* Bit Field:

A structure may contain fields whose length is specified as a number of bits.

* + Syntax of declaring a bit field structurevariable:

**Data\_type field\_name:bits\_number;**

Note1: The bit field can be used just like any other structure.

Note2: Most compilers support the int, char, short, long data types for a bit field.

* + Example:

struct Person{

unsigned char sex:1;

unsigned char married: 1;

unsigned char children:4;

char name[20];

};

Note1: Since the size of the sex and married fields is one bit,

their values can be either 0 or 1.

Note2: If the size of a bit is one bit, then its type must be def

ined as **unsigned** since a single bit can’t be signed.

Note3: Since the size of the children is four bits, it can take

values between 0 and 15.

Note4: Since the type of the bit fields is unsigned char, the

compiler allocates one byte. Actually 1+1+4 = 6 bits

will be used to store their values, while the other two

won’t be used.

* + The main advantage of using bit fields is to save memory space. For better memory saving, **declare all bit fields at the beginning of the structures**, not among the rest fields.
  + Example:

#include<stdio.h>

#include<iostream>

#include<string.h>

using namespace std;

struct Person{

unsigned char sex:1;

unsigned char married: 1;

unsigned char estate:6;

char name[20];

} p;

struct Person1{

unsigned int sex:1;

unsigned int married: 1;

unsigned int salary:30;

char name[20];

} p1;

int main(){

cout<<sizeof(Person)<<endl; //21

cout<<sizeof(Person1)<<endl; //24

p.sex=0;

p.married=0;

p.estate=5;

strcpy(p.name,"Mary");

printf("p.sex= %d\n",p.sex); /p.sex=0

cout<<"p.sex= "<<p.sex<<endl; //p.sex=

printf("p.married= %d\n",p.married);

printf("p.estate= %d\n\n",p.estate);

p1.sex=1;

p1.married=1;

p1.salary=100000000;

strcpy(p1.name,"John");

cout<<"p1.sex= "<<p1.sex<<endl;

cout<<"p1.married= "<<p1.married<<endl;

cout<<"p1.salary= "<<p1.salary<<endl;

cout<<p1.name<<endl;

/\* cin>>p1.salary;

cout<<"p1.salary= "<<p1.salary<<endl;

scanf(" %d",&(p.salary));

cout<<"p1.salary= "<<p1.salary<<endl;\*/

int x;

cout<<"Please enter salary:";

cin>>x;

p1.salary=x;

cout<<"p1.salary= "<<p1.salary<<endl;

return 0;

}

