

Solution/Model Answer to question paper of IAT-I
Programming in C++ - 10EC661
VI sem ECE A & B
Dr. P. N. Singh, Professor(CSE)

1 a) Explain difference in C & C++.

5

Ans:

- C++ is super set of C.
- Developer of C++ Bjarne Stroustrup named it first "C with Classes"
- C is Procedure Oriented programming language. C is also known as middle level language as it permits low level programming also.
- C++ is Object Oriented programming language & supports procedure oriented programming language also.
- Class is the important feature used in C++ which is not available in C.
- C is also known as middle level language as it permits low level programming also but C++ is regarded as high level language & object
- Pass by reference is permitted in C++ but in C pointer is used for pass by reference.
- C programs are compatible by C++ compiler.

1 b) Define the terms class & object.

5

Ans:

Class : A class is a group of similar objects. It represents object's type. A class is extensible program-code template for objects. A class is abstract data type (ADT).

Object: Anything having characteristics & behaviour is an object. All real world entities are objects.

"AN OBJECT IS AN INSTANCE OF A CLASS"

Example: Michale Jacson is an object of class "Rock-Musicians"

Padma is an object of class BE.

2 a) Explain #define directive with examples.

6

Ans:

#define is a pre-processor directive to define macros and symbolic constants.

#define directive causes the compiler to substitute the definition(token string) for each occurrence of symbolic constant or macro in the source program. Macros are expanded in which way they are defined.

Examples:

```
#define PI 3.14          /* each occurrence of PI will be replaced by 3.14 */
#define max(a,b) a>b?a:b /* macro will be expanded in which way they are defiened */
```

2 b) Write output of following code:

```
#include <iostream.h>
#define max(a,b) a>b?a:b
main( )
{
    int x=2, y=3;
    int z = 10+max(x,y);
    cout << z;
    return 0;
}
```

4

Ans:

z=10+2>3 ? 2 : 3;

10+2 = 12 is compared with 3 which is true.

(Arithmetic operator has greater precedence over relational operators)

So, $z = 2$ & the output will be 2

3. Differentiate while and do...while loop with examples.

10

Ans:

while statement : It is a loop statement.

syntax:

```
while (expression)
{
    Statement1;
    Statement2;
    .....;
}
```

do...while statement : It is also a loop statement.

syntax:

```
do
{
    Statement1;
    Statement2;
    .....;
}while (expression);
```

Difference between while & do...while loop

- *While* is a pre-tested loop/entry controlled loop and *do... while* is post-tested loop
- Post-tested loop must execute at least once because condition is checked at the end of loop

Examples:

int main()

```
{
    int n = 1;
    while (n <= 20)
    {
        cout << n << endl; // prints 1 to 20 & terminated when n becomes 21
        n++;
    }
    return 0;
}
```

int main()

```
{
    int n = 100;
    do
    {
        cout << n << endl; // prints 100 ( once only)
        n++;
    } while(n<=20); // condition is checked here so now nothing will be printed
    return 0;
}
```

4. Define user defined data types typedef and enum with examples.

10

Ans:

typedef : By typedef keyword one can define user defined data type.

```
typedef int purnank;
```

Now for int we can use purnank (int will also work)

Example:

```
purnank n = 20;
```

enum: By keyword enum we can define enumerated (listed) data type so that value of the variable can be taken from listing only.

It consists of a set of named values. Those may be called element or member or enumerator.

```
enum day { Sun, Mon, Tue, Wed, Thu, Fri, Sat };
```

Here day can be used as data type.

```
day holiday;
```

Value of holiday variable of day data type may be one from the list. Elements can be accessed by number also (by default starting from 0 to total elements – 1)

Example program (Program not asked but for depth of enum)

```
#include <iostream.h>
```

```
int main()
{
    enum day{sun,mon,tue,wed,thu,fri,sat};
    day holiday;
    int n;
    cout << "\nEnter day : ";
    cin >> n;
    switch (n)
    {
        case 0: cout << "Your holiday is sunday\n";
                break;
        case mon: cout << "Your holiday is Monday\n";
                break;
        case 2: cout << "Your holiday is Tuesday\n";
                break;
        case wed: cout << "Your holiday is Wednesday\n";
                break;
        case thu: cout << "Your holiday is Thursday\n";
                break;
        case fri: cout << "Your holiday is Friday\n";
                break;
        case sat: cout << "Your holiday is Saturday\n";
                break;
    }
    holiday=mon;
    cout << "\nNow holiday is " << holiday+1;
    n=thu;
    cout << "\nNow holiday is " << n;

    return 0;
}
```

5. Write a C++ program to check a string whether it is palindrome.

10

```
// Program to check a string whether it is palindrome
#include <iostream.h>

int main( )
{
    char str[50];
    int len, x,y;
    cout << "Enter string : ";
    cin.getline(str,50);    // to read string with blanks
    len=0;
    while(str[len] != '\0')
        len++;
    y=len-1;                // for last character before NULL
    int palin=1;
    for(x=0;x<=len/2;x++, y--)
        if(str[x] != str[y]) // if any mismatch occurs
        {
            palin=0;
            break;
        }
    if(palin)
        cout << "Yes, string " << str << " is a palindrome\n";
    else
        cout << "No, string " << str << " is not a palindrome\n";

    return 0;
}
```

Expected Output:

Enter String: able was I ere I saw elba

Yes, string able was I ere I saw elba is a palindrome

6. Generate prime numbers within given range by a C++ program.

10

Ans:

```
//Prime numbers within given range
#include <iostream>
#include <math>
using namespace std;
int main( )
{
    long int n,n1,n2,d;
    cout << "Enter range : from ? ";
    cin >> n1;
    cout << " to? ";
    cin >> n2;
    cout << "Prime numbers from " << n1 << " to " << n2 << ":\n";
    for(n=n1;n<=n2;n++)
    {
        prime=1;
        for(d=2;d<=sqrt(n);d++)
            if(n%d==0)    // testing divisibility from 2 to square root of n (integer)
```

```

        {
            prime=0;
            break;
        }
    if(prime) // if prime is still 1 means not exactly divided by any number
        cout << " " << n;
    }
    return 0;
}

```

Expected output:

**Enter range: from ? 2
to? 100**

Prime numbers from 2 to 100:

2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97

7. Write a C++ program to compute comb(n,r) i.e. $n!/(r!(n-r)!)$ using recursive function to find factorial.

10

Ans:

```

#include <iostream>
using namespace std;

long int factorial(int n)
{
    if(n==0 || n==1)
        return 1;
    else
        return n*factorial(n-1);
}

int main()
{
    int n,r,comb;
    cout << "Enter values of n & r : ";
    cin >> n >> r;
    comb=factorial(n)/(factorial(r) * factorial(n-r));
    cout << "Result = " << comb << "\n";
    return 0;
}

```

Expected output:

Enter values of n & r: 5 3

Result = 10

8. Define switch statement with example program.

10

Ans:

switch statement is conditional statement of c.

syntax:

switch (var)

```

{
    case 'expr1' : statement1;
                  statement2;
                  .....;
                  break;
    case 'expr2' : statement1;
                  statement2;
                  .....;
                  break;
    .....
    default :    statement1;
               statement2;
               .....;
}

```

If case is matched, related statements are executed and break terminates the construct. If break is not given, then it will execute statements for another case.

Example program:

```

/* to compute bonus based on grade */
#include <iostream>
using namespace std;
int main( )
{
    char grade;
    int bonus;
    cout << "Enter grade (a,b,c,d) of employee : "
    grade=cin.get();
    switch(grade)
    {
        case 'a' :
        case 'A': bonus = 5000;
                 break;
        case 'b':
        case 'B':
        case 'c':
        case 'C': bonus = 10000;
                 break;
        default: bonus=15000;
    }
    cout << "Bonus = Rs. " << bonus << "\n";
    return 0;
}

```

Expected Output:

Enter grade (a,b,c,d) of employee : c
Bonus = Rs. 10000

"Think like a person of action. Act like a person of thought."
