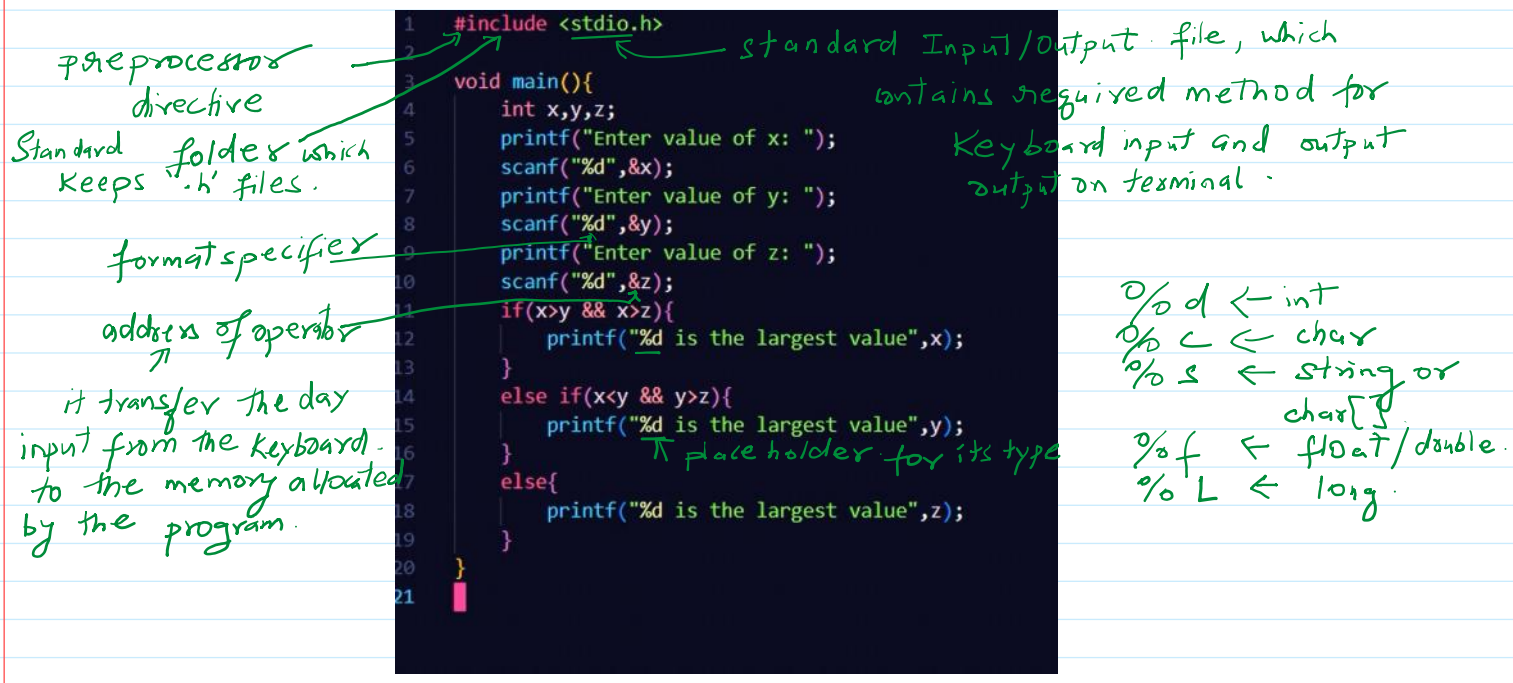
Introduction to C language

09 January 2026 21:00




```c
#include <stdio.h>

void main(){
    int x,y,z;
    printf("Enter value of x: ");
    scanf("%d",&x);
    printf("Enter value of y: ");
    scanf("%d",&y);
    printf("Enter value of z: ");
    scanf("%d",&z);
    if(x>y && x>z){
        printf("%d is the largest value",x);
    }
    else if(x<y && y>z){
        printf("%d is the largest value",y);
    }
    else{
        printf("%d is the largest value",z);
    }
}
```

Common Format Specifiers in C

Specifier	Data Type	Example Usage	Notes
%d or %i	int (signed)	printf("%d", 10);	%i works the same as %d in printf, but differs slightly in scanf for octal/hex input.
%u	unsigned int	printf("%u", 300);	Displays unsigned decimal.
%ld	long int	printf("%ld", 123456L);	For long integers.
%lu	unsigned long int	printf("%lu", 123456UL);	Unsigned long.
%lld	long long int	printf("%lld", 123456789LL);	For very large integers.
%llu	unsigned long long int	printf("%llu", 123456789ULL);	Unsigned long long.
%f	float / double	printf("%f", 3.14);	Default precision is 6 decimal places.
%lf	double	scanf("%lf", &var);	In printf, %f and %lf are the same.
%Lf	long double	printf("%Lf", var);	For extended precision.
%c	char	printf("%c", 'A');	Displays a single character.
%s	char[] (string)	printf("%s", "Hello");	Displays a string until \0.
%p	Pointer address	printf("%p", ptr);	Displays memory address.
%%	Literal %	printf("%%");	Prints % symbol.

Width, Precision, and Flags

Format specifiers can be **modified** to control output formatting:

C

```
printf("%5d", 42);     // Output: "   42" (width 5,
right-aligned)
printf("%-5d", 42);    // Output: "42   " (left-
```

```
aligned)
printf("%07d", 42);    // Output: "0000042" (zero-
padded)
printf("%.2f", 3.14159); // Output: "3.14" (2 decimal
places)
```

Example Program

C

```
#include <stdio.h>
int main() {
    int num = 42;
    float pi = 3.14159;
    char ch = 'A';
    char str[] = "Hello";
   printf("Integer: %d\n", num);
   printf("Float: %.2f\n", pi);
   printf("Character: %c\n", ch);
   printf("String: %s\n", str);
   printf("Pointer: %p\n", (void*)&num);
   printf("Percent sign: %%\n");
   return 0;
   }
```

Homework:
Create a presentation on the data-types of C-programming language