```
#include <iostream>
#include <string.h>
using namespace std;
class item
{
   char itemcode[6];
   char itemname[20];
   float rate;

public:
   item()
   {
      strcpy(itemcode, "abc");
      strcpy(itemname, "hello");
      rate = 10;
   }
   item(const char *a, const char *b, float r)
   {
      for (int i = 0; i < 6; i++)
         itemcode[i] = a[i];
      for (int i = 0; i < 20; i++)
         itemname[i] = b[i];

      rate = r;
   }

   void set_itemcode(const char *a)
   {
      for (int i = 0; i < 6; i++)
         itemcode[i] = a[i];
   }

   void set_itemname(const char *a)
   {
      for (int i = 0; i < 20; i++)
         itemname[i] = a[i];
   }
   void set_rate(float a)
   {
      rate = a;
   }

   char *get_itemcode()
   {
      return itemcode;
   }

   char *get_itemname()
   {
      return itemname;
   }
   float get_rate()
   {
      return rate;
   }
   void display()
```

```
    {
        int i;
        cout << "\n Itemcode : ";
        for (i = 0; i < strlen(itemcode); i++)
        {
            cout << itemcode[i];
        }
        cout << "\n Itemname : ";
        for (i = 0; i < strlen(itemname); i++)
        {
            cout << itemname[i];
        }
        cout << "\n   Rate   : " << rate;
    }
};

int main()
{
    item A1("ABh", "asd", 23.23), A2;
    // char a[12]= {"Abhishek"};
    // item A2("xyz","aaaa",25.50);

    // for(int i=0;i<12;i++)
    //     cout<<a[i];
    A1.display();
    A2.display();
    A2.set_itemcode("box");
    A2.set_itemname("cup001");
    A2.set_rate(49.99);
    A2.display();
    // for (int i = 0; i < 6; i++)
    //     cout << "\n code : " << A2.get_itemcode[i];
    cout << "\n Item name : " << A2.get_itemname();
    cout << "\n rate : " << A2.get_rate();

    A2.set_itemname("cupsABC");
    cout << "\n Item name : " << A2.get_itemname();
    return 0;
}
```

3 - Bank Account

```cpp
// class of bank account

#include <iostream>
using namespace std;

class BankAcc
{
    int balance;
    int Acc_no;
    string name;
    string type;

public:
    BankAcc()
    {
        name = "Unkown";
        type = "Null";
        Acc_no = 0;
        balance = 0;
    }
    BankAcc(int B, int A, string N, string T)
    {
        name = N;
        type = T;
        Acc_no = A;
        balance = B;
    }
    void display()
    {
        cout << "\n  Name    : " << name;
        cout << "\n Balance : " << balance;
    }
    void withdraw(int a)
    {
        if (balance == 0)
        {
            cout << "\n Account has zero balance.";
        }
        else
        {
            balance = balance - a;
        }
    }
    void deposit(int a)
    {
        balance = balance + a;
    }

    void init(int A, int B, string N, string T)
    {
        name = N;
        type = T;
        Acc_no = A;
        balance = B;
    }
    ~BankAcc()
    {
        cout << "\nDestructor is called..!!";
    }
};

int main()
{
    BankAcc c1, c2;
    c1.init(4096, 1000, "Ajay", "Saving");
    c1.display();
    c1.deposit(2500);
    c1.display();

    c1.withdraw(500);
    c1.display();

    c2.display();

    c2.withdraw(500);
    c2.display();

    return 0;
}
```

4 - Time class

```cpp
#include <iostream>
using namespace std;
class time
{
    int hh, mm, ss;

public:
    time()
    {
        hh = 12;
        mm = 00;
        ss = 00;
    }
    void set_time(int a, int b, int c)
    {
        if ((a >= 1) && (a <= 24))
        {
            hh = a;
        }
        if ((b >= 1) && (b <= 59))
        {
            mm = b;
        }
        if ((c >= 1) && (c <= 59))
        {
            ss = c;
        }
    }
    void add_time(int h)
    {
        for (int i = 1; i <= h; i++)
        {
            if ((hh + i) < 24)
            {
                hh = hh + 1;
            }
            else
            {
                hh = 0;
            }
        }
    }
    void display()
    {
        cout << "\n | " << hh << " : " << mm << " : " << ss << " | ";
    }

    void compare(time *b)
    {
        if (hh == b->hh)
        {
            if (mm > b->mm)
            {
                display();
            }
            else
            {
                if (mm < b->mm)
                {
                    b->display();
                }
                else
                {
                    cout << "\n Both are equal..!!";
                }
            }
        }
        if (hh != b->hh)
        {
            if (hh > b->hh)
            {
                display();
            }
            else
            {
                if (hh < b->hh)
                {
                    b->display();
                }
            }
        }
    }
};

int main()
{
    time t1, t2;
    t1.display();
    t2.display();

    t2.set_time(23, 45, 20);
    t2.display();
    t1.add_time(10);
    t1.display();
    t2.add_time(4);
    t2.display();

    t1.compare(&t2);

    return 0;
}
```

2 - Funtion Overloading

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

void display(char[]);
void display(char[], char[]);

int main()
{
    char s[10] = "Prudent";
    char s1[10] = "Academy";
    display(s);
    display(s, s1);
    return 0;
}

void display(char a[])
{
    for (int i = 0; i < 10; i++)
        cout << a[i];

    cout << "\n";
}

void display(char a[], char b[])
{
    for (int i = 0; i < 10; i++)
        cout << a[i];

    cout << " ";
    for (int i = 0; i < 10; i++)
        cout << b[i];

    cout << "\n";
}
```

Assignment – 2

pattern - 1

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

int main()
{
    int i, j;
    for (i = 1; i <= 5; i++)
    {
        cout << "\n";
        for (j = i; j <= 5; j++)
            cout << "* ";

        if (i > 1)
        {
            for (j = 1; j <= (4 * (i - 1)); j++)
                cout << " ";
        }

        for (j = i; j <= 5; j++)
            cout << " *";
    }

    //------------------------------------------------------

    for (i = 4; i >= 1; i--)
    {
        cout << "\n";
        for (j = i; j <= 5; j++)
            cout << "* ";

        if (i > 1)
        {
            for (j = 1; j <= (4 * (i - 1)); j++)
                cout << " ";
        }

        for (j = i; j <= 5; j++)
            cout << " *";
    }


    return 0;
}
```

1.
```
* * * * *   * * * * *
* * * *       * * * *
* * *           * * *
* *               * *
*                   *
* *               * *
* * *           * * *
* * * *       * * * *
* * * * *   * * * * *
```

pattern - 2

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

int main()
{
    int i, j;
    for (i = 1; i <= 9; i++)
    {
        cout << "\n";
        if ((i == 1) || (i == 5) || (i == 9))
        {
            cout << " ";
            for (j = 1; j <= 3; j++)
                cout << " *";
        }
        else
        {
            cout << "*";
            for (j = 1; j <= 7; j++)
                cout << " ";

            cout << "*";
        }
    }
    return 0;
}
```

2.

```
    * * *
  *       *
  *       *
  *       *
    * * *
  *       *
  *       *
  *       *
    * * *
```

pattern - 3

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

int main()
{
    int i, j;
    for (i = 1; i <= 3; i++)
    {
        cout << "\n";
        if (i > 1)
        {
            for (j = 1; j <= ((2 * i) - 2); j++)
                cout << " ";
        }
        for (j = 5; j >= (2 * i - 1); j--)
            cout << "* ";
    }
    //-------------------------------------------------------

    for (i = 3; i >= 1; i--)
    {
        cout << "\n";
        if (i > 1)
        {
            for (j = ((2 * i) - 2); j >= 1; j--)
                cout << " ";
        }
        for (j = 5; j >= (2 * i - 1); j--)
            cout << "* ";
    }
    return 0;
}
```

```
3.  * * * * *
      * * *
        *
        *
      * * *
    * * * * *
```