

MNNIT COMPUTER CODING CLUB

**CLASS-10**

**BASICS OF C**



# DYNAMIC MEMORY ALLOCATION

## void pointer

- The void pointer in C is a pointer which is not associated with any data types.
- It is a general-purpose pointer.
- It can point to any data type.

```
int a = 7;
```

```
float b = 7.6;
```

```
void *p;
```

```
p = &a;
```

```
printf("Integer variable is = %d", *( (int*) p) );
```

```
p = &b;
```

```
printf("\nFloat variable is = %f", *( (float*) p) );
```

# DYNAMIC MEMORY ALLOCATION

- **Malloc**


```
ptr = (cast-type*) malloc(byte-size)
```

- **Calloc**

```
ptr = (cast-type*) calloc(n, element-size);
```

## Malloc()

```
int* ptr = (int*) malloc ( 5* sizeof ( int ));
```

ptr =  → A large 20 bytes memory block is dynamically allocated to ptr


← 20 bytes of memory →

4 bytes



## Calloc()

```
int* ptr = (int*) calloc ( 5, sizeof ( int ));
```

ptr =  → 5 blocks of 4 bytes each is dynamically allocated to ptr

← 4b →

← 20 bytes of memory →

4 bytes



# STRUCTURE

A structure is a user-defined data type available in C that allows to combining data items of different kinds. Structures are used to represent a record.

Syntax:

```
struct [structure name]
{
    member definition;
    member definition;
    ...
    member definition;
};
```

```
struct struct_example
{
    int integer;
    float decimal;
    char name[20];
};
```

Creating an object:

```
struct struct_example s={10,10.0,"abcdef"};
```

Access(read/write):

```
s.integer
s.decimal
s.name
```

# UNION

A union is a special data type available in C that allows storing different data types in the same memory location. You can define a union with many members, but only one member can contain a value at any given time.

Syntax:

```
union [union name]
{
    member definition;
    member definition;
    ...
    member definition;
};
```

```
union union_example
{
    int integer;
    float decimal;
    char name[20];
};
```

Creating an object:

```
union union_example u;
```

Access(read/write):

```
u.integer
u.decimal
u.name
```

MNNIT COMPUTER CODING CLUB

**WHAT'S NEXT?**

