Macros

Macros are a powerful extension of #define statements. They are similar to functions in the sense that they use arguments.

We'll cover the following

- Variadic Macros

You can use `#define` statements in more advanced ways, which are sometimes called **macros** (a macro is typically used to define something that takes one or more arguments).

For example, you could define a macro to perform the square of a number like this:

```
#define SQUARE(n) n*n
```

Then you could use it in your code like this:

```
#include <stdio.h>
#define SQUARE(n) n*n

int main()
{
  int x = 3;
  int y = SQUARE(x);
  printf("%d",y);
  return 0;

}
```

Another common example of macros is to define a macro to return the maximum value of two arguments:

```
#define MAX(a,b) ( ((a) > (b)) ? (a) : (b) )
#include <stdio.h>

int main()
```

```
{
  int maxNumber=MAX(4,5);
  printf("The maximum number is %d", maxNumber);

 return 0;

}
```

Another one is to determine if a character is lower-case:

```
#define IS_LOWER_CASE(x) ( ((x) >= 'a') && ((x) <= 'z') )

int main()
{
 int answer= IS_LOWER_CASE('b');
  //the condition will either evaluate to 1(TRUE) or 0(FALSE)
 printf("%d",answer);
 return 0;
}
```

Variadic Macros

A Variadic macro is one with a variable number of arguments (one can also write variadic functions in C). Here is an example:

```
#define debugPrintf(...) printf("DEBUG: " __VA_ARGS__);
```

Now we could use this in the following way, either with one argument:

```
#define debugPrintf(...) printf("DEBUG: " __VA_ARGS__);
int main()
{
  debugPrintf("Hello World!\n");
  return 0;
}
```

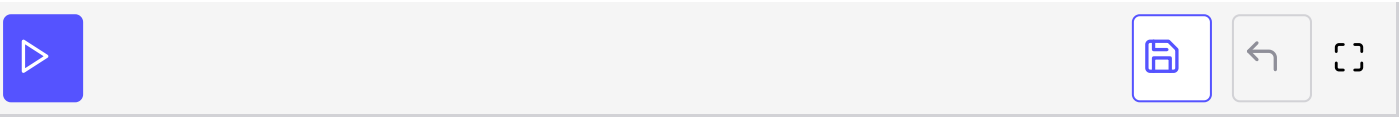
or with multiple arguments:

```
#define debugPrintf(...) printf("DEBUG: " __VA_ARGS__);

int main()
{
```

```
  int x=12;
  int y=13;
  debugPrintf("x=%d, y=%d\n", x, y);

  return 0;
}
```



There are countless other examples of cool things you can do with macros. Here is a link to some examples: Macros.

We can also tell the compiler how and we want to execute macros by using **conditional statements**. Now, we'll look at several different conditional statements.