

1.

```
#include <stdio.h>
void function_Array(int *const a)
{
    printf("%8d",a[a[0]]);
    printf("%8d",a[a[1-1]]);
    printf("%8d",a[a[1*0]]);
    printf("%8d",a[a[0/1]]);
    return;
}
int main(void)
{
    const int arr[5]={5-3*0-1,10,15,20,25};
    function_Array(arr);
    return 0;
}
```

- A. 25 25 25 25
- B. 25 4 4 4
- C. Compiler error
- D. 4 4 4 4
- E. 25 5 5 5

Answer: A

2.

```
#include<stdio.h>
void function_array(int a[], int size)
{
    ++2[a];
    printf(" %d ",- ++2[a]);
    return;
}
int main( void )
{
    int arr[4]={10,20,30,40};
    --2[arr];
    printf(" %d ",- --2[arr]);
    function_array(arr, 4);

    return 0;
}
```

- A. -29 -31
- B. -28 -30
- C. error
- D. 28 30
- E. 29 31

Answer: B

3.

```
#include <stdio.h>
void print(double a[])
{
    int n=sizeof(a)/sizeof(*a)+sizeof(&a)-(a[7] /1.1f);
    int i;
    for (i = 0; i < n; i++)
        printf(" %.1lf ", a[i]);
    return;
}
```

```
int main( void )
{
    double arr[] = {1.1,2.2,3.3,4.4,5.5,6.6,7.7,8.8} ;
    print(arr);
    return 0;
} //note :: consider 64 bit compilation.
```

- A. 1.1
- B. 1.1 2.2 3.3 4.4 5.5 6.6 7.7 8.8 0.0
- C. 1.1 2.2 3.3 4.4
- D. Compile time error
- E. 1.1 2.2

Answer: A

4.

```
#include<stdio.h>
float function_array(const double *ptr)
{
    return (float)(sizeof(*ptr)+1)/(sizeof(ptr[1]));
}
int main( void )
{
    double arr[]={1.2 , 2.3 , 3.4 , 4.5 , 5.6};
    float size=(float)(sizeof(arr)+1)/(sizeof(arr[1]));

    printf("%.4f %.4f",size, function_array(arr));
    return 0;
}
```

- A. 41.0000 5.1250
- B. 5.0000 1.2500
- C. 5.1250 1.1250
- D. 0.0000 0.0000
- E. Error

Answer: C

5.

```
#include<stdio.h>
void abc(int arr[])
{
    printf("%c", *++arr + ' ');
    printf("%c", *arr++ + ' ');
    return;
}
int main()
{
    int arr[10];
    arr[0] = 0101;
    arr[1] = 0106;
    arr[2] = 0113;
    arr[3] = 0125;

    abc(arr);

    return 0;
}
```

- A. ff
- B. ii
- C. jj
- D. gg
- E. ee

Answer :A