

1.

```
#include<stdio.h>
struct emp
{
    int age;
    struct emp *ptr;
};
int main(void)
{
    struct emp var={40,NULL};
    struct emp *ptr = &var;

    ptr->ptr = ptr;
    printf("%d %d %d",ptr->ptr->age,(*ptr).ptr->age, var.ptr->age);
    return 0;
}
```

- A. 40 40 40
- B. 40 NULL 40
- C. 0 0 0
- D. Garbage Garbage Garbage
- E. Error

Answer: A

2.

```
typedef struct date
{
    short int dd;
    short int mm;
    short int yy;
    typedef struct time
    {
        short int hr;
        short int min;
        short int sec;
    }TIME;
}DATE;
```

```
int main( void )
{
    DATE d1 = {16,8,2023,9,0,0};
    const struct date d2=d1;

    printf("date %d-%d-%d @ %d:%d:%d am\n",d2.dd,d2.mm, d2.yy,
        d2.TIME.hr, d2.TIME.min, d2.TIME.sec );

    return 0;
}
```

- A. date 16-8-2023 @ 9:0:0 am
- B. date 16-8-2023 @ 9:0:0 pm
- C. Compile time error
- D. Run time error
- E. date garbage values @ garbage values

Answer: C

3.

```
#include<stdio.h>
struct demo
{
    signed d1:2;
    unsigned d2:3;
    signed d3:4;
};

int main(void)
{
    struct demo ds;

    ds.d1 = 1;ds.d2 = 3; ds.d3=9;
    printf("%d %d %d", ds.d1,ds.d2,ds.d3);

    ds.d1 = 2;ds.d2 = 7;ds.d3=10;
    printf(" %d %d %d", ds.d1,ds.d2,ds.d3);

    return 0;
}
```

- A. 1 3 9 2 7 10
- B. 1 3 -7 -2 7 -6
- C. Error
- D. 1 3 2 -2 7 3
- E. 1 3 2 2 7 3

Answer: B

4.

```
#include<stdio.h>
```

```
typedef struct demo
```

```
{
```

```
    unsigned bit1:1;
```

```
    signed bit2:2;
```

```
    signed bit3:3;
```

```
    unsigned bit4:2;
```

```
}DEMO;
```

```
int main(void)
```

```
{
```

```
    DEMO d1;
```

```
    struct demo d2;
```

```
    printf("\n %d %d",sizeof(d1),sizeof(d2));
```

```
    return 0;
```

```
}
```

- A. 1 1
- B. 2 2
- C. 8 8
- D. 10 10
- E. 4 4

Answer: E

5.

```
#include<stdio.h>
struct Demo
{
    int num1:4;
    unsigned int num2:3;
    int num3:2;
};
int main(void)
{
    struct Demo d1;

    d1.num1=5;
    d1.num2=6;
    d1.num3=-1;
    printf("%d %d %d",d1.num1,d1.num2,d1.num3);

    printf(" ");
    d1.num1=10;
    d1.num2=9;
    d1.num3=3;

    printf("%d %d %d",d1.num1,d1.num2,d1.num3);

    return 0;
}
```

- A. 5 6 -1 -6 1 -1
- B. 5 6 0 -6 1 0
- C. 5 6 -1 -6 0 0
- D. 5 6 -1 -6 -6 -1
- E. 5 6 -1 -5 1 -1

Answer: A