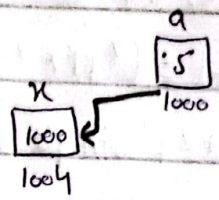
LAB #02

Data Structure & Algorithms

Write the dry run output of the codes.

Prog #01

```
{
① int a = 5;
②int *x;
③ x = &a;
cout<< "value of a = "<<a<<endl;
cout<<"Address of a = "<<&a <<endl;
cout<<"Address of x = "<<&x<<endl;
cout<"x Point to address = "<< x<<endl;
cout <<"x Point to value = "<< *x;
return 0;
}
```



Output:

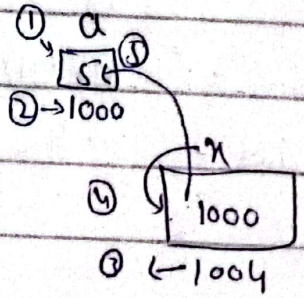
value of a = 5

Address of a = 1000

Address of x = 1004

x Point to address = 1000

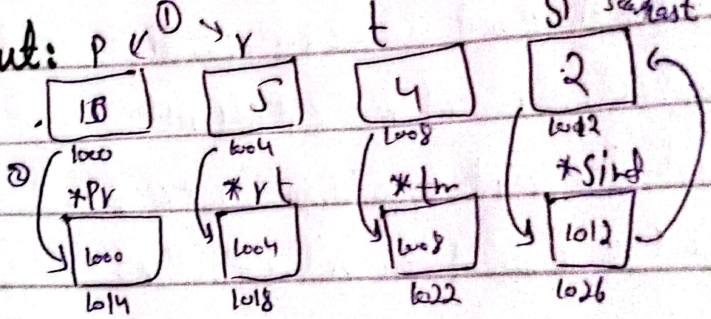
x Point to value = 5



Program #02

```
{
int P,r,t,si,*pr,*rt,*tm,*sint;
pr = &P;
rt = &r;
tm = &t;
sint = &si;
values enter from user
*sint = (*pr * *rt * *tm) /100;
cout << "simple interest=" << *sint;
return 0;
}
```

Output:



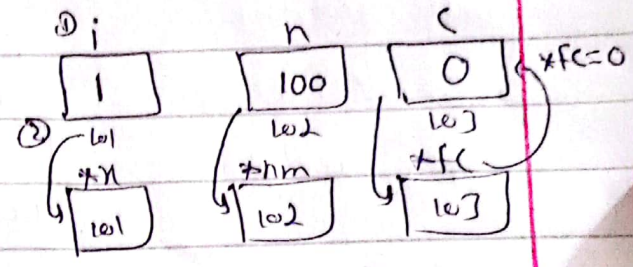
user enter values

Enter Principle = 10,
Enter Rate = 5
Enter time = 4
*sint = (10 * 5 * 4) /100
*sint = 200/100 = 2
si = 2

Program #03

Number is Prime or not.

```
{
int i,n,c, *x,*nm,*fc;
x=&i;
nm=&n;
fc=&c;
*fc=0;
```




```
Enter a number =10
for(*x=1; *x<=*nm; *x=*x+1)
 {
  if(*nm%*x=0)
    {
     *fc=*fc+1;
    }
  if (*fc==2)
  Cout<<"Prime Number";
 }else{
       cout<<"Not a Prime Number";
      }
return 0;
```

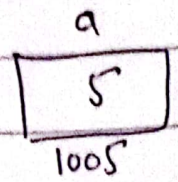
Output:

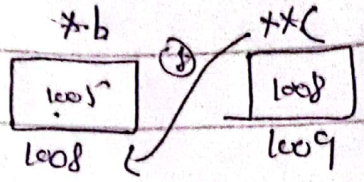
Enter a number=10

Not a Prime Number

Program #04

```
{
int a=5, *b, **c;
b = &a;   → 1005
c = &b;   → 1008
```





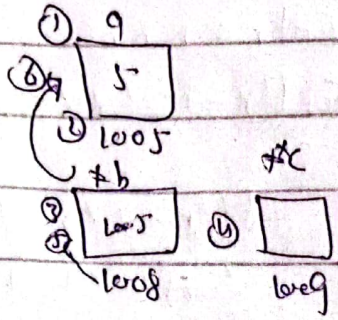
Output:

Value of a = 5

Address of a = 1005

Address of b = 1008

Address of c = 1009



b Points to address = 1005

b Points to the value = 5

C Point to the address = 1008

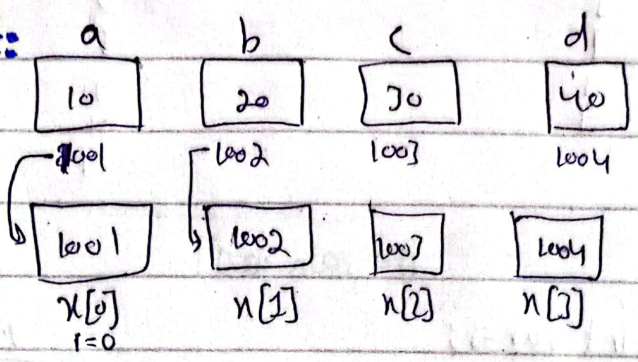
C Point to the value = 5

Program #05

```
int main()
{
int *x[4];
int a=10, b=20, c=30, d=40, i;
x[0] = &a;
x[1] = &b;
x[2] = &c;
x[3] = &d;

for(i=0; i<4; i++)
{
cout << "value of x[" << i << "]= " << x[i] << " and Points to = "
                                    << *x[i] << endl;
}
return 0;
}
```

Output:



value of x[0] = 1001 = 10

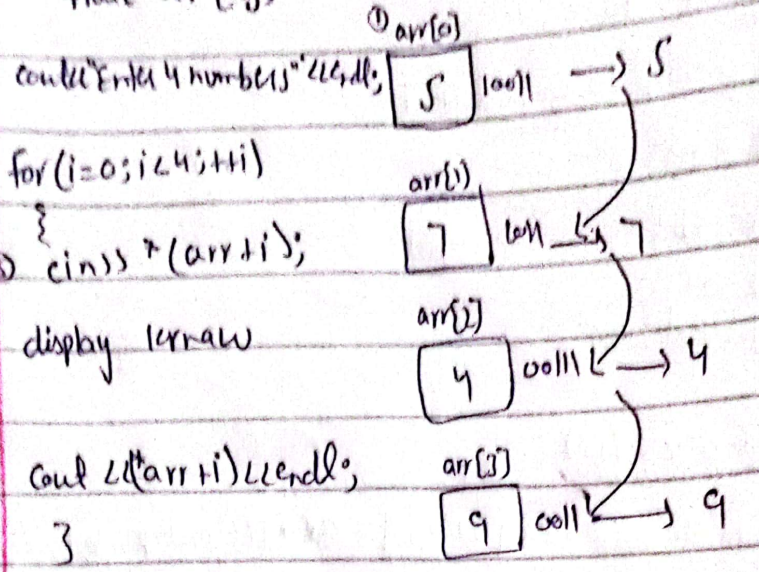
" " x[1] = 1002 = 20

" " x[2] = 1003 = 30

" " x[3] = 1004 = 40

Program #06

```
{
  float arr[4];
cout<<"Enter 4 numbers"<<endl;
for(i=0;i<4;++i)
{
cin>>*(arr+i);
display 1array
cout<<(*(arr+i))<<endl;
}
```



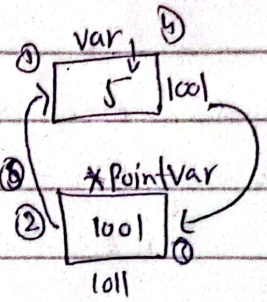
Output:

Enter 4 number
5
7
4
9

Display the data
5
7
4
9

Program #07

```
① int var=5;
② int *PointVar;
③ PointVar = &var
```



Output:

③ cout<<"var = "<<var<<endl =5

var = 5

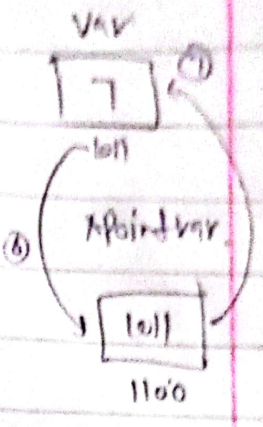
① *PointVar = 5

changing value of var to 7;

②var = 7

③ *PointVar = 7

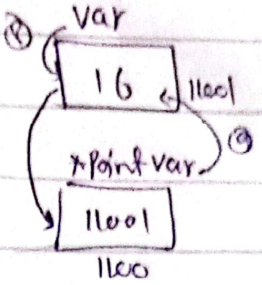
changing value of var to 16;



⑧ var = 16

⑨ *Pointvar = 16

⑩ *Pointvar = 16



Program #08

(Sum of even numbers)

{

①int i, n, s, d, *x, *nm, *sum, *dg;

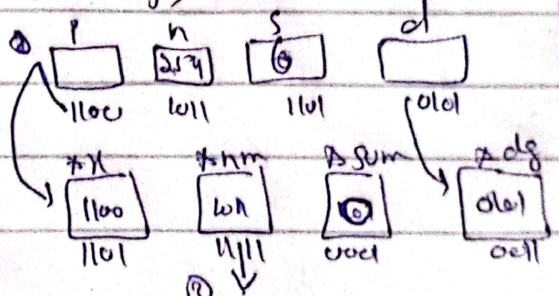
② x = &i;

nm = &n;

sum = &s;

dg = &d;

*sum = 0;



③ while (*nm > 0)

④ {

⑤ if(*dg % 2 == 0)

⑥ { *sum = *sum + *dg;

(1111 > 0)

④ (4 % 2 = 0)

⑤ s = 2 + 4 = 6

Output:

Enter a number 15649