ASSIGNMENT

12

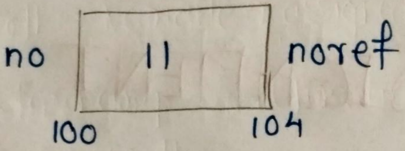
Q1] What is mean by reference in C++ ? Explain with example

Reference is a newly added datatype in C++. The concept of C++ reference is applicable in C++ & Java. Reference is considered as a Derived Data Type.

While we create a reference to an existing variable it is just considered as another name to the variable. It is just another name so that there is no separate memory allocation for reference

To create a reference we use '&' operator (reference operator) ie &=

Ex
```
int no = 11;
int &noref = no;
```



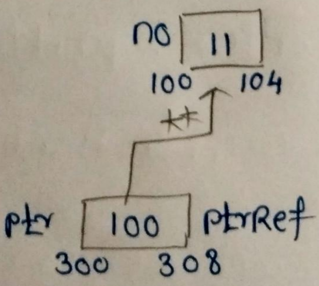
no is a variable of type integer & initialised with value 11. noref is a reference which refers to integer & currently it refers the variable no.

noref is anothername of an integer & the name of original variable is no.

We can also create reference which refers to any primitive data type

We can also create reference which refers to a pointer as a derived data type

Ex:
```
int * ptr = &no;
int * (&ptrRef)= ptr;
```



ptr is a pointer which points to integer & currently it holds the address of no. ptrRef is a reference which refers to integer pointer & currently it refers to the variable ptr.

When we create a reference there is no separate memory allocation for it. The name of reference as well as the name of original variable refers to the same memory location due to which the address & value are the same.

```
#include<iostream>
using namespace std;

int main()
{
    int no = 11;
    int &noref = no;
    int rahul = 99;
    int &pappu = rahul;
    int *ptr = &no;
    int *(&ptrRef) = ptr;
    cout<< "no:: "<< no << "\n"; //11
    cout<< "noref:: "<< noref << "\n"; //11
    cout<< "no:: "<< &no<< "\n";
    cout<< "noref:: " << &noref << "\n";
    cout<< "*ptr:: "<< *ptr<< "\n"; //11
cout << "*ptrref:: " << *ptrRef << "\n"; //11
return 0;
}
```

Q2] What is the difference b/w pointer & referenee.

Points	Pointer	Reference
Definition	A pointer is a variable that holds the memory address of another variable	A reference is an alias for an existing variable
Syntax	int *ptr = &x;	int &ref = x;
Null value	A pointer can be assigned nullptr or NULL	A reference cannot be null, it must be initialized
Reassignment	A pointer can be reassigned to point to another variable	A reference cannot be reassigned after initialization
Dereferencing	Requires dereferencing operator * to access the value (*ptr).	Automatically dereferenced
Size	A Pointer occupies memory	A reference does not occupy extra memory.
Modification	We can modify what the pointer points to	References are bound to one variable.

Q3] Explain different ways to call a function in C++.

There are three ways to call a function in C++ :-

1> Call by Value

2> Call by Address

3> Call by Refcrence

1> Call by Value

A copy of the actual argument is passed to the function. Changes made inside the function do not affect the original variable

2> Call by Address

The address of the actual argument is passed to the function. Changes made inside the function directly affect the original variable.

3> Call by Reference

A reference to the actual argument is passed to the function. Changes made inside the function directly affect the original variable.

```
#include <iostream>
using namespace std;
void callByValue(int x){
      x=20;
 }
void callByAddress (int *x){
      *x = 30;
 }
void callByReference (int &x) {
      x=40;
}
void main() {
    int num =10;
callByValue(num);
cout<<"After Call by Value: num = "<< num<<endl;

callByAddress (&num);
cout<<"After Call by Address: num = "<<num<<endl;

callByReference(num);
cout<<"After Call by Reference: num = "<<num<<endl;
return 0;
}
```

O/P:-

```
After Call by Value: num=10
After Call by Address: num=30
After Call by Reference: num=40
```

Q4] Draw Symbol Table

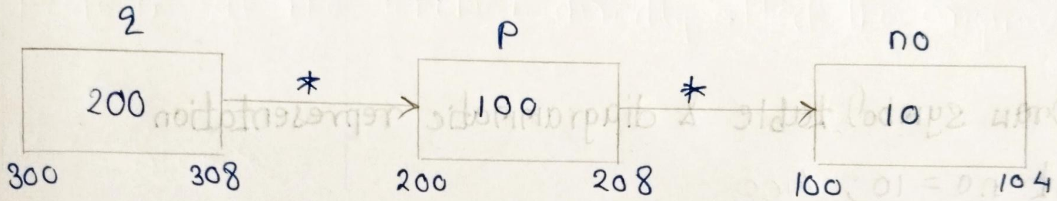
```
int no = 10; //100
int *p = &no; //200
int **q = &p; //300
```

Name	Address	Size	Value	Data Type	Another Name
no	100	4	10	int	*p, **q
*p	200	8	10	int	&no
**q	300	8	10	int	&p

Q5] Draw symbol table & diagrammatic representation

```
int no = 10; //100
int *p = &no; //200
int *q = &p; //300
```

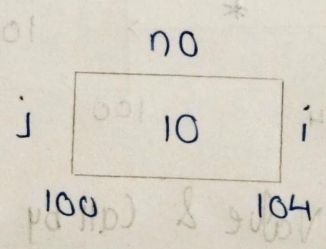
Name	Address	Size	Value	Data Type	Another Name
no	100	4	10	int	*p
*p	200	8	10	int	&no
*q	300	8	100	int	&p



Q6] Draw symbol table & diagrammatic representation

```
int no = 10;
int &i = no;
int &j = no;
```

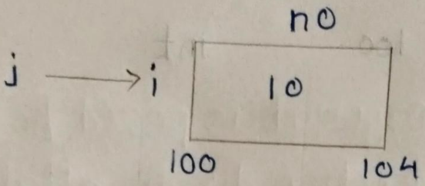
Name	Address	Size	Value	Data Type	Another Name
no	100	4	10	int	i, j



Q7] Draw symbol table & diagrammatic representation

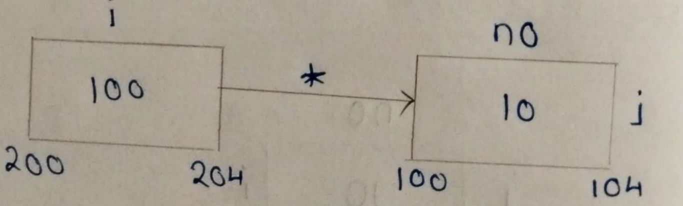
Q8]
```
int no=10; //100
int &i=no;
int &j=i;
```

Name	Address	Size	Value	Data Type	Another Name
no	100	4	10	int	i, j



Q~~8~~ 9] Draw symbol table & diagrammatic representation
```
int no=10; //100
int *i=&no; //200
int *(&j)=i;
```

Name	Address	Size	Value	Data Type	Another Name
no	100	4	10	int	*i, ~~j~~ *(&j)
*i	200	8	10	int	&no



Q10] What is mean by Call by Value & Call by Reference?

Refer Q3