

23-05-24

To print m-tables using arrays:-

→ main ()

```
{ int a[20], i, m, j;  
  printf ("enter m");  
  scanf ("%d", &m);  
  printf ("enter elements");  
  for (i=0; i<m; i++)  
    scanf ("%d", &a[i]);  
  for (i=0; i<m; i++)  
  {  
    for (j=0; j<=10; j++)  
    {  
      printf ("%d * %d = %d \n", a[i], j, a[i]*j);  
    }  
  }  
}
```


a:- Sum of individual digits:-

→ main ()

```
{ int a[20], i, n, m, s, x;
```

```
printf ("enter n");
```

```
scanf ("%d", &n);
```

```
printf ("enter elements: ");
```

```
for (i=0; i<n; i++)
```

```
scanf ("%d", &a[i]);
```

```
for (i=0; i<n; i++)
```

```
{
```

```
    s = 0; x = a[i];
```

```
    while (x != 0)
```

```
    {
```

```
        m = x / 10;
```

```
        s = s + m;
```

```
        x = x / 10;
```

```
    }
```

```
printf ("%d no digits sum is %d", a[i], s);
```

```
}
```


Q:- reverse a given number :-

→ main ()

```
{ int a[20], i, n, s, m;
```

```
printf ("enter n");
```

```
scanf ("%d", &n);
```

```
printf ("enter elements: ");
```

```
for (i=0; i<n; i++)
```

```
scanf ("%d", &a[i]);
```

```
for (i=0; i<n; i++)
```

```
{
```

```
    s = 0; m = a[i];
```

```
    while (m != 0)
```

```
    {
```

```
        m = m % 10;
```

```
        s = s * 10 + m;
```

```
        m = m / 10;
```

```
    }
```

```
}
```

```
printf ("%d reverse of given number is %d",
```

```
{
```

```
a[i], s);
```

```
}
```


Q:- Palindrome number:-

→ main ()

```
{
    int a[20], m, n, s = 0, x;
    printf("enter m");
    scanf("%d", &m);
    printf("enter elements: ");
    for (i = 0; i < m; i++)
        scanf("%d", &a[i]);
    for (i = 0; i < m; i++)
    {
        s = s * 10 + m % 10;
        x = m % 10;
        m = m / 10;
        while (m != 0)
        {
            m = m % 10;
            s = s * 10 + m;
            m = m / 10;
        }
        if (a[i] == s)
            printf("%d is palindrome (n", a[i]);
        else
            printf("%d is not palindrome (n", a[i]);
    }
}
```

m - number
Prime

Q:- m-numbers Prime / not prime :-

-> main ()

```
{
    int a[20], i, j, m, c=0;
    printf ("enter m ");
    scanf ("%d", &m);
    printf ("enter elements: ");
    for (i=0; i<m; i++)
        scanf ("%d", &a[i]);
    for (i=0; i<m; i++)
    {
        c=0;
        for (j=2; j<=a[i]/2; j++)
        {
            if (a[i]%j == 0)
                c++;
        }
        if (c == 0)
            printf ("%d is prime \n", a[i]);
        else
            printf ("%d is not prime \n", a[i]);
    }
}
```


Q:-

→ main ()

```
{ int a[20], i, m, key;
printf ("enter m: ");
scanf ("%d", &m);
printf ("enter elements");
for (i=0; i<m; i++)
{ if (a[i] == key)
{ printf ("
```

SORTING

count 5 = 0
count 10 = 0

5 8 10 10
↓ ↓
5 + 5

if (bill == 5)
++count 5;

```
→ if (bill == 5)
{ ++count 5;
}
else if (bill == 10)
{ --count 5;
++count 10;
}
else if (count 10 == 0)
{ --count 10;
--count 5;
}
if (count 5 < 0)
return false
```

bill > 10
++count 10;

```
SORTING
→ int main
{ int a[20], i, length, temp = 0;
Sorting =>
→ int main ( )
for (i=0; i<m; ++i)
{ for (j=i+1; j<m; ++j)
{ if (a[i] > a[j])
{ temp = a[i];
a[i] = a[j];
a[j] = temp;
}
}
}
```