h>

#define SIZE 4

int main(void) {
    int row = 0;
    while (row < SIZE) {
        printf("X");
        int col = 1;
        while (col < SIZE - 1) {
            if (row == 0 || row == SIZE - 1) {
                printf("X");
            } else {
                printf("O");
            }

            col++;
        }
        printf("X");
        row++;
        printf("\n");
    }
    return 0;
}
```

4

```
#include <stdio.h>

#define SIZE 4

int main(void) {
    int row = 0;
    while (row < SIZE) {
        int col = 0;
        while (col < SIZE) {
            if (col % 2 == 0) {
                printf("O");
            } else {
                printf("X");
            }

            col++;
        }
        row++;
        printf("\n");
    }
    return 0;
}
```

# Scanning in Until Ctrl-d

# While Loops

Write the loop conditions for each program

A

Enter a series of integers until you reach a negative number. Then, print out the sum of the positive integers.

C

Scan for numbers until end of input. As each number is scanned in, print the even ones.

B

Enter characters until the user presses 'q'. Then, display the count of characters entered.

D

Scan for integers keeping a cumulative sum, until the sum of entered integers reaches or exceeds the target sum provided by the user. Print the final sum.



Nicole Luong

# Structs and Enums

A decorative graphic consisting of several thick, light pink lines that intersect and loop, creating a stylized, abstract pattern on the left side of the slide.

# Lab Time!

Nicole Luong