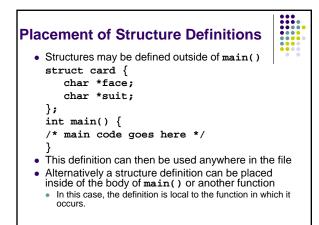
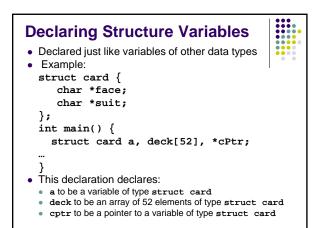


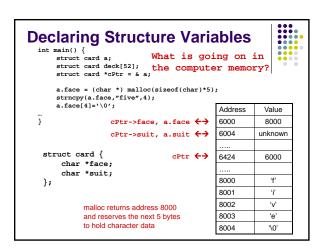
More About Structure Members

- Structure members can be variables of:
 Basic data types such as int, float, char
 - Arrays
 - Other structures (other than itself)
- Structure member can not be
 - variables of the same structure type
 - but could be pointers to the same type
- Important: A structure definition does not declare any variables or reserve any space in memory
- It creates a new *data type* that can, in turn, be used to used to declare variables

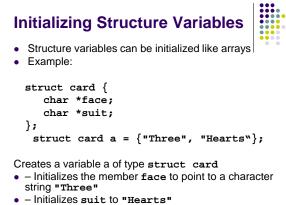




int main() { struct car struct car	Structure Vari d a; What is go d deck[52] the comput d *cPtr = & a;	ing on	in
}		Address	Value
struct card {	cPtr->face, a.face 🔶	6000	unknown
	cPtr->suit, a.suit ↔	6004	unknown
	deck[0].face ↔	6008	unknown
	deck[0].suit ↔	6012	unknown
	deck[1].face \leftarrow	6016	unknown
	deck[1].suit 🔶	6020	unknown
char *face;			
<pre>char *suit; };</pre>	deck[51].face ↔	6416	unknown
	deck[51].suit ↔	6420	unknown
	cPtr ↔	6424	6000
		6428	



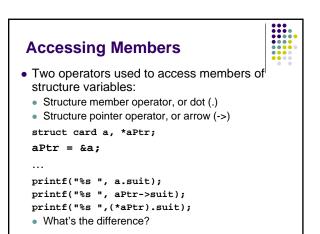


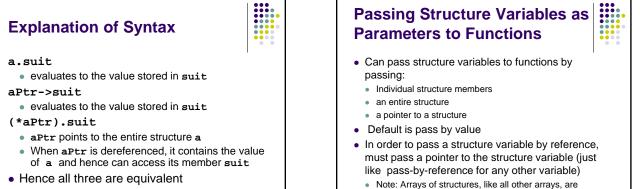


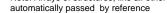
Structure Variable Initialization--Continued



- If there are fewer elements in the initializer list than members
 - Numerics members are initialized to 0
 - Pointers are initialized to NULL
- A structure variable may also be initialized by:
 - Assigning a structure variable of the same type: struct card b = a; /* copies all members of a to b*/
 - Assigning values to the individual members of the structure

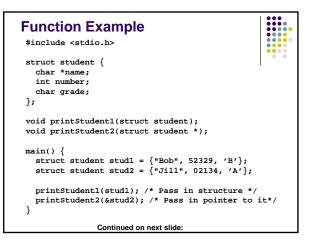


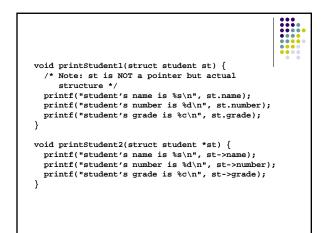




Function Example

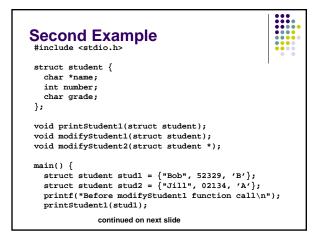
- Prints structure variable two ways
 - Passing the entire structure variable to a function (by value)
 - Passing a pointer to the structure variable to a functon
- One is call by value and the other is call by reference
- Note: Passing a structure as a parameter is different than passing an array
 - default is pass-by value for a structure variable

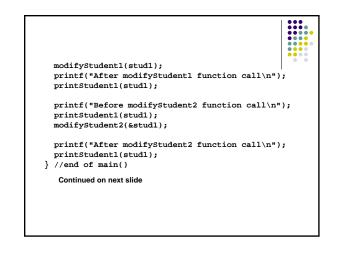


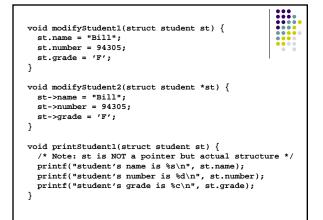


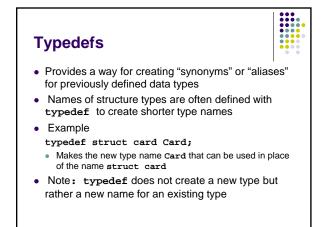
Another Function Example

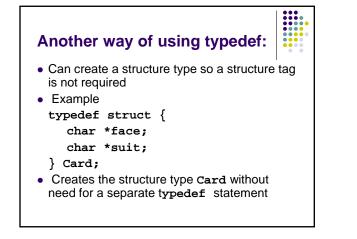
- Illustrates the difference between:
 - Passing structure variables by value
 - Passing structure variables by reference
- Prints structure variable before and after calls to various modifystuden() functions











Using a typedef to Declare Variables



- Now we can use the typedef Card to declare variables of type struct card
- Example:

Card deck[52]; /* Creates an array of 52 card structures*/



 Meaningful names help make programs self documenting

- Often typedef is used to create synonyms for the basic data types, too
- Example
 - typedef *char charPointer;
 - Creates new name for type *char