problem set 0

https://www.cs50.net/psets/0/

office hours

walkthrough

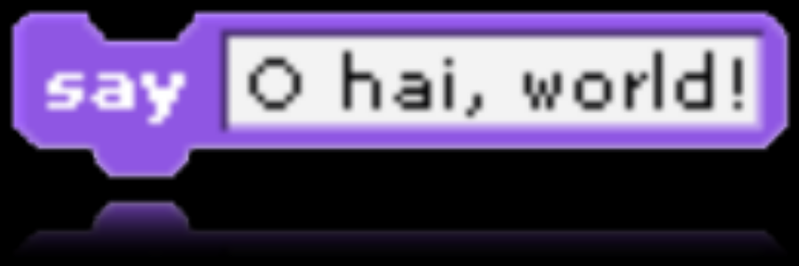
scribe notes

sectioning

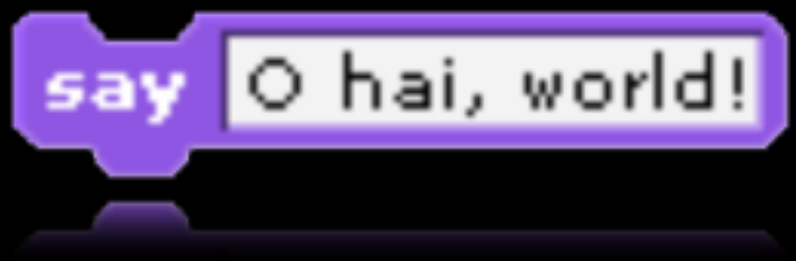

when clicked
say O hai, world!


```
int
main()
{
    printf("O hai, world!\n");
}
```

statements



statements




```
printf("O hai, world!\n");
```

loops

loops


```
while (true)
{
    printf("O hai!\n");
}
```

loops

loops


```
for (int i = 0; i < 10; i++)
{
    printf("O hai!\n");
}
```

variables

variables

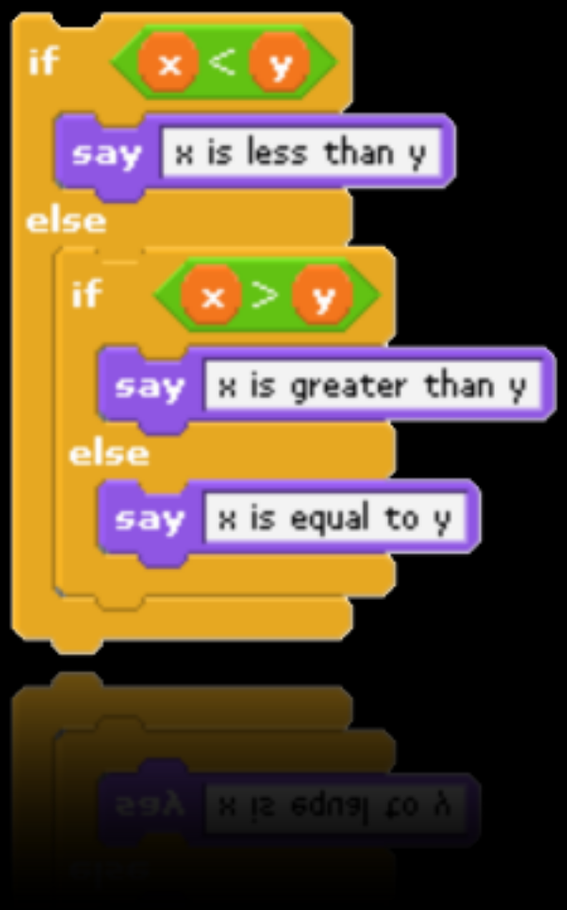

```
int counter = 0;
while (true)
 {
     printf("%d\n", counter);
     counter++;
 }
```

Boolean expressions

Boolean expressions


```
(x < y)
((x < y) && (y < z))
```

conditions


if x < y
say x is less than y
else
if x > y
say x is greater than y
else
say x is equal to y

conditions


```
if (x < y)
{
    printf("x is less than y\n");
}
else if (x > y)
{
    printf("x is greater than y\n");
}
else
{
    printf("x is equal to y\n");
}
```

arrays


```
string inventory[1];
inventory[0] = "Orange";
```

```c
#include <stdio.h>

int
main(void)
{
    printf("O hai, world!\n");
}
```

```
10000011 00000001 00010001 00000000 00111101 11111100 01110100 00111101
00000000 01000000 00000000 00000000 00000000 00000000 00000000 00000000
10010000 00000000 00000000 00000000 01010000 00000000 00000111 00110000
00001011 00000001 00001011 00000011 00001010 00000000 00000000 00000000
00000000 00100000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00100000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
01110000 00010000 00000000 00100000 00000001 00000000 00000000 00000000
00000000 00000000 00000000 00100000 00000001 00000000 00000000 00000000
00000000 00000000 00000000 01000000 00000001 00000000 00000000 00000000
00000000 00100000 00000000 01000000 00000001 00000000 00000000 00000000
11111111 11111111 11111111 11111111 11111111 11111111 11111111 11111111
10010000 10000000 00000000 01000000 00000001 00000000 00000000 00000000
00101110 01100100 01111001 01101110 01100001 01101101 01101001 01100011
10110000 00000100 00000000 00100000 00000001 00000000 00000000 00000000
10110000 00000100 00000000 00100000 00000001 00000000 00000000 00000000
10100000 00000001 00000000 00000000 00000000 00000000 00000000 00000000
10110000 00000100 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00000000 00000000 00000000
00000000 00000000 00000000 00000000 00000000 00100000 00000000 00000000
...
```

CS50 Appliance

CS50 Appliance 2.3 [Running]

Home

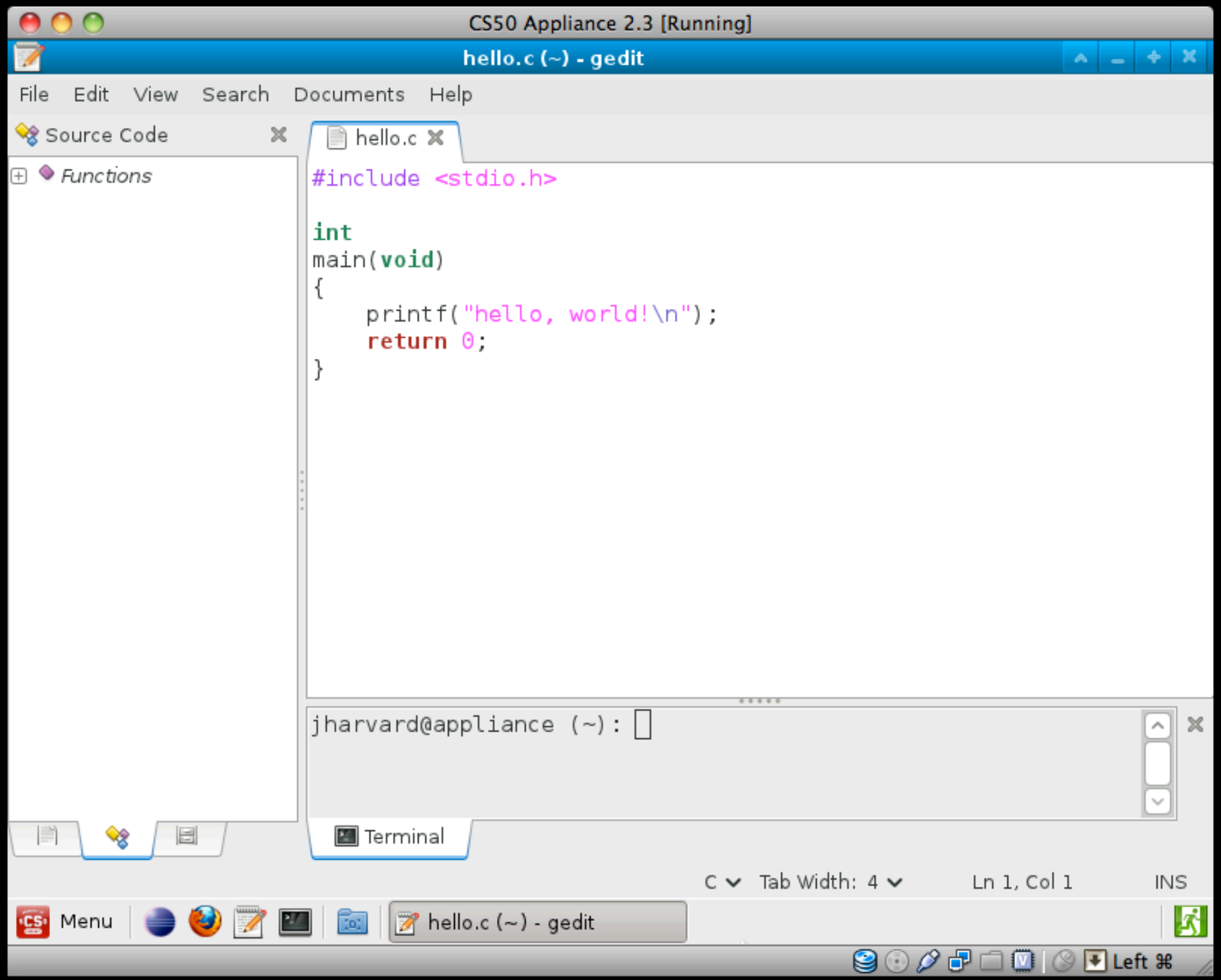
File System

Trash

Menu

Left ⌘

how to write a program



how to compile a program

```
gcc hello.c
```

how to run a program

```
./a.out
```

how to compile a program

```
gcc -o hello hello.c
```

how to run a program

```
./hello
```

how to compile a program

```
make hello
```

functions

main

Standard Library

stdio.h

```
printf

...
```

CS50 Library

cs50.h

```
GetChar

GetDouble

GetFloat

GetInt

GetLongLong

GetString
```

printf

%c %d %f %lld %s ...

escape sequences

```
\n   \r   \t   \'   \"   \\   \0   ...
```

math

\+ - * / %

primitive types

```
char   double   float   int   long long   ...
```

CS50 types

```
bool   string   ...
```

precedence

Operator	Description	Associativity

`()`
`[]`
`.`
`->`
`++ --`
	Parentheses (grouping)
	Brackets (array subscript)
	Member selection via object name
	Member selection via pointer
	Postfix increment/decrement (see Note 1)
		left-to-right

`++ --`
`+ -`
`! ~`
`(type)`
`*`
`&`
`sizeof`
	Prefix increment/decrement
	Unary plus/minus
	Logical negation/bitwise complement
	Cast (change *type*)
	Dereference
	Address
	Determine size in bytes
		right-to-left

`* / %`	Multiplication/division/modulus	left-to-right
`+ -`	Addition/subtraction	left-to-right
`<< >>`	Bitwise shift left, Bitwise shift right	left-to-right
`< <=` `> >=`	Relational less than/less than or equal to Relational greater than/greater than or equal to	left-to-right
`== !=`	Relational is equal to/is not equal to	left-to-right
`&`	Bitwise AND	left-to-right
`^`	Bitwise exclusive OR	left-to-right
`|`	Bitwise inclusive OR	left-to-right
`&&`	Logical AND	left-to-right
`||`	Logical OR	left-to-right
`?:`	Ternary conditional	right-to-left

`=`
`+= -=`
`*= /=`
`%= &=`
`^= |=`
`<<= >>=`
	Assignment
	Addition/subtraction assignment
	Multiplication/division assignment
	Modulus/bitwise AND assignment
	Bitwise exclusive/inclusive OR assignment
	Bitwise shift left/right assignment
		right-to-left

`,`	Comma (separate expressions)	left-to-right

http://www.difranco.net/cop2220/op-prec.htm

how to compile a program

```
gcc -o hello hello.c -lcs50
```

how to compile a program

```
make hello
```

to be continued...