* Array is a collection of same type of elements

int a[100];


```
struct student
{
    char name[20];
    int roll-no;
    float marks;
}
```

20B ← char name[20];
4B ← int roll-no;
4B ← float marks;

28B

- Declaration of a structure provides one more data type type in addition to the built in data types.
* declaration of a structure template does not reserve any space in m/m for the members;
* space is reserved only when variables of this structure type are declared.

```
struct student
{
};
```

28B 28B 28B

} stu1, stu2, stu3;

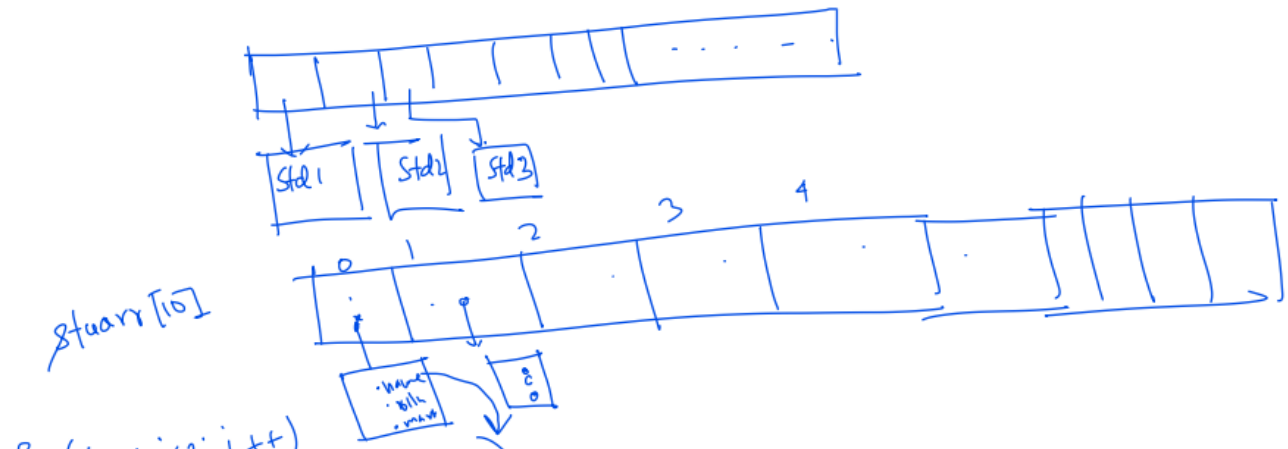
are variables of type struct student

struct student {

};

struct student stu1, stu2;

struct student stu2 = { "John", 24, 67.5 };



for(i=0; i<3; i++)

scanf("...", &stuarr[0].name)

for(i=1 &stuarr[1]

```
struct student
{
  char name[20];
  int rollno;
  int marks;
};
struct student stu, *ptr;
```

ptr is a pointer variable that can point to a variable of type struct student

```
ptr = &stu;
```

You can access the member of a structure by using

```
(*ptr).name
```

or

```
ptr->name
```

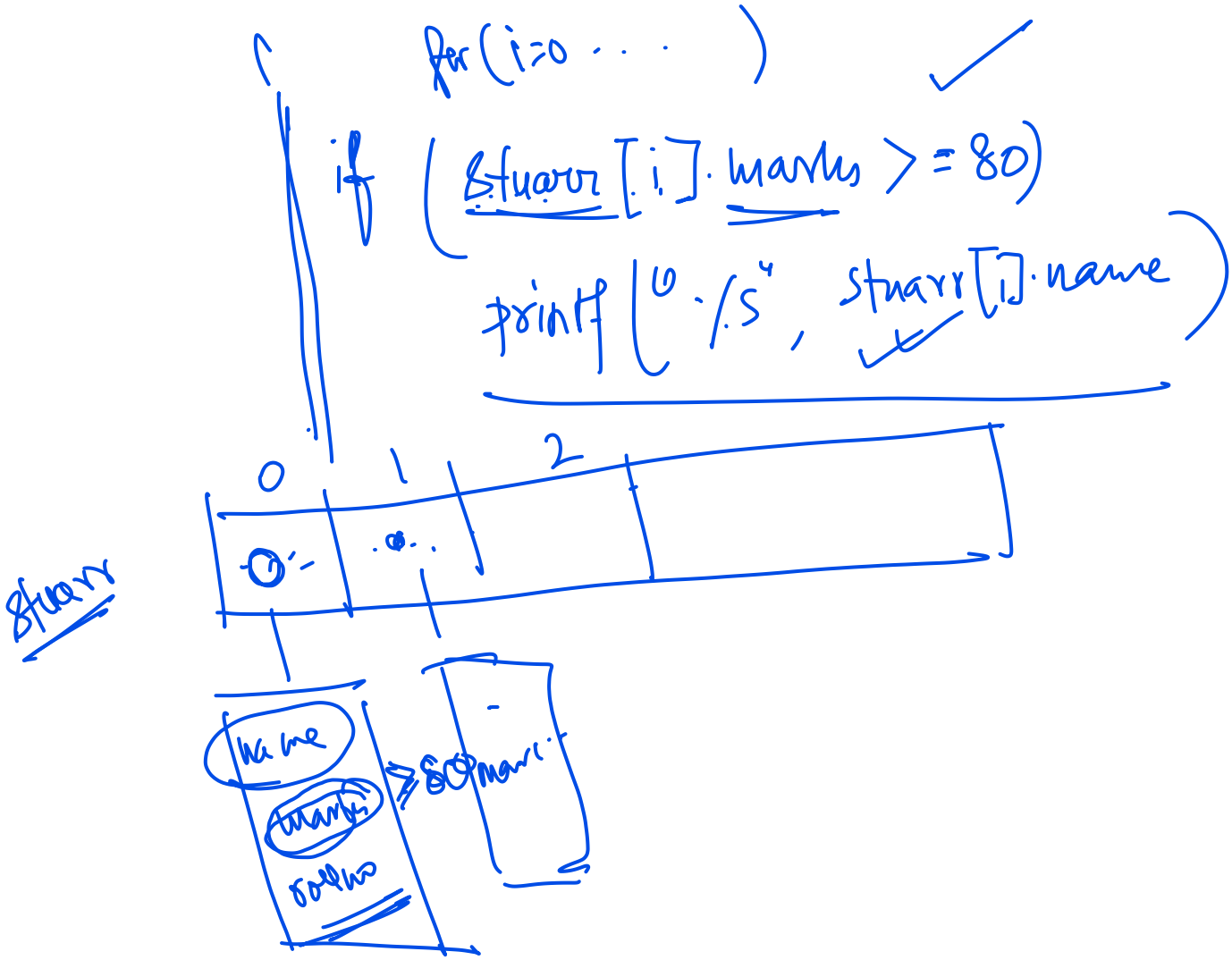
```
struct student
{
    char name[20];
    int rollno;
    struct date
    {
        int day;
        int month;
        int year;
    } birthdate;
    float marks;
} stu1, stu2;
```

```
stu1.birthdate.day;
stu1.birthdate.month;
```

LAB-ASSIGNMENT 6 (Structure)

SUBJECT: Computer Lab. SUBJECT CODE: CS111. SEMESTER: I

1. WAP of structure that reads and displays the information of student.
2. WAP to demonstrate the usage of Array of structure.
3. WAP to prints all students names who gets marks >= 80.
4. WAP to demonstrate the usages of nested structure.
5. WAP to demonstrate the usages of a structure pointer



```c
/*P11.1 Program to display the values of structure members*/
#include<stdio.h>
#include<string.h>
struct student
{
    char name[20];
      int rollno;
      float marks;
};
int main(void)
{
      struct student stu1={};
      struct student stu2,stu3;
      strcpy(stu2.name,"John");
      stu2.rollno=26;
      stu2.marks=98;
      printf("Enter name, rollno and marks for stu3 : ");
      scanf("%s %d %f", stu3.name, &stu3.rollno, &stu3.marks);
      printf("stu1 : %s  %d  %.2f\n",stu1.name,stu1.rollno,stu1.marks);
      printf("stu2 : %s  %d  %.2f\n",stu2.name,stu2.rollno,stu2.marks);
      printf("stu3 : %s  %d  %.2f\n",stu3.name,stu3.rollno,stu3.marks);
      return 0;
}
```

```c
/*P11.2 Program to assign a structure variable to another structure
variable*/
#include<stdio.h>
struct student
{
    char name[20];
      int rollno;
      float marks;
};
int main(void)
{
      struct student stu1={"Oliver"};
      struct student stu2;
      stu2 = stu1;
      printf("stu1:%s  %d  %.2f\n",stu1.name,stu1.rollno,stu1.marks);
      printf("stu2:%s  %d  %.2f\n",stu2.name,stu2.rollno,stu2.marks);
      printf("\n %d",sizeof(stu2));
      return 0;
}
```

```
/*P11.3 Array of structures*/
#include<stdio.h>
struct student
{
    char name[20];
      int rollno;
      float marks;
};
int main(void)
{
      int i;
      struct student stuarr[10];
      for(i=0; i<3; i++)
      {
            printf("Enter name,rollno and marks : ");

      scanf("%s%d%f",stuarr[i].name,&stuarr[i].rollno,&stuarr[i].marks);
      }
      for(i=0; i<3; i++)
            printf("%s %d
%f\n",stuarr[i].name,stuarr[i].rollno,stuarr[i].marks);
      return 0;
}
```

```
/*P11.4 Program to understand arrays within structures*/
#include<stdio.h>
struct student
{
    char name[20];
      int rollno;
      int submarks[4];
};
int main(void)
{
      int i,j;
      struct student stuarr[3];
      for(i=0; i<3; i++)
      {
            printf("Enter data for student %d\n", i+1);
            printf("Enter name : ");
            scanf("%s", stuarr[i].name);
            printf("Enter roll number : ");
            scanf("%d",&stuarr[i].rollno);
            for(j=0; j<4; j++)
            {
                  printf("Enter marks for subject %d : ",j+1);
                  scanf("%d", &stuarr[i].submarks[j]);
            }
      }
      for(i=0; i<3; i++)
      {
            printf("Data of student %d\n", i+1);
            printf("Name:%s, Roll
number:%d\nMarks:",stuarr[i].name,stuarr[i].rollno);
            for(j=0; j<4; j++)
                  printf("%d ",stuarr[i].submarks[j]);
            printf("\n");
      }
      return 0;
}
```

```c
/*P11.5 Program to understand pointers to structures*/
#include<stdio.h>
struct student
{
    char name[20];
      int rollno;
      int marks;
};
int main(void)
{
      struct student stu = {"Mary", 25, 68};
      struct student *ptr = &stu;
      printf("Name  - %s\t", (*ptr).name);
      printf("Rollno  - %d\t", ptr->rollno);
      printf("Marks  - %d\n", ptr->marks);
      return 0;
}
```

```
/*P11.6 Program to understand how structure members are sent to a
function */
#include<stdio.h>
#include<string.h>
struct student
{
    char name[20];
      int rollno;
      int marks;
};
void display(char name[],int rollno,int marks);
int main(void)
{
      struct student stu1={"John",12,87};
      struct student stu2;
      strcpy(stu2.name,"Mary");
      stu2.rollno=18;
      stu2.marks=90;
      display(stu1.name,stu1.rollno,stu1.marks);
      display(stu2.name,stu2.rollno,stu2.marks);
      return 0;
}
void display(char name[],int rollno,int marks)
{
      printf("Name   - %s\t",name);
      printf("Rollno - %d\t",rollno);
      printf("Marks  - %d\n",marks);
}
```

```
/*P11.7 Program to understand how a structure variable is sent to a
function*/
#include<stdio.h>
struct student
{
    char name[20];
      int rollno;
      int marks;
};
void display(struct student);
int main(void)
{
      struct student stu1={"John",12,87};
      struct student stu2={"Mary",18,90};
      display(stu1);
      display(stu2);
      return 0;
}
void display(struct student stu)
{
      printf("Name   - %s\t",stu.name);
      printf("Rollno - %d\t",stu.rollno);
      printf("Marks  - %d\n",stu.marks);
}
```

```
/*P11.8 Program to understand how a pointer to structure variable is sent
to a function */
#include<stdio.h>
struct student
{
    char name[20];
    int rollno;
    int marks;
};
void display(struct student *);
void inc_marks(struct student *);
int main(void)
{
      struct student stu1 = {"John",12,87};
      struct student stu2 = {"Mary",18,90};
      inc_marks(&stu1);
      inc_marks(&stu2);
      display(&stu1);
      display(&stu2);
      return 0;
}
void inc_marks(struct student *stuptr)
{
      (stuptr->marks)++;
}
void display(struct student *stuptr)
{
      printf("Name   - %s\t",stuptr->name);
      printf("Rollno - %d\t",stuptr->rollno);
      printf("Marks  - %d\n",stuptr->marks);
}
```