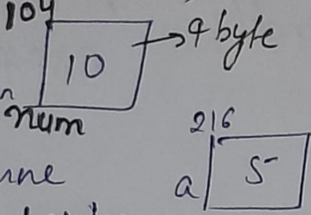
Pointers - class 1

Date- 4/10/2023

Pointers:- A special type of variable which store <u>address</u> of other variables.

for egs int num=10;

kissibhi memory location ko access karne ke liye us boxka address pta hona jaruri tabhi usko access kar sakte hai



Symbol table

num → 104
a → 216

Store ho jata hai address symbol table me.

Print address &num.
↳ 104.

reality me → hexadecimal ke form me return hota hai

```
int a=10
cout<<a       → 10
cout<<&a      → 104 address storehoga.
```

Pointer ke andar humesa

* Creation of pointer.

int *ptr = address

- int * → pointer to integer data
- ptr → pointername
- ptr → variable name

ptrisa pointer to integer data

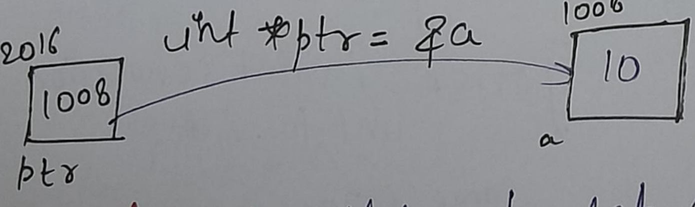
char * pointer to chardata

bool * pointer to bool data

long * pointer to long data

short * pointer to short data

```
int a=10
int *ptr= &a
```



✻ Access of pointer → Value stored at address stored in ptr

iske liye ek operator hota hai ↳ dereference operator → * →

cout *ptr
→10.

```
#include<iostream>
using namespace std;
int main(){
  int a=5;
  cout<<a<<endl;
 cout<<"address of a"<< &a <<endl;
 // pointer creation→
   int *ptr = &a;
  cout<< ptr<<endl;
  cout<< "Accessing:" << *ptr<<endl;
   cout << &ptr <<endl;
 return 0;
 }
```

output→

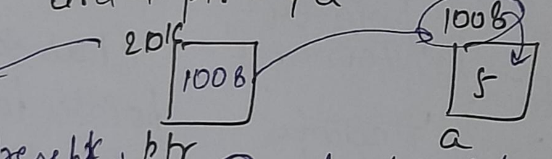
```
5
address of a! 0x7ffedec3b52c
 0x7ffedec3b52c

 Accessing: 5
  0x7ffedec3b52c.
```

int a=5

int * ptr = &a

ptr → 1008

&ptr → 2016



a → 5

&a → 1008.

*ptr → (I) address pe jao ptr (II) value print kar do.

Reference variable is just second name for same memory location. Symbol table me entry hoti hai.

But pointer takes memory and store address of another memory.

```
int main(){
  int a=5;
  int ptr=&a;
  int *ptr =a;
```

] error: invalid conversion from int to int.

" Same " error

```
int a=5
int *ptr= &a
```

→ sizeof(ptr) = 8

```
Char ch='k'
char* cptr=&ch;
```

↓

sizeof(cptr) = 8.

sizeof(ptr) → 8.

```
long a=10;
long* ptr=&a
```

pointer ke andar humesa address store hoga

depend karta system pe

machine dependent

H/w → 32 vs 64-bit
→ Why ptr size 8/4 ?

Dynamic Memory Allocation

```
int *ptr ;
cout << (*ptr);
```

→ Bad Practice

→ dereference

It gives
Runtime error

* Why it is Bad Practice?
when we create *ptr there will be random value or garbal value allocate ho jayegi and vo value kisi memory location ki hogi jo muhe allow nhi hai access karna. Ek illegal memory access karne jaa rhe hai jo ki bad practice hai.

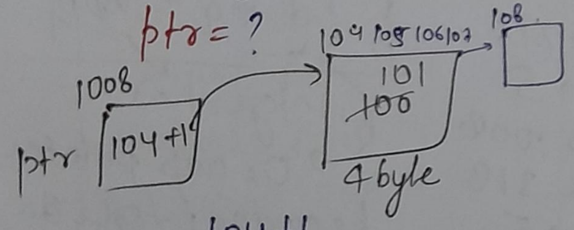
How to prevent this?
By using null pointer `int *ptr = 0;` → nullptr

Is time pe bhi error ayegyi lekin debugging karne ke liye code easy ho jayega.

Ques1→

```
int a=100
int *ptr = &a
a=a+1;
ptr = ptr+1;
```

a = ?
ptr = ?

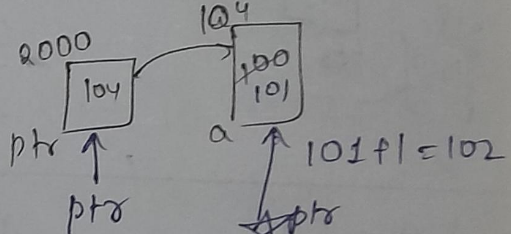


104+1
=108

1 byte is smallest addressable space.

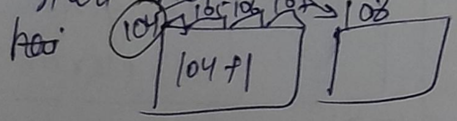
Ques2→
Value present at address stored in ptr

```
int a=100;
int *ptr=&a;
a=a+1
*ptr=*ptr+1
```



*ptr = 102
a = 101

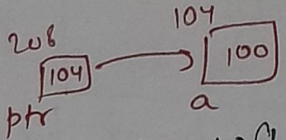
Agar koi intege 104 location se start hota hai toh ye 1 byte se increase hota hai



Toh jab hum address ke andar +1 kar rhe hai toh ve next address me jaa rha hai 108.

Ques3 →

```
int a = 100;
int *ptr = &a;
```



print →
- a = 100
- &a = 104
- *a = Error
- ptr = address → 104
- *ptr = 100 = (Two step process)
- &ptr = address → 208. [ptr wale box ka address.]

Pointer ko dereference kartehai

integer ko dereference nhi karte islye error aa rha hai

Two step process hota hai

- (*ptr)++ = 101
- ++(*ptr) = 102
- $*ptr = \frac{*ptr}{2} = \frac{102}{2} = 51$
- *ptr = *ptr-2 = 51 - 2 = 49

Ques4 →

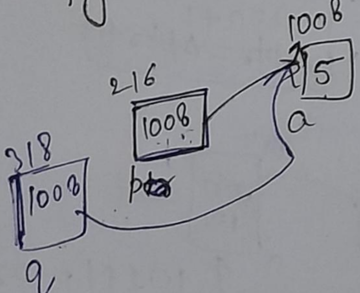
```
int a=5
int *p = &a;
int *q = p
```

a

&a

pointer copy

1008



print
- a → 5
- &a → 1008
- *a → Error → a is integer we can't dereference
- p → 1008
- &p → 216
- *p → 5
- a → 1008
- &q → 318
- *q → 5

```
int *q = *p;
```

integer hai

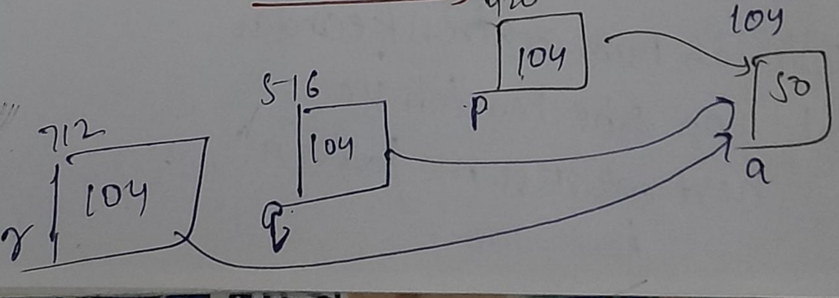
Or aap pointer ke andar integer da rakh rhe ho lekin pointer ke andar address store karte hai [it gives error]

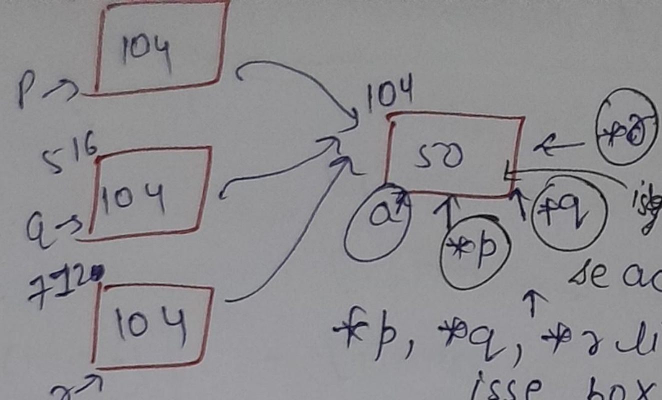
Ques→5

```
int a=50
int *p = &a;
int *q = p;
int *r = q
```

⟸

- a = 50
- &a = 104
- *a = error
- p = 104
- &p = 420
- *p = 50
- q = 104
- &q = 516
- *q = 50
- r = 104
- &r = 712
- *r = 50





~~*~~p = *p/2 = 25

~~*~~q = *q/2 = 25

r = ~~~~q/2 = 25

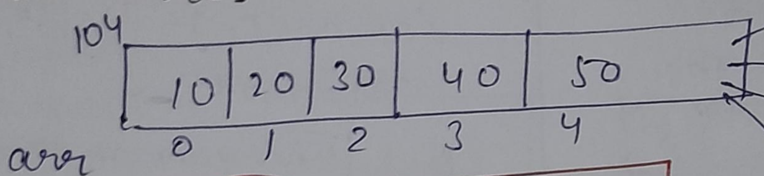
isse different value ~~se~~ name se access kar sakte hai

*p, *q, *r likhu vo saare ke saare isse box ko darsha rhe hai

NOTE →
⇒ No concept of pointer in Java

Pointer in Array :-

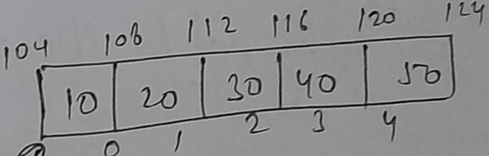
int arr[5] = {10, 20, 30, 40, 50}



→ arr → Base Address → 104
→ arr[0] → 10
→ &arr[0] → 104
→ &arr → B.A → 104

Imp | arr, &arr, &arr[0] → Base Address = 104

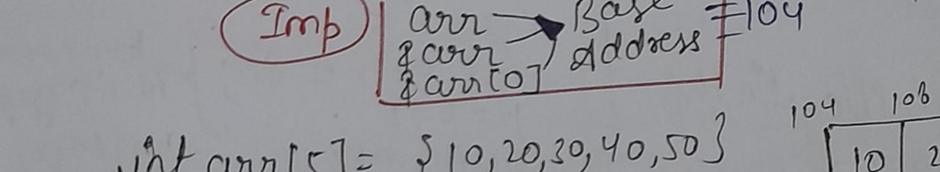
print int arr[5] = {10, 20, 30, 40, 50}



- → arr = BA
- → &arr = BA
- → arr[0] = 10
- → &arr[0] = BA
- → *arr = 10
- → *arr+1 = ~~11~~ ~~20~~ 11

- → *(arr)+1 = 11
- → *(arr+1) = 20   104+1 = 108   * → value print ~~hoga~~ hogya.
- → *(arr+2) ⇒ 104+2 = 112 → 30
- → *(arr+3) ⇒ 116 → 40

Imp



int size → 4
add → 104 + 0
104+0.4 = 1~~08~~
104+1×4 = 108
104+2 = 104+2×4 = 1~~0~~2

*arr → *arr+0 → *(arr+0)
*104   10
value stored at address 104 → 10
*arr[0] = 10

Important Observation.

*(arr+i) → arr[i]
i[arr]

*(arr+0) → arr[0]

*(arr+1) ⟷ arr[1]

arr[i]   *(arr+2) → arr[2]

↓

&*(arr+i)   *(arr+i) → arr[i]

↓

*(B.A+i)   *(i+arr) $\xrightarrow{or}$ i[arr]

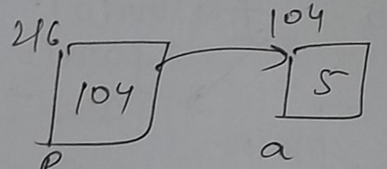
[(*arr)+1] ispe apply nhi hogi condition.

Notes

int a=5

int *p = &a

p=p+1

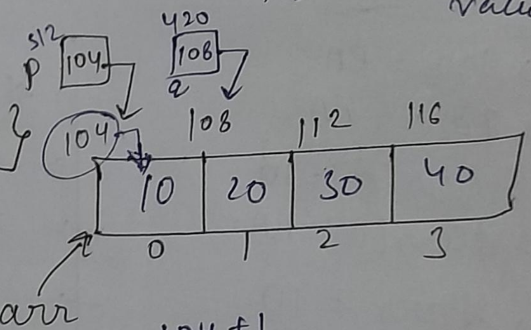


H.W. Imp. Ques Why not possible in this case? arr=arr+1.

```
int main(){
int arr[5]={1,2,3,4,5};
arr=arr+1; → This gives an error
}
int a=5;
int *ptr = &a;
p=p+1;
cout<<*p;
```

→ koi error nhi aayega Garbage value.

```
int arr[4]={10,20,30,40};
int *p=arr;
int *q=arr+1;
```



→ arr = BA=104

→ &arr = BA=104

→ arr[0]=10

→ &arr[0]=BA=104

p=104

&p=~~104~~512

*p=10

104+1

104+1×4=108.

q=108

&q=420

*q=20

*p+1 = 11

*(p)+2 = 12

*(q)+2 = 22

*(q+4) = ← Garbage value or segmentation fault or may be another.

int arr[4]= {10,20,30,40}

sizeof(arr) → 4x4=16.

int *p =arr

↳ sizeof(p) → 8

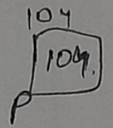
address ko store karta hai

int arr [10]={10,20,30}

int *p = arr

Cout<< p. → 104

print hoga.



Char ch[50]="love"

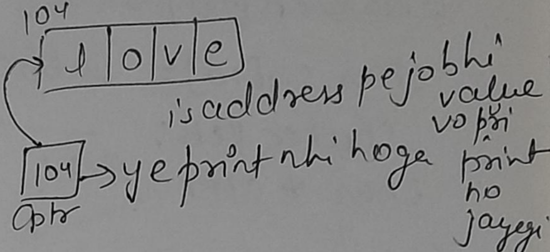
Char *cptr = ch or &ch

Dono case me base address hi milega

cout<<cptr;

↳ love

entire string print hogi



ye print nhi hoga

is address pe jo bhi value vo pri print ho jayegi

cptr → love

*cptr

*(cptr+0)

cptr[0] → phela character print hoga.

Ques char Ch[50]="love";

Char *cptr = Ch or &Ch → initialise nhi kar payegye 208 error dega

print

→ ch ⟹ love

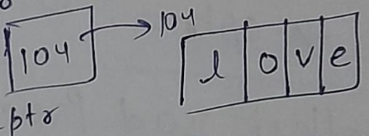
→ &ch → 104

→ ch[0] → l

→ &cptr → 208

→ *cptr → l

→ cptr → love



cptr =(cptr+0) → cptr[0]
↓ l

entire string print karte rehnge jab tak null character nhi mil jaye.

Ques char ch [30]="statement";

char * cptr = &ch[0];

→ ch ⟹ statement

→ &ch ⟹ 104

→ *(ch+3) ⟹ t ← Ch[3]

→ Cptr ⟹ statement

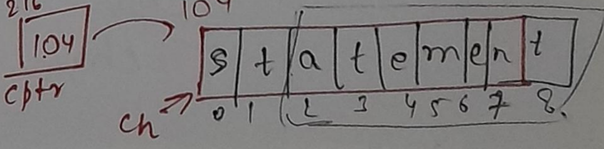
→ & cptr ⟹ 216

→ *(cptr+3) ⟹ t
cptr[3]

→ cptr +2 ⟹ atement

→ * cptr ⟹ cptr [0] → s

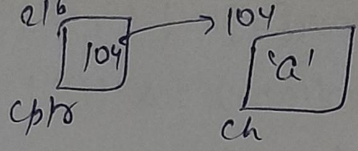
→ cptr +8 ⟹ t.



Que

```
char ch = 'a';
char * cptr = &ch

cout<<cptr
```

→ cptr



a

a print ~~usk~~ hoga uske baad kuch value print hongye jab tak null n mil jaye.

~~char ch[10] = "a";~~

```
char ch[10] = "Babbar";
cout<<&ch;            → address print hoga.
char * cptr = &ch;    → show error  cannot convert
```

```
int main(){
    char ch[10] = "Babbar";
    char *cptr = ~~ch~~& ch[0];   chal jayega lekin &ch problem de rha hai
    char * cptr = ch
}
```

Char * cptr = "Babbar"; → possible hai ki shi hai

Why Bad Practice

Jab ye word store kiye ~~ye create kiye toh~~ toh ye temporary storage me store hoga.

aap ek pointer ko temporary location pe point karwa rhe hai jiska koi ~~b~~ Vishwas nhi hai kab tak data present rehga or kabnhi

so iskye liye hum log data ko permanent storage me ~~store~~ copy karwayege uske bad uske ~~up~~ upar pointer ~~kare~~ create karegye

```
char ch[10] = "Babbar";
char *c = ch;
```

char ch[50] = "love";

char * cptr = ch

or

&ch → entire array ka starting address de rha hai.

cptr is a pointer to character data.

single character per point karne ke liye pointer bnaya hai par address kiska hai 10 character ka. isliye nhi create karne de rha.