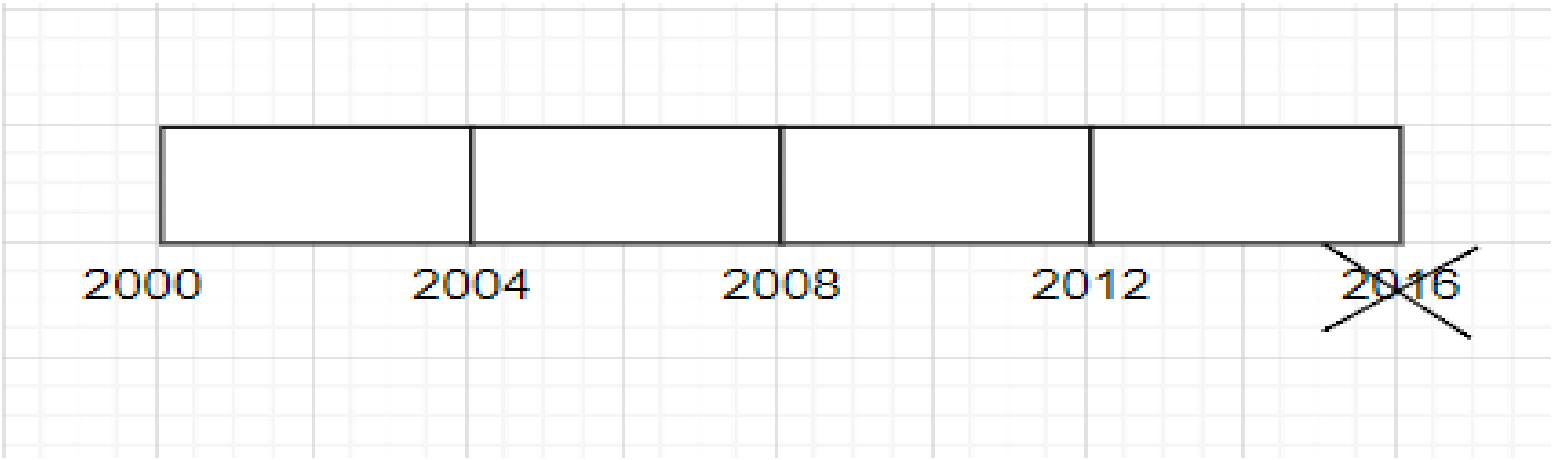
Pointer Arithmetic

- Through pointer arithmetic , we can access any no. of sequential storage. But it is necessary that access should be authorized .
- In below case, 2000-2004 , 2004-2008, 2008- 2012, 2012-2016 is authorized access but above 2016 there will be unauthorized access, that’s why if we try to access 2016 it will be unauthorized access.



Pointer Arithmetic

- Pointer arithmetic gives sequential access but the storage must be authorized otherwise it will go to accidental case

  (i.e 1. Program may run smoothly,

  2. Program may throw garbage value i.e unexcpected output

  3. Program may crash

So to get authorized access hence came array.)

ptr (2000)	(ptr + 0)	(ptr + 1)	(ptr + 2)	(ptr + 3)
char *	2000	2001	2002	2003
int*	2000	2004	2008	2012
double*	2000	2008	2016	2024
void*	2000	2000	2000	2000

C Pointers and Arrays

FirstBitSolutions.com
...Learn IT Bit by Bit...

1.Pointers and Arrays

```c
#include <stdio.h>
int main() {

  int i, x[6], sum = 0;

  printf("Enter 6 numbers: ");

  for(i = 0; i < 6; ++i) {
  // Equivalent to scanf("%d", &x[i]);
      scanf("%d", x+i);

  // Equivalent to sum += x[i]
      sum += *(x+i);
  }

  printf("Sum = %d", sum);

  return 0;
}
```

```
Enter 6 numbers: 2
3
4
4
12
4
Sum = 29
```

2. Arrays and Pointers

```c
#include <stdio.h>
int main() {

  int x[5] = {1, 2, 3, 4, 5};
  int* ptr;

  // ptr is assigned the address of the third element
  ptr = &x[2];

  printf("*ptr = %d \n", *ptr);   // 3
  printf("*(ptr+1) = %d \n", *(ptr+1)); // 4
  printf("*(ptr-1) = %d", *(ptr-1));  // 2

  return 0;
}
```

```
*ptr = 3
*(ptr+1) = 4
*(ptr-1) = 2
```

C Arrays

- An array is a collection of similar data items stored at contiguous memory locations and elements can be accessed randomly using indices of an array.
- Declaration of an array –

```
data_type array_name[array_size];
```

```
int marks[5];
```

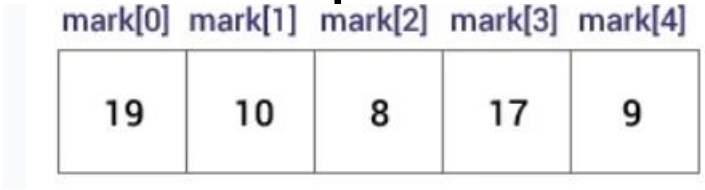
- Initialization of an array -

```
int mark[5] = {19, 10, 8, 17, 9};
```

or

```
int mark[] = {19, 10, 8, 17, 9};
```

- Pictorial representation -




```c
#include <stdio.h>

int main() {

  int values[5];

  printf("Enter 5 integers: ");

  // taking input and storing it in an array
  for(int i = 0; i < 5; ++i) {
     scanf("%d", &values[i]);
  }

  printf("Displaying integers: ");

  // printing elements of an array
  for(int i = 0; i < 5; ++i) {
     printf("%d\n", values[i]);
  }
  return 0;
}
```