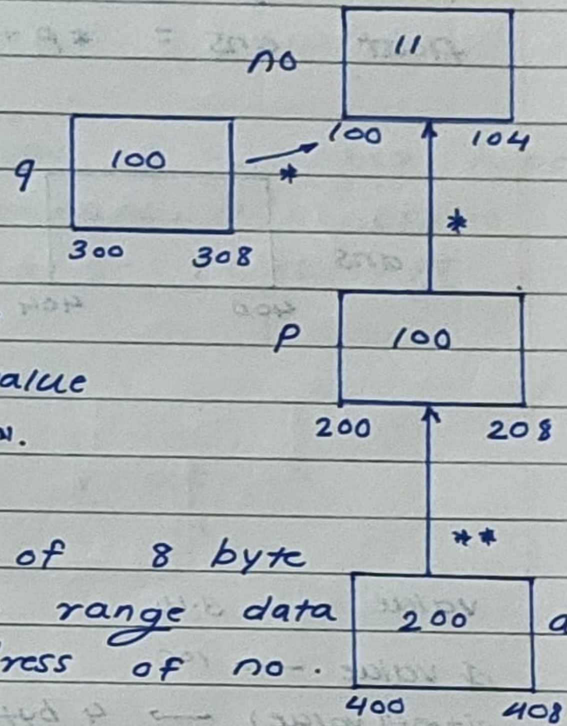
Page No.
Date

Assignment - 6

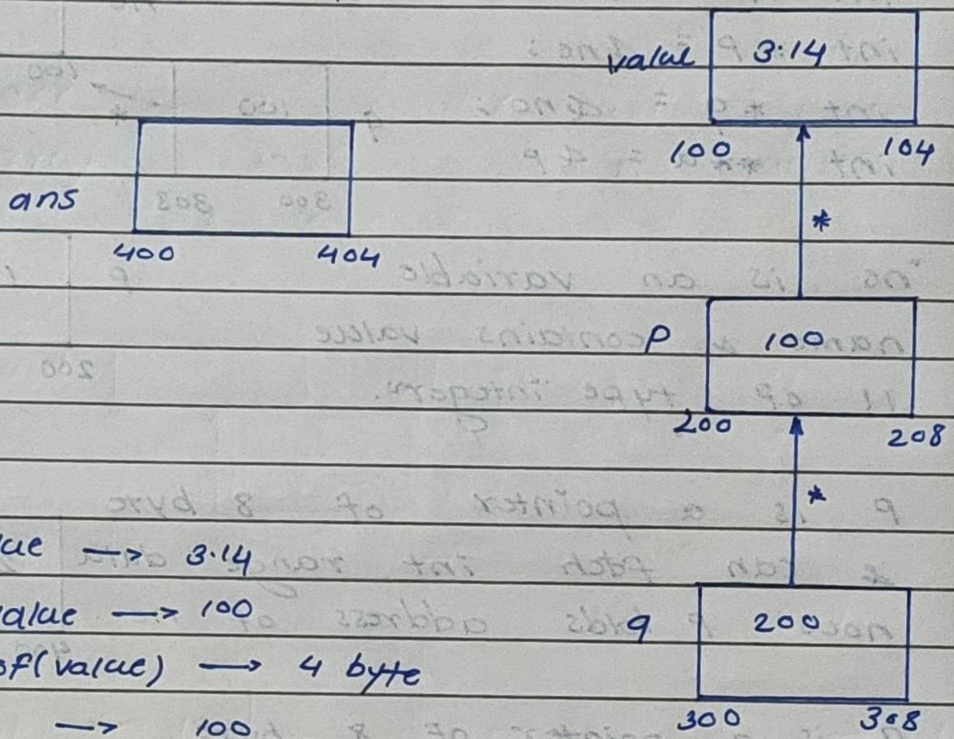
1.
```
int  no = 11 ;
int  * p = &no ;
int  * q = &no ;
int  ** a = &p
```



- "no is an variable name & contains value 11 of type integer".
- p is a pointer of 8 byte & can fetch int range data now p holds address of no.
- q is a pointer of 8 byte and can fetch int range data now, q holds address of no.
- a is a pointer of 8 byte & can fetch int range data & now, holds address of p.

no	→ 11	q	→ 100
&no	→ 100	&q	→ 300
sizeof(no)	→ 4	*q	→ 11
p	→ 100	Size of (q)	→ 8 byte
&p	→ 200	sizeof (*q)	→ 4
* p	→ 11	a	→ 200
Size of (P)	→ 8 byte	&a	→ 400
Sizeof (*P)	→ 4 byte	Size of (a)	→ 8
		Size of (*a)	→ 4

2. float value = 3.14 ;
float *p = & value;
float *q = p ;
float ans = *p + *q ;



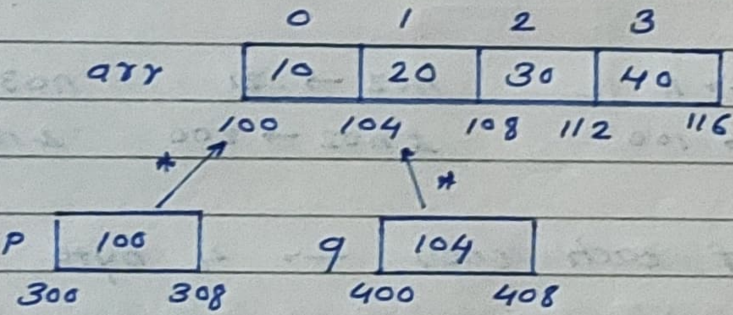
value → 3.14
& value → 100
sizeof(value) → 4 byte
p → 100
& p → 200
size of (p) → 8
Size of (*p) → 4
*p → 3.14
q → 200
& q → 300
*q → 100
size of (q) → 8
sizeof (*q) → 4

3. int arr [] = {10, 20, 30, 40}
int *P = arr;
int *q = arr+1;
int ans = *q - *P

- arr is the 1D array of 4 integer type elements. which contains 10, 20, 30, 40
size of arr is 4 * sizeof(int) = 16

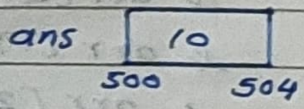
- P is a pointer which hold address of arr i.e arr (first base address)
we can say *p = &(arr[0]);

- *q is a pointer which increment arr by 1 block.



arr+1
= 100 + 1 x (int)
= 104

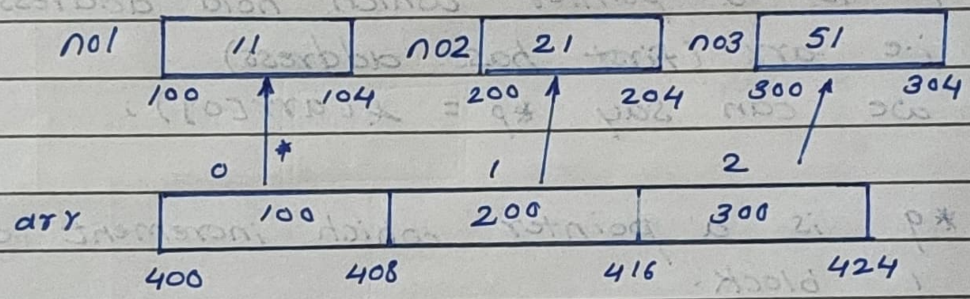
∴ *q − *P
= 20−10
= 10



4. int no1 = 11, no2 = 21 , no3 = 51 ;
// short hand declaration

int *arr [ ] = { &no1 , &no2 , &no3 } ;

" arr is a pointer and it is a 1D array which holds the address of 3 integer elements ".



no1 → 11 no2 → 21 no3 → 51
&no1 → 100 &no2 → 200 &no3 → 300

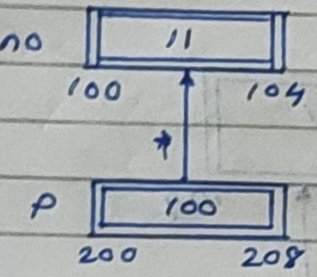
- size of each (no) → 4 byte
- size of (arr [3] → 24
- size of (* arr [3]) → 12
- & arr [0] → 400
  & arr [1] → 408
  & arr [2] → 416
  & arr [3] → Not determined

arr [0] → 100
arr [1] → 200
arr [2] → 300
* arr [1] → 21
* arr [2] → 51

5. const int no = 11;
cost int *p = &no;

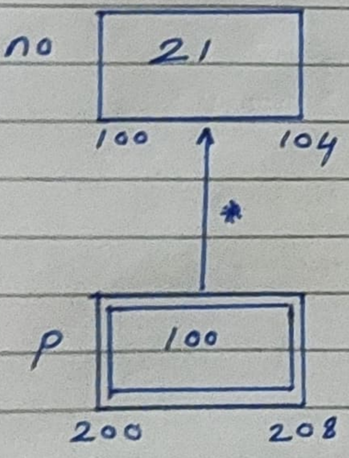
" no is an identifier which holds " value of const int type".

" p is a pointer holds address of no & it is of type int constant".



6. int no = 21;
int * const p = &no;

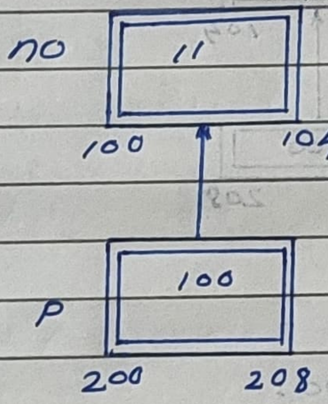
" Here p is constant pointer holds the * address of no which is integer".



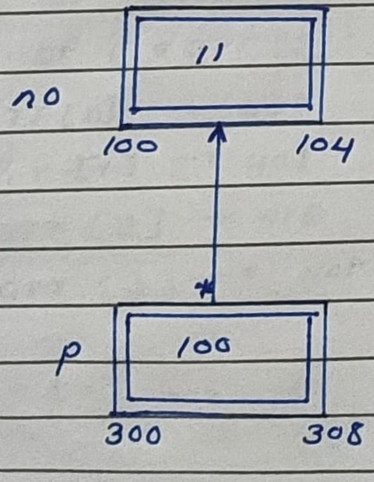
7. const int no = 11;
const int * const p = &no;

- no is a variable name which stores the value of const integer type i.e 11

- p is pointer of size 8 byte which is a constant type hold the 100 i.e address of no.

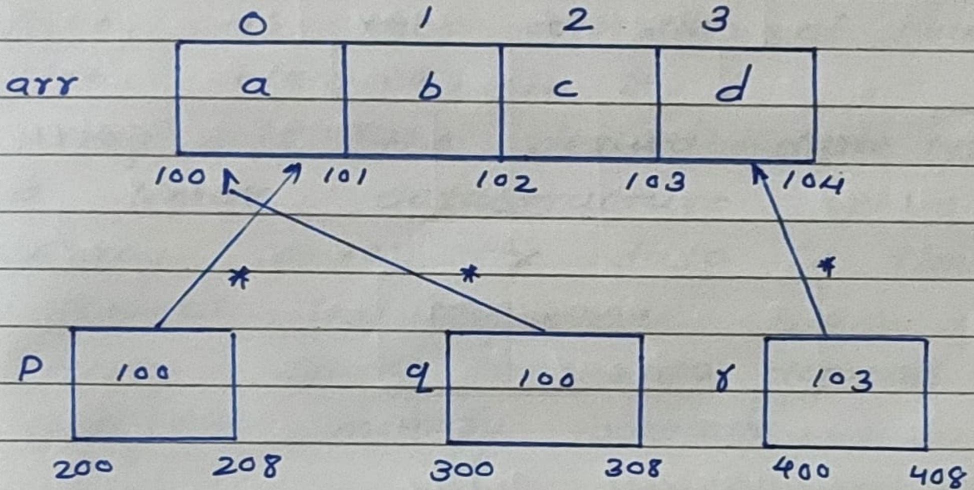


8. const int no = 11;
int const * const p = &no;

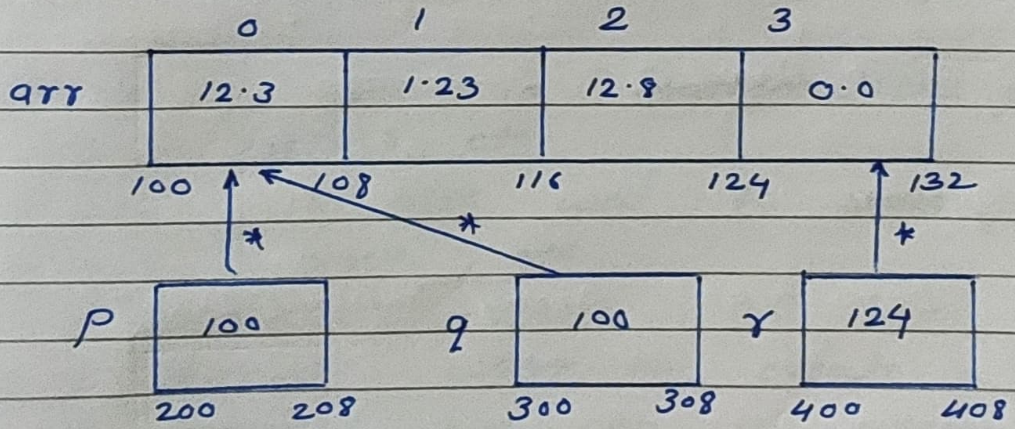


Page No.
Date

9.
```
char  arr [] = { 'a', 'b', 'c', 'd' };
char  *p     = arr ;
char  *q     = &(arr[0]) ;
char  *r     = &[arr[3]);
```



10.
```
double arr [] = { 12·3, 1·23. 12·8 };
double *p = arr;
char *q = &(arr[0]);
char * r = &(arr [3]);
```



0
1
2
3
arr
12·3
1·23
12·8
0·0
100
108
116
124
132
*
*
*
p
100
q
100
r
124
200
208
300
308
400
408