Uninitialized Variables

- A variable has no meaningful value until a program gives it one.
- For example, if the variable y has not been given a value, then the following assignment statement is an error.

```
int y ;
y = y+2 ;
```

→ Error

Variable Declaration With Initialization

You can initialize a variable at the time that you declare a variable.

syntax

```
Type_Name variable_Name1= Expression_for_value_1,
          variable_Name2= Expression_for_value_2,..;
```

```
int a = 2, b = 5 ;
```

~~int~~ double price = 100.5 ;

cout

- To display values of the variable as well as strings of text, you use an entity called cout and the << operator (sometimes called the *insertion* operator).
- Notice that strings must be included in double quotes.

Ex 1: What will be printed at the output after execution of the following statement, assuming the value of the variable square_area is 25?

```
int square_area=25;
cout << "The area is:"<<square_area;
```

OUTPUT

```
The area is: 25.
```

Ex 2: What will be printed at the output after execution of the following statement, assuming the value of the variable square_area is 25?

```
int square_area=25;
cout << "The area is:\n"<<square_area;
```

\n → NewLine Character.

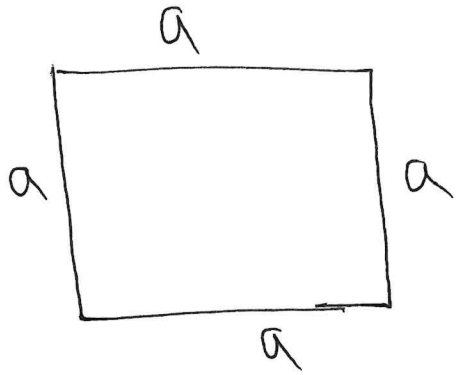
OUTPUT

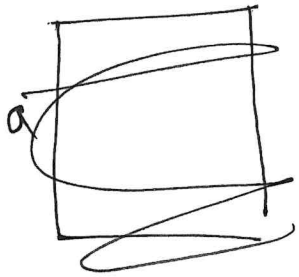
```
The area is:
25.
```

Write a Program to print the area of a square, where length of each side is 5. The equation for area of the square $A = a^2$, where a is the length of each side

$a = 5$

$A = a^2$






```
#include <iostream>

using namespace std;

int main()
{
    int a = 5, area;

    area = a*a;

    cout << "The area is:" << area<<endl;

    return 0;
}
```

5 * 5 = 25.

OUTPUT

```
The area is: 25.
```

\n and endl

* To start a new output line, you can include \n in a quoted string. Alternatively, you can start a new line by outputting endl.

```
cout << "Fall 2020\n";
cout << "EET 293\n";
```

OUTPUT

```
Fall 2026
EET 293
```

```
cout << "Fall 2020"<<endl;
cout << "EET 293"<<endl;
```

OUTPUT

```
Fall 2020
EET 293
```

cin Statements

- A cin statement sets variable equal to values typed in at the keyboard.
- When a program executes the input statement it waits for user to provide input. The user also needs to hit the Enter key so that the program accepts the input.

Extraction operator

```
cin >> Variable_1 >> Variable_2 >>.....;
```

Syntax .

~~cin >> value~~

```
int value;
cin >> Value ;
```

Example2

```
int price;
cout<<"Please enter the values:";

cin>>price;
```

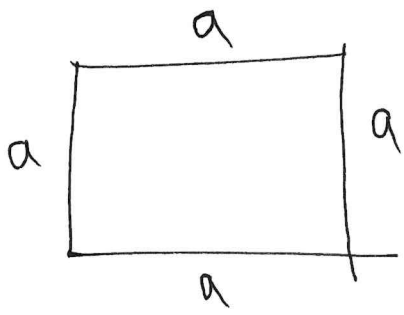
price

125.

OUTPUT .

```
Please enter the values: ~~105~~ 125
```

Write a Program to ask user to enter the length of the side and print the area of a square. The equation for area of the square $A = a^2$, where a is the length of each side



$A = a^2$

```
#include <iostream>

using namespace std;

int main()
{
    int a, area;

    cout<<"Enter the value of the side:";
    cin>>a;

    area= a*a;

    cout<<"The area is:"<<area;

    return 0;
}
```

$6 * 6 = 36$

$a = 6$

OUTPUT

```
Enter the value of the side: 6
The area is: 36
```

Arithmetic Operators

Operation	C++ Symbol	Example (C++ Symbol)	Mathematics Symbol	Example (Mathematics Symbol)
Addition	+	a + b	+	a + b
Subtraction	-	a - b	-	a - b
Multiplication	*	a * b	. or x	a. b or a x b
Division	/	a / b	fraction bar or ÷	$\frac{a}{b}$ or $a \div b$

Mathematical Expression	C++ Functions	Comments
$\sqrt{x}$	`sqrt (x)`	Square root of x
x^n	`pow (x, n)`	Power of x
e^x	`exp (x)`	Exponent of x
| x |	`abs (x)`	Absolute value |x|
sinx	`sin (x)`	Sine of x
cosx	`cos (x)`	Cosine of x

Example: Convert the following Math expression to C++ expression: $\frac{XY}{2}$

$$\frac{XY}{2}$$

$$= (X * Y)/2$$

Example: Convert the following Math expression to C++ expression: $(1 + \frac{r}{100})^n$

$$\left(1 + \frac{r}{100}\right)^n$$

$x^n = pow(x, n)$

$$= pow\left(1 + r/100, n\right)$$

Example: Convert the following Math expression to C++ expression: $\sqrt{a^2 + b^2}$

$$\sqrt{a^2 + b^2}$$

$\sqrt{x} \cong sqrt(x)$

```
sqrt ( a*a + b*b)
sqrt ( pow(a,2) + pow(b,2))
```

Write a program to print the value of y, where $y = \sqrt{a^2 + b^2}$. The value of a and b are 10 and 5, respectively.

$a = 10$, $b = 5$.

```
#include <iostream>
#include <cmath>

using namespace std;

int main()
{
    // Variable Declaration
    int a =10, b=5;
    double y;

    // Calculation of y
    y = sqrt(a*a+b*b);

    // printing the result
    cout << "The value of y is:" << y<<endl;
    return 0;
}
```

$= \sqrt{10^2 + 5^2} = \sqrt{125} = 11.1803$

OUTPUT

```
The Value of Y is: 11.1803
```

Comments

- You should add comment which explains your code. This helps programmers who read your code understand your intent.
- You use the // delimiter for short comments. If you have a longer comment, enclose it between /* and */ delimiters. The compiler ignores these delimiters and everything in between.

```
int r = 5 ;        // radius of the circle is 5 .

/* Name: Md Sayeed
   course: EET- 340
   semester: spring 2023 .
*/

PI = 3.1416
```

Naming Constants

- When a variable is defined with the reserved word const, its value can never change.
- Constants are generally written in capital letters to distinguish them visually from regular variables.

```
const Type_Name Variable_Name = Constant ;
```

Syntax

```
const  double  PI  =  3.1416 ;
```

Write a program to print the area of a circle, where the radius of the circle is 3. The equation of the circle is $A = \pi r^2$, where r is the radius.

```
#include <iostream>

using namespace std;

int main()
{
    // variable declaration
    int r=3;
    const double PI=3.14;
    double area;

    //calculate the area
    area = PI*r*r;

    //printing the area
    cout<<"The area is: "<<area;

    return 0;
}
```

$A = \pi r^2$

PI * r * r .

If it rains today, I will stay home.

Condition — Decision

if Statement

If statement is used to implement decision. When a condition is fulfilled, one set of statements is executed.

```
if (Boolean_Expression)
{
  statements
}
```

syntax .

```
if (rain)
{
   I will stay home.
}
```

Write a program to check a given grade. If the grade value is greater than or equal to 60, program prints "This student passed" on the console window.

```
#include <iostream>

using namespace std;

int main()
{
    int grade = 82;
    if (grade >= 60)
    {
        cout<<"This student passed ";
    }

    return 0;
}
```

82 >= 60 True .

OUTPUT

```
This student passed .
```

Practice Problem 1: Consider the following if statement to compute the discounted price. What is the discounted price if the original price is 90.

```
int originalPrice = 90, discountedPrice;
discountedPrice = originalPrice;

if ( originalPrice > 100 )
{
    discountedPrice = originalPrice -10 ;
}
```

= 90

90 > 10 → False

discounted price = 90

Practice Problem 2: Consider the following if statement to compute the discounted price. What is the discounted price if the original price is 100.

```
int originalPrice = 100, discountedPrice;
discountedPrice = originalPrice;

if ( originalPrice > 100 )
{
    discountedPrice = originalPrice -10 ;
}
```

100

100

100 > 100 → False.

discountedPrice = 100

if –else Statement

If statement is used to implement decision. When a condition is fulfilled, one set of statements is executed. Otherwise, another set of statements is executed.

```
if (Boolean_Expression)
  { statements1  }
else
  { statements2  }
```

Syntax

Write a program to check a given grade. If the grade value is greater than or equal to 60, program prints "This student passed" on the console window. Otherwise, program prints "This student failed" on the console window.

```
#include <iostream>

using namespace std;

int main()
{
    int grade = 52;
    if (grade >= 60)
    {
        cout<<"This student passed ";
    }
    else
    {
    cout<<"This student failed";
    }

    return 0;
}
```

52 >= 60 → False

OUTPUT

```
This student failed
```

Practice Problem 1. Consider the following if statement to compute the discounted price. What is the discounted price if the original price is 95.

```
int originalPrice = 95, discountedPrice;

if ( originalPrice > 100 )
      {  discountedPrice = originalPrice -20 ; }
else
      {  discountedPrice = originalPrice -10 ; }
```

(95 > 100) False

= 95 - 10 = 85

discounted price = 85.

Practice Problem 2. Consider the following if statement to compute the discounted price. What is the discounted price if the original price is 100.

```
int originalPrice = 100, discountedPrice;

if ( originalPrice > 100 )
      {  discountedPrice = originalPrice -20 ; }
else
      {  discountedPrice = originalPrice -10 ; }
```

95 → 100

100 > 100 False

= 100 - 10 = 90

discounted price = 90