```
//variables, operators, data types, arithmetic operations, conditions)
#include<iostream>
using namespace std;
int main()
  cout << "Aleena Fatima Khalid";
  cout << "\nMan, i missed cpp!";</pre>
  //types of variables:
     int - stores integers (whole numbers), without decimals, such as 123 or -123
     double - stores floating point numbers, with decimals, such as 19.99 or -19.99
     char - stores single characters, such as 'a' or 'B'. Char values are surrounded by single
quotes
     string - stores text, such as "Hello World". String values are surrounded by double quotes
     bool - stores values with two states: true or false
  */
  //declaring all variables and initalising
  int number = 0;
  double doubleNum = 0.0;
  char c = 'a';
  string s = "";
  bool flag = 0;
  //taking input
  cout << "Enter num:";
  cin >> number;
  cout << "Enter double num:";
  cin >> doubleNum;
  cout << "Enter char:";</pre>
```

```
cin >> c;
cout << "Enter string:";
cin >> s;
cout << number << " " << doubleNum << " " << c << " " << s << " ";
//arithemtic operations and conditions
if (number > 0 && number < 10)
{
  number = number + 10;
  cout << "\nNumber after addition of 10: " << number;
}
else if (number >= 10)
  number = number - 2;
  cout << "\nnumber after substraction of 2: " << number;</pre>
}
else
  cout << "\nNumber: " << number;</pre>
if (s == "hello")
  doubleNum = doubleNum* number;
  cout << "\nAfter multiplication: " << doubleNum;</pre>
}
else
{
  doubleNum = doubleNum / 2;
  cout << "\nAfter division: " << doubleNum;</pre>
}
return 0;
```

```
umber = number + 10;
  Microsoft Visual Studio Debug Console
 Aleena Fatima Khalid
 Man, i missed cpp!Enter num:5
 Enter double num:3.8
 Enter char:s
 Enter string:hi
5 3.8 s hi
Number after addition of 10: 15
 After division: 1.9
 G:\Users\HP\BTW Task 1\Debug\BTW Task 1.exe (process 19056) exited with code 0.
 Press any key to close this window . . .
 Microsoft Visual Studio Debug Console
Aleena Fatima Khalid
Man, i missed cpp!Enter num:15
Enter double num:5.8
Enter char:j
Enter string:hello
15 5.8 j hello
number after substraction of 2: 13
After multiplication: 75.4
G:\Users\HP\BTW Task 1\Debug\BTW Task 1.exe (process 17576) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options-:
le when debugging stops.
Press any key to close this window . . .
```

```
//Practice Switch, Continue, Break, For Loop, While Loop, do While Loop in C++
#include<iostream>
using namespace std;
int main() {
    int arr[5] = { 1,2,3,4,5 }, choice = 0;
```

```
for (int I = 0; I < 5; I + +) {
        cout << "\nenter a number between 1-5: ";
        cin >> choice;
        switch (choice) {
        case 1: {
                for (int i = 0; i < 5;i++)
                         cout << "\nIn for loop: number - " << arr[i];</pre>
                break;
        }
        case 2: {
                int j = 0;
                while (j < 5)
                         cout << "\n In while loop: number - " << arr[j];
                        j++;
                break;
        }
        case 3: {
                int k = 0;
                do {
                        cout << "\n In do while loop: number - " << arr[k];</pre>
                         k++;
                \} while (k < 5);
                break;
        }
        case 4: {
                for (int i = 0; i < 5;i++)
                         if (i == 4)
                                 continue;
                         cout << "\nIn for loop with continue: number - " << arr[i];
                break;
        }
        default:
                cout << "\nin default";</pre>
```

```
}
}
return 0;
   Microsoft Visual Studio Debug Console
  enter a number between 1-5: 1
  In for loop: number - 1
  In for loop: number - 2
  In for loop: number - 3
  In for loop: number - 4
  In for loop: number - 5
  enter a number between 1-5: 2
   In while loop: number - 1
   In while loop: number - 2
   In while loop: number - 3
   In while loop: number - 4
   In while loop: number - 5
   enter a number between 1-5: 3
   In do while loop: number - 1
   In do while loop: number - 2
   In do while loop: number - 3
   In do while loop: number - 4
   In do while loop: number - 5
   enter a number between 1-5: 4
  In for loop with continue: number - 1
  In for loop with continue: number - 2
  In for loop with continue: number - 3
  In for loop with continue: number - 4
  enter a number between 1-5: 5
  in default
  G:\Users\HP\BTW Task 2\Debug\BTW Task 2.exe (proce
  Press any key to close this window . . .
```

```
#include<iostream>
using namespace std;
class stationery {
       //private attributes
       static int id;
       string name;
       string color;
       double price;
public:
       //constructors
       stationery()
               id++; name = ""; color = ""; price = 0.0;
       stationery(string name, string color, double price)
       {
               id++;
               this->name = name;
               this->color = color;
               this->price = price;
       }
       stationery(stationery& copy)
               id++;
               name = copy.name;
               color = copy.color;
               price = copy.price;
       }
       //getter setters
       void setName(string name) { this->name = name; }
       void setColor(string color) { this->color = color; }
       void setPrice(double price) { this->price = price; }
```

```
string getName() const { return name; }
        string getColor() const { return color; }
        double getPrice() const { return price; }
        int getID() { return id; }
        //other class methods
        void typeOfStationery() { cout << "\nl am a stationery item."; }</pre>
};
int stationery::id = 0;
class pen : public stationery {
public:
        void typeOfStationery() { cout << "\n I am a pen."; }</pre>
};
class shop {
public:
        string shopName;
        stationery s1;
        pen p1;
        shop() { shopName = "";}
        shop(string s) { shopName = s; }
};
int main()
        stationery s2("inkpen", "purple", 35);
        cout << "First object ID : "<<s2.getID()<<endl;</pre>
        stationery s3(s2);
        cout << "\n second copied object: ";
        cout << s3.getID() << endl;
        cout << s3.getName() << endl;
        cout << s3.getColor() << endl;</pre>
        cout << s3.getPrice() << endl;</pre>
```

```
stationery s4("gelpen", "purple", 35);
    cout << "third object ID: " << s4.getID() << endl;
    shop sh1;
    cout << "\nAccessing stationery's child pen from shop: ";</pre>
    sh1.p1.typeOfStationery();
    return 0;
    Microsoft Visual Studio Debug Console
   First object ID : 1
s s second copied object: 2
   inkpen
//ppurple
strthird object ID : 3
douAccessing stationery's child pen from shop:
    I am a pen.
ic:G:\Users\HP\BTW Task 2\Debug\BTW Task 2.exe (proce
   Press any key to close this window . . .
```

```
#include<iostream>
using namespace std;
//pointers, arrays, linked lists
void arrayQueens()
```

```
{
        int queens[4][4] = {
                \{0,0,0,0\}
                \{0,0,0,0\},\
                \{0,0,0,0\},\
                \{0,0,0,0\}
        };
        cout << "4 queens problem initial board: \n";
        for (int i = 0; i < 4;i++) {
                cout << endl;
                for (int j = 0; j < 4; j++) {
                        cout << "|" << queens[i][j];
                cout << "|";
        }
        //4 queens problem solution
        queens[0][2] = 1;
        queens[1][0] = 1;
        queens[2][3] = 1;
        queens[3][1] = 1;
        cout << "\n\n4 queens problem solution board: \n";</pre>
        for (int i = 0; i < 4; i++) {
                cout << endl;
                for (int j = 0; j < 4; j++) {
                        cout << "|" << queens[i][j];
                cout << "|";
        }
        //using pointer notation - array pointer relationship
        int* q = NULL;
        q = &queens[0][0];
        cout << "\n\n4 queens problem solution board through pointers: \n";</pre>
        for (int i = 0; i < 4;i++) {
                cout << endl;
                for (int j = 0; j < 4; j++) {
                        cout << "|" << *(q + (i * 4) + j);
                cout << "|";
        }
}
```

```
//linked list
class Node {
public:
       int data;
       Node* next;
};
//print link list data
void printLinkedList(Node* n)
       cout << endl << "printing II: \n";
       while (n != NULL) {
              cout << n->data << " ";
               n = n->next;
       }
}
void ReverseLinkedList(Node* n) {
       Node* current = NULL, * prev = NULL, * next = NULL;
       current = n;
       while (current != NULL) {
               next = current->next;
               current->next = prev;
               prev = current;
               current = next;
       }
       n = prev;
       printLinkedList(n);
void LinkedList()
       Node* head = NULL;
       Node* second = NULL, * third = NULL, * fourth = NULL;
       //creating new nodes
       head = new Node();
       second = new Node();
       third = new Node();
       fourth = new Node();
       head->data = 10; head->next = second;
```

```
second->data = 20; second->next = third;
third->data = 30; third->next = fourth;
fourth->data = 40; fourth->next = NULL; //assign head again for doubly II

printLinkedList(head);
cout << "\nReversed: ";
ReverseLinkedList(head);
}
int main()
{

arrayQueens();
LinkedList();
return 0;
}</pre>
```

```
Microsoft Visual Studio Debug Console
4 queens problem initial board:
|0|0|0|0|
0 0 0 0
0 0 0 0
0 0 0 0
4 queens problem solution board:
|0|0|1|0|
1 0 0 0 0
[0]0[0]1
0 1 0 0
4 queens problem solution board through pointers:
|0|0|1|0|
1 0 0 0
|0|0|0|1|
0 1 0 0
printing 11:
10 20 30 40
Reversed:
printing 11:
40 30 20 10
```