1. Which of the following reduces the size of a structure?

2. What will be output of c code?
```c
#include<stdio.h>
int main()
{
    struct bitfield
    {
        signed int a:3;
        unsigned int b:13;
        unsigned int c:1;
    }; struct bitfield bit1={2,14,1};

    printf("%d",sizeof(bit1));
    return 0;
}
```

3. what is the ouput of following program
```c
#include<stdio.h>
int main()
{
        struct temp
            {
                int a : 13;
                int b : 8;
                int c ;
            }s;
    printf("%d", sizeof(s));
    return 0;
}
```

4. what is the output of following program
```c
void main()
{
struct student
{
int no;
char name[20];
};
struct student s;
s.no = 8;
printf("%d", s.no);
}
```

5. Members of a union are accessed as_____.

6. What would be the size of the following union declaration?
union uTemp
{
double a;
int b[10];
char c;
}u;
(Assuming size of double = 8, size of int = 4, size of char = 1)

7. What will be output of c code?
```c
#include<stdio.h>
int main() {
    struct bitfield {
        unsigned a:5;
        unsigned c:5;
        unsigned b:6;
    }bit;

    char *p;
    struct bitfield *ptr,bit1={1,3,3};

    p=&bit1;
    p++;

    printf("%d",*p);
    return 0;
}
```

8. what is the ouput of following program
```c
#include <stdio.h>
int main()
{
    union values
    {
        unsigned char a;
        unsigned char b;
        unsigned int  c;
    };

    union values val;
    val.a=1;
    val.b=2;
    val.c=300;
```

```c
        printf("%d,%d,%d",val.a,val.b,val.c);
        return 0;
}
```

9. what is the ouput of following program
```c
#include <stdio.h>
    struct S {
        int Var;
        struct S *Str;
                };
    int main(void)
{
        struct S S[] = { { 8, NULL }, { 4, &S[0] }, { 2, &S[1] } };
        printf("%d", S[2].Str->Str->Var);
        return 0;
    }
```

10. What is the output of C program with structures.?
```c
int main()
{
    struct tree
    {
        int h;
        int w;
    };
    struct tree tree1={10};
    printf("%d ",tree1.w);
    printf("%d",tree1.h);
    return 0;
}
```