

11c - Array of Pointer

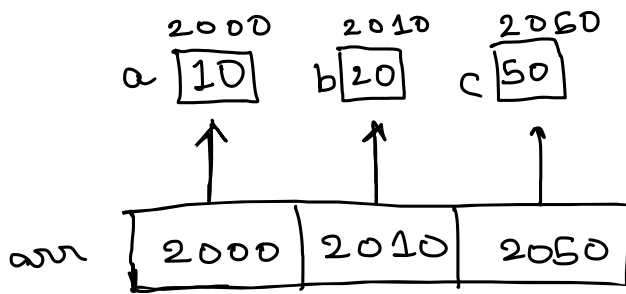
Saturday, 15 August 2026

17:56

```
int * arr [ 3 ] ;
```

or

```
int * (arr [ 3 ] ;
```



Accessing a value

If we want to access b i.e. 20
we can print:

→ b

→ *(arr [1])

→ ** (arr + 1)

first star points to
2010 i.e. the value
1

of arr[1].
Second one points
to the add value
that arr[1]
stores

→ arr[1][0]

Character Array

array_of_char_pointers.c > int main()

```
1  #include <stdio.h>
2
3  int main(){
4
5      char a = 'a', b = 'b', c = 'c';
6
7      char * arr[3];
8
9      arr[0] = &a;
10     arr[1] = &b;
11     arr[2] = &c;
12
13     for (int i = 0; i < 3; i++){
14         printf("Address %d has value %c\n", arr[i], *arr[i]);
15     }
16
17     return 0;
18 }
19
```

c - zsh

```
arnesh@Arneshs-MacBook-Air c % clear
arnesh@Arneshs-MacBook-Air c % ./a.out
Address 1872228027 has value a
Address 1872228026 has value b
Address 1872228025 has value c
arnesh@Arneshs-MacBook-Air c %
```

String Arrays

GO CLASSES **C Programming**

```
char *sports[5] = {  
    "golf",  
    "hockey",  
    "football",  
    "cricket",  
    "shooting"  
};
```

sports[1][0] = ?

Index	Address	String	Null Terminator Address
sports[0]	1000	g o l f \0	1004
sports[1]	1005	h o c k e y \0	1011
sports[2]	1012	f o o t b a l l \0	1020
sports[3]	1021	c r i c k e t \0	1028
sports[4]	1029	s h o o t i n g \0	1037

www.goclasses.in

array_of_string_pointers.c

```
1  #include <stdio.h>  
2  
3  int main() {  
4  
5      char a[] = "Hi";  
6      char b[] = "Hello";  
7      char c[] = "Bye";  
8  
9      char *arr[3];  
10  
11     arr[0] = a;  
12     arr[1] = b;  
13     arr[2] = c;  
14  
15     for (int i = 0; i < 3; i++) {  
16         printf("Address: %p, String: %s\n",
```

```

17         (void *)arr[i],
18         arr[i]);
19     }
20
21     return 0;
22 }

```

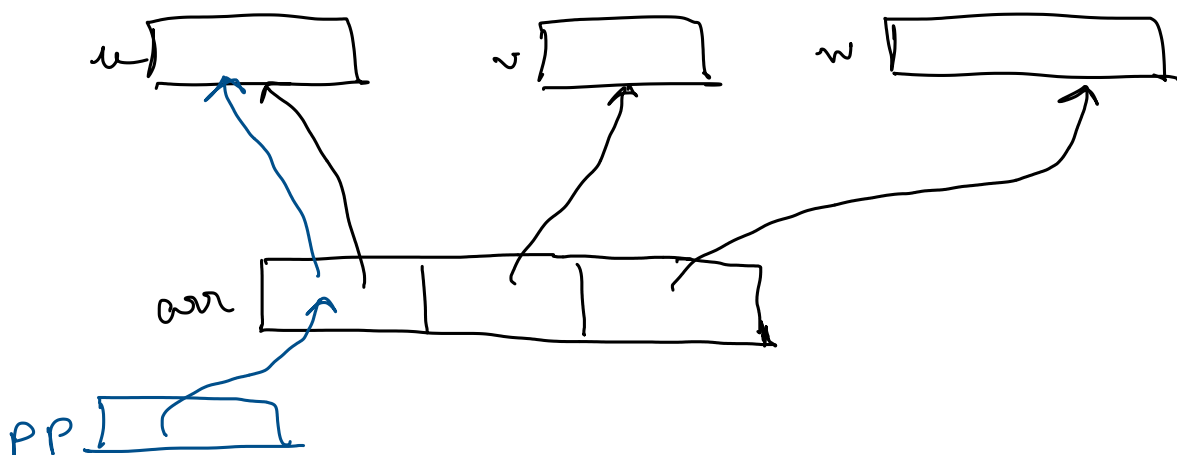
c - zsh

Last login: Mon Aug 17 14:28:45 on ttys001
arnesh@Arneshs-MacBook-Air c % gcc array_of_string_pointers.c
arnesh@Arneshs-MacBook-Air c % ./a.out
Address: 0x16ef9eeb8, String: Hi
Address: 0x16ef9eeb0, String: Hello
Address: 0x16ef9eeac, String: Bye
arnesh@Arneshs-MacBook-Air c %

Tips :

• `char * arr [ ] = { u , v , w } ;`

This means $\downarrow$ it's an array of pointers



• `char ** pp = arr`

`** pp` becomes double ptr
 it

since " points to arr[0]
which points to u[0].

NOTE: Make diagram while solving
the questions.