

Q2. Trace the C program and write the output into the corresponding boxes given below.

Note that;

- Four numbers will be displayed in the output and while you are writing the output, you must fill the given boxes below according to the display order of these numbers.
- Every box must contain only one number.

answer box

```
#include <stdio.h>
int main()
{
    int R=0;
    do
    {
        switch (R)
        {
            case 0: printf("%d\n",R);
            case 1: R += 1;
                    break;
            case 2: printf("2\n");
            case 3: printf("%d\n",R * 2);
            case 4: R = R + 3;
                    break;
            default: R = R * 3;
                    printf("%d\n",R / 2);
        }
    }
    while(R<=10);
    return 0;
}
```

Q3. Trace the following C program and write the output into the corresponding box, which is divided into cells, below. Assume that you have 4-cell to 8-cell display screen and each cell shows one character on the screen. Note: if any character is not printed, let the corresponding cell empty.

```
#include <stdio.h>
int main()
{
    int n=10;
    int j=10;

    for( n=7; n>=1; n=n-2)
    {
        for(j=0; j<n; j++)
            printf("%d", n-1);
        printf("C\n");
    }
    return 0;
}
```

OUTPUT:

	1	2	3	4	5	6	7	8
1								
2								
3								
4								

Q4. Trace the following C program and write the output into the corresponding box below.

```
#include <stdio.h>
int main()
{
    int Number=0;
    while(Number<5)
    {
        switch(Number)
        {
            default: printf("Magusa\n");
            case 0: printf("Girne\n");
                    break;
            case 3: printf("Lefkosa\n");
                    ++Number;
                    break;
            case -1: printf("Iskele\n");
                    break;
            case 1: printf("Karpaz\n");
        }
        Number++;
    }
    return 0;
}
```

OUTPUT:

Q5.

Complete the missing parts in the following C program to produce the output below. Be aware of odd numbers will be displayed one time, and even numbers will be displayed according to its value. For example: 1 is an odd number, and displayed one time; 2 is an even number and displayed 2 times; 3 is an odd number and displayed one time; 4 is even number, and displayed 4 times; etc.

Notes: Do not use any variable other than the variables declared in the program.

OUTPUT:

122344445666666788888888

```
Code
#include <stdio.h>

int main()
{
    int N=8;
    int i=1;
    int j;
    while(i<=N)
    {
        if(i%2==0)
        {
            for( ..... )

                printf( ..... );
        }
        else
            printf( ..... );
        i++;
    }
    return 0;
}
```

Q6. Consider the following C program whose description and functionality is given below:

The program calculates the tax pay for 100 employees. The tax pay for each employee depends on the age and marital status (**m_status=1** for single, and **m_status=2** for married) of the employee. Data (**age**, **salary**, and **m_status**) is entered by an operator who must have a valid security code between 1 and 9999. The operator cannot enter data if his security code is not valid, and is asked to re-enter his code every time he enters it wrongly. There are 4 different tax rates, **rate1**, **rate2**, **rate2**, and **rate4**. The tax rate which of an employee is determined using the following table and the tax to be paid is given by:

$$\text{tax} = \text{salary} * \text{tax rate}$$

Complete the missing parts of the code such that the program functions correctly.

Age	Marital status	Tax rate
<30	single	rate3
<30	married	rate4
≥30	single	rate1
≥30	married	rate2

```
#include <stdio.h>

#define rate1 0.05
#define rate2 0.03
#define rate3 0.01
#define rate4 0.0

main(){
int age, salary, emp_no, m_status, sec_code;
float tax;

do{
    printf("\n Please enter security code");
    scanf(" %d", &sec_code);
} while( ..... ); // a

emp_no = 1;
while( ..... ) { // b
    printf("\n For the employee with number %d ", emp_no);
    printf("\n enter age, salary and marital status ");
    scanf( ..... ); // c
    if( ..... ) // d
        tax=salary*rate1;
    else if( ..... ) // e
        tax=salary*rate2;
    else if( ..... ) // f
        tax= ..... ; // g
    else
        tax= ..... ; // h
    printf("\n Tax pay for employee %d = %.2f, emp_no, tax);
    emp_no ++;
}
}
```

Q7.

Write down the output of the following C program.

output

```
#include <stdio.h>
int main()
{
    int x;
    for (x=0; x<5; x++)
    {
        switch (x)
        {
            case 0: printf("Zero\n");
                    x=x+1;
                    break;
            case 1: printf("One\n");
                    x += 2;
                    break;
            case 2:  printf("Two\n");
                    x ++;
            default: printf("%d\n",x);
                    printf("None\n");
        }
    }
    return 0;
}
```

Q8.

a) Write the following loops using while structure:

<pre>for (i=0, sum=0; i<5; i++) { scanf ("%d", &n); sum +=n; }</pre>	<pre>for (; i<10; i++) printf ("%10d%10d", i, i*i*i);</pre>
---	--

b) Write the following loops using for structure.

<pre> graph TD Start(()) --> i1[i=1] i1 --> Cond{i < 20} Cond -- T --> S["s = s + i^2"] S --> I["i = i + 2"] I --> Cond Cond -- F --> Exit(()) </pre>	<pre>int i,k,s; s=0; i=20; while(i<30 && s<1000){ s=s+i*i; i=i+1; }</pre>	<pre>int i=2, s=1000; while(1) { i=i*2-1; s=s-i; if(i>20) break; if(s<500) break; }</pre>
--	---	---