**Union assignments**

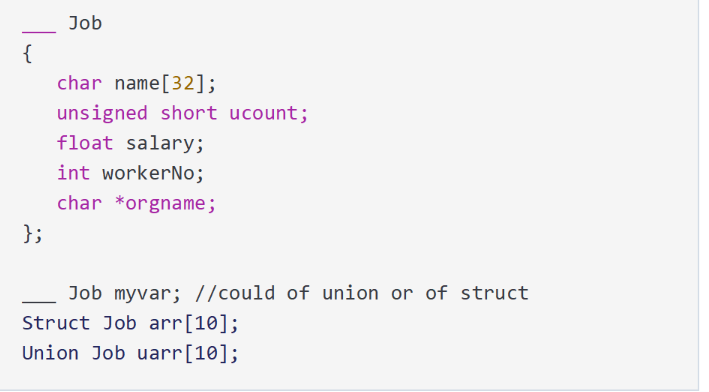
Refer the code below and comment on size of the given structure considering

a. Structure as union

b. Structure as struct

c. arr

d. uarr



**A. Structure as Union**

* The size of the union is determined by the largest member, which is name[32].
* Size of Job union: 32 bytes (on both 32-bit and 64-bit systems).
* Size of arr[10] (Array of unions): 10 \* 32 = 320 bytes.

**B. Structure as Struct**

* In a struct, each member has its own memory, and padding is added for alignment.
* Size of Job struct: 32 (name) + 2 (ucount) + 4 (salary) + 4 (workerNo) + 4 (orgname) = 46 bytes, padded to 48 bytes.
* Size of uarr[10] (Array of structs): 10 \* 48 = 480 bytes.

**C. Array of Structs (arr)**

* The array arr is an array of 10 Job structs.
* Size of arr[10]: 10 \* sizeof(Job) = 10 \* 48 = 480 bytes.

**D. Array of Unions (uarr)**

* The array uarr is an array of 10 Job unions.
* Size of uarr[10]: 10 \* sizeof(Job) = 10 \* 32 = 320 bytes.

**2.) Refer Job datastructure in Q#1 above. Using uarr, perform below operations.**

a. Read and store salary

b. Read and store workerNo

Comment on values of output if salary and workerNo are printed in order. Justify your statement.

A screenshot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

In a **union**, all members use the same memory space, so storing a value in one member will overwrite any previously stored values in other members. Therefore, only the last stored value is valid, and any attempt to access other members will yield incorrect or garbage values.

3.)**Refer Job datastructure in Q#1 above. Assume that myvar is a structure variable. If I need to place 2 bytes (i.e 0x0102) as ucount using a char \*ptr then list all possible statements that can be used in \_\_\_\_\_.**

[Let solutions include cases such as

i. using base address of ucount

ii. using relative address of ucount w.r.t to base address of myvar]

A screenshot of a computer program

Description automatically generated

A black background with white text

Description automatically generated

**Using the base address of ucount:**

* By casting the address of ucount to a char \*, we can directly access and modify the individual bytes of ucount.

**Using the relative address of ucount (from the base address of myvar):**

* By calculating the address of ucount relative to the base address of myvar (after the name field), we can again access and modify the bytes of ucount.