In [15]:

```
#Name: T.Kavyasree
#Roll.No: 25201318
```

In [3]:

```
#count the number of trailing zeros in a binary number
def count(N):
    count=0
    if N==0:
        return 0
    while(N&1)==0:
        count+=1
        N=N>>1
    return count
N=int(input("Enter a number : "))
print("The number of trailing zeros are",count(N))
```

```
The number of trailing zeros are 3
```

In [9]:

```
#finds the exponential of x power y
def exponential(x,y):
    if y==0:
        return 1
    else:
        return x*exponential(x,y-1)
x=int(input("Enter a number : "))
y=int(input("Enter a number : "))
print("The Exponential is",exponential(x,y))
```

```
The Exponential is 81
```

In [1]:

```
#simulates a bank ledger system using journaling, where transactions are first
#with recovery support in case of a crash.
import os
Ledger="Ledger.txt"
journal="Journal.txt"
def add_transaction(account,amount):
    transaction=f"{account},{amount}"
    with open(journal,"a") as j:
        j.write(transaction)
    with open(Ledger,"a") as l:
        l.write(transaction)
    open(journal,"w").close()
    print("Transaction added",transaction.strip())
def recover():
    if os.path.exists(journal) and os.path.getsize(journal)>0:
        print("journal is not empty")
        with open(journal,'r') as j:
            with open(Ledger,"a") as l:
                for line in j:
                    l.write(line)
        open(journal,"w").close()
        print("journal cleared")
def crash(account,amount):
    print("starting transaction")
    transaction=f"{account},{amount}"
    with open(journal,"a") as j:
        j.write(transaction)
    print("written to journal")
    print("system crashed")
    exit()
account=input("Enter account number")
amount=int(input("Enter the amount to be deposited"))
add_transaction(account,amount)
#crash(account,amount)
#recover()
```

```
Transaction added A-501,500
```

```
In [ ]:
```