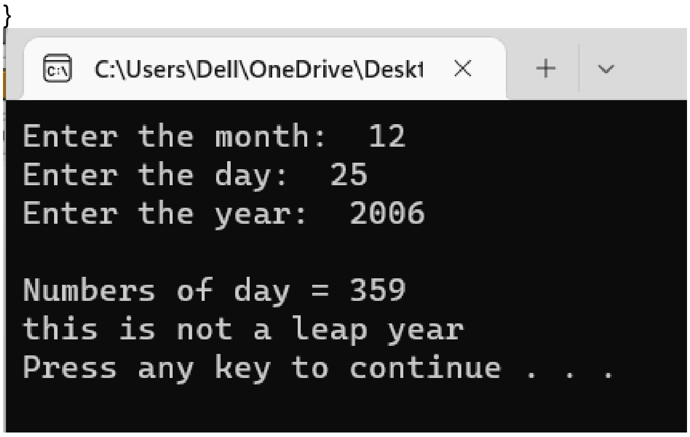
Task 1:

```
#include <iostream>
using namespace std;
bool leap(int year)
{
    if (!(year % 400) && !(year % 100))
    {
        return true;
    }
    else if (year % 100 && !(year % 4))
    {
        return true;
    }
    return false;
}
int tempday(int mont, int day, bool l)
{
    int num = day, i = 1;
    while (i < mont)
    {
        switch (i)
        {
        case 1:
        case 3:
        case 5:
        case 7:
        case 8:
        case 10:
        case 12:
        {
            num += 31;
            break;
        }
        case 4:
        case 6:
        case 9:
        case 11:
        {
            num += 30;
            break;
        }
        case 2:
        {
            if (l)
            {
                num += 29;
            }
            else
            {
                num += 28;
            }
            break;
        }
        }
        i++;
    }
    return num;
}
void lyear(int month, int day, int year)
{
    bool l = leap(year);
    cout << "\nNumbers of day = " << tempday(month, day, l) << endl;
```

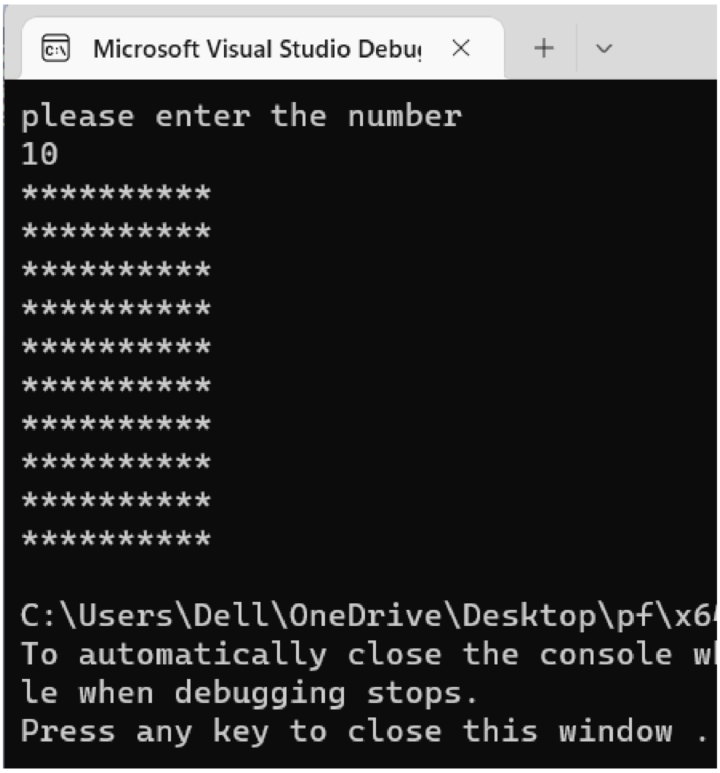
```
        if (l)
        {
                cout << "this is a leap year" << endl;
        }
        else
        {
                cout << "this is not a leap year" << endl;
        }
}
int main()
{
        int m, d, y;
        cout << "Enter the month:  ";
        cin >> m;
        cout << "Enter the day:  ";
        cin >> d;
        cout << "Enter the year:  ";
        cin >> y;
        lyear(m, d, y);
        system("pause");
}
```



```
Enter the month:  12
Enter the day:  25
Enter the year:  2006

Numbers of day = 359
this is not a leap year
Press any key to continue . . .
```

Task 2:

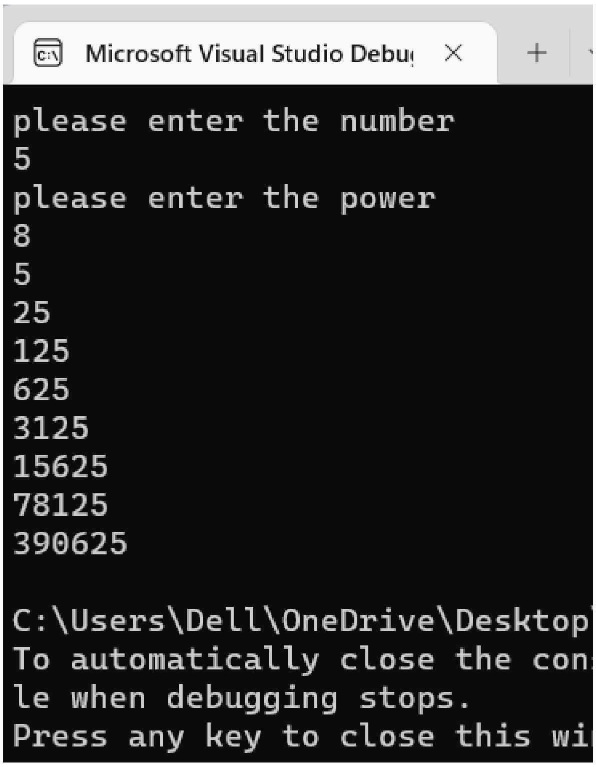
```
#include<iostream>
using namespace std;
void square(int n)
{
        for (int i = 0; i < n; i++)
        {
                for (int j = 0; j < n; j++)
                {
                        cout << "*";
                }
                cout << endl;
        }
}
int main()
{
        int num;
        cout << "please enter the number   " << endl;
        cin >> num;
        square(num);
        return 0;

}
```



```
please enter the number
10
**********
**********
**********
**********
**********
**********
**********
**********
**********
**********

C:\Users\Dell\OneDrive\Desktop\pf\x6
To automatically close the console wh
le when debugging stops.
Press any key to close this window .
```

Task 3:

```
#include<iostream>
using namespace std;
void powercalc(int x,int p)
{
	int num = 1;
	for (int i = 1; i <= p; i++)
	{
		num = num*x;
			cout <<num ;
			cout << endl;
	}
}
int main()
{
	int x,p;
	cout << "please enter the number   " << endl;
	cin >> x;
	cout << "please enter the power   " << endl;
	cin >> p;
	powercalc(x, p);
	return 0;

}
```

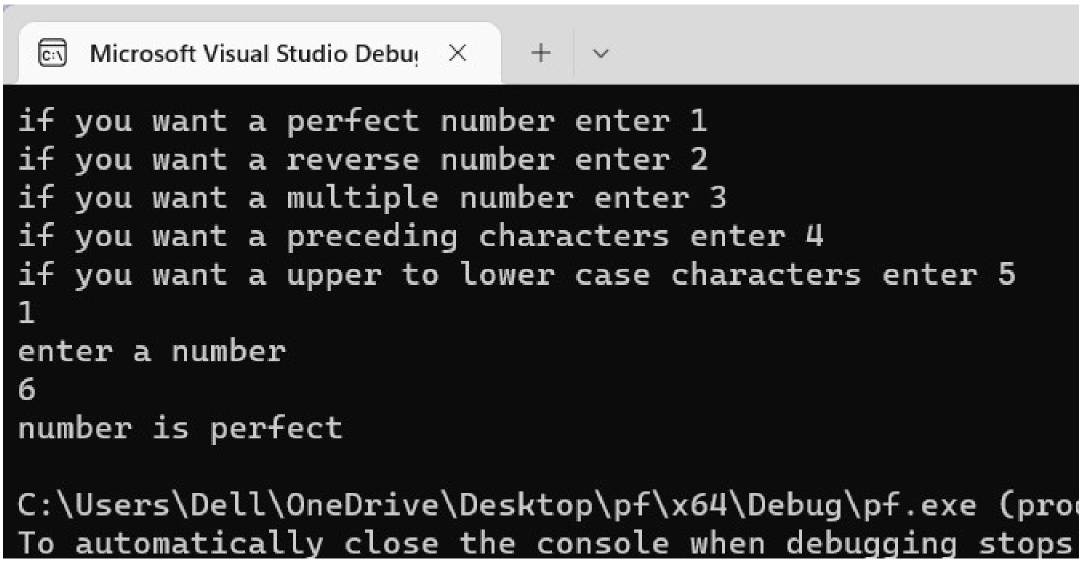


```
please enter the number
5
please enter the power
8
5
25
125
625
3125
15625
78125
390625

C:\Users\Dell\OneDrive\Desktop
To automatically close the con
le when debugging stops.
Press any key to close this wi
```

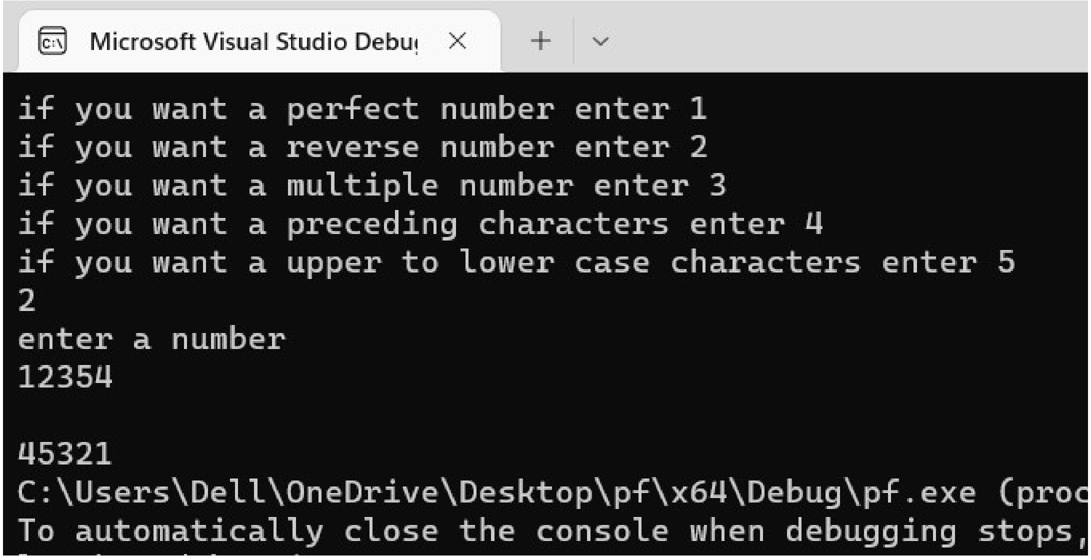
Task 4:

```
#include<iostream>
using namespace std;
bool multiple(int n1, int n2)
{
	if (n1 % n2 == 0)
	{
		cout << "yes";
		return true;
	}
	else
	{
		return false;
	}

}
char Upper_to_lower2(int n4)
{
	char ch4 = n4 + 32;
	cout << ch4 << endl;
	return 0;

}
char Upper_to_lower(int n1, int n2, int n3)
{

	char ch1, ch2, ch3;
	ch1 = n1 + 32; ch2 = n2 + 32; ch3 = n3 + 32;
	cout << ch1 << endl << ch2 << endl << ch3 << endl;
	return 0;
}
char preceding(int n4, int n5, int n6)
{
	n4++; n5++; n6++;
	char ch1, ch2, ch3;
	ch1 = n4; ch2 = n5; ch3 = n6;
	cout << ch1 << endl << ch2 << endl << ch3 << endl;
	return 0;
}
int reverse(long long int n2)
{
	int temp = 0, r;
	while (n2 != 0)
	{
		r = n2 % 10;
		temp = temp * 10 + r;
		n2 /= 10;

	}
	cout << endl;
	return  temp;

}
void is_perfect(long long int n1)
{
	long long int temp = 0;
	for (long long int i = 1; i < n1; i++)
	{
		if (n1 % i == 0)
		{
			temp += i;
		}
	}

	if (temp == n1)
	{
		cout << "number is perfect  \n";
	}
	else
	{
		cout << "number is not perfect  \n";
	}

}
int main()
{
	char ch1, ch2, ch3, ch4;
	int temp, n4, n5, n6, n7;
	long long int num1, num2;
	cout << "if you want a perfect number enter 1\nif you want a reverse number enter 2\nif you want a multiple number enter 3\nif you want a preceding characters enter 4\nif you want a upper to lower case characters enter 5\n";
	cin >> temp;
	switch (temp)
	{
	case 1:
	{
		cout << "enter a number\n";
		cin >> num1;
		is_perfect(num1);
	}
	break;
	case 2:
	{
		cout << "enter a number\n";
		cin >> num2;
		cout << reverse(num2);
	}
	break;
```

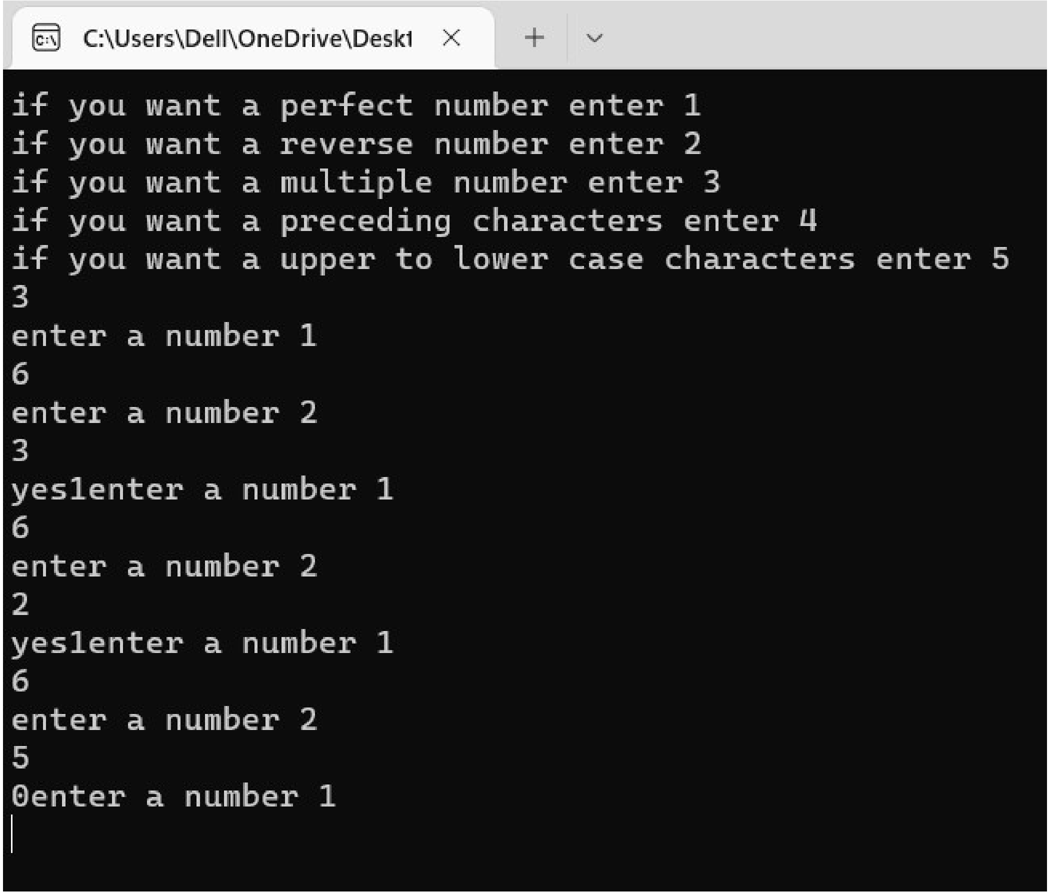
```
    case 3:
    {
        do
        {
            cout << "enter a number 1\n";
            cin >> num1;
            cout << "enter a number 2\n";
            cin >> num2;
            cout << multiple(num1, num2);

        } while (true);

        return 0;
    }
    break;
    case 4:
    {
        cout << "enter character 1\n";
        cin >> ch1;
        cout << "enter character 2\n";
        cin >> ch2;
        cout << "enter character 3\n";
        cin >> ch3;
        n4 = ch1;
        n5 = ch2;
        n6 = ch3;
        cout << preceding(n4, n5, n6);

        return 0;
    }
    break;
    case 5:
    {
        cout << "enter character 1\n";
        cin >> ch1;
        cout << "enter character 2\n";
        cin >> ch2;
        cout << "enter character 3\n";
        cin >> ch3;
        cout << "enter character 4\n";
        cin >> ch4;
        n4 = ch1;
        n5 = ch2;
        n6 = ch3;
        n7 = ch4;
        cout << Upper_to_lower(n4, n5, n6);
        cout << Upper_to_lower2(n7);
        return 0;

    }
    break;
    default:
        cout << "wrong input";
    }


    return 0;



}
```



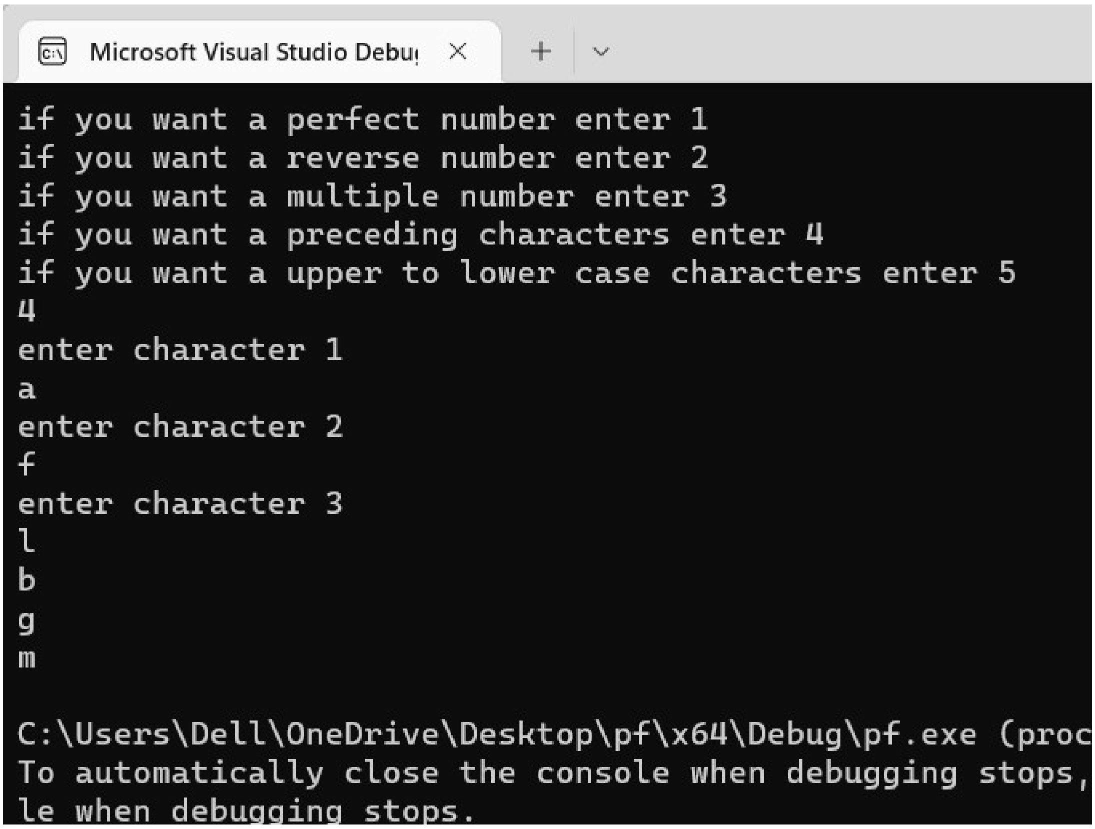
```
if you want a perfect number enter 1
if you want a reverse number enter 2
if you want a multiple number enter 3
if you want a preceding characters enter 4
if you want a upper to lower case characters enter 5
1
enter a number
6
number is perfect

C:\Users\Dell\OneDrive\Desktop\pf\x64\Debug\pf.exe (proc
To automatically close the console when debugging stops
```



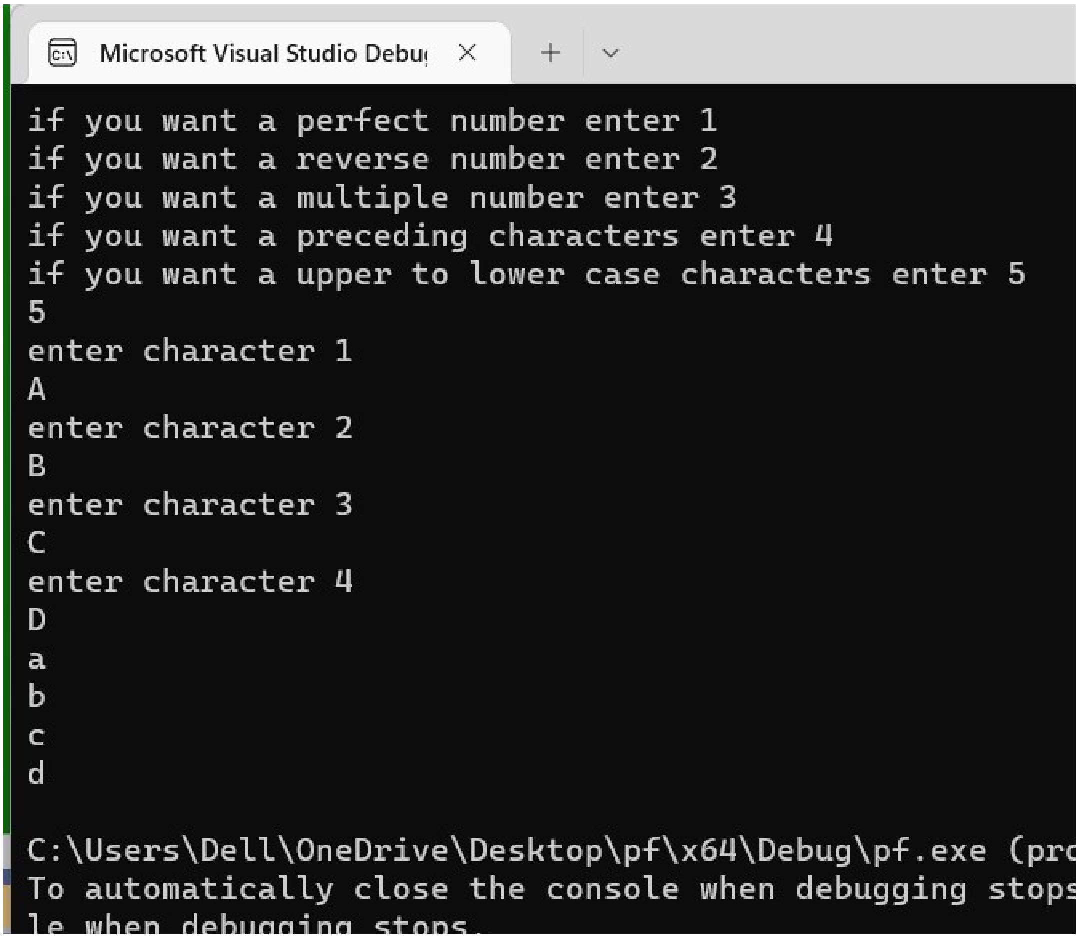
```
if you want a perfect number enter 1
if you want a reverse number enter 2
if you want a multiple number enter 3
if you want a preceding characters enter 4
if you want a upper to lower case characters enter 5
2
enter a number
12354

45321
C:\Users\Dell\OneDrive\Desktop\pf\x64\Debug\pf.exe (proc
To automatically close the console when debugging stops,
```



```
if you want a perfect number enter 1
if you want a reverse number enter 2
if you want a multiple number enter 3
if you want a preceding characters enter 4
if you want a upper to lower case characters enter 5
3
enter a number 1
6
enter a number 2
3
yes1enter a number 1
6
enter a number 2
2
yes1enter a number 1
6
enter a number 2
5
0enter a number 1
```



```
if you want a perfect number enter 1
if you want a reverse number enter 2
if you want a multiple number enter 3
if you want a preceding characters enter 4
if you want a upper to lower case characters enter 5
4
enter character 1
a
enter character 2
f
enter character 3
l
b
g
m

C:\Users\Dell\OneDrive\Desktop\pf\x64\Debug\pf.exe (proc
To automatically close the console when debugging stops,
le when debugging stops.
```



```
if you want a perfect number enter 1
if you want a reverse number enter 2
if you want a multiple number enter 3
if you want a preceding characters enter 4
if you want a upper to lower case characters enter 5
5
enter character 1
A
enter character 2
B
enter character 3
C
enter character 4
D
a
b
c
d

C:\Users\Dell\OneDrive\Desktop\pf\x64\Debug\pf.exe (pro
To automatically close the console when debugging stops
le when debugging stops.
```

Task 4 (c):

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

char preceding(char& ch1, char& ch2, char& ch3)
{
	int n4 = ch1;
	int n5 = ch2;
	int n6 = ch3;
	n4++; n5++; n6++;
	ch1 = n4; ch2 = n5; ch3 = n6;
	return 0;
}

int main()
{
	char ch1, ch2, ch3;
	int temp, n4, n5, n6;



	cout << "enter character 1\n";
	cin >> ch1;
	cout << "enter character 2\n";
	cin >> ch2;
	cout << "enter character 3\n";
	cin >> ch3;

	cout << preceding(ch1, ch2, ch3);
	cout << ch1 << endl << ch2 << endl << ch3 << endl;
```

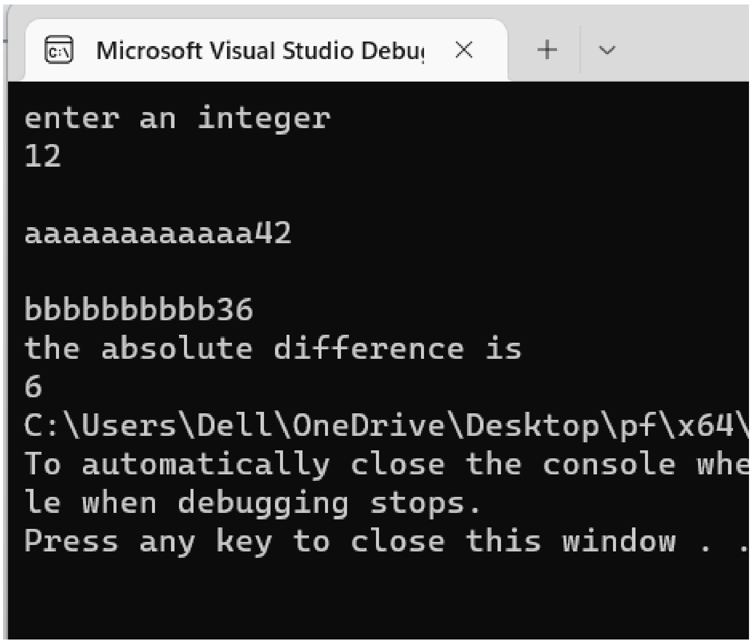
```
        return 0;
}
```

Task 4(d):

```
#include<iostream>
using namespace std;
char Upper_to_lower2(char& ch4)
{
        int     n4 = ch4;
        ch4 = n4 + 32;

        return 0;

}
char Upper_to_lower(char& ch1, char& ch2, char& ch3)
{
        int n1 = ch1;
        int n2 = ch2;
        int n3 = ch3;



        ch1 = n1 + 32; ch2 = n2 + 32; ch3 = n3 + 32;

        return 0;
}


int main()
{
        char ch1, ch2, ch3, ch4;
        int n4, n5, n6, n7;
        long long int num1, num2;


        cout << "enter character 1\n";
        cin >> ch1;
        cout << "enter character 2\n";
        cin >> ch2;
        cout << "enter character 3\n";
        cin >> ch3;
        cout << "enter character 4\n";
        cin >> ch4;

        cout << Upper_to_lower(ch1, ch2, ch3);
        cout << Upper_to_lower2(ch4);
        cout << ch1 << endl << ch2 << endl << ch3 << endl << ch4 << endl;
        return 0;




}
```

Task 5:

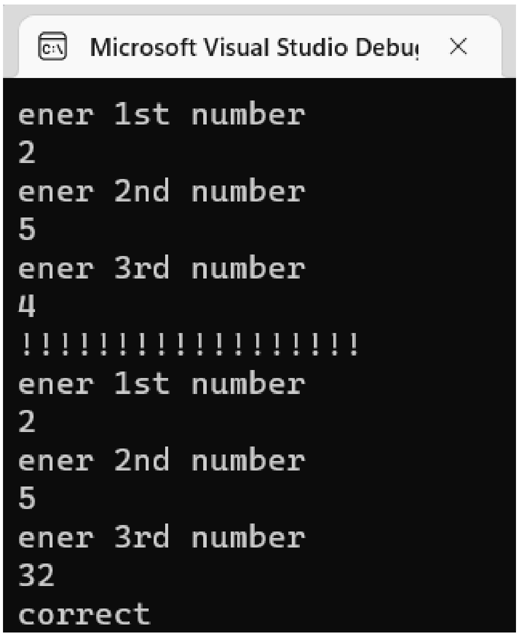
```
#include<iostream>
using namespace std;
void upper_bound(int num)
{
        int temp1 = 0, temp2 = 0,difference;
        for (int i = 0; i<=num; i++)
        {
                if (i % 2 == 0)
                temp1 += i;
        }
        cout << "\naaaaaaaaaaaa" << temp1<<endl;
        for (int j = 1; j <=num; j++)
        {
                if(j%2!=0)
                temp2 += j;
        }
        cout << "\nbbbbbbbbbb" << temp2<<endl;

        difference = temp1 - temp2;
        cout << "the absolute difference is    \n" << difference;
}
int main()
{
        int n;
        cout << "enter an integer \n";
        cin >> n;
        upper_bound(n);
        return 0;
}
```



```
enter an integer
12

aaaaaaaaaaaa42

bbbbbbbb36
the absolute difference is
6
C:\Users\Dell\OneDrive\Desktop\pf\x64\
To automatically close the console whe
le when debugging stops.
Press any key to close this window . .
```

Task 6:

```
#include<iostream>
using namespace std;
bool karatosDestruction(double n1, double n2, double n3);
int main()
{
        int num1, num2, num3;
        cout << "ener 1st number \n";
        cin >> num1;
        cout << "ener 2nd number \n";
        cin >> num2;
        cout << "ener 3rd number \n";
        cin >> num3;
        cout << karatosDestruction(num1, num2, num3);
        return 0;
}

bool karatosDestruction(double n1, double n2, double n3)
{
        int temp = 1;

        for (int i = 1; i <= n2; i++)
        {
            temp *= n1;
        }
        if (temp == n3)
        {
            cout << "correct" << endl;
            return true;
        }
        else if(temp!=n3)
        {
            cout << "!!!!!!!!!!!!!!!!!!" << endl;
            return main();
        }
}
```



```
ener 1st number
2
ener 2nd number
5
ener 3rd number
4
!!!!!!!!!!!!!!!!!!
ener 1st number
2
ener 2nd number
5
ener 3rd number
32
correct
```