**Union assignments**

**Mandatory**

**1. 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**

**\_\_\_ Job**

**{**

**char name[32];**

**unsigned short ucount;**

**float salary;**

**int workerNo;**

**char \*orgname;**

**};**

**\_\_\_ Job myvar; //could of union or of struct**

**Struct Job arr[10];**

**Union Job uarr[10];**

Char--32 bytes

unsigned short--2 bytes

float--4 bytes

int --4 bytes

char \*--4 bytes

Total :46 bytes

a. structure as union: 32 bytes

b. structure as struct :46 bytes

c. arr : 10\*46 bytes=460

d. uarr: 10\*32 bytes=320

**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.**

**In unions only 32 bytes are created for the structure. When workerNo is given the salary is overridden so we do not get the salary output**

**A screen shot of a computer program

Description automatically generated**

**A screen shot of a computer

Description automatically generated**

**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]**

**int main()**

**{**

**char \*ptr = &myvar;**

**\_\_\_\_\_\_\_\_\_\_\_ = 0x01;**

**\_\_\_\_\_\_\_\_\_\_\_ = 0x02**

**}**

i) \*ptr =0x01;

\*(ptr+1)= 0x02

ii) As ucount is at 32 we direct