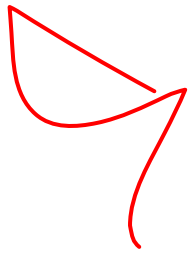
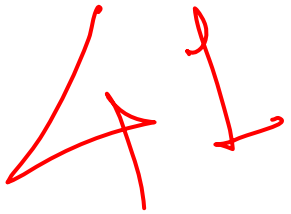


Let’s solve it



Can i do arrayname++ ?

NO

Reason: Arrayname is a 'constant' pointer.

Hence, it cannot be updated.

int *ptr = arr;

ptr++; ? Yes

Constant Pointer i.e arrayname Vs Pointer to constent

strcpy (char *dest,

const char *source);

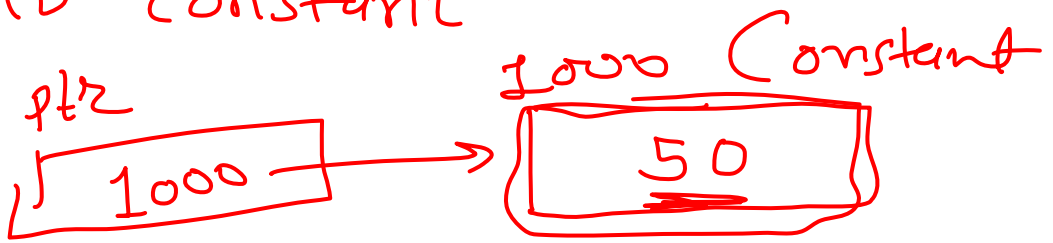
*source = 'x'; ✗

*(source)++; ✗

source is an example of pointer to constant.

strcpy is a reliable function, bcoz it will not corrupt the memory of data pointed by source pointer.

pointer to constant



(*ptr)++; // make it SI

const int x = 50; // x++ ? No, bcz x is constant.

int y = 100;

int *ptr = &y;

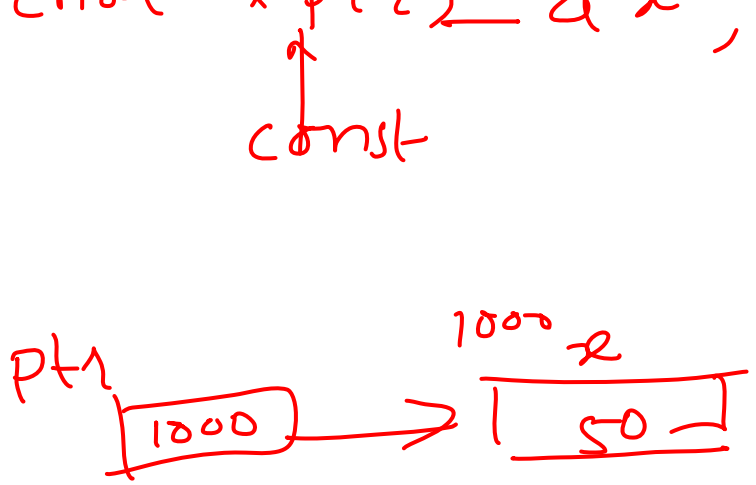
(*ptr)++; // Not allowed - pointer to constant.

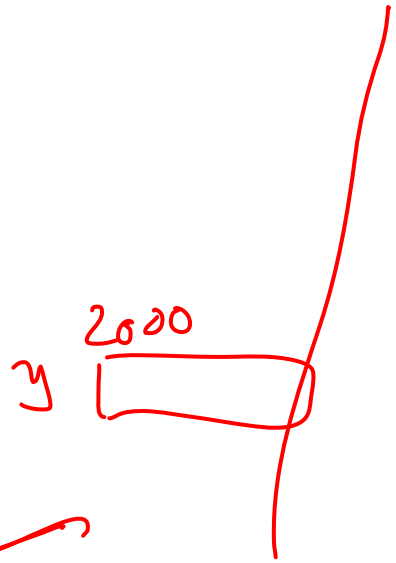
ptr = &x; ptr = &y;

Constant Pointer (i.e array name)

```
char *ptr = &x;
```

const





ptr++;

1040,
1000

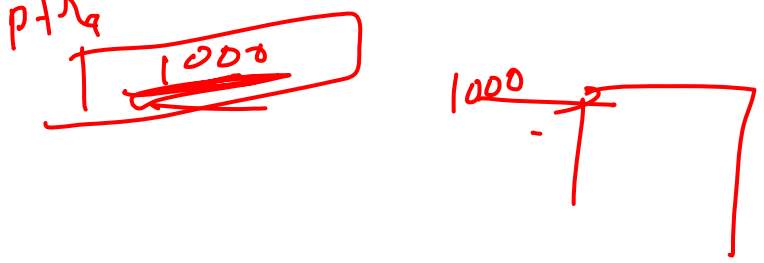
✗ Bcoz, ptr is constant pointer.

(*ptr)++; // 51

ptr = &y; // ptr 2000

Not allowed. Bcoz, the ptr itself is constant.

void *ptr;



x = *ptr;

printf( ) You donot know what datatype to read.

You know from where to read.
from address 1000 starting

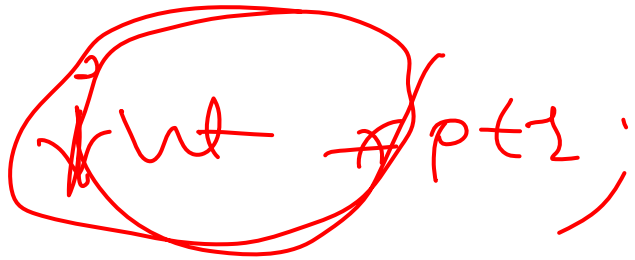
void * malloc(int);

memory allocation

dynamic memory allocation.

Heap memory area.

Type casting of pointers



- *ptr
- (int*) ptr
- void *

---

int *ptr = &x;

int ** pptr = &ptr;