

# The Expressive Mr C

ESC101: Foundations of Computing

Purushottam Kar

# Announcements

- No ESC101 lecture tomorrow, no ESC101 lab tomorrow
- Extra ESC101 lecture on Saturday 18 Aug, 12noon, L20
- Extra ESC101 lab for sections B10, B11, B12, B14 on Saturday 18 Aug, 2PM, New Core Labs CC-01, CC-02



# Recap

3



# Recap

3

Saw several datatypes int, long, float, double



# Recap

3

Saw several datatypes int, long, float, double  
Can be printed (printf) and input (scanf)



# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)



# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)

When using mixed-type expressions, be careful about typecasting errors



# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)

When using mixed-type expressions, be careful about typecasting errors

2.0 is not the same as 2 for Mr C





# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)

When using mixed-type expressions, be careful about typecasting errors

2.0 is not the same as 2 for Mr C

Be careful about using correct notation in printf, scanf



# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)

When using mixed-type expressions, be careful about typecasting errors

2.0 is not the same as 2 for Mr C

Be careful about using correct notation in printf, scanf

Use %d for int, %ld for long, %f or %e for float and double



# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)

When using mixed-type expressions, be careful about typecasting errors

2.0 is not the same as 2 for Mr C

Be careful about using correct notation in printf, scanf

Use %d for int, %ld for long, %f or %e for float and double

Do not use %d for float – will get strange answers



# Recap

3

Saw several datatypes int, long, float, double

Can be printed (printf) and input (scanf)

Can use math operations on them +, -, /, \*, % (only int, long)

When using mixed-type expressions, be careful about typecasting errors

2.0 is not the same as 2 for Mr C

Be careful about using correct notation in printf, scanf

Use %d for int, %ld for long, %f or %e for float and double

Do not use %d for float – will get strange answers

Can experiment in free time but be careful during labs/exams/quizzes



# Today

# 13



# Today

13

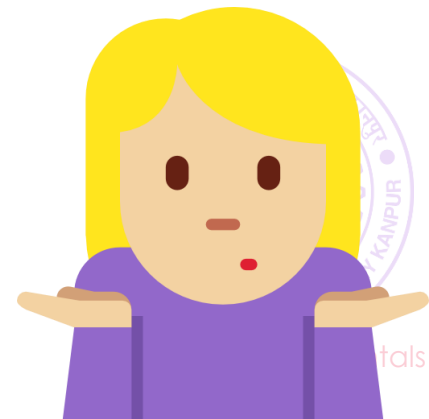
Find out if Mr C went to primary school



# Today

13

Find out if Mr C went to primary school

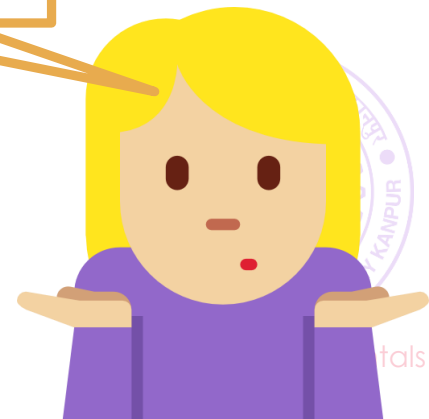


# Today

13

Find out if Mr C went to primary school

Does he know the  
BODMAS rule?





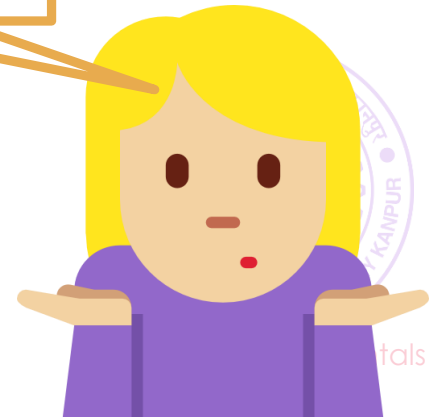
# Today

13

Find out if Mr C went to primary school

Learn how to solve Aadhar Leak problem more beautifully

Does he know the  
BODMAS rule?



# Today

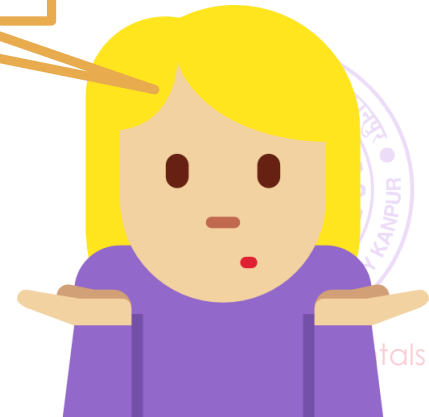
13

Find out if Mr C went to primary school

Learn how to solve Aadhar Leak problem more beautifully

If pin is 6744 and date of birth is 08/11/1978, password is 674411

Does he know the  
BODMAS rule?



# Today

13

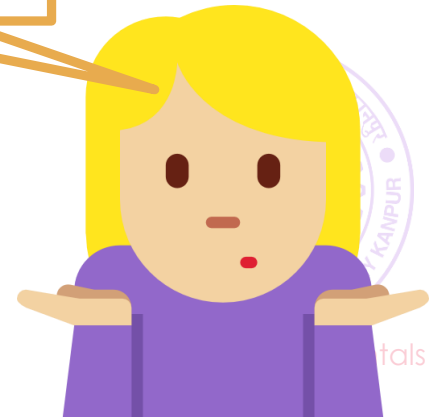
Find out if Mr C went to primary school

Learn how to solve Aadhar Leak problem more beautifully

If pin is 6744 and date of birth is 08/11/1978, password is 674411

If pin is 6744 and date of birth is 08/05/1978, password is 674405 (not 67445)

Does he know the  
BODMAS rule?



# Today

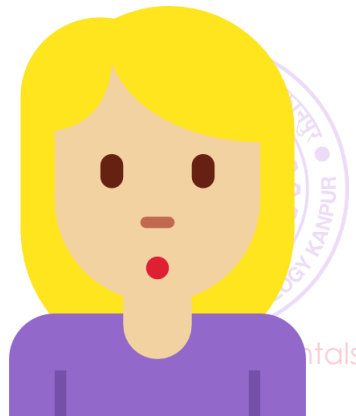
13

Find out if Mr C went to primary school

Learn how to solve Aadhar Leak problem more beautifully

If pin is 6744 and date of birth is 08/11/1978, password is 674411

If pin is 6744 and date of birth is 08/05/1978, password is 674405 (not 67445)



# Today

13

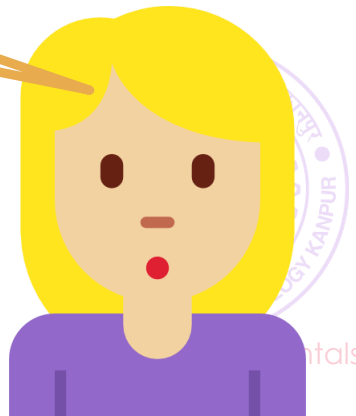
Find out if Mr C went to primary school

Learn how to solve Aadhar Leak problem more beautifully

If pin is 6744 and date of birth is 08/11/1978, password is 674411

If pin is 6744 and date of birth is 08/05/1978, password is 674405 (not 67445)

That would be  
really nice!



# Today

13

Find out if Mr C went to primary school

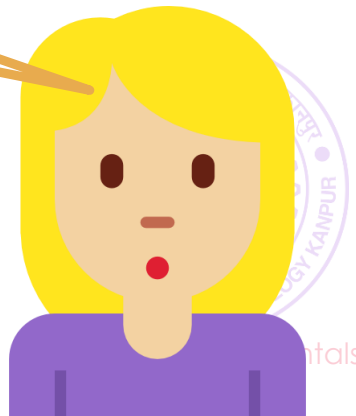
Learn how to solve Aadhar Leak problem more beautifully

If pin is 6744 and date of birth is 08/11/1978, password is 674411

If pin is 6744 and date of birth is 08/05/1978, password is 674405 (not 67445)

But first, let us finish the unfinished business from last class 😊

That would be  
really nice!



# Math Formulae and Expressions

23



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math





# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}(\text{double}y, 2.0) + \text{pow}(\text{double}z, 2.0))$$



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}(\text{double}y, 2.0) + \text{pow}(\text{double}z, 2.0))$$

Same answer, use math.h



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C





# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C

Oh! So two expressions  
can be added together to  
get another expression!



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

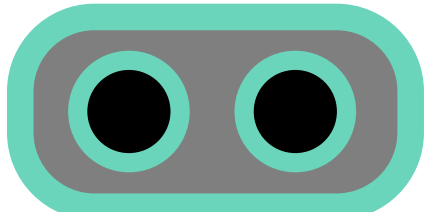
$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C

Oh! So two expressions  
can be added together to  
get another expression!



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

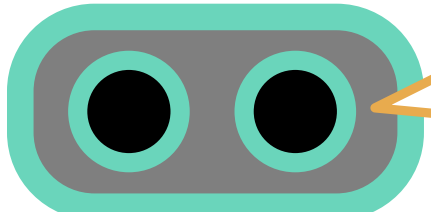
$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C

Oh! So two expressions can be added together to get another expression!

Yes, take two expressions and do operations like addition, multiplication, or assignment (=) with them and a new expression will emerge



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

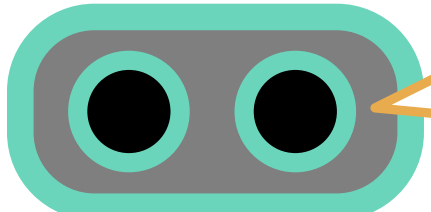
$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C

Oh! So two expressions can be added together to get another expression!

So is  $z$  an expression?

Yes, take two expressions and do operations like addition, multiplication, or assignment (=) with them and a new expression will emerge



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

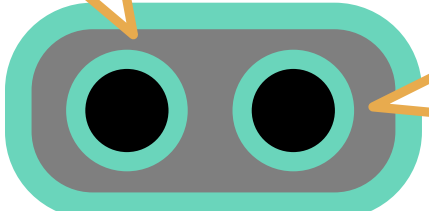
$z * z$  is also an expression for Mr C

Oh! So two expressions can be added together to get another expression!

It sure is!

Yes, take two expressions and do operations like addition, multiplication, or assignment (=) with them and a new expression will emerge

So is  $z$  an expression?



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C

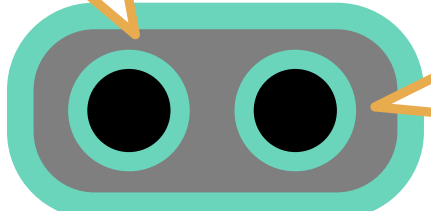
Oh! So two expressions can be added together to get another expression!

It sure is!

Yes, take two expressions and do operations like addition, multiplication, or assignment (=) with them and a new expression will emerge

So is  $z$  an expression?

So is 5 an expression?



# Math Formulae and Expressions

23

We use math formulae all the time in physics, chem, math

$$a = b / 5$$

$$x = y * y + z * z$$

$$x = (\text{int})(\text{pow}((\text{double})y, 2.0) + \text{pow}((\text{double})z, 2.0))$$

Same answer, use math.h

Mr C calls these formulae *expressions*

$x = y * y + z * z$  is an expression for Mr C

$y * y + z * z$  is also an expression for Mr C

$y * y$  is also an expression for Mr C

$z * z$  is also an expression for Mr C

Oh! So two expressions can be added together to get another expression!

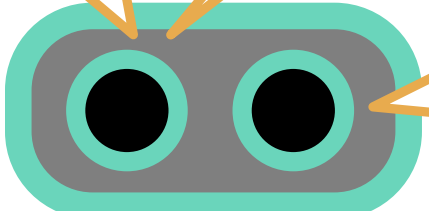
It sure is!

It sure is!

Yes, take two expressions and do operations like addition, multiplication, or assignment (=) with them and a new expression will emerge

So is  $z$  an expression?

So is 5 an expression?





# Expressions and Statements

41



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C  
a = b / 5 would be an incorrect thing to write in a C program



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



# Expressions and Statements

41

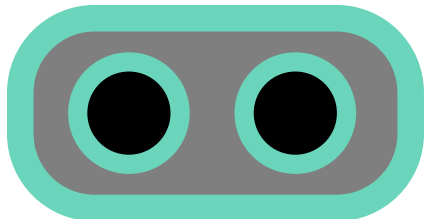
Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



# Expressions and Statements

41

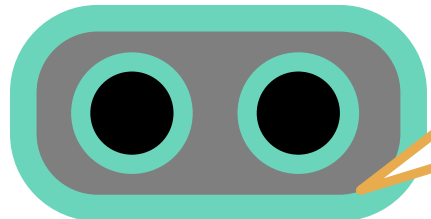
Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



It makes me sad since I have to throw away the results of a calculation





# Expressions and Statements

41

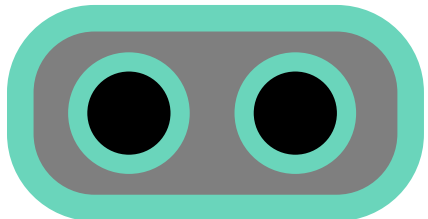
Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



# Expressions and Statements

41

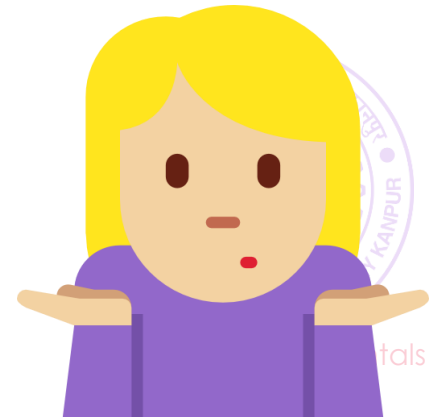
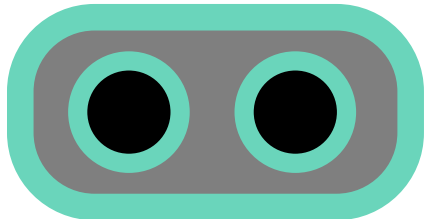
Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C

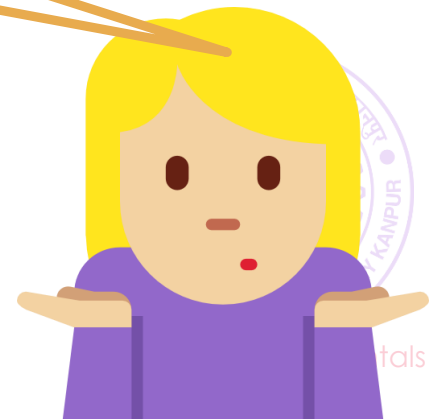
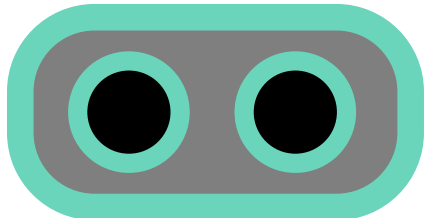
$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one

So I can take any expression,  
put a semicolon at the end,  
and it becomes a statement?



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C

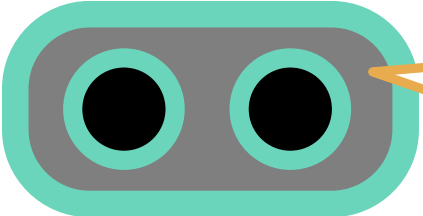
$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

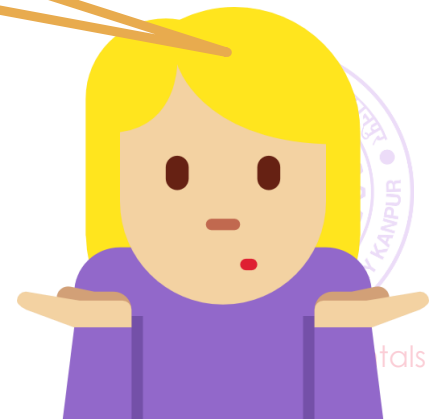
$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one

So I can take any expression,  
put a semicolon at the end,  
and it becomes a statement?



It sure  
does!



# Expressions and Statements

41

Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

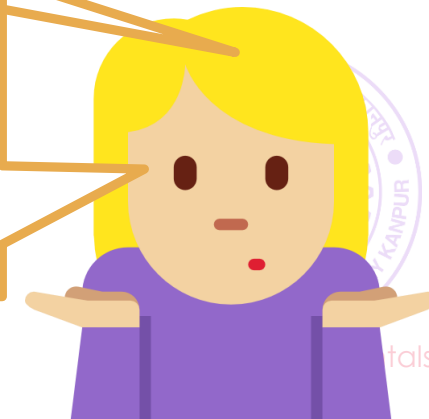
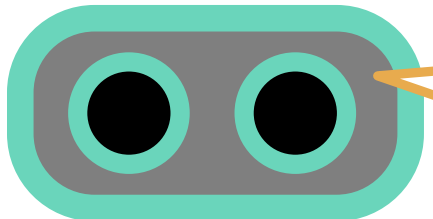
$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one

So I can take any expression,  
put a semicolon at the end,  
and it becomes a statement?

Hmm ... are there any  
examples where I do not  
put a semicolon at the  
end of an expression?

It sure  
does!



# Expressions and Statements

41

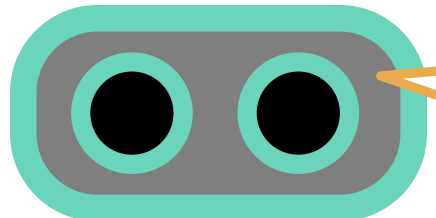
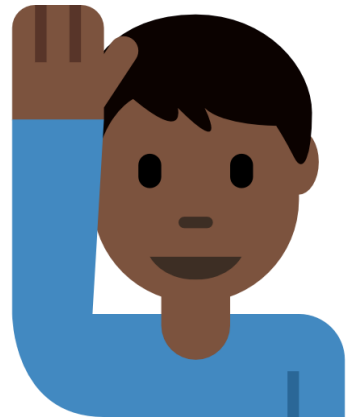
Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

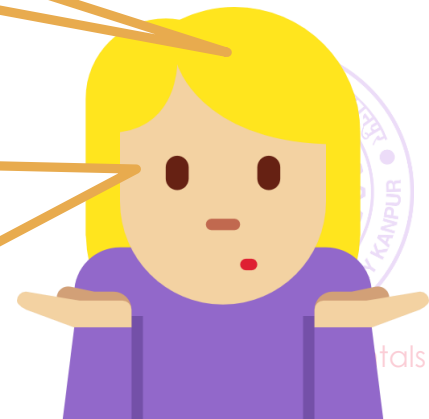
$b / 5;$  is also a statement in C although a bit useless one



It sure does!

So I can take any expression, put a semicolon at the end, and it becomes a statement?

Hmm ... are there any examples where I do not put a semicolon at the end of an expression?



# Expressions and Statements

41


Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

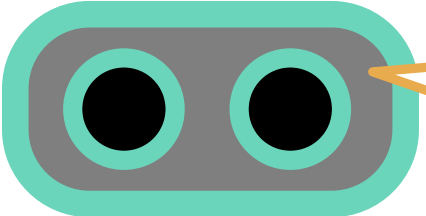
$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



```
printf("%d",b/5);
```

So I can take any expression,  
put a semicolon at the end,  
and it becomes a statement?



It sure  
does!

Hmm ... are there any  
examples where I do not  
put a semicolon at the  
end of an expression?

# Expressions and Statements

41


Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

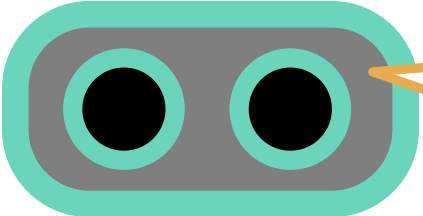
$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



```
printf("%d",b/5);
```

$b/5$  is an expression  
sitting inside a statement



It sure  
does!

So I can take any expression,  
put a semicolon at the end,  
and it becomes a statement?

Hmm ... are there any  
examples where I do not  
put a semicolon at the  
end of an expression?



# Expressions and Statements

41


Remember, we use semicolon ; as a full-stop in C

$a = b / 5$  would be an incorrect thing to write in a C program

$a = b / 5;$  is the correct way to divide  $b$  by 5 and store result in  $a$

$a = b / 5;$  is called a *statement* in C

$b / 5;$  is also a statement in C although a bit useless one



```
printf("%d",b/5);
```

$b/5$  is an expression  
sitting inside a statement

So I can take any expression,  
put a semicolon at the end,  
and it becomes a statement?



It sure  
does!

Correct!  

```
printf("%d",b/5);
```

  
is a statement

Hmm ... are there any  
examples where I do not  
put a semicolon at the  
end of an expression?

# A Quiz for Mr C

58



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`  
What is the result of `a = b + c + d;`



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

It does not matter. The result is the same 😊





# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

It does not matter. The result is the same 😊



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

It does not matter. The result is the same 😊



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`

It does not matter. The result is the same 😊



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

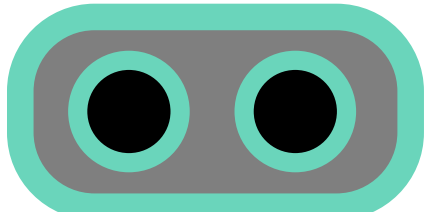
Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

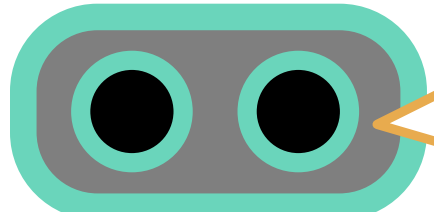
Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



All binary math operators `+`, `-`, `/`, `*`, `%` behave this way. This behaviour is called left associativity



# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

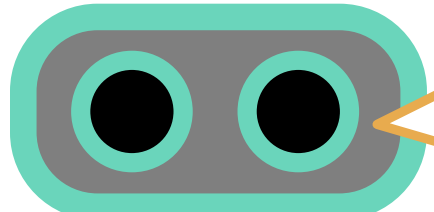
Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



All binary math operators `+`, `-`, `/`, `*`, `%` behave this way. This behaviour is called left associativity

Brackets and evaluation go from left to right





# A Quiz for Mr C

58

Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

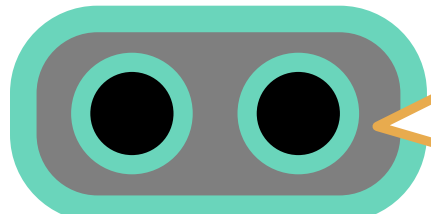
Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



All binary math operators `+`, `-`, `/`, `*`, `%` behave this way. This behaviour is called left associativity

Brackets and evaluation go from left to right

What if I want the second answer?



# A Quiz for Mr C



Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

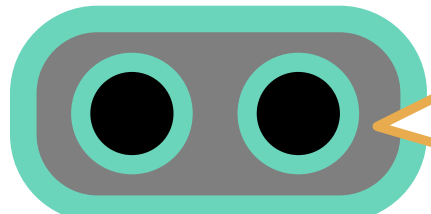
Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



All binary math operators `+`, `-`, `/`, `*`, `%` behave this way. This behaviour is called left associativity

Brackets and evaluation go from left to right

What if I want the second answer?



# A Quiz for Mr C

Use brackets yourself! 😊



Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

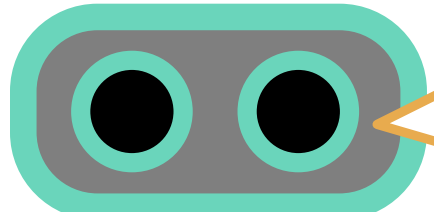
Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`



All binary math operators `+`, `-`, `/`, `*`, `%` behave this way. This behaviour is called left associativity

What if I want the second answer?

Brackets and evaluation go from left to right



# A Quiz for Mr C

Use brackets yourself! 😊



Suppose we have `int a, b = 5, c = 3, d = 2;`

What is the result of `a = b + c + d;`

Is it computed as `a = b + (c + d);` ?

Is it computed as `a = (b + c) + d;` ?

What about the result of `a = b - c - d;`

If computed as `a = (b - c) - d;` the answer is `a = 0`

If computed as `a = b - (c - d);` the answer is `a = 4;`

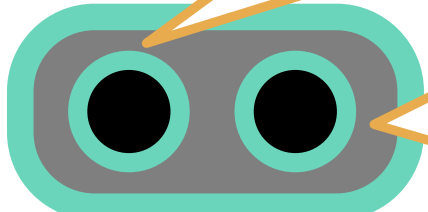


Exactly! It is always a good idea to put brackets. Less confusion, less chance of error

All binary math operators `+, -, /, *, %` behave this way. This behaviour is called left associativity

Brackets and evaluation go from left to right

What if I want the second answer?



# Multi-operator expressions

77



# Multi-operator expressions

77

If multiple operators inside a formula (expression),  
BODMAS rule applies



# Multi-operator expressions

77

If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left



# Multi-operator expressions

77

If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left

HIGH  
PRECEDENCE



LOW  
PRECEDENCE





# Multi-operator expressions

77

If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left

HIGH  
PRECEDENCE



LOW  
PRECEDENCE

$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$



# Multi-operator expressions

77

If multiple operators inside a formula (expression),  
BODMAS rule applies

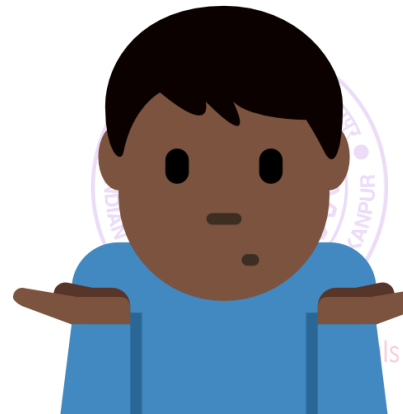
Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left

HIGH  
PRECEDENCE



LOW  
PRECEDENCE

$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$



# Multi-operator expressions

77

If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left

HIGH  
PRECEDENCE



Any operators  
with Right  
Associativity?

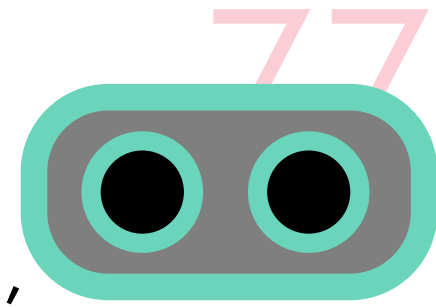


$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$

LOW  
PRECEDENCE

# Multi-operator expressions

If multiple operators inside a formula (expression),  
BODMAS rule applies



Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left

HIGH  
PRECEDENCE



Any operators  
with Right  
Associativity?



$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$

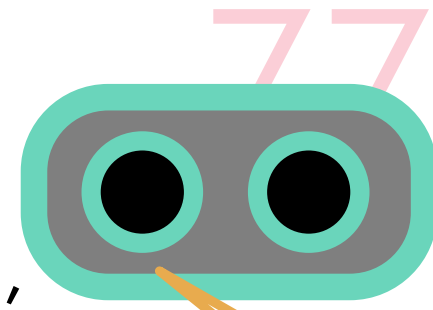
LOW  
PRECEDENCE

# Multi-operator expressions

If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left

HIGH  
PRECEDENCE



Yup!

Any operators  
with Right  
Associativity?



$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$

LOW  
PRECEDENCE

# Multi-operator expressions

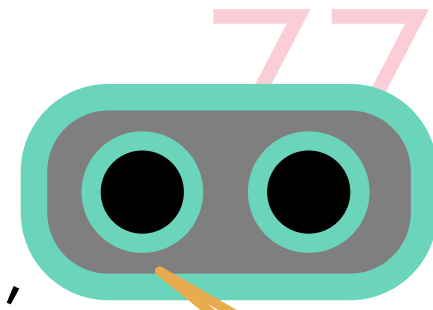
If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Unary negation	-	Right
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left
Assignment	=	Right

$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$

HIGH  
PRECEDENCE

LOW  
PRECEDENCE



Yup!

Any operators  
with Right  
Associativity?



# Multi-operator expressions

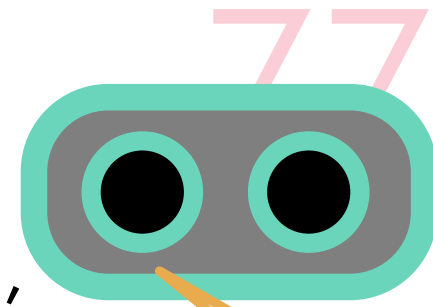
If multiple operators inside a formula (expression),  
BODMAS rule applies

Operator Name	Symbol/Sign	Associativity
Bracket	()	Left
Unary negation	-	Right
Multiplication/division/ remainder	*, /, %	Left
Addition/subtraction	+, -	Left
Assignment	=	Right

$a + b - c * d \% e / f$  is same as  $(a+b) - (((c * d) \% e) / f)$

HIGH  
PRECEDENCE

LOW  
PRECEDENCE



Yup!

Aha! = is also  
an operator

Any operators  
with Right  
Associativity?



# Multi-operator Multi-type formulae





# Multi-operator Multi-type formulae

Be very very careful about these



# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide

Do I have to  
memorize  
this table?



# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



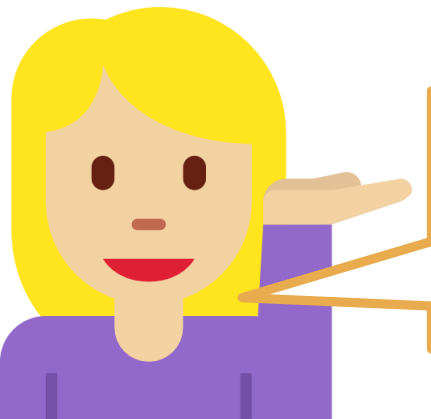
Do I have to  
memorize  
this table?



# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



Write it down in your notebook. Allowed in labs, quizzes, exams

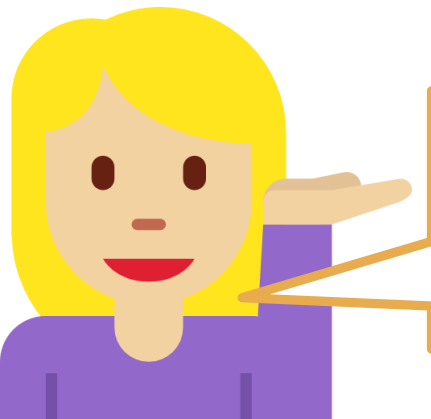


Do I have to memorize this table?

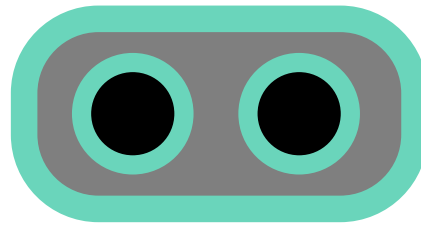
# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



Write it down in your notebook. Allowed in labs, quizzes, exams



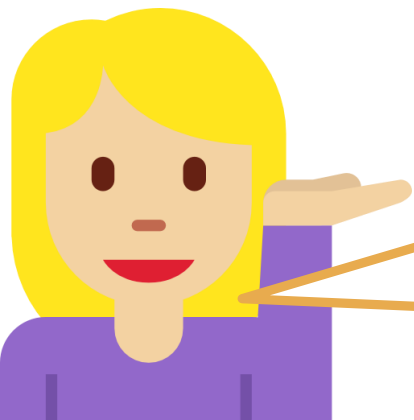
Do I have to memorize this table?



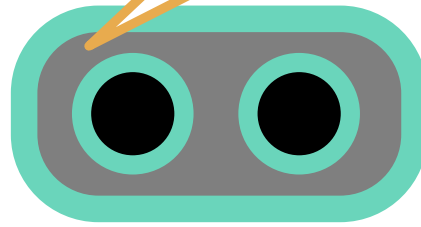
# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



Write it down in your notebook. Allowed in labs, quizzes, exams



Don't forget to practice!

Do I have to memorize this table?

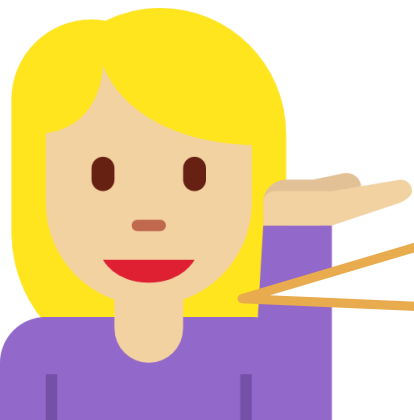




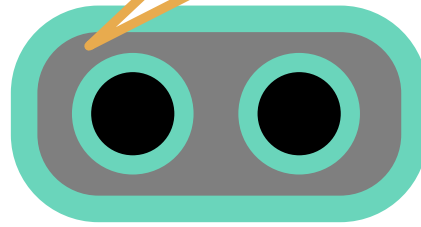
# Multi-operator Multi-type formulae 88

Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide



Write it down in your notebook. Allowed in labs, quizzes, exams



Don't forget to practice!

Mr. C, you are starting to sound just like my parents

Do I have to memorize this table?

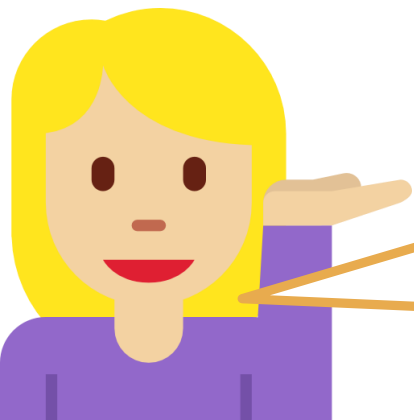


# Multi-operator Multi-type formulae 88

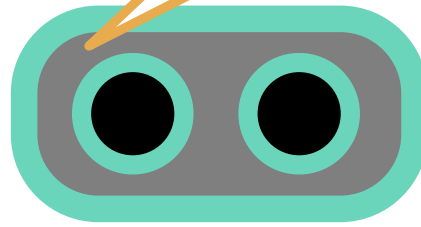
Be very very careful about these

Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide

Always a good idea to put brackets. Others can also then read your code more easily. Less surprises.



Write it down in your notebook. Allowed in labs, quizzes, exams



Don't forget to practice!

Mr. C, you are starting to sound just like my parents

Do I have to memorize this table?



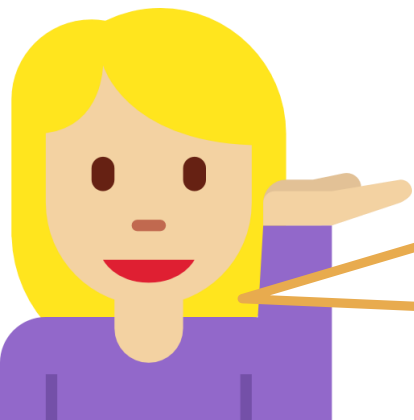
# Multi-operator Multi-type formulae 88

Be very very careful about these

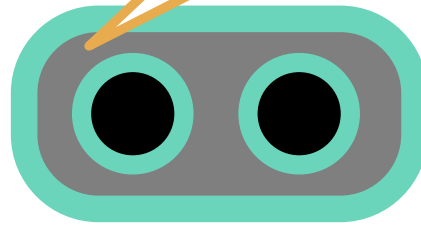
Mr C will apply his automatic typecasts, but only in the order defined in the BODMAS table in the previous slide

Always a good idea to put brackets. Others can also then read your code more easily. Less surprises.

Also a good idea to typecast yourself.



Write it down in your notebook. Allowed in labs, quizzes, exams



Don't forget to practice!

Mr. C, you are starting to sound just like my parents

Do I have to memorize this table?



# Aadhar Leak



# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411



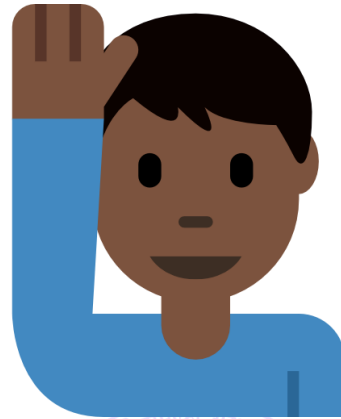
# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405



# Aadhar Leak

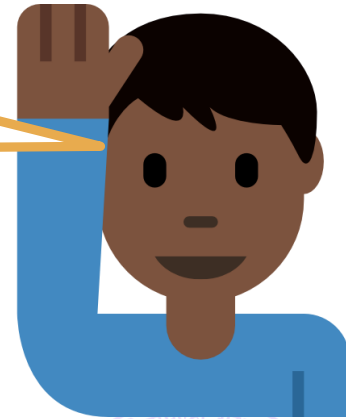
- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405



# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

Just print a 0  
before the month!

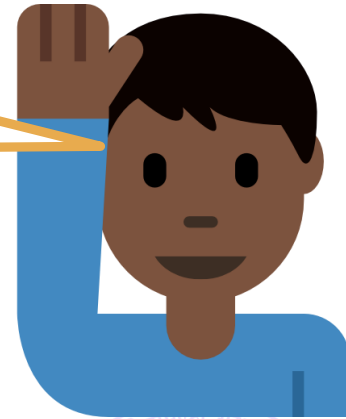




# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

Just print a 0  
before the month!

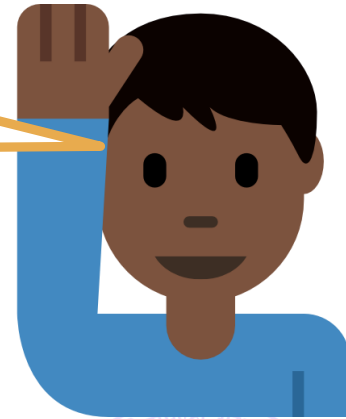


# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

Just print a 0  
before the month!

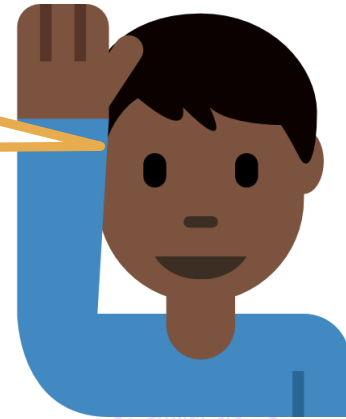


# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>  
int main(){
```

Just print a 0  
before the month!



# Aadhar Leak

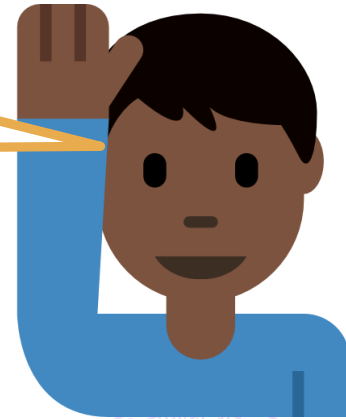
- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

```
int main(){
```

```
    int p, d, m, y;
```

Just print a 0  
before the month!



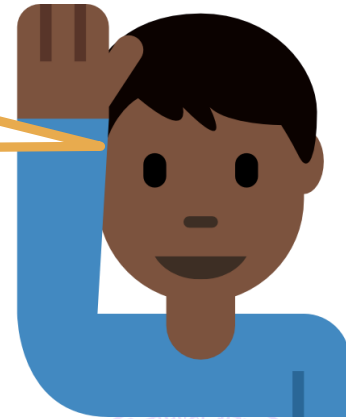
# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>

int main(){
    int p, d, m, y;
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

Just print a 0  
before the month!



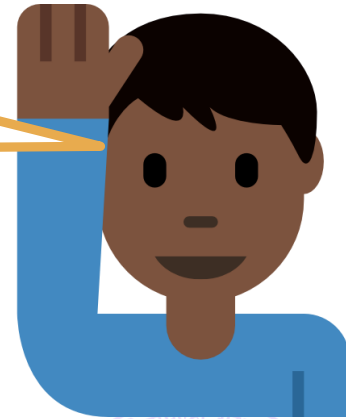
# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>

int main(){
    int p, d, m, y;
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
    printf("%d0%d",p,m);
```

Just print a 0  
before the month!



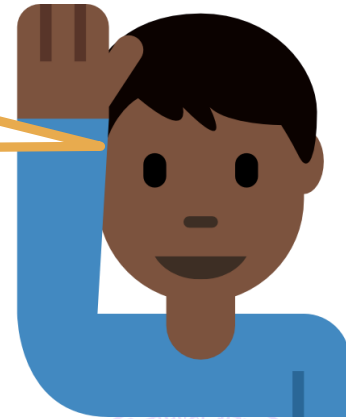
# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>

int main(){
    int p, d, m, y;
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
    printf("%d0%d",p,m);
}
```

Just print a 0  
before the month!



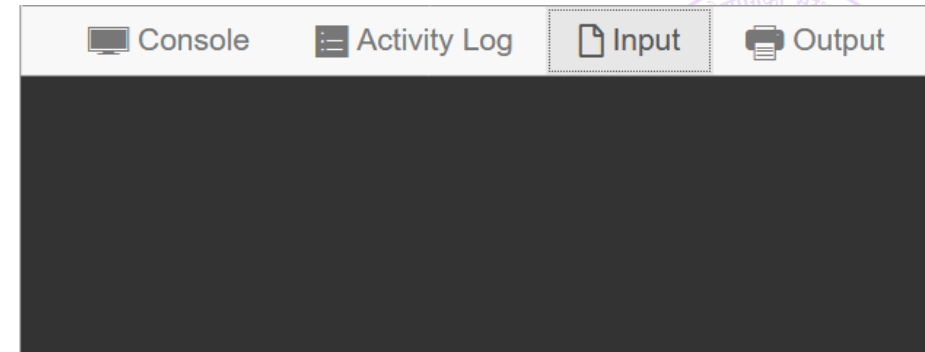
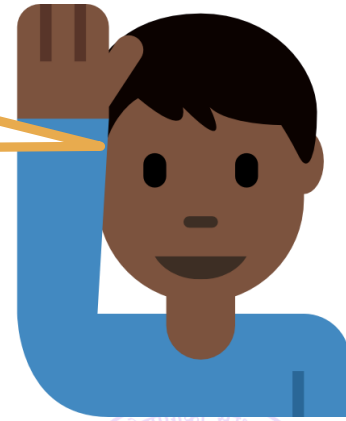
# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>

int main(){
    int p, d, m, y;
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
    printf("%d0%d",p,m);
}
```

Just print a 0  
before the month!





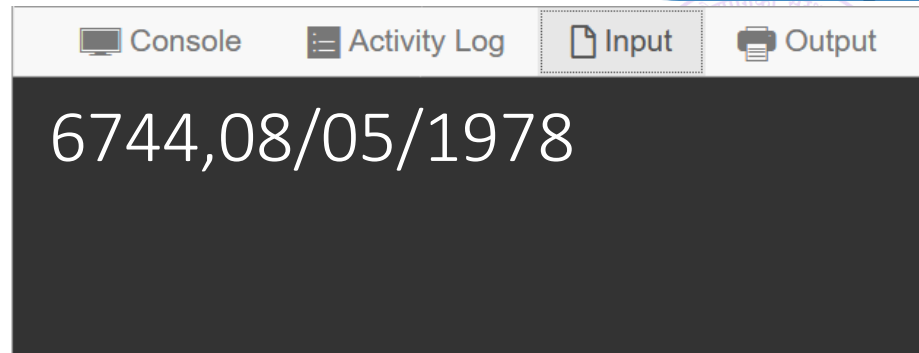
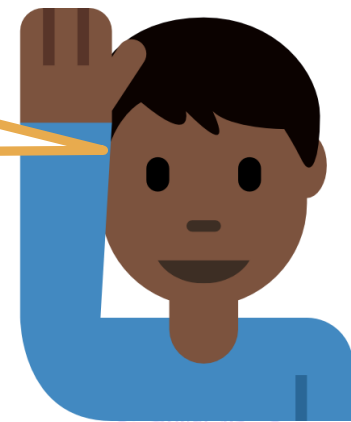
# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>

int main(){
    int p, d, m, y;
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
    printf("%d0%d",p,m);
}
```

Just print a 0  
before the month!



# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

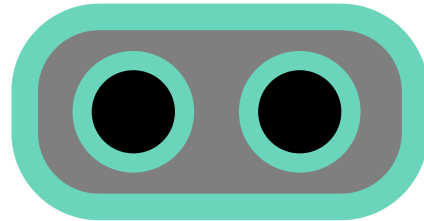
```
int main(){
```

```
    int p, d, m, y;
```

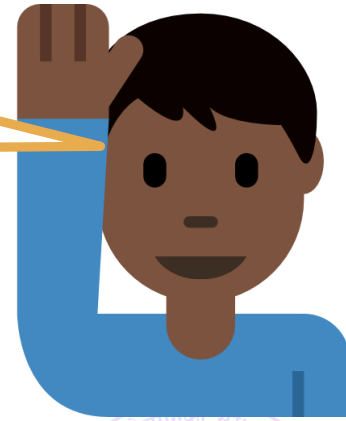
```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

```
    printf("%d0%d",p,m);
```

```
}
```



Just print a 0  
before the month!



Console

Activity Log

Input

Output

6744,08/05/1978

# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

```
int main(){
```

```
    int p, d, m, y;
```

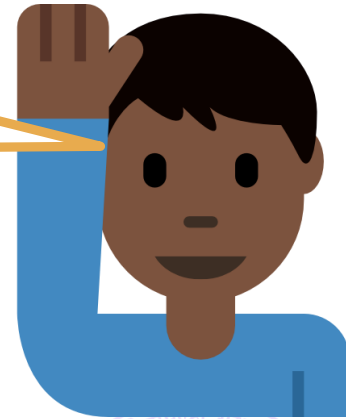
```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

```
    printf("%d0%d",p,m);
```

```
}
```

674405

Just print a 0  
before the month!



Console

Activity Log

Input

Output

6744,08/05/1978

# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

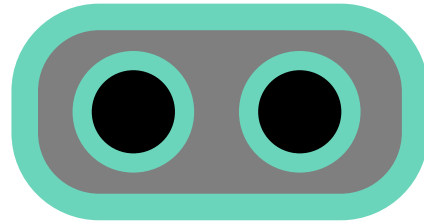
```
int main(){
```

```
    int p, d, m, y;
```

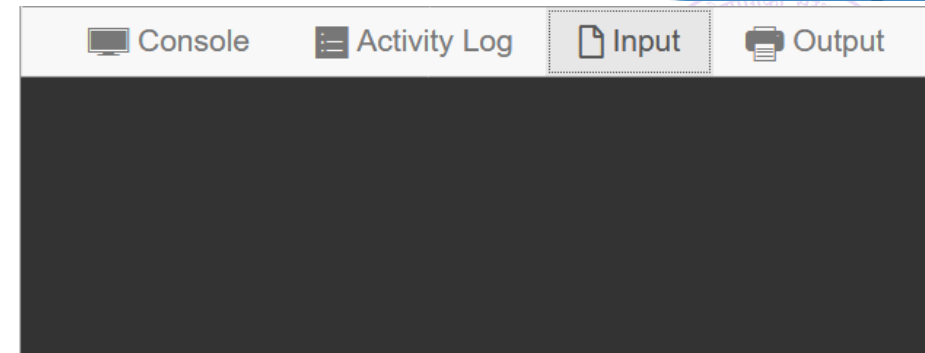
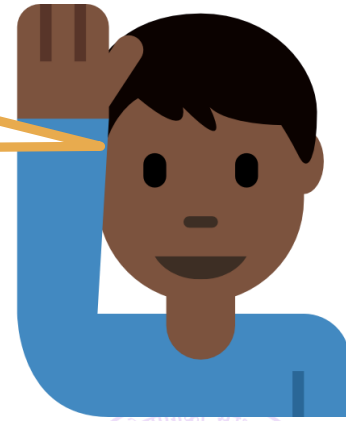
```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

```
    printf("%d0%d",p,m);
```

```
}
```



Just print a 0  
before the month!



# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

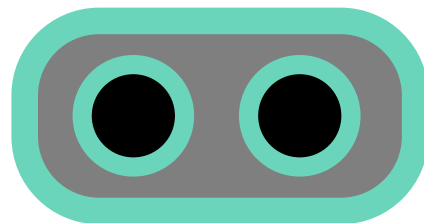
```
int main(){
```

```
    int p, d, m, y;
```

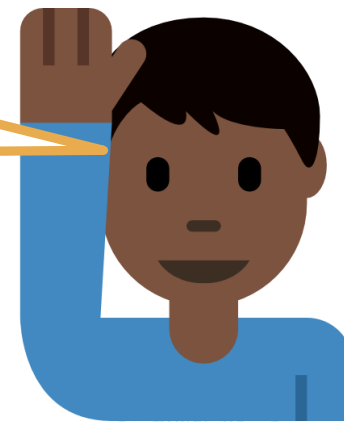
```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

```
    printf("%d0%d",p,m);
```

```
}
```



Just print a 0  
before the month!



Console

Activity Log

Input

Output

6744,08/11/1978

# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

```
int main(){
```

```
    int p, d, m, y;
```

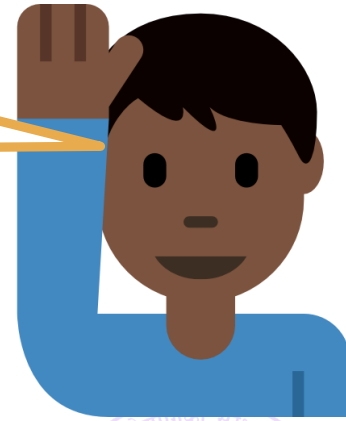
```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

```
    printf("%d0%d",p,m);
```

```
}
```

6744011

Just print a 0  
before the month!



Console

Activity Log

Input

Output

6744,08/11/1978

# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

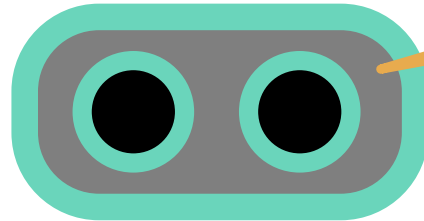
```
int main(){
```

```
    int p, d, m, y;
```

```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

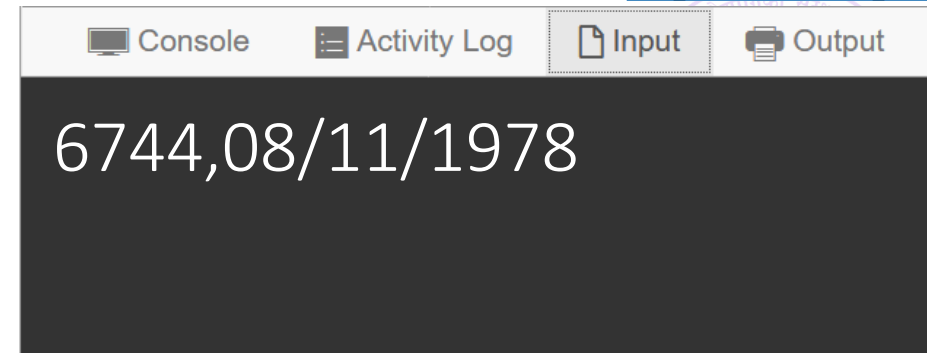
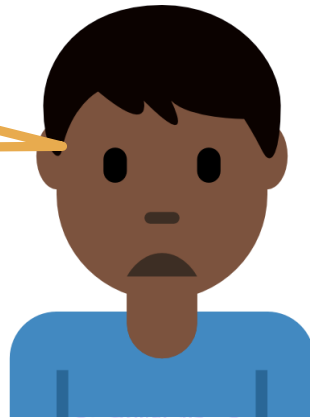
```
    printf("%d0%d",p,m);
```

```
}
```



6744011

Just print a 0  
before the month!



# Aadhar Leak

- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

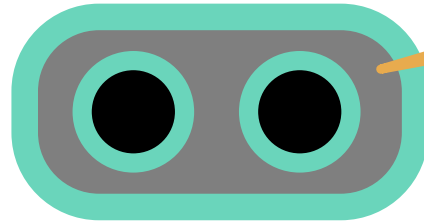
```
int main(){
```

```
    int p, d, m, y;
```

```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

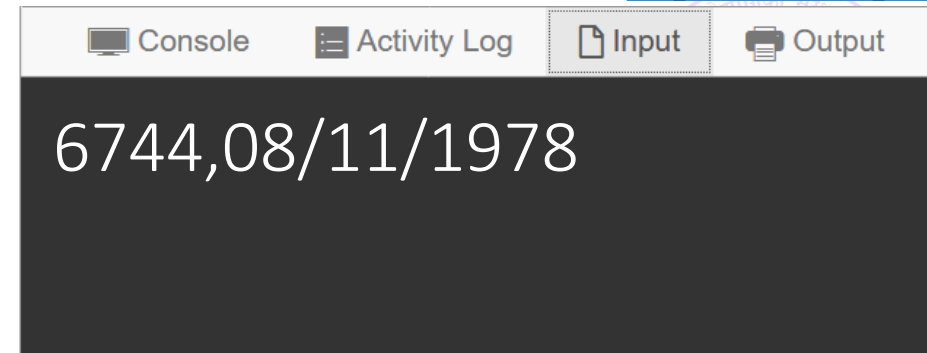
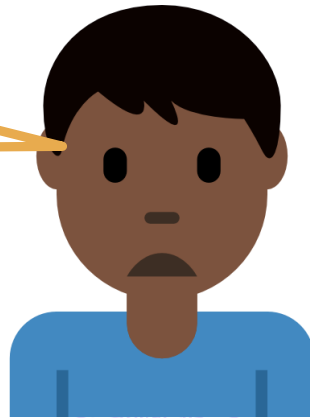
```
    printf("%d0%d",p,m);
```

```
}
```



6744011

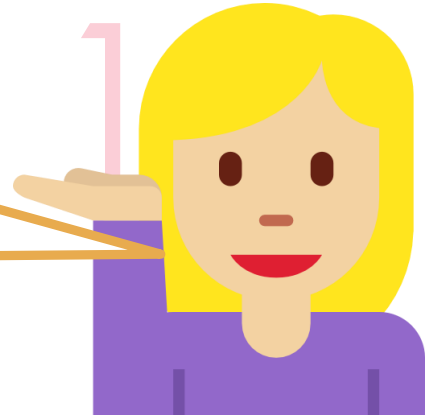
Just print a 0  
before the month!





# Aadhar Leak

You want to print a zero only when the month number is less than 10



- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

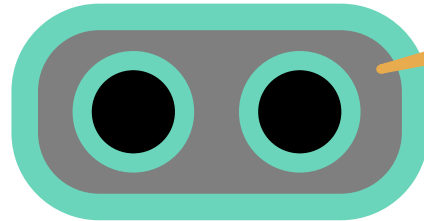
```
int main(){
```

```
    int p, d, m, y;
```

```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

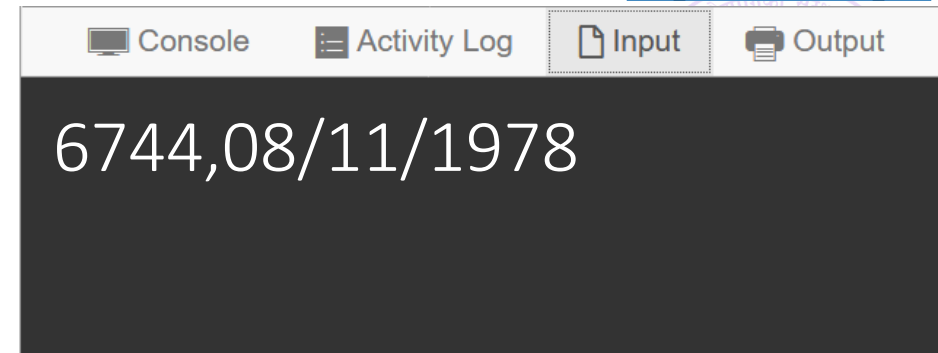
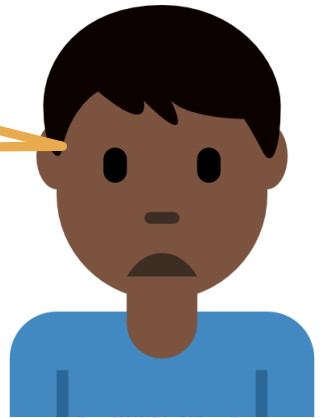
```
    printf("%d0%d",p,m);
```

```
}
```



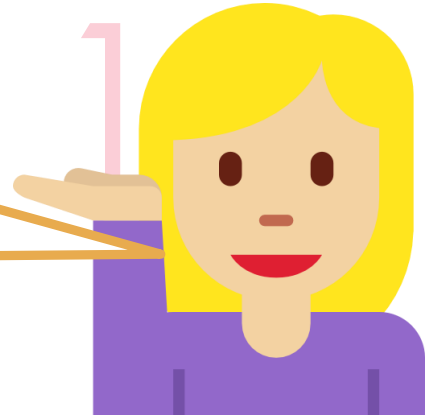
6744011

Just print a 0 before the month!



# Aadhar Leak

You want to print a zero only when the month number is less than 10



- If input is 6744,08/11/1978, password is 674411
- If input is 6744,08/05/1978, password is 674405

```
#include <stdio.h>
```

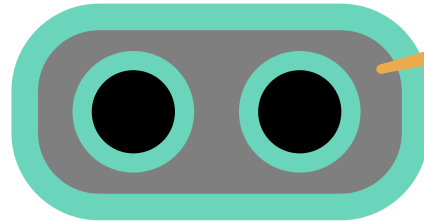
```
int main(){
```

```
    int p, d, m, y;
```

```
    scanf("%d,%d/%d/%d",&p,&d,&m,&y);
```

```
    printf("%d0%d",p,m);
```

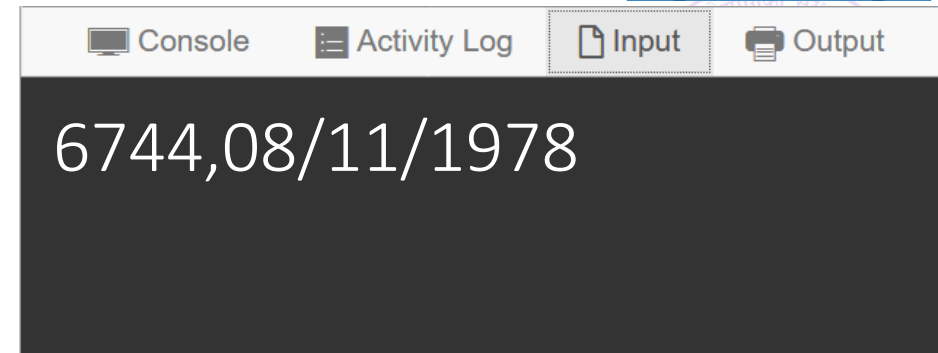
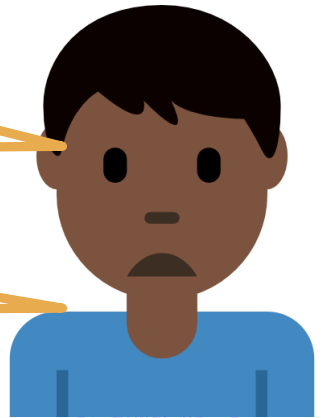
```
}
```



6744011

Just print a 0 before the month!

How do I do that?



# The much awaited if statement 😊



# The much awaited if statement 😊123

**HOW WE USUALLY SPEAK TO A HUMAN**



# The much awaited if statement 😊123

## HOW WE USUALLY SPEAK TO A HUMAN



# The much awaited if statement 😊123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?



# The much awaited if statement 😊123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello



# The much awaited if statement 😊123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

m is an integer variable.





# The much awaited if statement 😊123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

m is an integer variable.

Please ask me for value of m.



# The much awaited if statement 😊 123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0



# The much awaited if statement 😊 123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

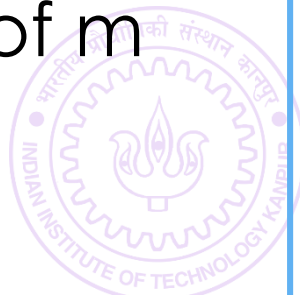
Hello

m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0

Now please print value of m



# The much awaited if statement 😊 123

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

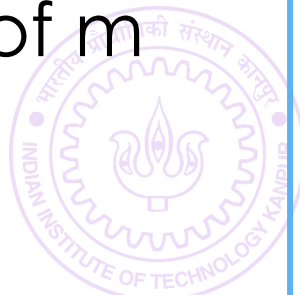
m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0

Now please print value of m

Goodbye



# The much awaited if statement 😊 123

**HOW WE MUST SPEAK TO MR. COMPILER**

**HOW WE USUALLY SPEAK TO A HUMAN**

Do you speak English?

Hello

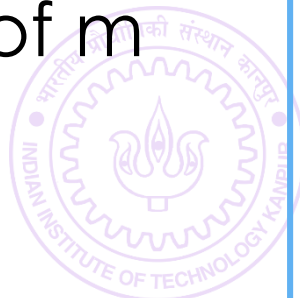
m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0

Now please print value of m

Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

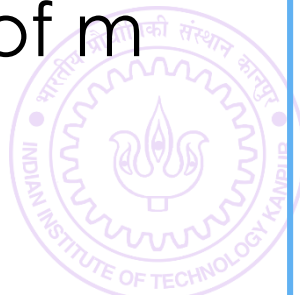
m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0

Now please print value of m

Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0

Now please print value of m

Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

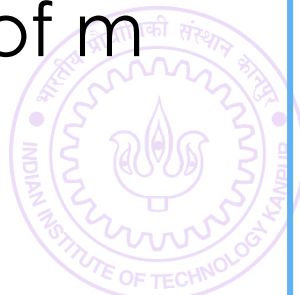
m is an integer variable.

Please ask me for value of m.

If the value of m is less than 10,  
then please print a 0

Now please print value of m

Goodbye





# The much awaited if statement 😊 123

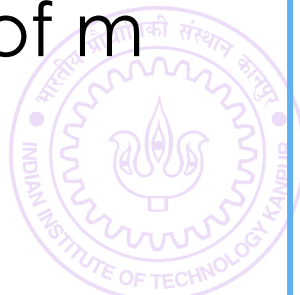
## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>

int main(){
    int m;
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊 123

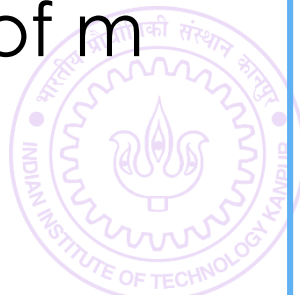
## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>

int main(){
    int m;
    scanf("%d",&m);
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



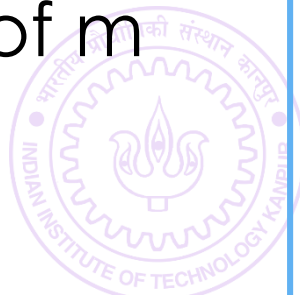
# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



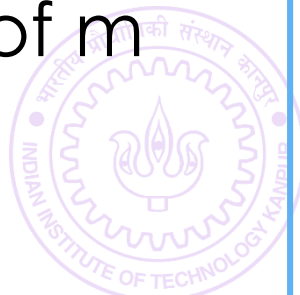
# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



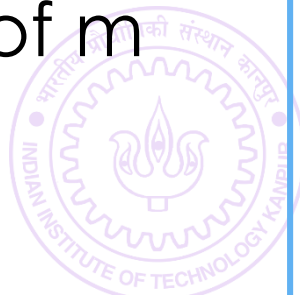
# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
}
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



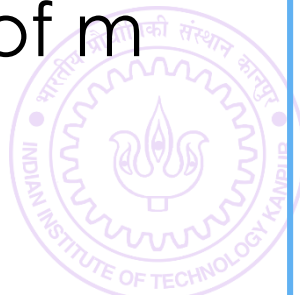
# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



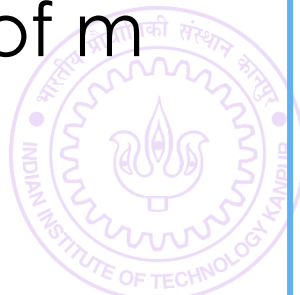
# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



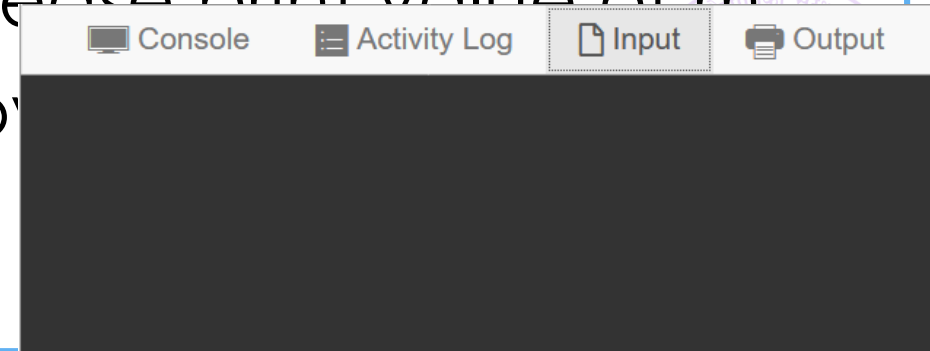
# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye





# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye

Console

Activity Log

Input

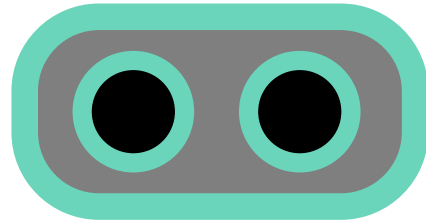
Output

5

# The much awaited if statement 😊123

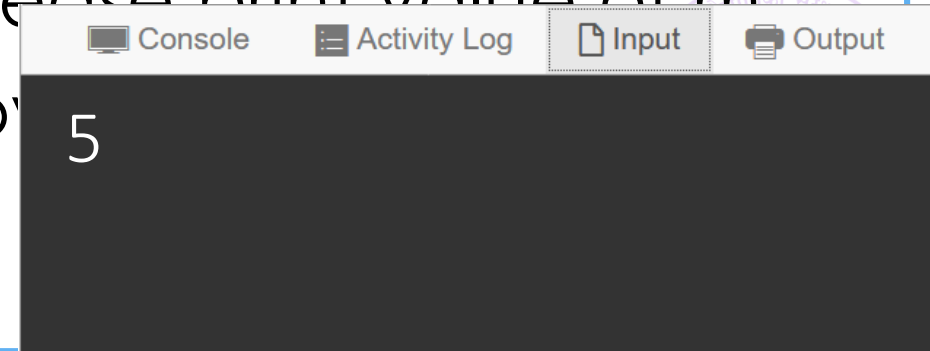
## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



## HOW WE USUALLY SPEAK TO A HUMAN

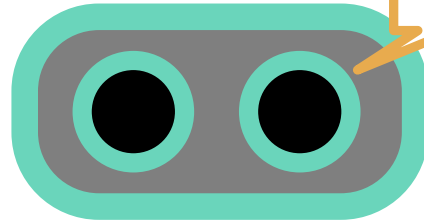
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊123

## HOW WE MUST SPEAK TO MR. COMPILER

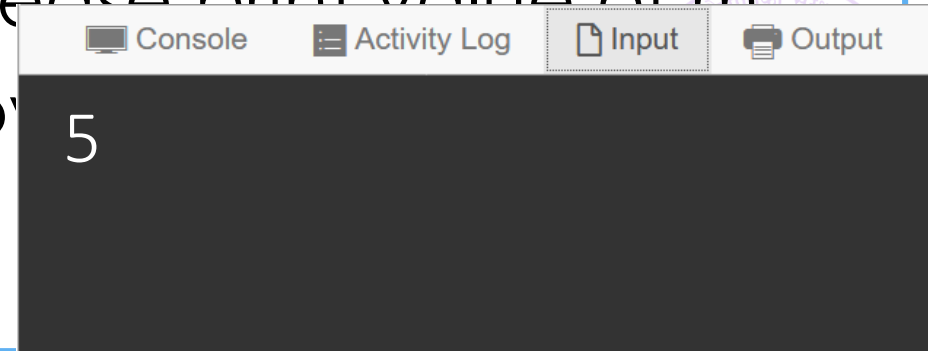
```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



05

## HOW WE USUALLY SPEAK TO A HUMAN

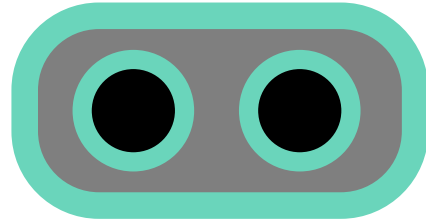
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊123

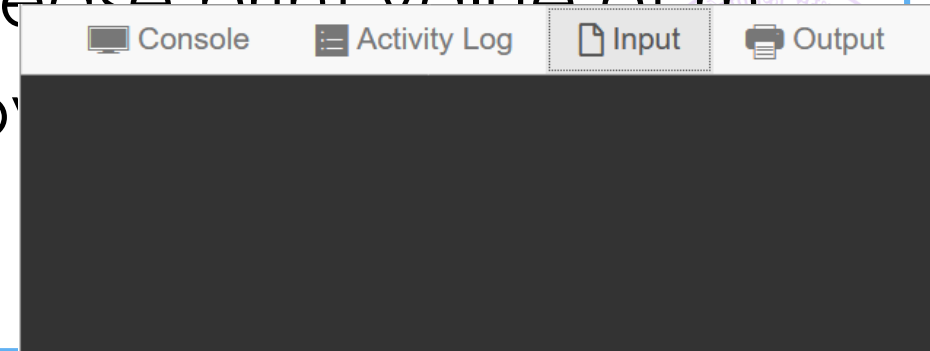
## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



## HOW WE USUALLY SPEAK TO A HUMAN

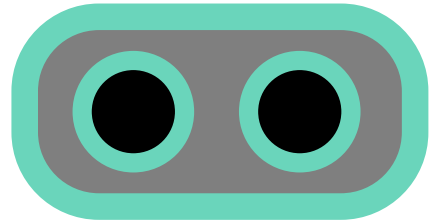
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊123

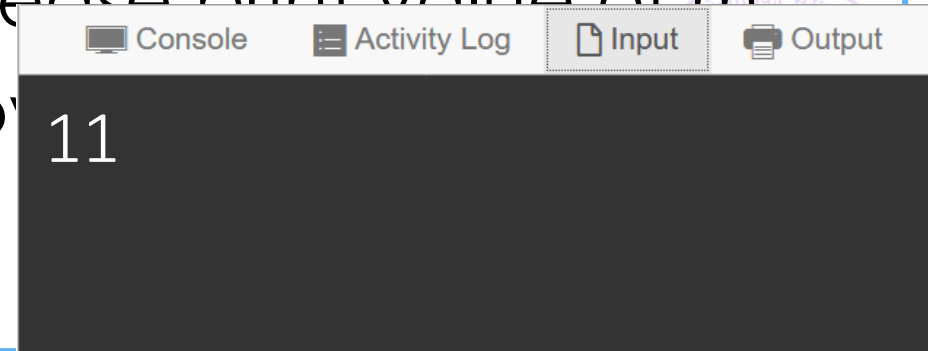
## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



## HOW WE USUALLY SPEAK TO A HUMAN

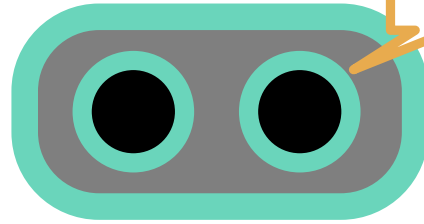
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

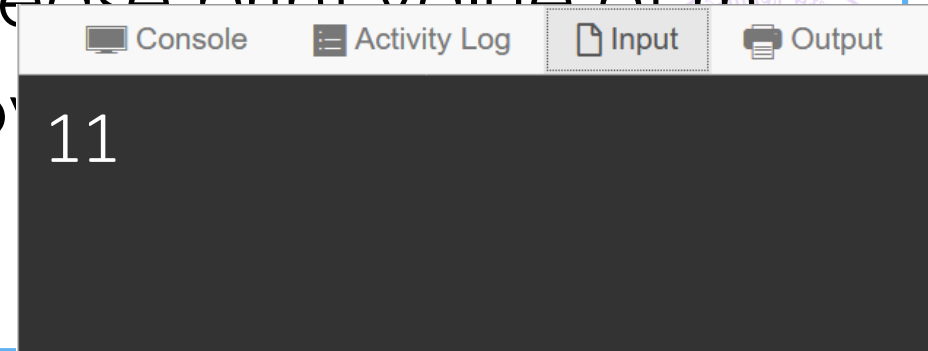
```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



11

## HOW WE USUALLY SPEAK TO A HUMAN

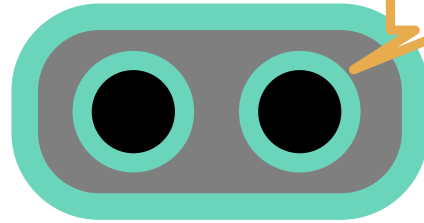
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0")
    }
    printf("%d",m);
}
```

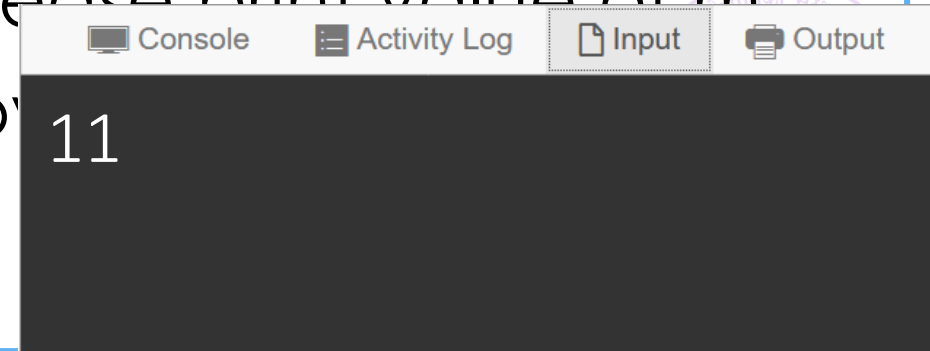


11

This statement is  
always executed  
whether  $m < 10$  or not

## HOW WE USUALLY SPEAK TO A HUMAN

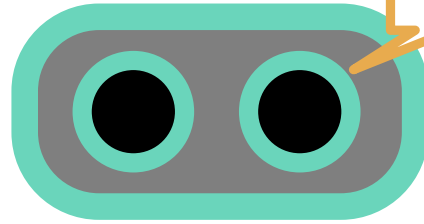
Do you speak English?  
Hello  
 $m$  is an integer variable.  
Please ask me for value of  $m$ .  
If the value of  $m$  is less than 10,  
then please print a 0  
Now please print value of  $m$ .  
Goodbye



# The much awaited if statement 😊123

## HOW WE MUST SPEAK TO MR. COMPILER

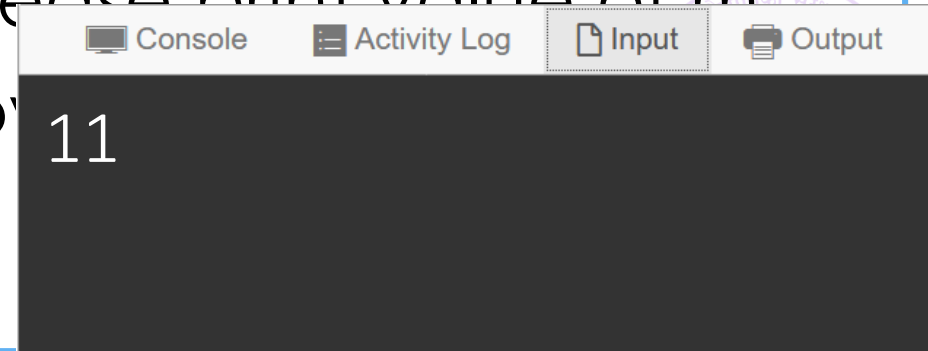
```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



11

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye

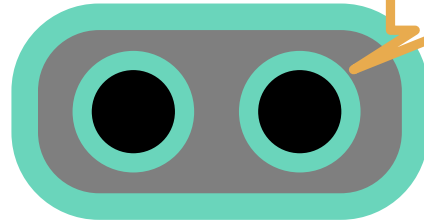




# The much awaited if statement 😊123

## HOW WE MUST SPEAK TO MR. COMPILER

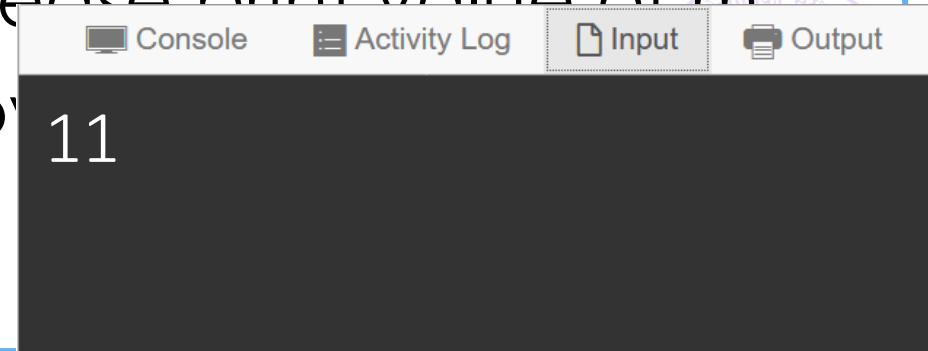
```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



11

## HOW WE USUALLY SPEAK TO A HUMAN

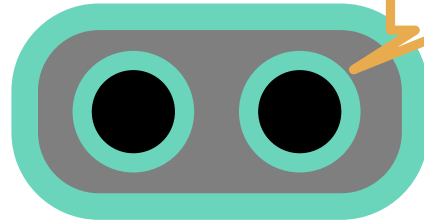
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d", m);
}
```

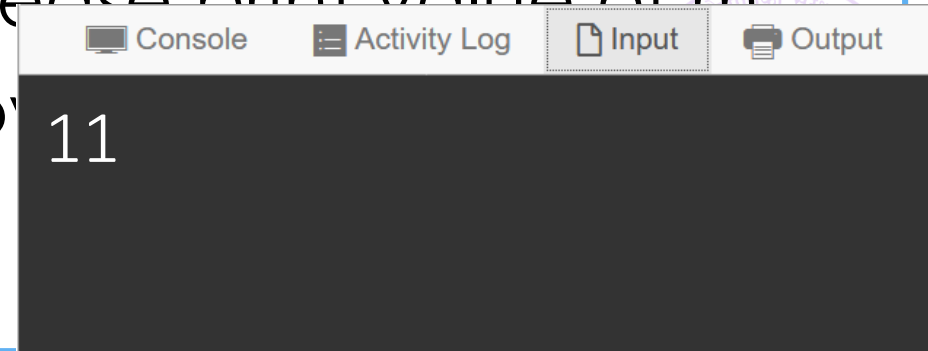


11

These curly brackets are used to tell Mr. C what all we want him to do if  $m < 10$

## HOW WE USUALLY SPEAK TO A HUMAN

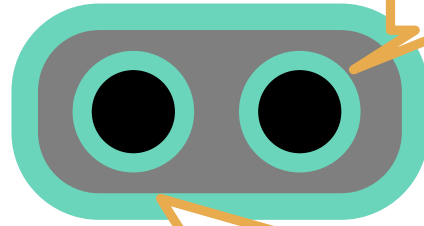
Do you speak English?  
Hello  
m is an integer variable.  
Please ask me for value of m.  
If the value of m is less than 10,  
then please print a 0  
Now please print value of m  
Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



11

Essential in case you want me to do many things if  $m < 10$ , not just print one 0

These curly brackets are used to tell Mr. C what all we want him to do if  $m < 10$

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

m is an integer variable.

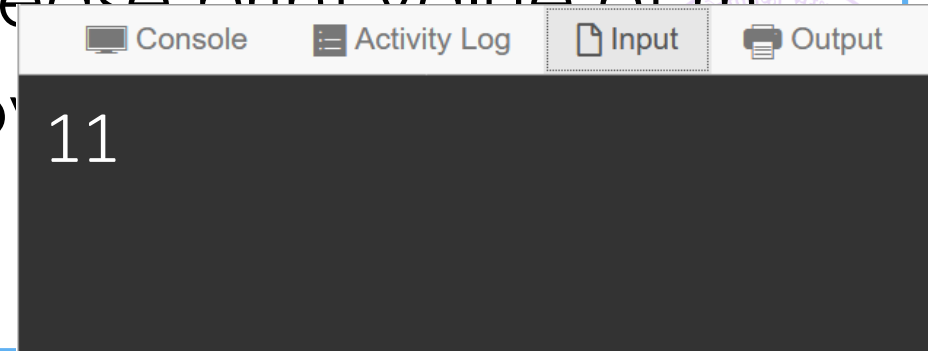
ask me for value of m.

value of m is less than 10,

then please print a 0

Now please print value of m

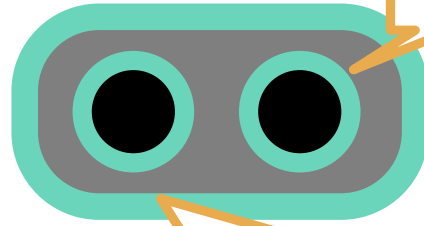
Goodbye



# The much awaited if statement 😊 123

## HOW WE MUST SPEAK TO MR. COMPILER

```
#include <stdio.h>
int main(){
    int m;
    scanf("%d",&m);
    if(m < 10){
        printf("0");
    }
    printf("%d",m);
}
```



11

Essential in case you want me to do many things if  $m < 10$ , not just print one 0

Missing brackets - common mistake

These curly brackets are used to tell Mr. C what all we want him to do if  $m < 10$

## HOW WE USUALLY SPEAK TO A HUMAN

Do you speak English?

Hello

m is an integer variable.

ask me for value of m.

value of m is less than 10,

please print a 0

please print value of m

Goodbye

Console Activity Log Input Output

11

# On Saturday Lecture

157



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$





# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$

What if someone by mistake enters a negative number?



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$

What if someone by mistake enters a negative number?

What if we want something special if  $m = 10$



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$

What if someone by mistake enters a negative number?

What if we want something special if  $m = 10$

Seven days in a week Monday = 1, Sunday = 7



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$

What if someone by mistake enters a negative number?

What if we want something special if  $m = 10$

Seven days in a week Monday = 1, Sunday = 7

Can we print Weekday if input is 1 – 5 and Weekend if input is 6 – 7?



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$

What if someone by mistake enters a negative number?

What if we want something special if  $m = 10$

Seven days in a week Monday = 1, Sunday = 7

Can we print Weekday if input is 1 – 5 and Weekend if input is 6 – 7?

Can we print if the input is a day on which there is an ESC101 lecture?



# On Saturday Lecture

157

We asked Mr. C to do something special if  $m < 10$

We asked him to print an extra 0 in front of the month

Can we ask him to do something special if  $m$  is not  $< 10$

What if someone by mistake enters a negative number?

What if we want something special if  $m = 10$

Seven days in a week Monday = 1, Sunday = 7

Can we print Weekday if input is 1 – 5 and Weekend if input is 6 – 7?

Can we print if the input is a day on which there is an ESC101 lecture?

You are now ready to solve the bonus problem (all of you!)

