This is CS50.

cs50.brianyu.me

Week 1

- C
- Compiling
- Strings
- Variables
- Types
- Loops
- Conditions
- Imprecision
- Overflow

What questions do you have?

Today

Variables and Types

Loops and Conditions

Functions

Part One

Variables and Types

```c
#include <stdio.h>

int main(void)
{
    printf("hello, world\n");
}
```

Compiling

```
#include <stdio.h>

int main(void)
{
    printf("hello, world\n");
}
```

hello.c

```
01111111 01000101 01001100
01000110 00000010 00000001
00000001 00000000 00000000
00000000 00000000 00000000
00000000 00000000 00000000
00000000 00000010 00000000
00111110 00000000 00000001
00000000 00000000 00000000
10110000 00000101 01000000
00000000 00000000 00000000
...
```

hello

Types

- `bool`
- `char`
- `double`
- `float`
- `int`
- `long`
- `string`
- `...`

Variables

```
int x = 28;
```

Variables

```
int x = 28;
```

variable
name

Variables

```
int x = 28;
```

type

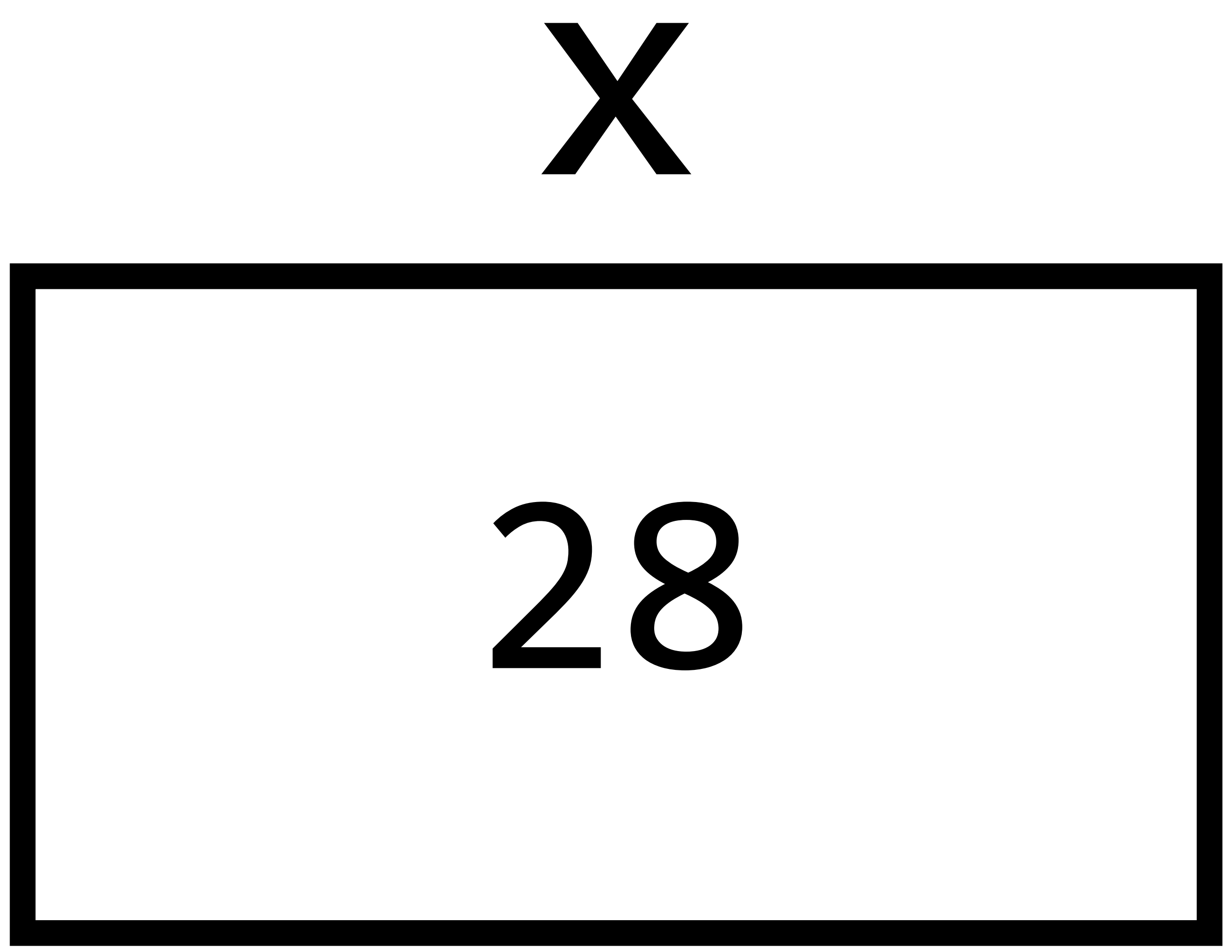
Variables

```
int x = 28;
```

value

Variables

```
int x = 28;
```

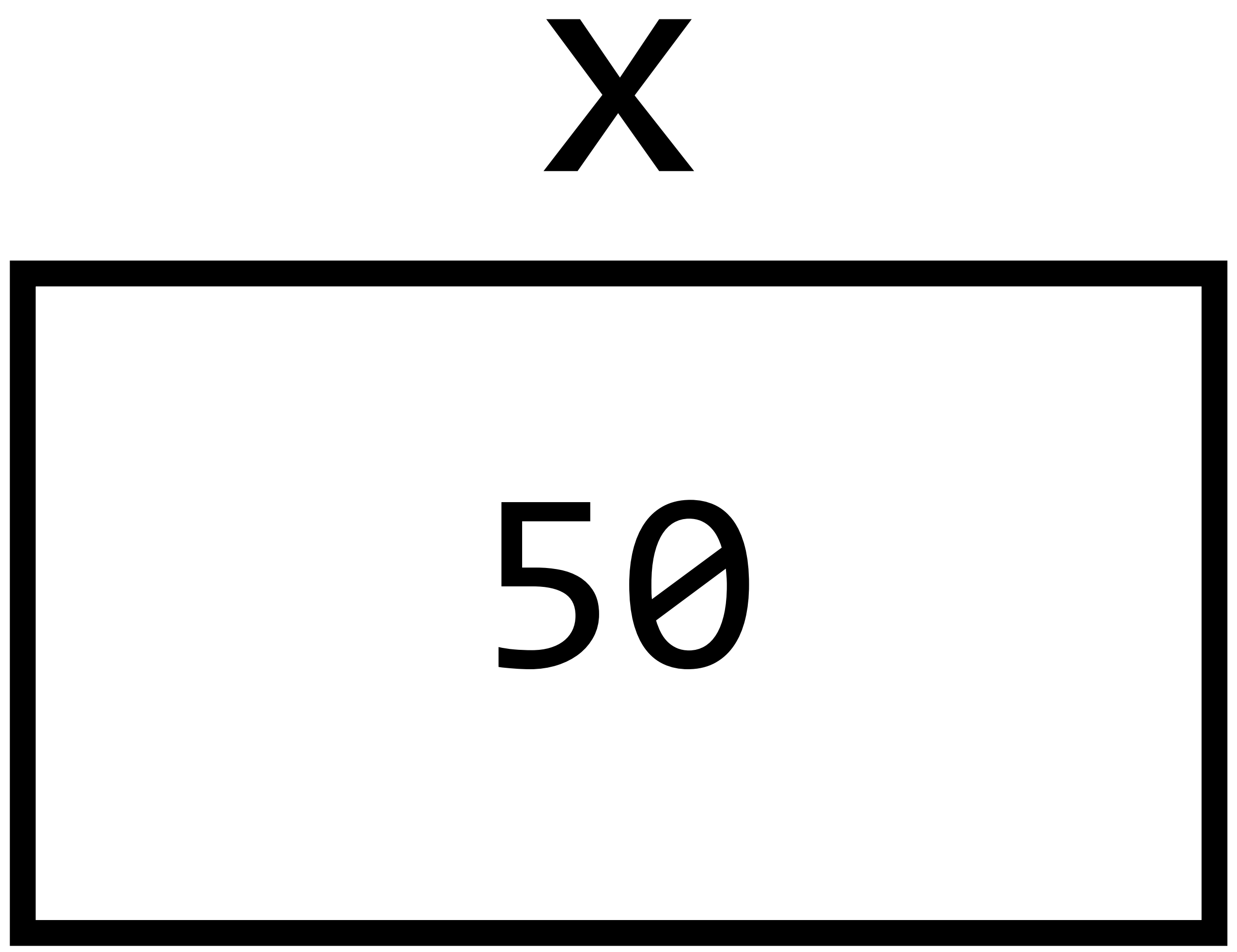


x

28

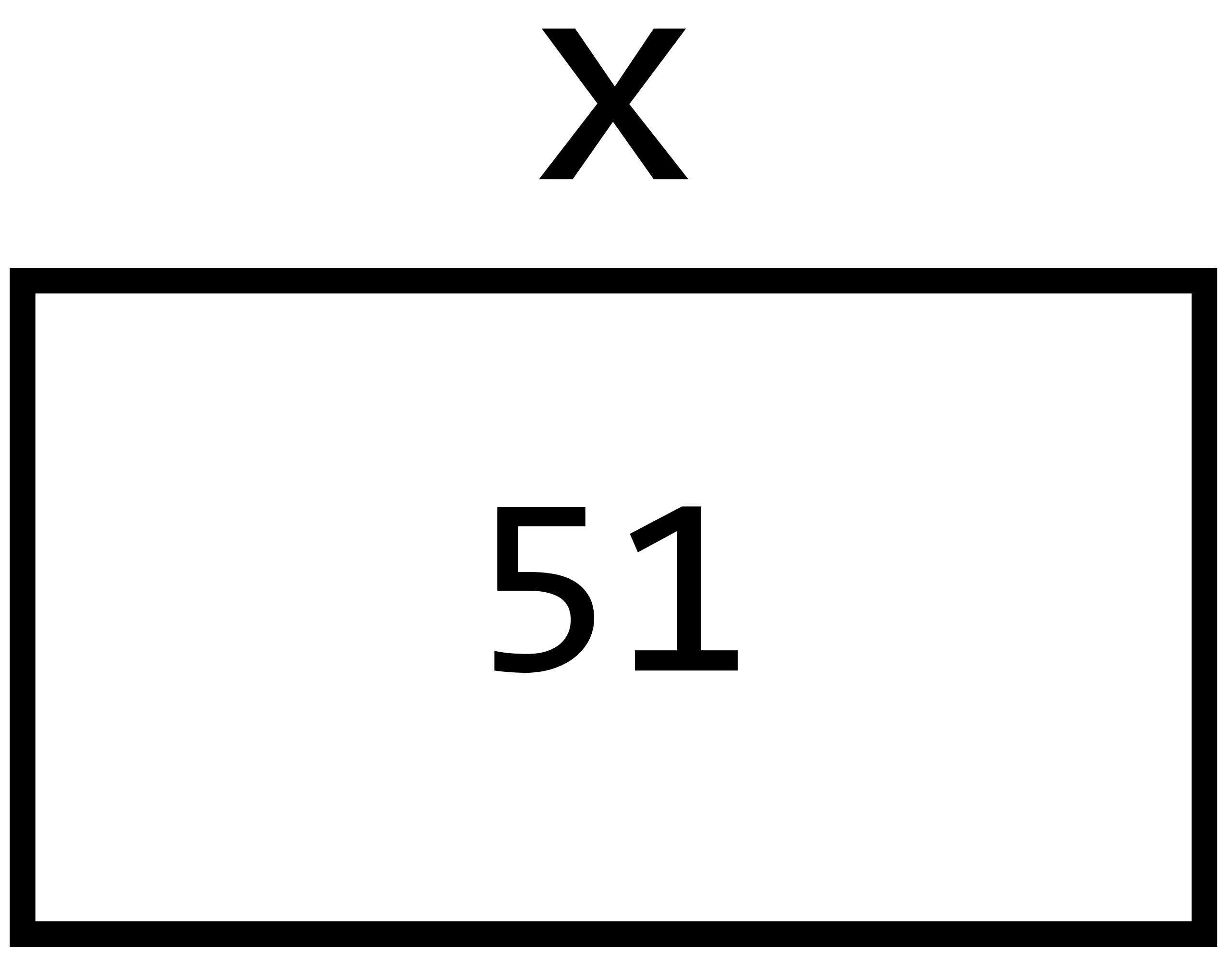
Variables

```
int x = 28;
x = 50;
```



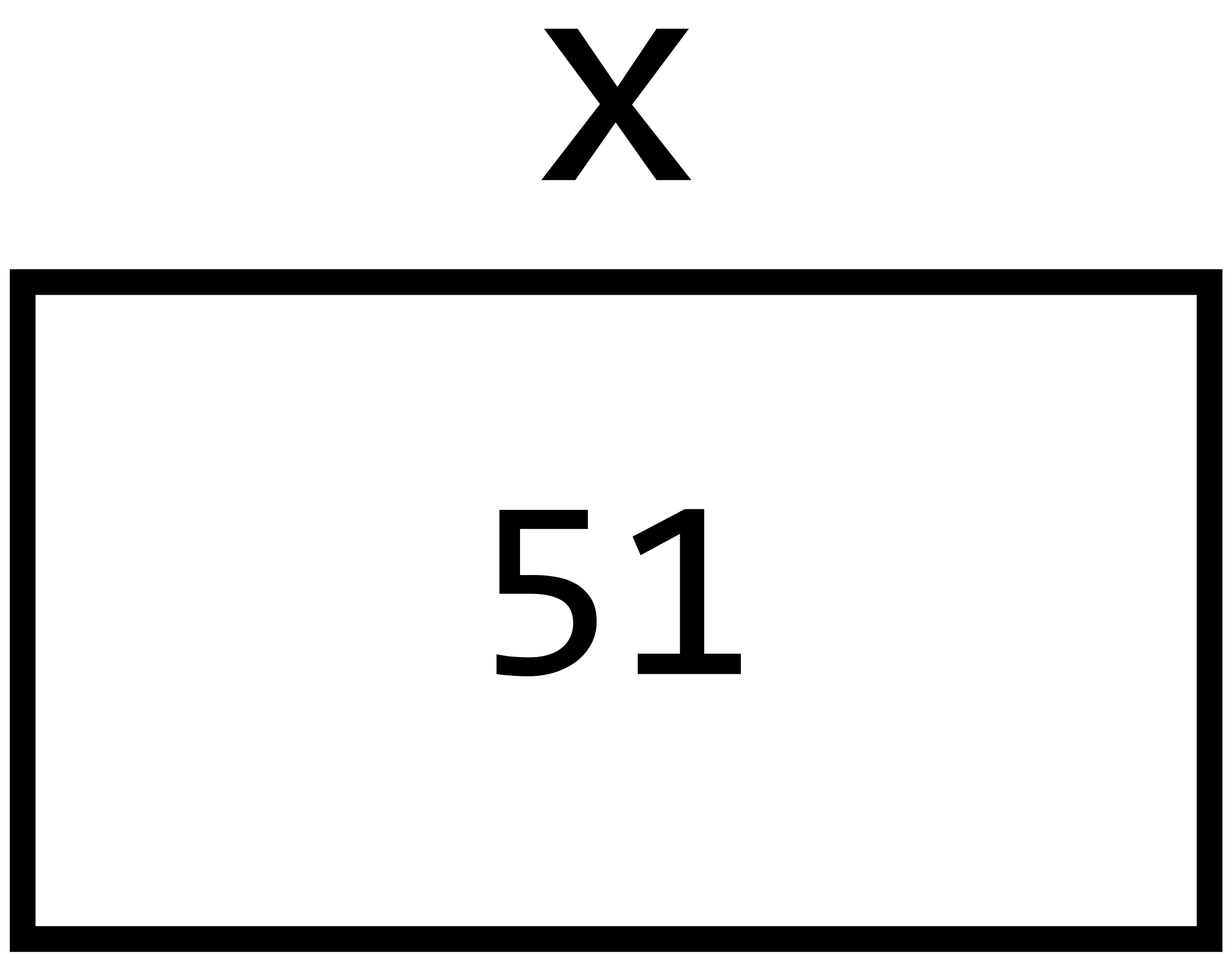
Variables

```
int x = 28;
x = 50;
x = x + 1;
```



Variables

```
int x = 28;
x = 50;
x += 1;
```

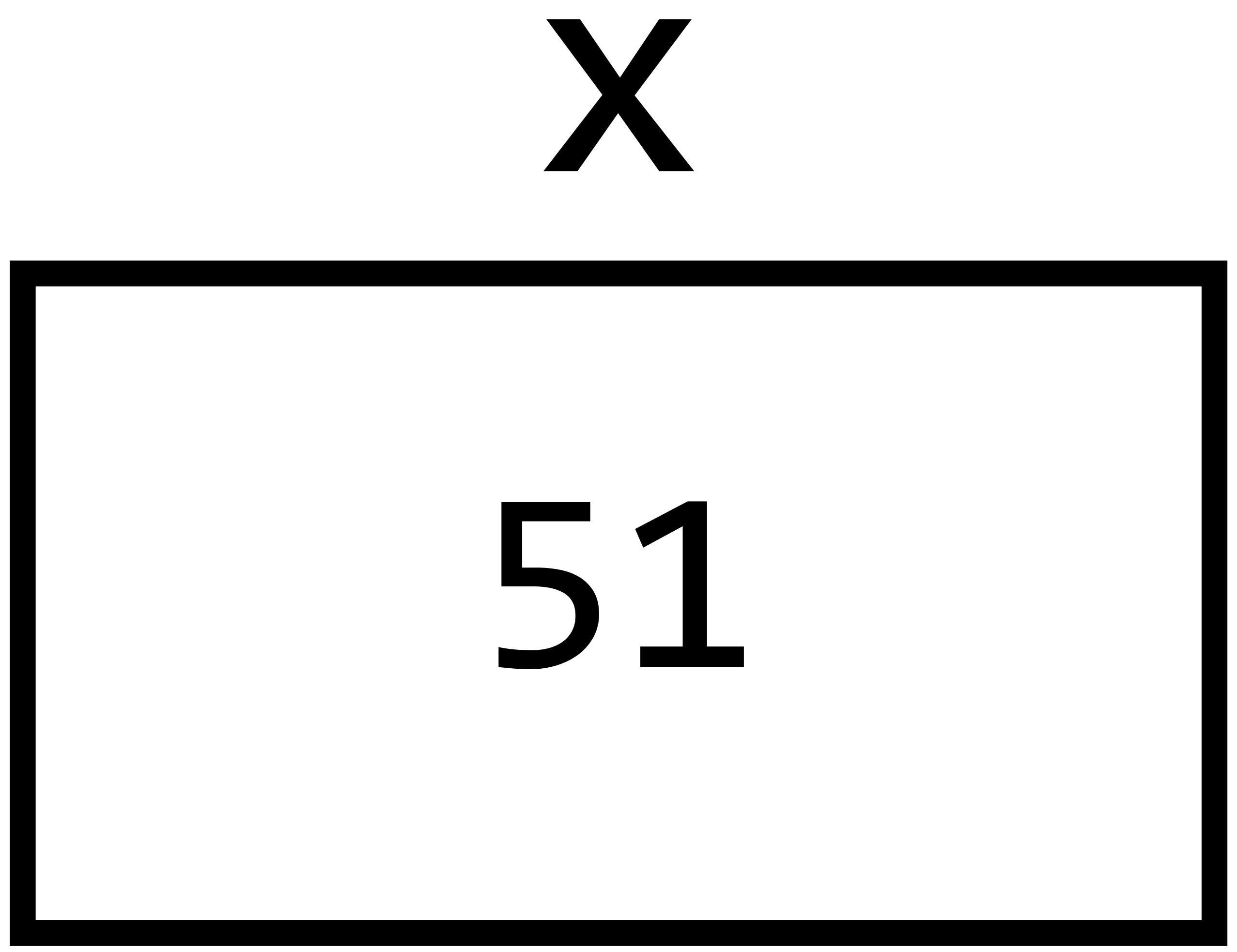


x

51

Variables

```
int x = 28;
x = 50;
x++;
```



x

51

Getting Input

```
int x = get_int("Number: ");
```

Printing Values

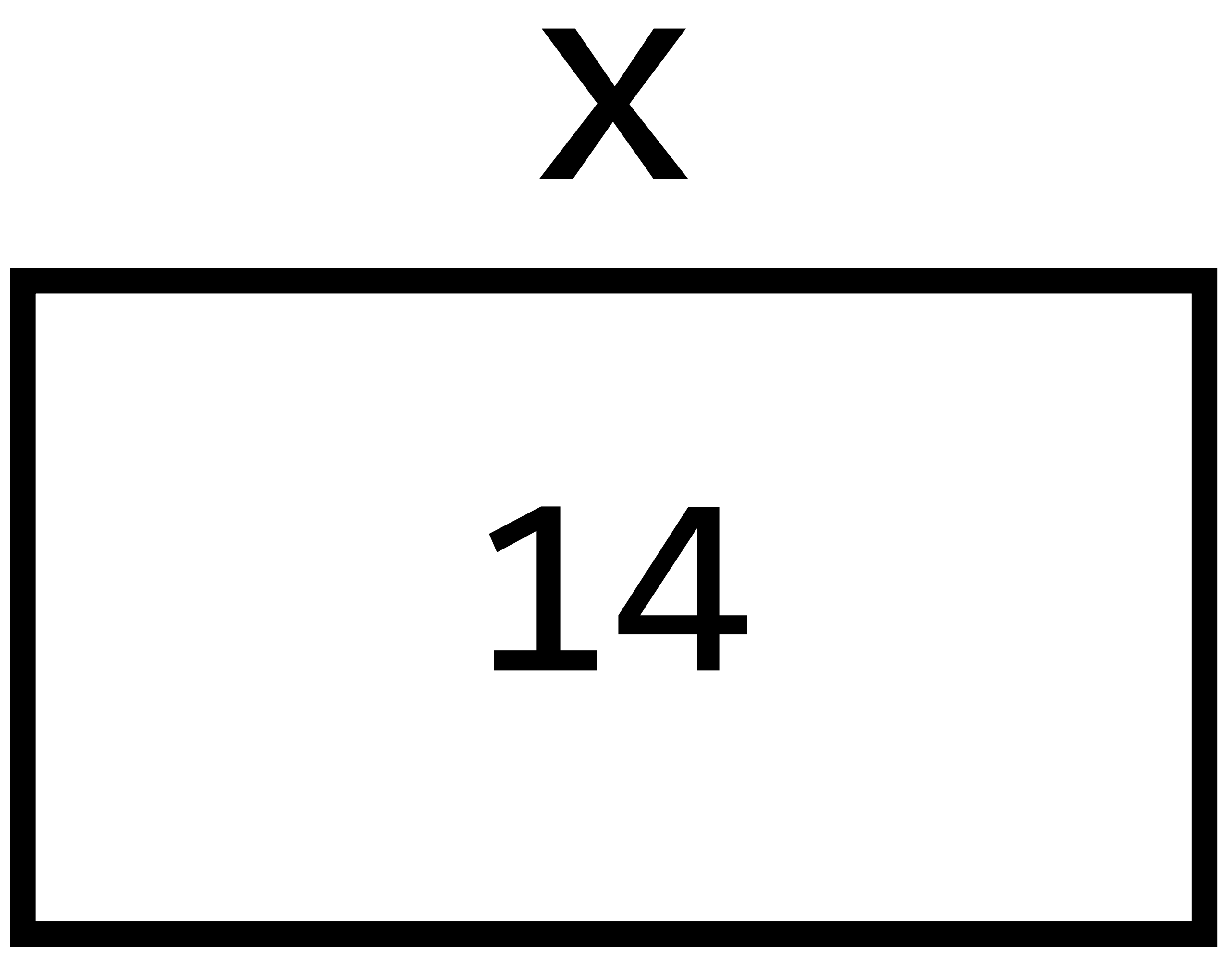
```
printf("Hello, world!");
```

Printing Values

```
int x = 50;
printf("The value is %i", x);
```

Operators

```
int x = 10 + 4;
```

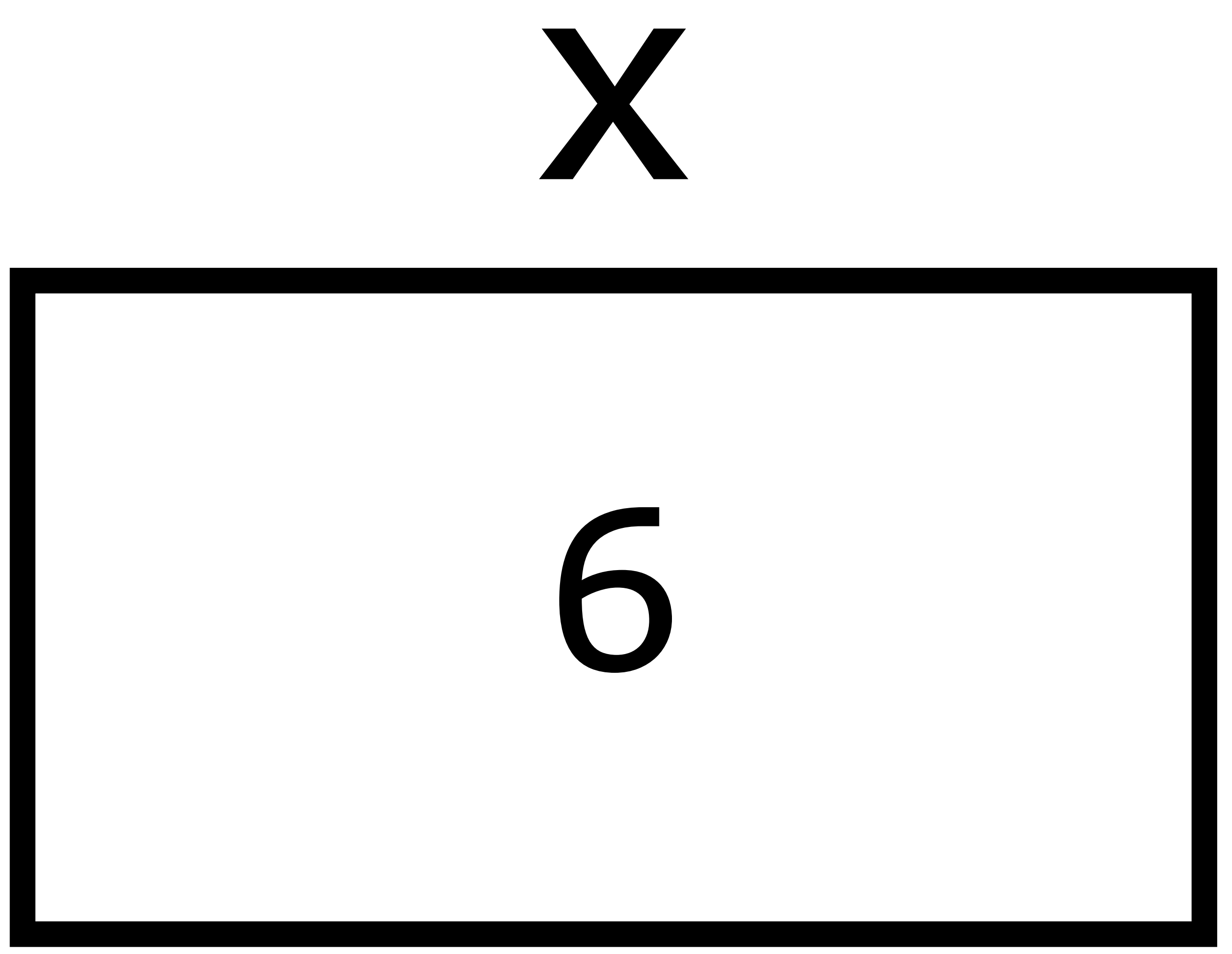


x

14

Operators

```
int x = 10 - 4;
```

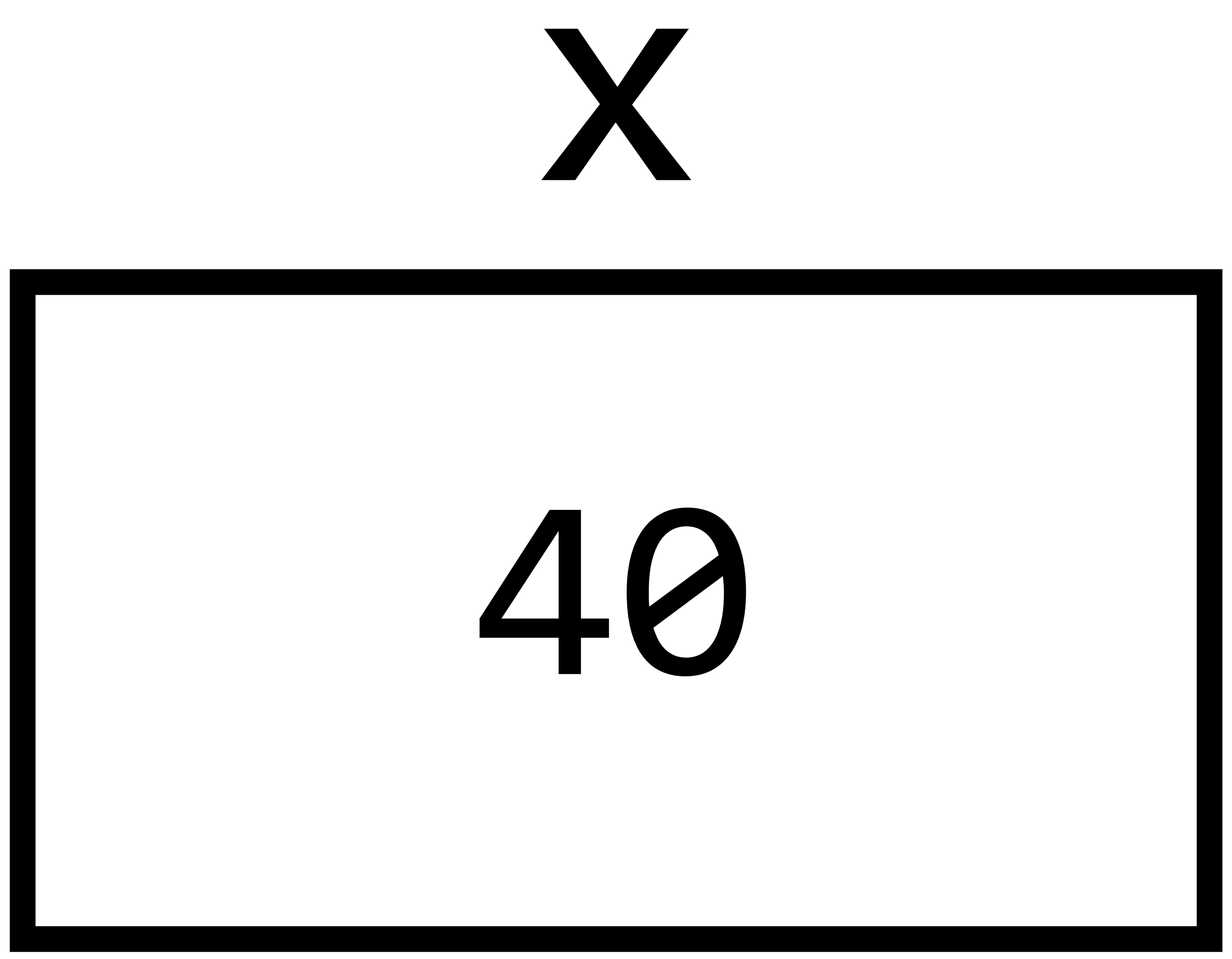


x

6

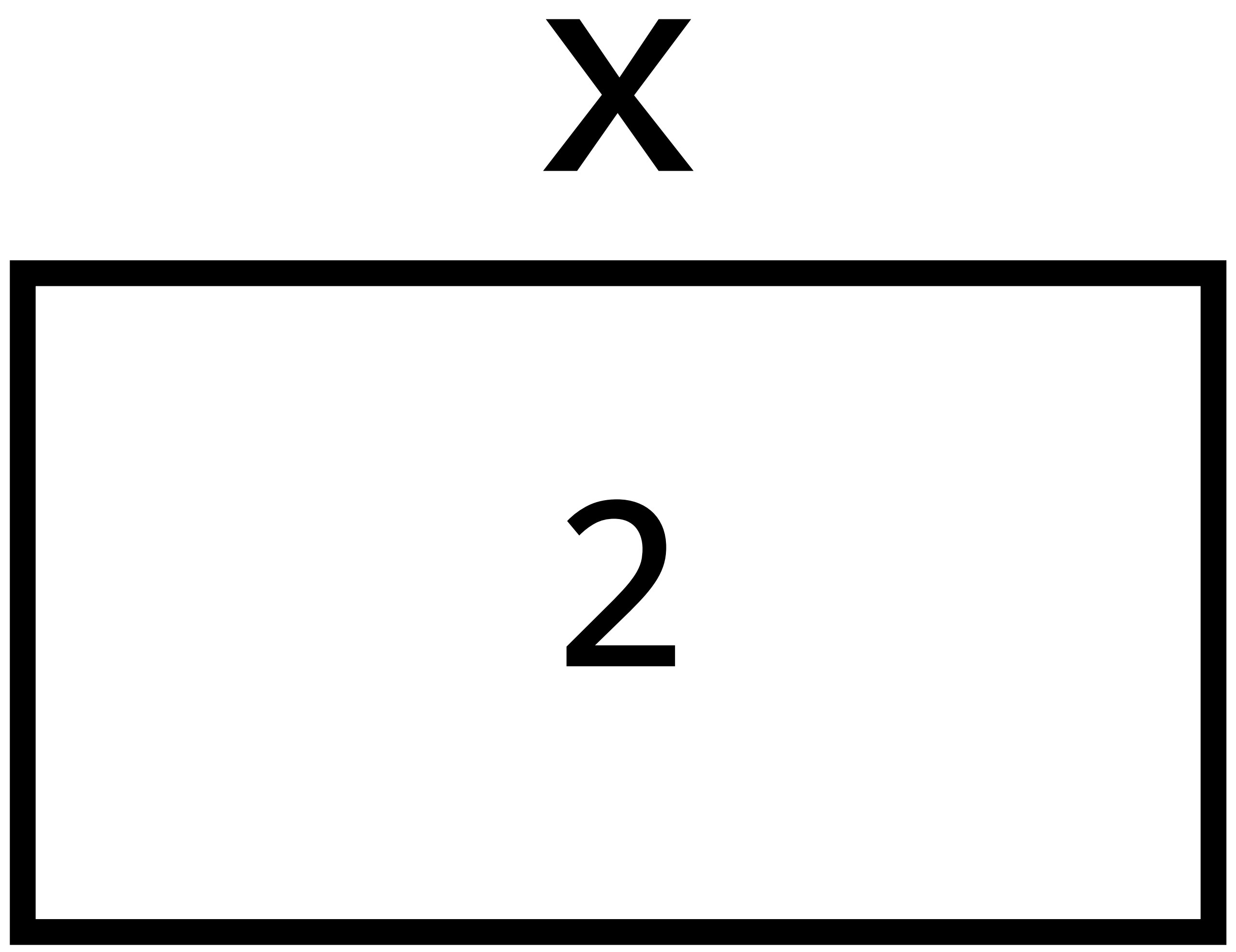
Operators

```
int x = 10 * 4;
```



Operators

```
int x = 10 / 4;
```

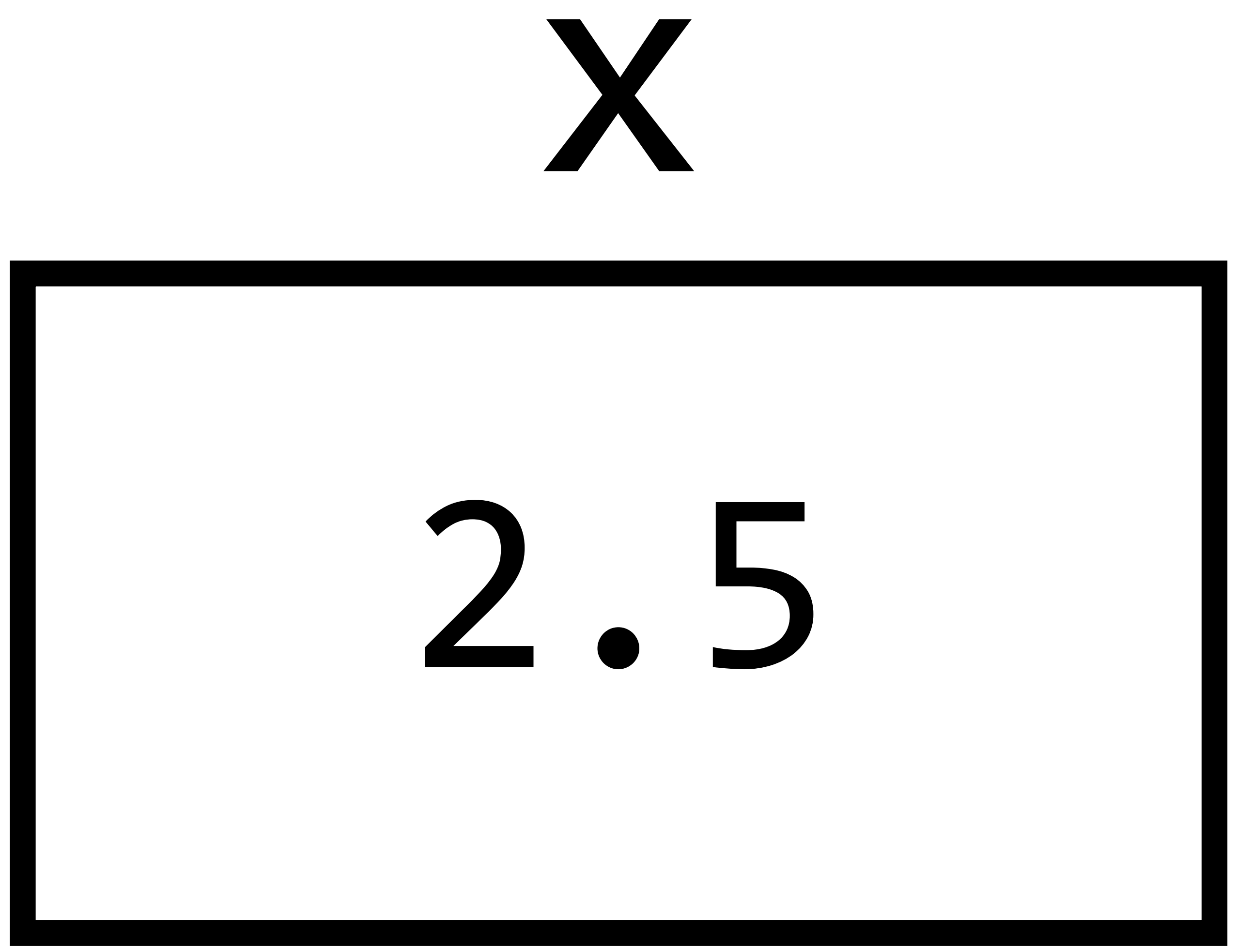


x

2

Operators

```
float x = 10.0 / 4.0;
```



x

2.5

Printing Values

```
string name = "Emma";
printf("Hello, %s", name);
```

sandbox.cs50.io

```
help50
```

check50

style50

Part Two

Loops and Conditions

Conditions

```
if (x > 0)
{
   printf("x is positive\n");
}
```

Conditions

```
if (x > 0)
{
   printf("x is positive\n");
}
else
{
    printf("x is not positive\n");
}
```

Conditions

```
if (x > 0)
{
   printf("x is positive\n");
}
else if (x < 0)
{
    printf("x is negative\n");
}
else
{
    printf("x is 0\n");
}
```

Loops

```
int x = 0;
while (x < 10)
{
    printf("%i\n", x);
    x++;
}
```

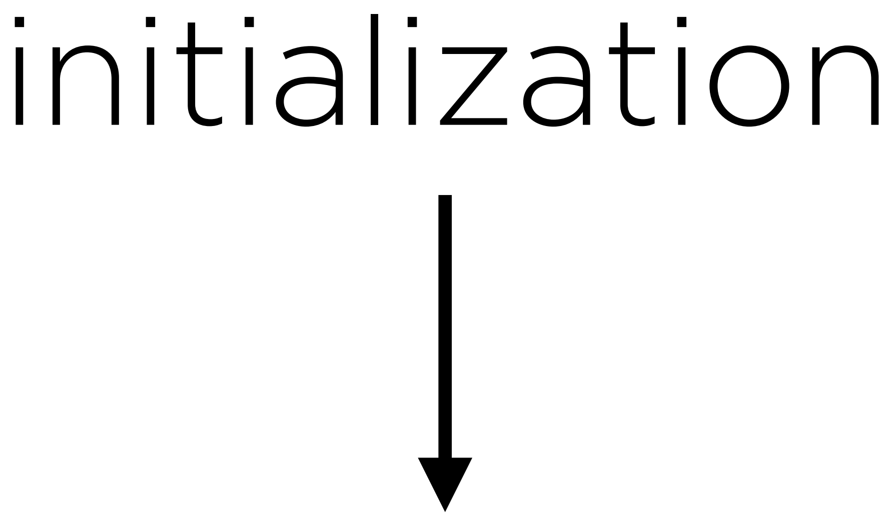
Loops

```
int x = 0;
while (true)
{
    printf("%i\n", x);
    x++;
}
```

Loops

```
for (int i = 0; i < 10; i++)
{
    printf("%i\n", i);
}
```

Loops



initialization

```
for (int i = 0; i < 10; i++)
{
    printf("%i\n", i);
}
```

Loops

condition

```
for (int i = 0; i < 10; i++)
{
    printf("%i\n", i);
}
```

Loops

increment

```
for (int i = 0; i < 10; i++)
{
    printf("%i\n", i);
}
```

Loops

```
for (int i = 0; i < 10; i++)
{
    printf("%i\n", i);
}
```

Exercise: Average

Write a program **`average.c`** that asks the user to provide ten integers as input and computes the sum.

Exercise: Multiplication Table

Write a program `multiplication.c` that prints out multiplication facts for multiplying all numbers from 1 to 10 by each other.

Sample Output:

```
1 * 1 = 1
1 * 2 = 2
1 * 3 = 3
...
10 * 9 = 90
10 * 10 = 100
```

Part Three

Functions

Functions

```
void hello(void)
{
    printf("Hello!");
}
```

Functions

function name

```
void hello(void)
{
    printf("Hello!");
}
```

Functions

inputs

```
void hello(void)
{
    printf("Hello!");
}
```

Functions

output type

```
void hello(void)
{
    printf("Hello!");
}
```

Functions

```
void hello(void)
{
    printf("Hello!");
}
```

function body

Functions

```
void hello(void)
{
    printf("Hello!");
}
```

Functions

```
void hello(int count)
{
    for (int i = 0; i < count; i++)
    {
        printf("Hello!");
    }
}
```

Functions

```
int square(int x)
{
    return x * x;
}
```

Functions

```
int sum(int x, int y)
{
    return x + y;
}
```

Policies

Academic Honesty

- https://cs50.harvard.edu/college/2020/spring/syllabus/#academic-honesty
- "... be reasonable..."
- "... when asking for help, you may show your code to others, but you may not view theirs..."

Academic Honesty

- **Regret clause.** If you commit some act that is not reasonable but bring it to the attention of the course’s heads within 72 hours, the course may impose local sanctions that may include an unsatisfactory or failing grade for work submitted, but the course will not refer the matter for further disciplinary action except in cases of repeated acts.

Late Policy

- Late submissions (of quizzes, problem sets, the test, and the final project's milestones) will be penalized at a rate of 0.1% per minute.
- However, you may grant yourself one 3-day (72-hour) extension during the term for any one problem set. (Form on course website.)

Problem Set 1

Problem Set 1

- Hello
- One of:
    - Mario (Less)
    - Mario (More)
- One of:
    - Cash
    - Credit

This is CS50.