str = { Kushaal from Yelahanka }

while (str[i] != '\0') {
  if (str[i] == ' ') {
    words ++;
  }
  i++;
}

Represent an Employee → parameters / properties / attributes

- char[20] 20 → name
- int → 4 → ID
- int 4 → age
- char[20] 20 → address

Emp (48 bytes) → emp

Structures & Unions

User-defined data types:

```
struct Employee {
    char name[20];
    int age;
    int empID;
    char email[20];
};
```

```
union Employee {
    char name[20];
    int age;
    int empID;
    char email[20];
};
```

Struct

Size = 48 (20+20+4+4)

Size = Σ All vars

$V_1 \rightarrow M_1$
$V_2 \rightarrow M_2$
$V_3 \rightarrow M_3$

Union

Size = 20

Size = sizeof(largest)

$V_1, V_2, V_3 \rightarrow M$ (one value)

Student → struct & union (separate files)

- name — c 20
- age — i
- branch — c 20
- USN — c [1PC26CSE096] 50
- email — c 20

Struct | Union

struct { Online Store }

items : book : shirt

(Make it memory efficient)

shirt

```
char brand[20];
char size[10];
char color[10];
int price;
char type[20];
```

casual / formal

book

```
char title[20]
char author[20]
char genre[10]
int price;
char type[20];
```

ebook, hardcover