Array(অ্যারে)

1(1). অ্যারের মাধ্যমে যোগফল নির্ণয়।

1(2). অ্যারের মাধ্যমে কাঙ্কিত সংখ্যাটি প্রিন্ট।

1.(3). কোনো অ্যারের সবগুলো উপাদান একসঙ্গে প্রিন্ট।

2(1). অ্যারের মাধ্যমে যোগফল এবং এভারেজ নির্ণয়।

2(2) .অ্যারের মাধ্যমে ইউজার হুতে ইনপুট নিয়ে যোগফল এবং এভারেজ নির্ণয়।

2(3). পাঁচ জন ছাত্রের firstsemester(fsx25%), secondsememster(ssx25%), thirdsemester(tsx50%) এর গণিত এর প্রাপ্ত নম্বর নিম্নরূপ। টোটাল মার্কস হিসাব করতে হবে।

2(4). পাঁচ জন ছাত্রের firstsemester(fsx25%), secondsememster(ssx25%), thirdsemester(tsx50%) এর গণিত এর প্রাপ্ত নম্বর নিম্নরূপ। কোন নম্বর কত জন পেলো সেটি হিসাব করতে হবে।

2(5). 0 - 10 এর মধ্যে অ্যারের সংখ্যা গুলো কতবার করে আছে সেটি প্রিন্ট করতে হবে।

2(6). 2D অ্যারের সাহায্যে টোটাল মার্কস নির্নয়।

2(7). 2D array ব্যবহার করে 1 - 10 পর্যন্ত সংখ্যাগুলোর নামতা বের করা।

2(8). সার্কভুক্ত ৭ টি দেশের নাম একটি অ্যারেতে রাখতে হবে।

2(9). সার্কভুক্ত ৭ টি দেশের নাম আলাদা আলাদা ভাবে প্রিণ্ট করা।

2(10). সার্কভুক্ত ৭ টি দেশের নাম কোন ঘরে কোন ক্যারেক্টার আছে সেটি নির্নয় করা।

2(11). Sum of rows.

2(12). Sum of columns.

2(13). প্রথম অ্যারের রো গুলো দ্বিতীয় অ্যারের কলাম আকারে প্রিন্ট করতে হবে।

3(1). অ্যারের মাধ্যমে সবচেয়ে বড় সংখ্যাটি নির্ণয়।

3(2). অ্যারের মাধ্যমে ইউজার হতে ইনপুট নিয়ে সবচেয়ে বড় সংখ্যাটি নির্ণয়।

4. Fibonacci series using array.

5(1). Linear search(লিনিয়ার সার্চ).

5(2). Linear search from the user.

5(3). Binary search(বাইনারি সার্চ).

5(4). Binary search from the user.

6(1). Array - 1 এর উপাদান গুলো Array - 2 এ কপি করা।

6(2). Array - 1 এর উপাদান গুলো Array - 2 এ কপি করা।(ইঊজার হতে ইনপুট নিয়ে)

7(1). 2D অ্যারের মাধ্যমে রো এবং কলাম প্রিন্ট করা।

7(2). 2D অ্যারের মাধ্যমে রো এবং কলাম প্রিন্ট করা।(ইউজার হতে ইনপুট নিয়ে)

8. Array - র সাহায্যে simple matrix তৈরি করা।

9(1). দুটি ম্যাটিক্স এর যোগ।

9(2). দুটি ম্যাটিক্স এর বিয়োগ।

10(1). দুটি ম্যাটিক্স এর গুণন।

10(2). দুটি ম্যাটিক্স এর গুণন এবং গুণফল নির্ণয়।

11. Transpose Matrix(ট্রান্সপোস ম্যাটিক্স)

12. Diagnal element এর যোগফল নির্ণয়।

13. Sum of upper and lower triangle element.

14. Line in Array.

15. Column in Array.

16. Above the Main Diagonal.

17. Below the Main Diagonal.

18. Above the Secondary Diagonal.

19. Below the Secondary Diagonal.

20. Top Area.

21. Inferior Area.

22. Left Area.

23. Right Area.

24(1).Ascending and Descending order.

24(2).Ascending and Descending order from the user

25. একটি অ্যারের মধ্যে কোনো একটি নিদিষ্ট উপাদান আছে কিনা সেটি নির্নয় করা,

এবং উপাদানটি থাকলে সেটি কততম ইনডেক্সে আছে সেটিও নির্নয় করা।

1(1).অ্যারের মাধ্যমে যোগফল নির্ণয়।

#include <stdio.h>

int main()

{

int num[] = { 10, 20, 30, 40, 50 };

int sum;

sum = num[0] + num[1] + num[2] + num[3] + num[4];

printf("The sum is = %d\n", sum);

}

Output:

The sum is = 150

1(2).অ্যারের মাধ্যমে কাঙ্কিত সংখ্যাটি প্রিন্ট।

#include <stdio.h>

int main()

{

int num[] = { 10, 20, 30, 40, 50 };

int sum;

sum = num[0] + num[1] + num[2] + num[3] + num[4];

printf("The sum is = %d\n", sum);

printf("I wanted to print number = %d\n", num[2]);

}

Output:

The sum is = 150

I wanted to print number = 30

1.3 কোনো অ্যারের সবগুলো উপাদান যদি একসঙ্গে দেখতে চাই তাহলে নিম্নরূপ ভাবে করতে হবে।

#include <stdio.h>

int main()

{

int num[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

int i;

for (i = 0; i < 10; i++)

{

printf("%dth element is = %d\n", i + 1, num[i]);

}

}

Or,

#include <stdio.h>

int main()

{

int num[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

int i;

for (i = 9; i > 0; i--)

{

printf("%dth element is = %d\n", i + 1, num[i]);

}

}

Output:

1th element is = 1

2th element is = 2

3th element is = 3

4th element is = 4

5th element is = 5

6th element is = 6

7th element is = 7

8th element is = 8

9th element is = 9

10th element is = 10

Output:

10th element is = 10

9th element is = 9

8th element is = 8

7th element is = 7

6th element is = 6

5th element is = 5

4th element is = 4

3th element is = 3

2th element is = 2

1th element is = 1

2(1).অ্যারের মাধ্যমে যোগফল এবং এভারেজ নির্ণয়।

Output:

The sum is = 150

The average is = 30.00

#include <stdio.h>

int main()

{

int num[5] = { 10, 20, 30, 40, 50 };

int i, sum = 0;

for (i = 0; i < 5; i++)

{

sum = sum + num[i];

}

printf("The sum is = %d\n", sum);

printf("The average is = %.2f\n", (float)sum / 5);

}

2(2).অ্যারের মাধ্যমে ইউজার হুতে ইনপুট নিয়ে যোগফল এবং এভারেজ নির্ণয়।

Output:

How many numbers = 5

Enter numbers = 10 20 30 40 50

The sum is = 150

The average is = 30.00

#include <stdio.h>

int main()

{

int num[10];

int n, i, sum = 0;

printf("How many numbers = ");

scanf("%d", &n);

printf("\nEnter numbers = ");

for (i = 0; i < n; i++)

{

scanf("%d", &num[i]);

}

for (i = 0; i < n; i++)

{

sum = sum + num[i];

}

printf("The sum is = %d\n", sum);

printf("The average is = %.2f\n", (float)sum / n);

}

2.3 পাঁচ জন ছাত্রের firstsemester(fsx25%), secondsememster(ssx25%), thirdsemester(tsx50%)

এর গণিত এর প্রাপ্ত নম্বর নিম্নরূপ। টোটাল মার্কস হিসাব করতে হবে।

#include <stdio.h>

int main()

{

int fsmarks[5] = { 80, 87, 82, 88, 89 },

ssmarks[5] = { 88, 90, 91, 98, 99 },

tsmarks[5] = { 89, 85, 82, 98, 91 };

int i;

double totalmarks[5];

for (i = 0; i < 5; i++)

{

totalmarks[i] = fsmarks[i] / 4.0 + ssmarks[i] / 4.0 + tsmarks[i] / 2.0;

}

for (i = 0; i < 5; i++)

{

printf("Roll number = %d\tMarks = %.2lf\n", i + 1, totalmarks[i]);

}

}

Output:

Roll number = 1 Marks = 86.50

Roll number = 2 Marks = 86.75

Roll number = 3 Marks = 84.25

Roll number = 4 Marks = 95.50

Roll number = 5 Marks = 92.50

2.4 পাঁচ জন ছাত্রের firstsemester(fsx25%), secondsememster(ssx25%), thirdsemester(tsx50%)

এর গণিত এর প্রাপ্ত নম্বর নিম্নরূপ। কোন নম্বর কত জন পেলো সেটি হিসাব করতে হবে।

#include <stdio.h>

int main()

{

int totalmarks[] = { 88, 84, 81, 88, 83 };

int i, count, marks;

for (marks = 80; marks <= 100; marks++)

{

count = 0;

for (i = 0; i < 5; i++)

{

if (totalmarks[i] == marks)

{

count++;

}

}

printf("Marks = %d\tCount = %d\n", marks, count);

}

}

Output:

Marks = 80 Count = 0

Marks = 81 Count = 1

Marks = 82 Count = 0

Marks = 83 Count = 1

Marks = 84 Count = 1

Marks = 85 Count = 0

Marks = 86 Count = 0

Marks = 87 Count = 0

Marks = 88 Count = 2

Marks = 89 Count = 0

Marks = 90 Count = 0

Marks = 91 Count = 0

Marks = 92 Count = 0

Marks = 93 Count = 0

Marks = 94 Count = 0

Marks = 95 Count = 0

Marks = 96 Count = 0

Marks = 97 Count = 0

Marks = 98 Count = 0

Marks = 99 Count = 0

Marks = 100 Count = 0

2.5 0-10 এর মধ্যে অ্যারের সংখ্যা গুলো কতবার করে আছে সেটি প্রিন্ট করতে হবে।

#include <stdio.h>

int main()

{

int totalmarks[] = { 2, 3, 4, 5, 6, 7, 8, 2, 3, 7, 5 };

int i, count, marks;

for (marks = 0; marks <= 10; marks++)

{

count = 0;

for (i = 0; i < 11; i++)

{

if (totalmarks[i] == marks)

{

count++;

}

}

printf("Marks = %d\tCount = %d\n", marks, count);

}

}

Output:

Marks = 0 Count = 0

Marks = 1 Count = 0

Marks = 2 Count = 2

Marks = 3 Count = 2

Marks = 4 Count = 1

Marks = 5 Count = 2

Marks = 6 Count = 1

Marks = 7 Count = 2

Marks = 8 Count = 1

Marks = 9 Count = 0

Marks = 10 Count = 0

2.6 2D অ্যারের সাহায্যে টোটাল মার্কস নির্নয়।

#include <stdio.h>

int main()

{

int marks[4][10] =

{

{80, 70, 92, 78, 58, 83, 85, 66, 99, 81}, //row 0

{75, 67, 55, 98, 91, 84, 79, 61, 90, 89}, //row 1

{98, 67, 75, 89, 81, 83, 80, 90, 88, 77}, //row 2

{0,0,0,0,0,0,0,0,0,0} //row 3

};

int i;

for (i = 0; i < 10; i++)

{

marks[3][i] = marks[0][i] / 4.0 + marks[1][i] / 4.0 + marks[2][i] / 2.0;

printf("Roll number = %d Total marks = %d\n", i + 1, marks[3][i]);

}

}

Output:

Roll number = 1 Total marks = 87

Roll number = 2 Total marks = 67

Roll number = 3 Total marks = 74

Roll number = 4 Total marks = 88

Roll number = 5 Total marks = 77

Roll number = 6 Total marks = 83

Roll number = 7 Total marks = 81

Roll number = 8 Total marks = 76

Roll number = 9 Total marks = 91

Roll number = 10 Total marks = 81

marks[4] means marks[0] to marks[3]

marks[0] -> firstsememster

marks[1] -> secondsemester

marks[2] -> finalsemester

marks[3] -> totalmarks

**From the user**

for (int i = 0; i < 4; i++) {

for (int j = 0; j < 10; j++) {

scanf("%d", &marks[i][j]);

}

}

2.7 2D array ব্যবহার করে 1-10 পর্যন্ত সংখ্যাগুলোর নামতা বের করা।

#include <stdio.h>

int main()

{

int namta[10][10];

int i, j;

for (i = 0; i < 10; i++)

{

for (j = 0; j < 10; j++)

{

namta[i][j] = (i + 1) \* (j + 1);

}

}

for (i = 0; i < 10; i++)

{

for (j = 0; j < 10; j++)

{

printf("%d X %d = %d\n", i + 1, j + 1, namta[i][j]);

}

printf("\n");

}

}

Output:

1 X 1 = 1

1 X 2 = 2

1 X 3 = 3

1 X 4 = 4

1 X 5 = 5

1 X 6 = 6

1 X 7 = 7

1 X 8 = 8

1 X 9 = 9

1 X 10 =10

এভাবে একে একে ১০ পর্যন্ত প্রিন্ট হবে।

2.8 সার্কভুক্ত ৭ টি দেশের নাম একটি অ্যারেতে রাখতে হবে।

#include <stdio.h>

int main()

{

char saarc[7][100] = { "Bangladesh", "India", "Japan", "Korea", "Nepal", "Ugands", "Canada" };

Output:

Bangladesh

India

Japan

Korea

Nepal

Ugands

Canada

int i;

for (i = 0; i < 7; i++)

{

printf("%s\n", saarc[i]);

}

}

2.9 সার্কভুক্ত ৭ টি দেশের নাম আলাদা আলাদা ভাবে প্রিণ্ট করা।

#include <stdio.h>

#include <string.h>

int main()

{

char saarc[7][100] = { "Bangladesh", "India", "Japan", "Korea", "Nepal", "Ugands", "Canada" };

int i, j, length;

for (i = 0; i < 7; i++)

{

Output:

B a n g l a d e s h

I n d i a

J a p a n

K o r e a

N e p a l

U g a n d s

C a n a d a

length = strlen(saarc[i]);

for (j = 0; j < length; j++)

{

printf("%c ", saarc[i][j]);

}

printf("\n");

}

}

2.10 সার্কভুক্ত ৭ টি দেশের নাম কোন ঘরে কোন ক্যারেক্টার আছে সেটি নির্নয় করা।

#include <stdio.h>

#include <string.h>

int main()

{

char saarc[7][100] = { "Bangladesh", "India", "Japan", "Korea", "Nepal", "Ugands", "Canada" };

int i, j, length;

for (i = 0; i < 7; i++)

{

length = strlen(saarc[i]);

for (j = 0; j < length; j++)

{

printf("(%d, %d) = %c\n", i, j, saarc[i][j]);

}

printf("\n");

}

}

(4, 0) = N

(4, 1) = e

(4, 2) = p

(4, 3) = a

(4, 4) = l

(5, 0) = U

(5, 1) = g

(5, 2) = a

(5, 3) = n

(5, 4) = d

(5, 5) = s

(6, 0) = C

(6, 1) = a

(6, 2) = n

(6, 3) = a

(6, 4) = d

(6, 5) = a

Output:

(0, 0) = B

(0, 1) = a

(0, 2) = n

(0, 3) = g

(0, 4) = l

(0, 5) = a

(0, 6) = d

(0, 7) = e

(0, 8) = s

(0, 9) = h

(1, 0) = I

(1, 1) = n

(1, 2) = d

(1, 3) = i

(1, 4) = a

(2, 0) = J

(2, 1) = a

(2, 2) = p

(2, 3) = a

(2, 4) = n

(3, 0) = K

(3, 1) = o

(3, 2) = r

(3, 3) = e

(3, 4) = a

2.11 Sum of rows

Output:

Sum of row 1 : 34

Sum of row 2 : 29

Sum of row 3 : 27

Sum of row 4 : 22

Sum of row 5 : 21

#include <stdio.h>

int main()

{

int num[5][5] =

{

{6, 4, 7, 8, 9},

{3, 7, 1, 9, 9},

{8, 6, 4, 2, 7},

{2, 4, 2, 5, 9},

{4, 1, 6, 7, 3}

};

int i, j, sum = 0;

for (i = 0; i < 5; i++)

{

for (j = 0; j < 5; j++)

{

sum = sum + num[i][j];

}

printf("Sum of row %d: %d\n", i + 1, sum);

sum = 0;

}

}

2.12 Sum of columns

Output:

Sum of column 1 : 23

Sum of column 2 : 22

Sum of column 3 : 20

Sum of column 4 : 31

Sum of column 5 : 37

#include <stdio.h>

int main()

{

int num[5][5] =

{

{6, 4, 7, 8, 9},

{3, 7, 1, 9, 9},

{8, 6, 4, 2, 7},

{2, 4, 2, 5, 9},

{4, 1, 6, 7, 3}

};

int i, j, sum = 0;

for (j = 0; j < 5; j++)

{

for (i = 0; i < 5; i++)

{

sum = sum + num[i][j];

}

printf("Sum of column %d: %d\n", j + 1, sum);

sum = 0;

}

}

Output:

The element of first array is =

1 2 3 4 5

10 20 30 40 50

100 200 300 400 500

1000 2000 3000 4000 5000

10000 20000 30000 40000 50000

The element of second array is =

1 10 100 1000 10000

2 20 200 2000 20000

3 30 300 3000 30000

4 40 400 4000 40000

5 50 500 5000 50000

2.13 প্রথম অ্যারের রো গুলো দ্বিতীয় অ্যারের কলাম আকারে প্রিন্ট করতে হবে।

#include <stdio.h>

int main()

{

int ara1[5][5] =

{

{1, 2, 3, 4, 5},

{10, 20, 30, 40, 50},

{100, 200, 300, 400, 500},

{1000, 2000, 3000, 4000, 5000},

{10000, 20000, 30000, 40000, 50000}

};

int ara2[5][5];

int i, j;

printf("The element of first array is = \n");

for (i = 0; i < 5; i++)

{

for (j = 0; j < 5; j++)

{

printf("%d ", ara1[i][j]);

}

printf("\n");

}

for (i = 0; i < 5; i++)

{

for (j = 0; j < 5; j++)

{

ara2[i][j] = ara1[j][i];

}

}

printf("\n");

printf("The element of second array is = \n");

for (i = 0; i < 5; i++)

{

for (j = 0; j < 5; j++)

{

printf("%d ", ara2[i][j]);

}

printf("\n");

}

}

3(1) - অ্যারের মাধ্যমে সবচেয়ে বড় সংখ্যাটি নির্ণয়।

#include <stdio.h>

int main()

{

int num[] = { 2, 4, 5, 6, 1, 8, 9 };

int i, position;

int max = num[0];

for (i = 1; i < 7; i++)

{

if (num[i] > max)

{

max = num[i];

position = i;

}

}

printf("The maximum number is = %d\n", max);

printf("The position is = %d\n", position);

}

Output:

The maximum number is = 9

The position is = 6

3(2) - অ্যারের মাধ্যমে ইউজার হতে ইনপুট নিয়ে সবচেয়ে বড় সংখ্যাটি নির্ণয়।

#include <stdio.h>

int main()

{

int num[10], n, i, position;

printf("How many numbers = ");

scanf("%d", &n);

printf("Please enter numbers = ");

for (i = 0; i < n; i++)

{

scanf("%d", &num[i]);

}

int max = num[0];

for (i = 1; i < n; i++)

{

if (num[i] > max)

{

max = num[i];

position = i;

}

}

printf("The maximum number is = %d\n", max);

printf("The position of maximum number is = %d\n", position);

}

Output:

How many numbers = 5

Please enter numbers = 11 34 56 67 78

The maximum number is = 78

The position of maximum number is = 4

3(3). যদি বলতো সবচেয়ে ছোট সংখ্যাটি নির্নয় করো তাহলে শুধু এই লাইনটি চেঞ্জ করলেই

হবেঃ if (num[i] < max)

4. Fibonacci series using array.

#include <stdio.h>

int main()

{

int n, i, num[100];

printf("How many fibonacci numbers = ");

scanf("%d", &n);

num[0] = 0;

num[1] = 1;

for (i = 2; i < n; i++)

{

num[i] = num[i - 2] + num[i - 1];

}

for (i = 0; i < n; i++){

printf("%d ", num[i]);

}

}

Output:

How many fibonacci numbers = 7

0 1 1 2 3 5 8

5(1) - Linear search(লিনিয়ার সার্চ).

#include <stdio.h>

int main()

{

int num[] = { 4, 5, 6, 8, 9, 11, 12 };

int value, position = -1, i;

printf("Enter the value you want to search = ");

scanf("%d", &value);

for (i = 0; i < 7; i++)

{

if (value == num[i])

{

position = i + 1;

break;

}

}

if (position == -1)

printf("Value is not found\n");

else

printf("Value is found at position %d", position);

}

Output:

Enter the value you want to search = 11

Value is found at position 5

5(2) - Linear search from the user.

#include <stdio.h>

int main()

{

int num[5];

int value, position = 0, i, n;

printf("Enter number = ");

scanf("%d", &n);

for (i = 0; i < n; i++)

{

scanf("%d", &num[i]);

}

printf("Enter the value you want to search = ");

scanf("%d", &value);

for (i = 0; i < n; i++)

{

if (value == num[i])

{

position = position + i;

break;

}

}

if (position == -1)

printf("Value is not found\n");

else

printf("Value is found at position %d", position);

}

Output:

Enter number = 5

10 20 30 40 50

Enter the value you want to search = 20

Value is found at position 1

5(3) - Binary search(বাইনারি সার্চ).

#include <stdio.h>

int main()

{

int num[] = { 1,4,6,8,9,11,14,15,20,25,33,83,87,97,99,100 };

int value = 97;

int lowindex = 0;

int highindex = 15;

int midindex;

while (lowindex <= highindex)

{

midindex = (lowindex + highindex) / 2;

if (value == num[midindex])

{

break;

}

else if (value > num[midindex])

{

lowindex = midindex + 1;

}

else

{

highindex = midindex - 1;

}

}

if (lowindex > highindex)

printf("%d is not in the number\n", value);

else

printf("Value is found at position = %d\n", midindex);

}

Output:

Value is found at position = 13

5(4) - Binary search from the user.

Output:

Enter number = 5 12 23 34 45 56

Enter the value you want to search = 34

Value is found at position = 2

#include <stdio.h>

int main()

{

int num[5], i, n;

printf("Enter number = ");

scanf("%d", &n);

for (i = 0; i < n; i++)

{

scanf("%d", &num[i]);

}

int value;

printf("Enter the value you want to search = ");

scanf("%d", &value);

int lowindex = 0;

int highindex = n;

int midindex;

while (lowindex <= highindex)

{

midindex = (lowindex + highindex) / 2;

if (value == num[midindex])

{

break;

}

else if (value > num[midindex])

{

lowindex = midindex + 1;

}

else

{

highindex = midindex - 1;

}

}

if (lowindex > highindex)

printf("%d is not in the number\n", value);

else

printf("Value is found at position = %d\n", midindex);

}

6(1).Array - 1 এর উপাদান গুলো Array - 2 এ কপি করা।

#include <stdio.h>

int main()

{

int num1[5] = { 10, 20, 30, 40, 50 };

int num2[5], i;

printf("num1 = ");

for (i = 0; i < 5; i++)

{

printf("%d ", num1[i]);

}

//copy started.

for (i = 0; i < 5; i++)

{

num2[i] = num1[i];

}

printf("\n\nnum2 = ");

for (i = 0; i < 5; i++)

{

printf("%d ", num2[i]);

}

}

Output:

num1 = 10 20 30 40 50

num2 = 10 20 30 40 50

6(2).Array - 1 এর উপাদান গুলো Array - 2 এ কপি করা।(ইঊজার হতে ইনপুট নিয়ে)

#include <stdio.h>

int main()

{

int num1[10];

int num2[10], i, n;

printf("How many numbers = ");

scanf("%d", &n);

printf("Please enter numbers = ");

for (i = 0; i < n; i++)

{

scanf("%d", &num1[i]);

}

printf("num1 = ");

for (i = 0; i < n; i++)

{

printf("%d ", num1[i]);

}

//copy started.

for (i = 0; i < n; i++)

{

num2[i] = num1[i];

}

printf("num2 = ");

for (i = 0; i < n; i++)

{

printf("%d ", num2[i]);

}

}

Output:

How many numbers = 5

Please enter numbers = 12 23 34 45 56

num1 = 12 23 34 45 56

num2 = 12 23 34 45 56

কোনো একটি অ্যারেতে ১০ টি উপাদান আছে সেগুলো বিপরীতক্রমে রাখতে হবে।

অর্থাৎ দশম উপাদান হবে প্রথম, নবম উপাদান হবে দ্বিতীয়......এইভাবে।

#include <stdio.h>

int main()

{

int ara[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

for (int i = 9; i >= 0; i--) {

printf("%d ", ara[i]);

}

}

or,

#include <stdio.h>

int main()

{

int ara[10] = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 };

int i, j, temp;

for (i = 0, j = 9; i < 10; i++, j--) {

temp = ara[i];

ara[i] = ara[j];

ara[j] = temp;

}

for (i = 9; i >= 0; i--) {

printf("%d ", ara[i]);

}

}

Output:

10 9 8 7 6 5 4 3 2 1

7(1) - 2D অ্যারের মাধ্যমে রো এবং কলাম প্রিন্ট করা।

#include <stdio.h>

int main()

{

int A[3][4] = { {1, 2, 3, 4}, {2, 3, 4, 5}, {3, 4, 5, 6} };

int i, j; /\*i means row and j means column\*/

for (i = 0; i < 3; i++)

{

for (j = 0; j < 4; j++)

{

printf("%d ", A[i][j]); //2 space.

}

printf("\n");

}

}

Output:

1 2 3 4

2 3 4 5

3 4 5 6

7(2) - 2D অ্যারের মাধ্যমে রো এবং কলাম প্রিন্ট করা।(ইউজার হতে ইনপুট নিয়ে)

Output:

Enter rowand columns = 3 3

A[0][0] = 2

A[0][1] = 3

A[0][2] = 4

A[1][0] = 5

A[1][1] = 6

A[1][2] = 7

A[2][0] = 7

A[2][1] = 8

A[2][2] = 9

The matrix is =

2 3 4

5 6 7

7 8 9

#include <stdio.h>

int main()

{

int A[10][10];

int i, j, row, column;

printf("Enter row and columns = ");

scanf("%d %d", &row, &column);

printf("\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++)

{

printf("A[%d][%d] = ", i, j);

scanf("%d", &A[i][j]);

}

printf("\n");

}

printf("The matrix is = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++)

{

printf("%d ", A[i][j]); //2 space.

}

printf("\n");

}

}

8. Array - র সাহায্যে simple matrix তৈরি করা।

Output:

Enter row and columns for A matrix = 2 2

A[0][0] = 3

A[0][1] = 4

A[1][0] = 6

A[1][1] = 7

The element of A matrix is =

3 4

6 7

Enter row and columns for B matrix = 2 3

B[0][0] = 5

B[0][1] = 6

B[0][2] = 7

B[1][0] = 8

B[1][1] = 9

B[1][2] = 1

The element of B matrix is =

5 6 7

8 9 1

#include <stdio.h>

int main()

{

int A[10][10], B[10][10];

int i, j, row, column;

printf("Enter row and columns for A matrix = ");

scanf("%d %d", &row, &column);

printf("\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++)

{

printf("A[%d][%d] = ", i, j);

scanf("%d", &A[i][j]);

}

printf("\n");

}

printf("The element of A matrix is = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++)

{

printf("%d ", A[i][j]); //2 space.

}

printf("\n");

}

printf("\nEnter row and columns for B matrix = ");

scanf("%d %d", &row, &column);

printf("\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++)

{

printf("B[%d][%d] = ", i, j);

scanf("%d", &B[i][j]);

}

printf("\n");

}

printf("The element of B matrix is = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++)

{

printf("%d ", B[i][j]); //2 space.

}

printf("\n");

}

}

9(1).দুটি ম্যাটিক্স এর যোগ।

Output:

Enter rowand columns for A matrix = 2 2

A[0][0] = 2

A[0][1] = 3

A[1][0] = 4

A[1][1] = 5

The element of A matrix is =

2 3

4 5

Enter row and columns for B matrix = 2 2

B[0][0] = 6

B[0][1] = 7

B[1][0] = 8

B[1][1] = 9

The element of B matrix is =

6 7

8 9

The sum of A + B =

8 10

12 14

#include <stdio.h>

int main()

{

int A[10][10], B[10][10], C[10][10];

int i, j, row, column;

printf("Enter row and columns for A matrix = ");

scanf("%d %d", &row, &column);

printf("\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++) {

printf("A[%d][%d] = ", i, j);

scanf("%d", &A[i][j]);

}

printf("\n");

}

printf("The element of A matrix is = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++) {

printf("%d ", A[i][j]); //2 space.

}

printf("\n");

}

printf("\nEnter row and columns for B matrix = ");

scanf("%d %d", &row, &column);

printf("\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++) {

printf("B[%d][%d] = ", i, j);

scanf("%d", &B[i][j]);

}

printf("\n");

}

printf("The element of B matrix is = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++) {

printf("%d ", B[i][j]); //2 space.

}

printf("\n");

}

printf("\nThe sum of A + B = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < column; j++){

printf("%2d ", C[i][j] = A[i][j] + B[i][j]);

}

printf("\n");

}

}

**9(2). দুটি ম্যাটিক্স এর বিয়োগ।(পুরোটাই সেম, শুধু লাষ্টে + এর পরিবর্তে – হবে)**

10(1) - দুটি ম্যাটিক্স এর গুণন।

#include <stdio.h>

int main()

{

int first[10][10], second[10][10];

int r1, c1, r2, c2, i, j;

printf("Enter rows and columns for first matrix = ");

scanf("%d %d", &r1, &c1);

printf("Enter rows and columns for second matrix = ");

scanf("%d %d", &r2, &c2);

while (c1 != r2)

{

printf("\nError!Column of first matrix is not equal to row of second matrix\n");

printf("\nEnter rows and columns for first matrix = ");

scanf("%d %d", &r1, &c1);

printf("Enter rows and columns for second matrix = ");

scanf("%d %d", &r2, &c2);

}

printf("\nEnter element for first matrix = \n");

for (i = 0; i < r1; i++)

{

for (j = 0; j < c1; j++) {

printf("First[%d][%d] = ", i, j);

scanf("%d", &first[i][j]);

}

}

printf("\nFirst matrix is = \n");

for (i = 0; i < r1; i++)

{

for (j = 0; j < c1; j++) {

printf("%d ", first[i][j]);

}

printf("\n");

Output:

Enter rowsand columns for first matrix = 2 2

Enter rows and columns for second matrix = 2 2

Enter element for first matrix =

First[0][0] = 3

First[0][1] = 4

First[1][0] = 5

First[1][1] = 6

First matrix is =

3 4

5 6

Enter element for second matrix =

Second[0][0] = 7

Second[0][1] = 8

Second[1][0] = 9

Second[1][1] = 2

Second matrix is =

7 8

9 2

}

printf("\nEnter element for second matrix = \n");

for (i = 0; i < r2; i++)

{

for (j = 0; j < c2; j++) {

printf("Second[%d][%d] = ", i, j);

scanf("%d", &second[i][j]);

}

}

printf("\nSecond matrix is = \n");

for (i = 0; i < r2; i++)

{

for (j = 0; j < c2; j++)

{

printf("%d ", second[i][j]);

}

printf("\n");

}

}

10(2) - দুটি ম্যাটিক্স এর গুণন এবং গুণফল নির্ণয়।

#include <stdio.h>

int main()

{

int first[10][10], second[10][10], result[10][10];

int r1, c1, r2, c2, i, j, k, sum = 0;

printf("Enter rows and columns for first matrix = ");

scanf("%d %d", &r1, &c1);

printf("Enter rows and columns for second matrix = ");

scanf("%d %d", &r2, &c2);

while (c1 != r2)

{

printf("\nError....Column of first matrix is not equal to row of second matrix\n");

printf("\nEnter rows and columns for first matrix = ");

scanf("%d %d", &r1, &c1);

printf("Enter rows and columns for second matrix = ");

scanf("%d %d", &r2, &c2);

}

printf("\nEnter element for first matrix = \n");

for (i = 0; i < r1; i++)

{

for (j = 0; j < c1; j++)

{

printf("First[%d][%d] = ", i, j);

scanf("%d", &first[i][j]);

}

}

printf("\nFirst matrix is = \n");

for (i = 0; i < r1; i++)

{

for (j = 0; j < c1; j++)

{

printf("%d ", first[i][j]);

}

printf("\n");

}

printf("\nEnter element for second matrix = \n");

for (i = 0; i < r2; i++)

{

for (j = 0; j < c2; j++)

{

printf("Second[%d][%d] = ", i, j);

scanf("%d", &second[i][j]);

}

}

printf("\nSecond matrix is = \n");

for (i = 0; i < r2; i++)

{

for (j = 0; j < c2; j++)

{

printf("%d ", second[i][j]);

}

printf("\n");

}

for (i = 0; i < r1; i++)

{

for (j = 0; j < c2; j++)

{

for (k = 0; k < c1; k++)

{

sum = sum + first[i][k] \* second[k][j];

}

result[i][j] = sum;

sum = 0;

}

}

printf("\nResult matrix = \n");

for (i = 0; i < r1; i++)

{

for (j = 0; j < c2; j++)

{

printf("%d ", result[i][j]);

}

printf("\n");

}

}

Output:

Enter rowsand columns for first matrix = 2 2

Enter rows and columns for second matrix = 3 2

Error....Column of first matrix is not equal to row of second matrix

Enter rows and columns for first matrix = 2 2

Enter rows and columns for second matrix = 2 2

Enter element for first matrix =

First[0][0] = 4

First[0][1] = 5

First[1][0] = 6

First[1][1] = 7

First matrix is =

4 5

6 7

Enter element for second matrix =

Second[0][0] = 1

Second[0][1] = 2

Second[1][0] = 3

Second[1][1] = 4

Second matrix is =

1 2

3 4

Result matrix =

19 28

27 40

11. Transpose Matrix(ট্রান্সপোস ম্যাটিক্স)

Output:

Enter rowand columns = 2 2

Please enter numbers :

A[0][0] = 4

A[0][1] = 5

A[1][0] = 6

A[1][1] = 7

Entered matrix =

4 5

6 7

Transpose matrix =

4 6

5 7

#include <stdio.h>

int main()

{

int A[10][10], transpose[10][10];

int i, j, row, col;

printf("Enter row and columns = ");

scanf("%d %d", &row, &col);

printf("\nPlease enter numbers:\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("A[%d][%d] = ", i, j);

scanf("%d", &A[i][j]);

}

printf("\n");

}

printf("\nEntered matrix = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("%d ", A[i][j]); //2 space.

}

printf("\n");

}

//now transpose the matrix.

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

transpose[i][j] = A[j][i];

}

}

printf("\nTranspose matrix = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("%d ", transpose[i][j]); //2 space.

}

printf("\n");

}

}

12. Diagnal element এর যোগফল নির্ণয়।

Output:

Enter rowand columns = 3 3

Please enter numbers :

A[0][0] = 1

A[0][1] = 2

A[0][2] = 3

A[1][0] = 4

A[1][1] = 5

A[1][2] = 6

A[2][0] = 7

A[2][1] = 8

A[2][2] = 9

Entered matrix =

1 2 3

4 5 6

7 8 9

Diagonal Elements =

1 5 9

Sum of diagonal element is = 15

#include <stdio.h>

int main()

{

int A[10][10];

int i, j, row, col, sum = 0;

printf("Enter row and columns = ");

scanf("%d %d", &row, &col);

printf("\nPlease enter numbers:\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("A[%d][%d] = ", i, j);

scanf("%d", &A[i][j]);

}

printf("\n");

}

printf("\nEntered matrix = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("%d ", A[i][j]);

}

printf("\n");

}

printf("\nDiagonal Elements = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

if (i == j)

{

printf("%d ", A[i][j]);

sum = sum + A[i][j];

}

}

}

printf("\nSum of diagonal element is = %d\n", sum);

}

13. Sum of upper and lower triangle element.

Output:

Enter rowand columns = 3 3

Please enter numbers :

A[0][0] = 1

A[0][1] = 2

A[0][2] = 3

A[1][0] = 4

A[1][1] = 5

A[1][2] = 6

A[2][0] = 7

A[2][1] = 8

A[2][2] = 9

Entered matrix =

1 2 3

4 5 6

7 8 9

Sum of lower triangle element = 19

Sum of upper triangle element = 11

#include <stdio.h>

int main()

{

int A[10][10];

int i, j, row, col, uppersum = 0, lowersum = 0;

printf("Enter row and columns = ");

scanf("%d %d", &row, &col);

printf("\nPlease enter numbers:\n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("A[%d][%d] = ", i, j);

scanf("%d", &A[i][j]);

}

printf("\n");

}

printf("\nEntered matrix = \n");

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

printf("%d ", A[i][j]);

}

printf("\n");

}

for (i = 0; i < row; i++)

{

for (j = 0; j < col; j++)

{

if (i > j)

lowersum = lowersum + A[i][j];

else if (j > i)

uppersum = uppersum + A[i][j];

}

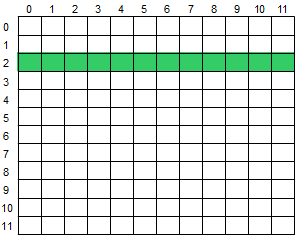
}

printf("\n\nSum of lower triangle element = %d\n", lowersum);

printf("\n\nSum of upper triangle element = %d\n", uppersum);

}

14 – Line in Array.



#include <stdio.h>

int main()

{

double N[12][12], sum = 0;

int i, j, k = 2;

char ch[2];

scanf("%d %s", &k, &ch);

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (j = 0; j < 12; j++)

{

sum = sum + N[k][j];

}

if (ch[0] == 'S')

printf("%.1lf\n", sum);

else if (ch[0] == 'M')

printf("%.1lf\n", sum / 12);

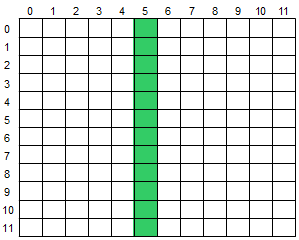
return 0;

}

অর্থাৎ আমি ১৪৪ টা সংখ্যা ইনপুট নিবো। তারপর রো ২ এর মধ্যে যে সংখ্যাগুলো

থাকবে সেগুলোর যোগফল(S হলে) বা এভারেজ(M হলে) বের করবো।

15– Column in Array.



#include <stdio.h>

int main()

{

double N[12][12], sum = 0;

int i, j, k = 5;

char ch[2];

scanf("%d %s", &k, &ch);

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (i = 0; i < 12; i++)

{

sum = sum + N[i][k];

}

if (ch[0] == 'S')

printf("%.1lf\n", sum);

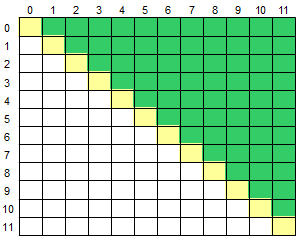
else if (ch[0] == 'M')

printf("%.1lf\n", sum / 12);

return 0;

}

16 – Above the Main Diagonal.



#include <stdio.h>

int main()

{

double N[12][12], sum = 0;

char c[2];

scanf("%s", &c);

int i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

if (j > i)

{

sum = sum + N[i][j];

}

}

}

if (c[0] == 'S')

printf("%.1lf\n", sum);

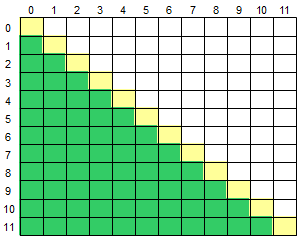
else

printf("%.1lf\n", sum / 66.0);

return 0;

}

17 – Below the Main Diagonal.



#include <stdio.h>

int main()

{

double N[12][12], sum = 0;

char c[2];

scanf("%s", &c);

int i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

if (i > j)

{

sum = sum + N[i][j];

}

}

}

if (c[0] == 'S')

printf("%.1lf\n", sum);

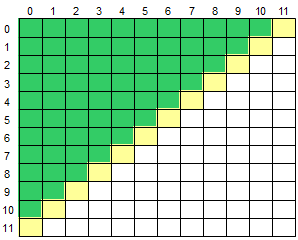
else

printf("%.1lf\n", sum / 66.0);

return 0;

}

18 – Above the Secondary Diagonal.



#include <stdio.h>

int main()

{

double N[12][12], sum = 0;

char c[2];

scanf("%s", &c);

int n = 1, i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (i = 10; i >= 0; i--)

{

for (j = 0; j < n; j++)

{

sum = sum + N[i][j];

}

n++;

}

if (c[0] == 'S')

printf("%.1lf\n", sum);

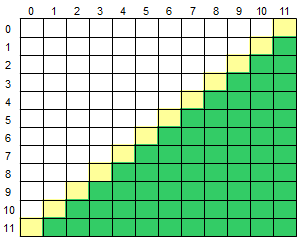
else

printf("%.1lf\n", sum / 66.0);

return 0;

}

19 – Below the Secondary Diagonal.



#include <stdio.h>

int main()

{

double N[12][12], sum = 0;

char c[2];

scanf("%s", &c);

int n = 11, i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (i = 1; i < 12; i++)

{

for (j = n; j < 12; j++)

{

sum = sum + N[i][j];

}

n--;

}

if (c[0] == 'S')

printf("%.1lf\n", sum);

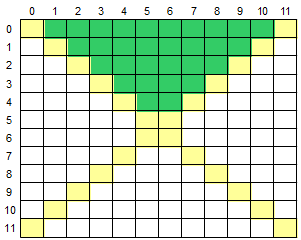
else

printf("%.1lf\n", sum / 66.0);

return 0;

}

20 – Top Area.



#include <stdio.h>

int main()

{

double M[12][12], sum = 0.0;

char ch[2];

scanf("%s", &ch);

int n = 11, a = 1, i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &M[i][j]);

}

}

for (i = 0; i < 5; i++)

{

for (j = a; j < n; j++)

{

sum = sum + M[i][j];

}

n--;

a++;

}

if (ch[0] == 'S')

printf("%.1lf\n", sum);

else

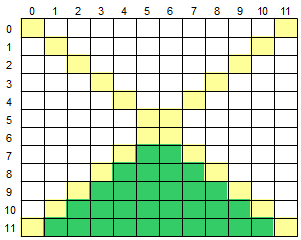
printf("%.1lf\n", sum / 30.0);

return 0;

}

21 – Inferior Area.

#include <stdio.h>



int main()

{

double M[12][12], sum = 0.0;

char ch[2];

scanf("%s", &ch);

int n = 11, a = 1, i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &M[i][j]);

}

}

for (i = 11; i > 6; i--)

{

for (j = a; j < n; j++)

{

sum += M[i][j];

}

n--;

a++;

}

if (ch[0] == 'S')

printf("%.1lf\n", sum);

else

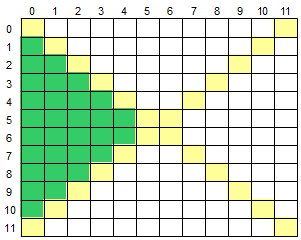
printf("%.1lf\n", sum / 30.0);

return 0;

}

22 – Left Area.

#include <stdio.h>



int main()

{

double M[12][12], sum = 0.0;

char ch[2];

scanf("%s", &ch);

int n = 11, a = 1, i, j;

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &M[i][j]);

}

}

for (j = 0; j < 5; j++)

{

for (i = a; i < n; i++)

{

sum = sum + M[i][j];

}

n--;

a++;

}

if (ch[0] == 'S')

printf("%.1lf\n", sum);

else

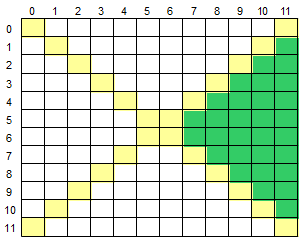
printf("%.1lf\n", sum / 30.0);

return 0;

}

23 – Right Area.

#include <stdio.h>



int main()

{

double N[12][12], sum = 0;

char ch[2];

int i, j, n = 1, m = 10;

scanf("%s", ch);

for (i = 0; i < 12; i++)

{

for (j = 0; j < 12; j++)

{

scanf("%lf", &N[i][j]);

}

}

for (j = 11; j > 6; j--)

{

for (i = n; i <= m; i++)

{

sum = sum + N[i][j];

}

n++;

m--;

}

if (ch[0] == 'S')

printf("%.1lf\n", sum);

else

printf("%.1lf\n", sum / 30.0);

return 0;

}

24(1). Ascending and Descending order.

Output:

Asecnding order

2 3 5 7 9

Descending order

9 7 5 3 2

#include <stdio.h>

int main()

{

int num1[] = { 5, 7, 3, 2, 9 };

int num2[5];

int i, j, minimum, miniposition;

//i for num1 and j for num2.

for (j = 0; j < 5; j++)

{

minimum = 1000;

for (i = 0; i < 5; i++)

{

if (minimum > num1[i])

{

minimum = num1[i];

miniposition = i;

}

}

num1[miniposition] = 1000;

num2[j] = minimum;

}

printf("Asecnding order\n");

for (i = 0; i < 5; i++)

{

printf("%d ", num2[i]);

}

printf("\n");

printf("Descending order\n");

for (i = 4; i >= 0; i--)

{

printf("%d ", num2[i]);

}

}

24(2). Ascending and Descending order from the user

Output:

Enter how many numbers = 5

2 8 5 3 1

Asecnding order

1 2 3 5 8

Descending order

8 5 3 2 1

#include <stdio.h>

int main()

{

int num1[5];

int num2[5];

int i, j, minimum, miniposition, n;

//i for num1 and j for num2.

printf("Enter how many numbers = ");

scanf("%d", &n);

for (i = 0; i < n; i++)

{

scanf("%d", &num1[i]);

}

for (j = 0; j < n; j++)

{

minimum = 1000;

for (i = 0; i < n; i++)

{

if (minimum > num1[i])

{

minimum = num1[i];

miniposition = i;

}

}

num1[miniposition] = 1000;

num2[j] = minimum;

}

printf("Asecnding order\n");

for (i = 0; i < n; i++)

{

printf("%d ", num2[i]);

}

printf("\n");

printf("Descending order\n");

for (i = n - 1; i >= 0; i--)

{

printf("%d ", num2[i]);

}

}

25. একটি অ্যারের মধ্যে কোনো একটি নিদিষ্ট উপাদান আছে কিনা সেটি নির্নয় করা,

এবং উপাদানটি থাকলে সেটি কততম ইনডেক্সে আছে সেটিও নির্নয় করা।

#include <stdio.h>

int main()

{

int num[] = { 1,4,6,8,9,11,14,15,20,25,33,83,87,97,99,100 };

int value = 97;

int lowindex = 0;

int highindex = 15;

int midindex;

while (lowindex <= highindex)

{

midindex = (lowindex + highindex) / 2;

if (value == num[midindex])

break;

else if (value > num[midindex])

lowindex = midindex + 1;

else

highindex = midindex - 1;

}

if (lowindex > highindex)

printf("%d is not in the number\n", value);

else

printf("%d is found in the number.\n"

"It is the % dth element of the number\n", value, midindex);

}

Output:

97 is found in the number.

It is the 13th element of the number

int num[30], n;

printf("How many number : ");

scanf("%d", &n);

printf("Enter %d numbers\n", n);

for (int h = 0; h < n; h++) {

scanf("%d", &num[h]);

}

int value;

printf("Enter the value you want to search : ");

scanf("%d", &value);