1. Please translate the following program fragment into **three-address code**.

```
i=1;
product=1;
count=0;
while (i<=10) {
    j=1;
    while (j<=10) {
        if (a[i,j] <> 0) //"<>" means "not equal to"
        {
            product=product*a[i,j];
            count=count+1;
        }
        j=j+1;
    }
     i=i+1;
}
```

Notes: Here we assume that the declaration of array A is array [1..10,1..10],

each data element of array A would **only use 1 storage unit,** and the start address of array A's storage area is addrA.

2. Please translate the following program fragment into **quadruple.**

```
i=2;
m=0;
loop=0;
while (loop==0 && i<=10) {
    j=1;
    while (loop ==0 && j<=i)
     if (a[i,j] != a[j,i]) //"!=" means "not equal to"
          {
               loop=1;
               m=1;
          }
      else j=j+1;
    if (loop==0) i=i+1;
}
```

Notes: Here we assume that the declaration of array A is array [1..10,1..10], each data

element of array A would **only use 1 storage unit,** and the start address of array A’s storage area is addrA.