

: *enum*

enum ها در واقع اعضای غیر عددی مجموعه ها هستند. مثلاً در مورد اطلاعات یک دانشجو، مهمان

بودن یا نبودن یک داده غیر عددی است که در مجموعه دو عضوی قرار دارد.

یا در مورد موجودات خصوصیات فناپذیری یک داده غیر عددی است اینگونه داده ها توسط نوع

enum شبیه سازی می شوند مثلاً

type

```
StudentType = ( stGuest , stMember );
```

```
Mortality = ( Mortal , lmmortal );
```

این روش مشابه نسبت دادن عدد به هر عضو غیر عددی یک مجموعه است با این تفاوت که اعداد

مناسبی به هر عضو نسبت می دهد. در بسیاری موارد با تعریف ثوابت همین عمل را شبیه سازی می کنند با

کار بیشتر در زمینه نوشتن برنامه با مزایای استفاده از هر کدام از روشها آشنا خواهید شد.

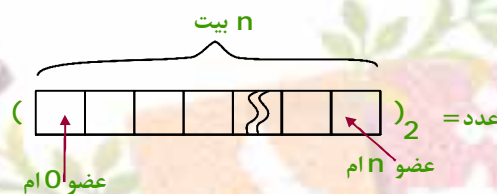
نا گفته نماند انتخاب عدد مناسب منظور انتخاب عددی است که بتوان عضویت در یک مجموعه را با

آن شبیه سازی کرد. برای این کار روش زیر استفاده می شود:

یک مجموعه یک عدد n بیتی است که حضور یا عدم حضور عضو خاص در آن یک یا صفر شدن بیت

مربوطه در عدد مذکور است. این مشابه روشی است که برای محاسبه تعداد زیر مجموعه های مجموعه

استفاده می کردید.



لذا نسبت دادن عدد مناسب نسبت دادن عدد بصورت توانی از 2 است.

پس می توان دو عبارت بالا را با ثابت های زیر شبیه سازی کرد.

```
const                               نشانه عدد مبنای 16
                                   ↓
stGuest : Cardinal = $00000001;
stMember : Cardinal = $00000002;
```

enum ها در تعریف مجموعه کاربرد دارند تابع های *low* و *high* برای یافتن اولین و آخرین

اعضای *enum* استفاده می شوند.

