

# مبانی برنامه نویسی

مدرس: سعدون عزیزی

[s.azizi@uok.ac.ir](mailto:s.azizi@uok.ac.ir)

مرکز آموزش های الکترونیکی

تابستان ۹۶

## سرفصل مطالب

- آشنایی با کامپیوتر و الگوریتم
- مقدمه‌ای بر برنامه‌نویسی C
- محاسبات
- ورودی/خروجی
- حلقه‌ها
- دستورات شرطی
- توابع
- آرایه‌ها
- کاراکترها و رشته‌ها
- اشاره‌گرها
- ساختار
- فایل‌ها

## مقدمه

- ساختار یک نوع داده مرکب است که شامل مجموعه ای از چند نوع داده است
- به عنوان مثال، اطلاعات دانشجوی یک واحد اطلاعاتی مرکب است که شامل داده هایی نظیر نام، نام خانوادگی، شماره دانشجویی، معدل و ... می باشد

# تعريف ساختار

```
struct tag
{
    تعريف اعضا
};
```

مثال: □

```
struct student
{
    char first_name[20], last_name[20];
    int id;
    float ave;
};
```

## استفاده از ساختار به عنوان یک نوع داده

```
struct date
```

```
{  
    int year, month, day;  
};
```

```
struct student
```

```
{  
    char first_name[20], last_name[20];  
    int id;  
    struct date birth_date;  
    float ave;  
};
```

## معرفی متغیر از نوع ساختار

```
#include <stdio.h>

struct student
{
    char first_name[20], last_name[20];
    int id;
    float ave;
};

main()
{
    struct student st;
}
```

## مقداردهی اولیه

```
#include <stdio.h>

struct student
{
    char first_name[20], last_name[20];
    int id;
    float ave;
};

main()
{
    struct student st = {"Sadoon", "Azizi", 88112003, 18.5};
}
```

## دسترسی به اعضای ساختار

□ برای دسترسی به اعضای ساختار به صورت زیر عمل می کنیم:

عضو مورد نظر . نام متغیر

علامت . بیانگر عملگر عضو است.

مثال در محیط ++C-Dev



## کپی کردن و مقایسه ساختارها

□ به راحتی می توان توسط دستور زیر از یک ساختار را در ساختار دیگری کپی کرد:

```
struct date today, holiday;
```

```
today = holiday;
```

□ ولی نمی توان دو ساختار را به طور مستقیم با یکدیگر مقایسه کرد.

```
if (today == holiday) // خطا در مقایسه
```

```
printf("***");
```

برای این منظور، باید تک تک اعضای دو ساختار را دو به دو با هم مقایسه کرد.

# ساختار و آرایه

□ عناصر یک آرایه می توانند از نوع ساختار باشند.

```
struct student
{
    char fname[20],lname[20];
    int id;
    float ave;
};
```

```
struct student st[20];
```

## اشاره گر به ساختار

```
struct student
```

```
{
```

```
    char fname[20], lname[20];
```

```
    int id;
```

```
    float ave;
```

```
};
```

```
struct student st={"Sadoon", "Azizi", 88112003, 18.50}, *p;
```

```
p=&st;
```

## ساختار و تابع

- ساختارها هم مانند سایر متغیرها این قابلیت را دارد که به صورت پارامتر به یک تابع ارسال گردند و یا به عنوان جواب تابع به محل فراخوانی بازگردانده شوند.
- یک ساختار را می توان هم با استفاده از فراخوانی با مقدار و هم فراخوانی با ارجاع در اختیار تابع قرار داد.

مثال در محیط ++C-Dev

## ساختار خودارجاع

□ اگر عضوی از ساختار از نوع اشاره گر به خود ساختار باشد به آن ساختار خود ارجاع می گویند

```
struct book
{
    char title[50];
    struct book *next;
};
```

مثال در محیط Dev-C++