

**PRACTICAL NO. 1**

- AIM:** a) Find the position of an element from 10 numbers .
- b) Find whether the number is even/odd without using the % operator.
- c) Write a program to demonstrate the switch case.
- 

**THEORY:**

Find the position of an element from 10 numbers

The linear search algorithm works by iterating through each element in the list until it finds the target element. If the target element is found, the position/index of the element is returned. If the target element is not found, then -1 is returned to indicate that the element is not present in the list.

**Algorithm:**

1. Declare and initialize an array.
2. Calculate the length of the declared array.
3. Loop through the array by initializing the value of variable "i" to 1 (because first even positioned element lies on  $i = 1$ ) then incrementing its value by 2, i.e.,  $i = i + 2$ .
4. Print the elements present in even positions.

**Code:**

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int arr[10] = {2, 4, 6, 8, 10, 12, 14, 16, 18, 20};
```

```
    int elem = 12;
```

```
    int pos = -1;
```

```
    for (int i = 0; i < 10; i++) {
```

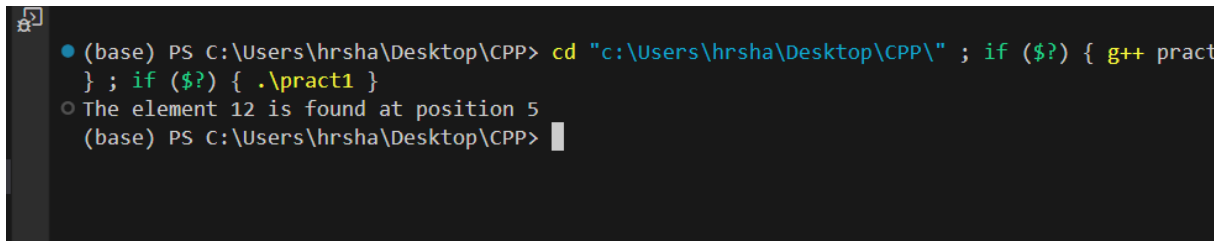
```

    if (arr[i] == elem) {
        pos = i;
        break;
    }
}

if (pos != -1) {
    cout << "The element " << elem << " is found at position " << pos << endl;
} else {
    cout << "The element " << elem << " is not found in the array" << endl;
}

return 0;
}

```



```

(base) PS C:\Users\hrsha\Desktop\CPP> cd "c:\Users\hrsha\Desktop\CPP\" ; if ($?) { g++ pract1.cpp ; if ($?) { .\pract1 } }
The element 12 is found at position 5
(base) PS C:\Users\hrsha\Desktop\CPP>

```

---

Find whether the number is even/odd without using the % operator.

### Theory

The modulus operator % returns the remainder of a division operation. To determine whether a number is even or odd without using the modulus operator, we can use integer division and multiplication to check if the result is equal to the original number.

### Algorithm

1. Read the number from the user.
2. Divide the number by 2.

3. Multiply the result by 2.
4. Check if the result is equal to the original number.
5. If the result is equal to the original number, then the number is even. Otherwise, it is odd.
6. Print the result.

### Code

```
#include <iostream>

using namespace std;

int main() {

    int num;

    cout << "Enter a number: ";

    cin >> num;

    if ((num / 2) * 2 == num) {

        cout << num << " is even" << endl;

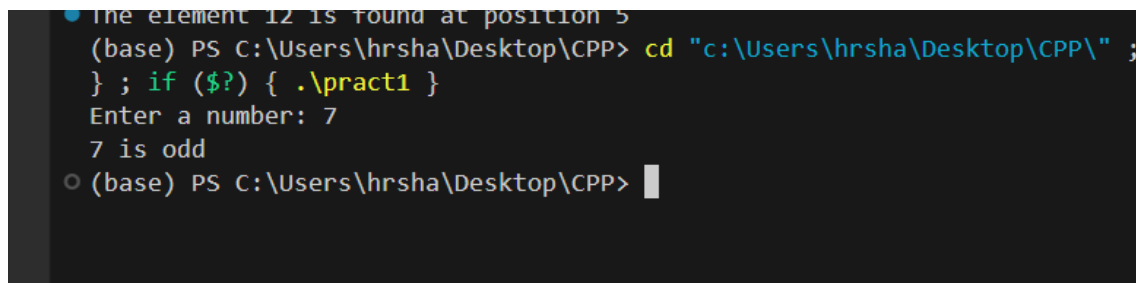
    } else {

        cout << num << " is odd" << endl;

    }

    return 0;

}
```



```
• The element 12 is found at position 5
(base) PS C:\Users\hrsha\Desktop\CPP> cd "c:\Users\hrsha\Desktop\CPP\" ;
} ; if ($?) { .\pract1 }
Enter a number: 7
7 is odd
○ (base) PS C:\Users\hrsha\Desktop\CPP> █
```

---

Write a program to demonstrate the switch case.

**Theory:**

The switch case statement is a control flow statement used in programming languages like C, C++, Java, and others. It allows you to execute different sections of code based on the value of a variable or expression.

**Algorithm:**

1. Start the program.
2. Declare a variable 'choice' to store user input.
3. Display a menu of options to the user.
4. Read the user's input and store it in the 'choice' variable.
5. Use a switch case statement to execute different code blocks based on the value of 'choice'.
6. Case 1: Display a message saying "You chose option 1".
7. Case 2: Display a message saying "You chose option 2".
8. Case 3: Display a message saying "You chose option 3".
9. Default case: Display a message saying "Invalid choice".
10. End the switch case statement.
11. End the program.

**Code**

```
#include <iostream>

#include <iostream>

using namespace std;
```

```
int main() {

    int num1, num2;

    cout << "Enter the num 1 : ";

    cin >> num1;
```

```
cout << "Enter the num 2 : ";

cin >> num2;


int choice;

cout << "Menu:\n";

cout << "1. Add\n";

cout << "2. Subtract\n";

cout << "3. Multiply\n";

cout << "4. Divide\n";

cout << "Enter your choice: ";

cin >> choice;


switch(choice) {

    case 1:

        cout << "Add selected" << endl;

        cout << num1 + num2;

        break;

    case 2:

        cout << "Subtract selected" << endl;

        cout << num1 - num2;

        break;

    case 3:

        cout << "Multiply selected" << endl;

        cout << num1 * num2;
```

```

        break;

    case 4:

        cout << "Divide selected" << endl;

        cout << num1 / num2;

        break;

    default:

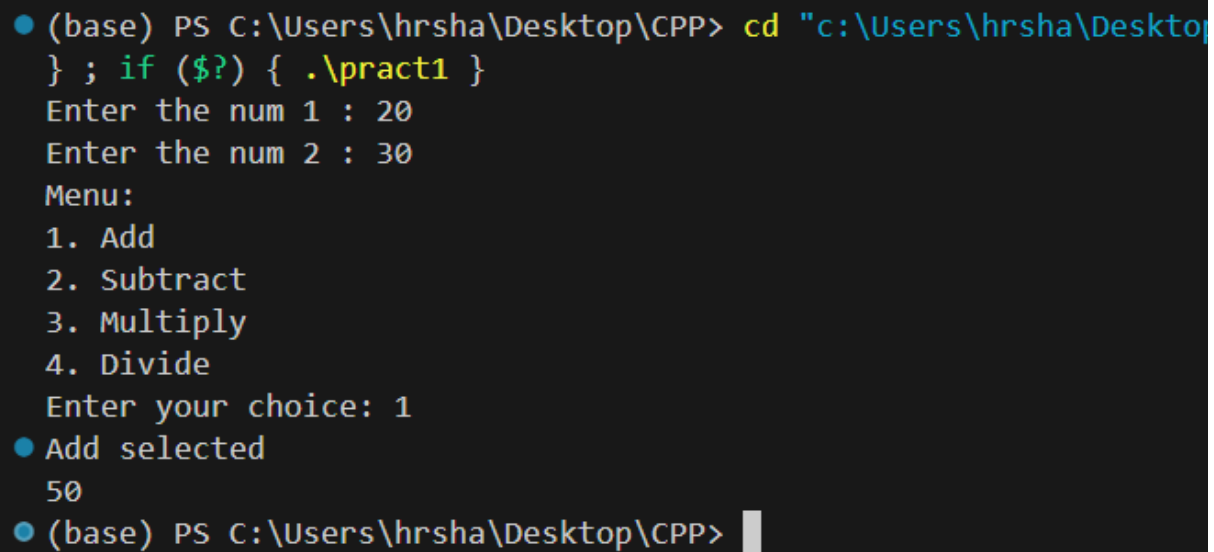
        cout << "Invalid choice" << endl;

    }

    return 0;

}

```



```

● (base) PS C:\Users\hrsha\Desktop\CPP> cd "c:\Users\hrsha\Desktop\CPP" ; if ($?) { .\pract1 }
Enter the num 1 : 20
Enter the num 2 : 30
Menu:
1. Add
2. Subtract
3. Multiply
4. Divide
Enter your choice: 1
● Add selected
50
● (base) PS C:\Users\hrsha\Desktop\CPP>

```

## CONCLUSION:

Thus we have successfully executed programs .