* Created By: Golam Kibria
* S\_ID: 203-15-14522
* Section: 0-2
* Department of CSE

|  |  |  |
| --- | --- | --- |
| NO | Contents | Page |
| 01 | Basic Program | 2 |
| 02 | Condition | 25 |
| 03 | Loop | 38 |
| 04 | Pattern | 50 |
| 05 | Series | 80 |
| 06 | Function | 94 |
| 07 | Array | 104 |
| 08 | String | 151 |
| 09 | Pointer | 169 |
| 10 | File | 190 |
| 11 | Structure and Union | 225 |
| 12 | Recursion | 241 |
| 13 | Bitwise Operation | 255 |
| 14 | 52 programming problem | 271 |
| 15 | Solution of Noel Kalicharan sir’s book program | 328 |

Basic(বেসিক)

1 – Takes two integer and prints them.

2 – Prints float, doubleand character.

3 – Takes float numbers and prints them.

4 – Takes integer and float number and prints them.

5 – Convert character into ASCII value.

6 – Size of operator.

7 – Takes two integer and display sum, average.

8(1) – Lower to Uppercase letter.

8(2) – Lower to Uppercase letter using library function.

9(1) – Upper to Lowercase letter.

9(2) – Upper to Lowercase letter using library function.

10(1) – Decimal to Octal.

10(2) – Octal to Decimal.

10(3) – Decimal to Hexa.

10(4) – Hexa to Decimal.

10(5) – Octal to Hexa.

10(6) – Hexa to Octal.

11 – Add, Sub, Division, Multiplication, Remainder.

12 – Area of a Triangle.

13 – Area of a Rectangle.

14 – Area of a triangle given three arms length.

15 – Area of a Circle.

16(1) – Celcious to Farenheit.

16(2) – Farenheit to Celcious.

17(1) – Swapping two number using temporary variable.

17(2) – Swapping two number without temporary variable.

18 – Quadratic equation(ax2 + bx + c).

19 – Absolute value print.

20(1) – Square root.

20(2) – Suare root from the user

21(1) – Power.

21(2) – Power from the user.

22(1) – log(), log10(), exp.

22(2) – sin, cos, tan.

23 – Round, Trunc, Ceil, Floor.

24 – Assignment operator.

25(1) – Unary Operator.

25(2) – Unary Operator.

26. Print a character.

27. Simple sum, sub, mul, div.

28. Simple sum, sub, add, mul using character sign.

29. Interger to Character.

30. সুদ আসল

31. সমীকরণ হতে x এর মান নির্নয়।

1 – Takes two integer and prints them.

Output:

Please enter two integer number = 10 20

Numbers are = 10, 20

#include <stdio.h>

int main()

{

int num1, num2;

printf("Please enter two integer number = ");

scanf("%d %d", &num1, &num2);

printf("Numbers are = %d, %d", num1, num2);

return 0;

}

2 – Prints float, doubleand character.

Output:

Number 1 is = 10.500000

Number 2 is = 10.55

Character is = A

#include <stdio.h>

int main()

{

float num1 = 10.5;

double num2 = 10.555;

char ch = 'A';

printf("Number 1 is = %f\n", num1);

printf("Number 2 is = %.2lf\n", num2);

printf("Character is = %c\n", ch);

}

3 – Takes float numbers and prints them.

Output:

Enter two float number = 12.43223 45.7879

Numbers are = 12.43, 45.79

#include <stdio.h>

int main()

{

float num1, num2;

printf("Enter two float number = ");

scanf("%f %f", &num1, &num2);

printf("Numbers are = %.2f, %.2f", num1, num2);

}

4 – Takes integer and float number and prints them.

#include <stdio.h>

int main()

{

int num1;

float num2;

printf("Enter a integer and a float number = ");

scanf("%d %f", &num1, &num2);

printf("Numbers are = %d, %.2f\n", num1, num2);

}

Output:

Enter a integerand a float number = 23 24.5678

Numbers are = 23, 24.57

5 – Convert character into ASCII value.

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character = ");

scanf("%c", &ch);

printf("The ASCII value is = %d\n", ch);

}

Output:

Enter any character = d

The ASCII value is = 100

6 – Size of operator.

Output:

Size of integer = 4 bytes

Size of float = 4 bytes

Size of double = 8 bytes

Size of character = 1 bytes

#include <stdio.h>

int main()

{

int i;

float f;

double d;

char ch;

printf("Size of integer = %d bytes\n", sizeof(i));

printf("Size of float = %d bytes\n", sizeof(f));

printf("Size of double = %d bytes\n", sizeof(d));

printf("Size of character = %d bytes\n", sizeof(ch));

}

7 – Takes two integer and display sum, average.

#include <stdio.h>

int main()

{

float num1, num2, sum, avg;

printf("Please enter two number = ");

scanf("%f %f", &num1, &num2);

sum = num1 + num2;

avg = sum / 2;

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

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

}

Output:

Please enter two number = 12.5 17.543

The sum is = 30.04

The average is = 15.02

8(1) – Lower to Uppercase letter.

#include <stdio.h>

int main()

{

char lower;

printf("Enter any lowercase letter = ");

scanf("%c", &lower);

printf("The uppercase letter is = %c\n", lower - 32);

}

Output:

Enter any lowercase letter = a

The uppercase letter is = A

8(2) – Lower to Uppercase letter using library function.

#include <stdio.h>

int main()

{

char lower, upper;

printf("Enter any lowercase letter = ");

scanf("%c", &lower);

upper = toupper(lower);

printf("The uppercase letter is = %c\n", upper);

}

Output:

Enter any lowercase letter = a

The uppercase letter is = A

9(1) – Upper to Lowercase letter.

#include <stdio.h>

int main()

{

char upper;

printf("Enter any uppercase letter = ");

scanf("%c", &upper);

printf("The lowercase letter is = %c\n", upper + 32);

}

Output:

Enter any uppercase letter = A

The lowercase letter is = a

9(2) – Upper to Lowercase letter using library function.

#include <stdio.h>

int main()

{

char upper, lower;

printf("Enter any uppercase letter = ");

scanf("%c", &upper);

lower = tolower(upper);

printf("The lawercase letter is = %c\n", lower);

}

Output:

Enter any uppercase letter = A

The lawercase letter is = a

10(1) – Decimal to Octal.

Output:

Enter any decimal number = 10

The Octal number is = 12

#include <stdio.h>

int main()

{

int num;

printf("Enter any decimal number = ");

scanf("%d", &num);

printf("The Octal number is = %o\n", num);

}

10(2) – Octal to Decimal.

Output:

Enter any octal number = 10

The decimal number is = 8

#include <stdio.h>

int main()

{

int num;

printf("Enter any octal number = ");

scanf("%o", &num);

printf("The decimal number is = %d\n", num);

}

10(3) – Decimal to Hexa.

Output:

Enter any decimal number = 15

The hexadecimal number is = f

#include <stdio.h>

int main()

{

int num;

printf("Enter any decimal number = ");

scanf("%d", &num);

printf("The hexadecimal number is = %x\n", num);

}

10(4) – Hexa to Decimal.

Output:

Enter any hexadecimal number = f

The decimal number is = 15

#include <stdio.h>

int main()

{

int num;

printf("Enter any hexadecimal number = ");

scanf("%x", &num);

printf("The decimal number is = %d\n", num);

}

10(5) – Octal to Hexa.

Output:

Enter any octal number = 17

The hexadecimal number is = f

#include <stdio.h>

int main()

{

int num;

printf("Enter any octal number = ");

scanf("%o", &num);

printf("The hexadecimal number is = %x\n", num);

}

10(6) – Hexa to Octal.

Output:

Enter any hexadecimal number = f

The Octal number is = 17

#include <stdio.h>

int main()

{

int num;

printf("Enter any hexadecimal number = ");

scanf("%x", &num);

printf("The Octal number is = %o\n", num);

}

11 – Add, Sub, Division, Multiplication, Remainder.

Output:

Enter two number = 10 5

Sum is = 15.00

Sub is = 5.00

Mul is = 50.00

Div is = 2.00

Mod is = 0

#include <stdio.h>

#include <math.h>

int main()

{

double num1, num2, result;

printf("Enter two number = ");

scanf("%lf %lf", &num1, &num2);

result = num1 + num2;

printf("Sum is = %.2lf\n", result);

result = num1 - num2;

printf("Sub is = %.2lf\n", result);

result = num1 \* num2;

printf("Mul is = %.2lf\n", result);

result = num1 / num2;

printf("Div is = %.2lf\n", result);

(int)result = (int)num1 % (int)num2;

printf("Mod is = %d\n", result);

}

12 – Area of a Triangle.

Output:

Enter baseand height = 6.8 5.4

The area of triangle is = 18.36

#include <stdio.h>

int main()

{

float base, height, area;

printf("Enter base and height = ");

scanf("%f %f", &base, &height);

area = 0.5 \* base \* height; //(float)1/2

printf("The area of triangle is = %.2f\n", area);

}

13 – Area of a Rectangle.

Output:

Please enter lengthand width = 10.4 5.6

The area of rectangul is = 58.24

#include <stdio.h>

int main()

{

float length, width, area;

printf("Please enter length and width = ");

scanf("%f %f", &length, &width);

area = length \* width;

printf("The area of rectangul is = %.2f\n", area);

}

14 – Area of a triangle given three arms length.

(ত্রিভূজ এর তিন বাহুর দৈর্ঘ্য দেওয়া থাকলে ত্রিভূজের ক্ষেত্রফল নির্ণয় করার সি প্রোগ্রাম)

Output:

Enter the value of a band c = 15 10 5

The area is = 639.00

#include <stdio.h>

int main()

{

double a, b, c, s, area;

printf("Enter the value of a b and c = ");

scanf("%lf %lf %lf", &a, &b, &c);

s = (a + b + c) / 2;

area = sqrt(s \* (s - a) \* (s - b) \* (s - c));

printf("The area is = %.2lf\n", area);

}

15 – Area of a Circle.

Output:

Enter the radius of the circle = 5.2

The area is = 84.95

#include <stdio.h>

int main()

{

float radius, area;

printf("Enter the radius of the circle = ");

scanf("%f", &radius);

area = 3.1416 \* radius \* radius;

printf("The area is = %.2f\n", area);

}

16(1) – Celcious to Farenheit.

#include <stdio.h>

int main()

{

float f, c;

printf("Enter the celcious temperatre = ");

scanf("%f", &c);

f = (c \* 1.8) + 32;

printf("The Farenheit temperature is = %.2f\n", f);

}

Output:

Enter the celcious temperatre = 37

The Farenheit temperature is = 98.60

16(2) – Farenheit to Celcious.

#include <stdio.h>

int main()

{

float f, c;

printf("Enter the farenheit temperatre = ");

scanf("%f", &f);

c = (f - 32) / 1.8;

printf("The Celcious temperature is = %.2f\n", c);

}

Output:

Enter the farenheit temperatre = 100

The Celcious temperature is = 37.78

17(1) – Swapping two number using temporary variable.

#include <stdio.h>

int main()

{

int num1 = 10;

int num2 = 5;

int temp;

temp = num1;

num1 = num2;

num2 = temp;

printf("After swapping number 1 is = %d\n", num1);

printf("After swapping number 2 is = %d\n", num2);

}

Output:

After swapping number 1 is = 5

After swapping number 2 is = 10

17(2) – Swapping two number without temporary variable.

#include <stdio.h>

int main()

{

int num1 = 10, num2 = 5;

num1 = num1 - num2; //10-5 = 5

num2 = num1 + num2; //5+5 = 10

num1 = num2 - num1; //10-5 = 5

printf("So, the number 1 is = %d\n", num1);

printf("So, the number 2 is = %d\n", num2);

}

Output:

So, the number 1 is = 5

So, the number 2 is = 10

18 – Quadratic equation(ax2 + bx + c).

#include <stdio.h>

int main()

{

double a, b, c, d, x1, x2;

printf("Enter the value of a, b, c = ");

scanf("%lf %lf %lf", &a, &b, &c);

d = sqrt(b \* b - 4 \* a \* c);

x1 = (-b + d) / 2;

x2 = (-b - d) / 2;

printf("X1 = %.2lf\n", x1);

printf("X2 = %.2lf\n", x2);

}

Output:

Enter the value of a, b, c = 2.3 4.5 6.7

X1 = 5482325.75

X2 = -5482330.25

19 – Absolute value print.

#include <stdio.h>

int main()

{

double result = abs(-7);

printf("The value is = %.0lf\n", result);

}

Output:

The value is = 7

20(1) – Square root.

#include <stdio.h>

#include <math.h>

int main()

{

double result = sqrt(49);

printf("The value is = %.2lf\n", result);

}

Output:

The value is = 7.00

20(2) – Suare root from the user.

#include <stdio.h>

#include <math.h>

int main()

{

double num, result;

printf("Enter any number = ");

scanf("%lf", &num);

result = sqrt(num);

printf("The value is = %.2lf\n", result);

}

Output:

Enter any number = 196

The value is = 14.00

21(1) – Power.

#include <stdio.h>

#include <math.h>

int main()

{

double result = pow(5, 2);

printf("The value is = %.2lf\n", result);

}

Output:

The value is = 25.00

21(2) – Power from the user.

#include <stdio.h>

#include <math.h>

int main()

{

double num1, num2, result;

printf("Please enter two number = ");

scanf("%lf %lf", &num1, &num2);

result = pow(num1, num2);

printf("The valur is = %.2lf\n", result);

}

Output:

Please enter two number = 5 2

The valur is = 25.00

22(1) – log(), log10(), exp.

#include <stdio.h>

#include <math.h>

int main()

{

double a = 10.5, b = 1, c = 2;

double result1 = log(a);

double result2 = log(b);

double result3 = exp(c);

printf("log(%.2lf) = %.2lf\n", a, result1);

printf("log(%.2lf) = %.2lf\n", b, result2);

printf("exp(%.2lf) = %.2lf\n", c, result3);

}

Output:

log(10.50) = 2.35

log(1.00) = 0.00

exp(2.00) = 7.39

22(2) – sin, cos, tan.

#include <stdio.h>

#include <math.h>

int main()

{

double a = 2, b = 3, c = 4;

double result1 = sin(a);

double result2 = cos(b);

double result3 = tan(c);

printf("Sin(%.2lf) = %.2lf\n", a, result1);

printf("cos(%.2lf) = %.2lf\n", b, result2);

printf("tan(%.2lf) = %.2lf\n", c, result3);

}

Output:

Sin(2.00) = 0.91

cos(3.00) = -0.99

tan(4.00) = 1.16

23 – Round, Trunc, Ceil, Floor.

#include <stdio.h>

#include <math.h>

int main()

{

double a = 5.55, b = 5.25, c = 5.25, d = 5.25;

double resulta = round(a);

double resultb = trunc(b);

double resultc = ceil(c);

double resultd = floor(d);

printf("Round-%.2lf = %.2lf\n", a, resulta);

printf("Trunc-%.2lf = %.2lf\n", b, resultb);

printf("Ceil-%.2lf = %.2lf\n", c, resultc);

printf("Floor-%.2lf = %.2lf\n", d, resultd);

}

Output:

Round - 5.55 = 6.00

Trunc - 5.25 = 5.00

Ceil - 5.25 = 6.00

Floor - 5.25 = 5.00

round - দশমিক এর পর ৫ এর বড় হলে পরের পূর্নসংখ্য প্রিন্ট করবে।

trunc - দশমিক এর পরের সংখ্যগুলো বাদ পড়ে যাবে।

ceil - দশমিক এর পর কেবল পূর্নসংখ্য প্রিন্ট করবে(যেমন - ২.৩ থাকলে ৩)।

floor - দশমিক এর আগের কেবল পূর্নসংখ্য প্রিন্ট করবে(যেমন - ২.৩ থাকলে ২)।

24 – Assignment operator.

Output:

7

50

#include <stdio.h>

int main()

{

int a = 5, b = 10;

a += 2; //a = a+2 = 7.

b \*= 5; //b = b\*5 = 50.

printf("%d\n%d\n", a, b);

}

25(1) – Unary Operator.

Output:

y1 = 10

y2 = -10

y3 = 10

y4 = 9

y5 = 10

#include <stdio.h>

int main()

{

int x1 = 10, x2 = 10, x3 = 10, x4 = 10, x5 = 10;

int y1 = +x1, y2 = -x2, y3 = x3--, y4 = --x4, y5 = x5++;

printf("y1 = %d\n", y1); //10

printf("y2 = %d\n", y2); //-10

printf("y3 = %d\n", y3); //10

printf("y4 = %d\n", y4); //9

printf("y5 = %d\n", y5); //10

}

25(2) – Unary Operator.

Output:

10

11

12

12

12

11

10

#include <stdio.h>

int main()

{

int x = 10;

printf("%d\n", x++); //10

printf("%d\n", x); //11

printf("%d\n", ++x); //12

printf("%d\n", x); //12

printf("%d\n", x--); //12

printf("%d\n", x); //11

printf("%d\n", --x); //10

}

26. Print a character.

#include <stdio.h>

int main()

{

char ch;

printf("Enter the first letter of your name = ");

//scanf("%c", &ch);

ch = getchar();

printf("The first letter is = %c\n", ch);

}

Output:

Enter the first letter of your name = d

The first letter is = d

27. Simple sum, sub, mul, div.

#include <stdio.h>

int main()

{

double num1, num2;

printf("Enter the first number = ");

scanf("%lf", &num1);

printf("Enter the second number = ");

scanf("%lf", &num2);

printf("%lf + %lf = %lf\n", num1, num2, num1 + num2);

printf("%lf - %lf = %lf\n", num1, num2, num1 - num2);

printf("%lf \* %lf = %lf\n", num1, num2, num1 \* num2);

printf("%lf / %lf = %.2lf\n", num1, num2, num1 / num2);

}

Output:

Enter the first number = 20.7

Enter the second number = 10.3

20.700000 + 10.300000 = 31.000000

20.700000 - 10.300000 = 10.400000

20.700000 \* 10.300000 = 213.210000

20.700000 / 10.300000 = 2.01

28. Simple sum, sub, add, mul using character sign.

#include <stdio.h>

int main()

{

double num1, num2, value;

char sign;

printf("Enter the value of first number = ");

scanf("%lf", &num1);

printf("Enter the value of second number = ");

scanf("%lf", &num2);

value = num1 + num2;

sign = '+';

printf("%.2lf %c %.2lf = %.2lf\n", num1, sign, num2, value);

value = num1 - num2;

sign = '-';

printf("%.2lf %c %.2lf = %.2lf\n", num1, sign, num2, value);

value = num1 \* num2;

sign = '\*';

printf("%.2lf %c %.2lf = %.2lf\n", num1, sign, num2, value);

value = num1 / num2;

sign = '/';

printf("%.2lf %c %.2lf = %.2lf\n", num1, sign, num2, value);

}

Output:

Enter the value of first number = 10.5

Enter the value of second number = 5.5

10.50 + 5.50 = 16.00

10.50 - 5.50 = 5.00

10.50 \* 5.50 = 57.75

10.50 / 5.50 = 1.91

29. Integer to Character.

Output:

C E c

#include <stdio.h>

int main()

{

char ch1 = 67, ch2 = 69, ch3 = 99;

printf("%c %c %c\n", ch1, ch2, ch3);

}

30. দশ হাজার টাকার ঋন নিলে ৩৫% সুদ। তাহলে ৫ বছর পর সুদ-আসলে মোট

কত টাকা দিতে হবে। এবং প্রতি মাসে কত টাকা করে দিতে হবে।

#include <stdio.h>

int main()

{

double loanamount, interestrate, years, totalamount, monthlyamount;

printf("Enter the loan amount = ");

scanf("%lf", &loanamount);

printf("Enter the interest rate = ");

scanf("%lf", &interestrate);

printf("Enter total years = ");

scanf("%lf", &years);

totalamount = loanamount + loanamount \* interestrate / 100;

monthlyamount = totalamount / (years \* 12);

printf("Total amount = %.2lf\n", totalamount);

printf("Monthly amount = %.2lf\n", monthlyamount);

}

Output:

Enter the loan amount = 10000

Enter the interest rate = 35

Enter total years = 5

Total amount = 13500.00

Monthly amount = 225.00

31. সমীকরণ থেকে x এর মান নির্নয়।

#include <stdio.h>

int main()

{

double a1, a2, b1, b2, c1, c2, x, y, d;

printf("Enter a1 = ");

scanf("%lf", &a1);

printf("Enter a2 = ");

scanf("%lf", &a2);

printf("Enter b1 = ");

scanf("%lf", &b1);

printf("Enter b2 = ");

scanf("%lf", &b2);

printf("Enter c1 = ");

scanf("%lf", &c1);

printf("Enter c2 = ");

scanf("%lf", &c2);

d = (a1 \* b2 - a2 \* b1);

if ((int)d == 0) //or if(d == 0.0)

{

printf("Value of x and y cannot be determined\n");

}

else

{

x = (b2 \* c1 - b1 \* c2) / d;

y = (a1 \* c2 - a2 \* c1) / d;

}

printf("X = %.2lf\n", x);

printf("Y = %.2lf\n", y);

}

Output:

Enter a1 = 1

Enter a2 = 2

Enter b1 = 3

Enter b2 = 4

Enter c1 = 5

Enter c2 = 6

X = -1.00

Y = 2.00

Here,

x = (b2c1-b1c2)/(a1b2-a2b1)

y = (a1c2-a2c1)/(a1b2-a2b1)

Condition(if/else/switch)

1. Determine a number even or odd.

2. Multiple Common Statement.

3. Large number between the two number.

4. Determine marks.

5. Determine a number is positive or negative.

6. Vowel or Consonant using relational operator.

7. Vowel or Consonant using logical operator.

8. Large number between the three number.

9. Leap Year.

10(1).Pass or Fail.

10(2).Letter grade.

11. Capital letter or Small letter.

12(1). Local Variable

12(2). Vowel or Consonant using switch.

13. Read a digit and display it’s spelling.

14. Menu Based temperature.

15. Switch calculator.

16. Taking input from user using conditional operator.

17. Bitwiseand, or , xor .

18. Uses of goto keyword.

19. OR operator

20. Grid Traversal – গ্রিড ট্রাভার্সাল বা রোবটের ভ্রমণ

1. Determine a number even or odd.

Output:

Enter the number = 4

This is even number

#include <stdio.h>

int main()

{

int num;

printf("Enter the number = ");

scanf("%d", &num);

if (num % 2 == 0)

printf("This is even number\n");

else

printf("This is odd number\n");

}

2. Multiple Common Statement.

Output:

Good Morning

Golam Kibria

#include <stdio.h>

int main()

{

int time = 10;

if (time == 10)

{

printf("Good Morning\n");

printf("Golam Kibria\n");

}

else

{

printf("Sorry it's not morning\n");

printf("Golam Kibria you can sleep little more\n");

}

}

3. Large number between the two number.

#include <stdio.h>

int main()

{

int num1, num2;

printf("Please enter two number = ");

scanf("%d %d", &num1, &num2);

if (num1 > num2)

printf("Large number is = %d\n", num1);

else if (num2 > num1)

printf("Large number is = %d\n", num2);

else

printf("Numbers are equal\n");

}

Output:

Please enter two number = 10 5

Large number is = 10

4. Determine marks.

#include <stdio.h>

int main()

{

int num;

printf("Please enter a number = ");

scanf("%d", &num);

if (num >= 80) printf("Your grade is A+");

else if (num >= 70) printf("Your grade is A");

else if (num >= 60) printf("Your grade is A-");

else if (num < 33) printf("FAIL!!");

}

Output:

Please enter a number = 70

Your grade is A

5. Determine a number is positive or negative.

Output:

Please enter a number = 6

The number is positive

#include <stdio.h>

int main()

{

int num;

printf("Please enter a number = ");

scanf("%d", &num);

if (num > 0) printf("The number is positive\n");

else if (num < 0) printf("The number is negative\n");

else printf("The number is zero\n");

}

6. – Vowel or Consonant using relational operator.

Output:

Enter any character = x

Consonant

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character = ");

scanf("%c", &ch);

if (ch == 'a') printf("Vowel");

else if (ch == 'e') printf("Vowel");

else if (ch == 'i') printf("Vowel");

else if (ch == 'o') printf("Vowel");

else if (ch == 'u') printf("Vowel");

else if (ch == 'A') printf("Vowel");

else if (ch == 'E') printf("Vowel");

else if (ch == 'I') printf("Vowel");

else if (ch == 'O') printf("Vowel");

else if (ch == 'U') printf("Vowel");

else printf("Consonant");

}

7. – Vowel or Consonant using logical operator.

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character = ");

scanf("%c", &ch);

if (ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u' || ch == 'A' || ch == 'E' || ch == 'I' || ch == 'O' || ch == 'U')

printf("Vowel\n");

else printf("Consonant\n");

}

Output:

Enter any character = a

Vowel

8. Large number between the three number.

#include <stdio.h>

int main()

{

int num1, num2, num3;

printf("Enter the three value = ");

scanf("%d %d %d", &num1, &num2, &num3);

if (num1 > num2 && num1 > num3)

printf("%d is the large number\n", num1);

else if (num2 > num1 && num2 > num3)

printf("%d is the large numbre\n", num2);

else if (num3 > num1 && num3 > num2)

printf("%d is the large number\n", num3);

}

Output:

Enter the three value = 12 23 31

31 is the large number

9. Leap Year.

#include <stdio.h>

int main()

{

int year;

printf("Enter a year = ");

scanf("%d", &year);

if (year % 400 == 0)

printf("Leap Year\n");

else if (year % 4 == 0 && year % 100 != 0)

printf("Leap Year");

else printf("Not a leap year\n");

}

Output:

Enter a year = 2000

Leap Year

10(1). Pass or Fail.

#include <stdio.h>

int main()

{

int marks;

printf("Please enter your marks = ");

scanf("%d", &marks);

if (marks >= 33)

printf("pass");

else

printf("Fail");

}

Output:

Please enter your marks = 78

pass

10(2). Letter grade.

#include <stdio.h>

int main()

{

int marks;

printf("Please enter your marks = ");

scanf("%d", &marks);

if (marks > 100 || marks < 0) printf("Invalid Marks\n");

else if (marks >= 80 && marks <= 100) printf("A+");

else if (marks >= 70 && marks <= 79) printf("A");

else if (marks >= 60 && marks <= 69) printf("A-");

else if (marks >= 50 && marks <= 59) printf("B");

else if (marks >= 33 && marks <= 49) printf("D");

else printf("Fail");

}

Output:

Please enter your marks = 87

A +

11. Capital letter or Small letter.

#include <stdio.h>

int main()

{

char ch;

printf("Please inter a character = ");

scanf("%c", &ch);

if (ch >= 'A' && ch <= 'Z')

printf("Capital Letter\n");

else if (ch >= 'a' && ch <= 'z')

printf("Small Letter\n");

else

printf("Not a letter\n");

}

Output:

Please inter a character = A

Capital Letter

12(1). Local Variable

#include <stdio.h>

int main()

{

int a = 10;

printf("The value of a = %d\n", a);

/\*This is into the main function so it's called local variable\*/

}

Output:

The value of a = 10

12(2). Vowel or Consonant using switch.

#include <stdio.h>

int main()

{

char ch;

printf("Enter any character = ");

scanf("%c", &ch);

switch (ch)

{

case 'a':

case 'b':

case 'e':

case 'i':

case 'u':

case 'A':

case 'E':

case 'I':

case 'O':

case 'U':

printf("Vowel\n");

break;

default:

printf("Consonant\n");

}

}

Output:

Enter any character = a

Vowel

13. Read a digit and display it’s spelling.

#include <stdio.h>

int main()

{

int digit;

printf("Enter any digit = ");

scanf("%d", &digit);

switch (digit)

{

case 0:

printf("Zero\n");

break;

case 1:

printf("One\n");

break;

case 2:

printf("Two\n");

break;

case 3:

printf("Three\n");

break;

case 4:

printf("Four\n");

break;

case 5:

printf("Five\n");

break;

case 6:

printf("Six\n");

break;

case 7:

printf("Seven\n");

break;

case 8:

printf("Eight\n");

break;

case 9:

printf("Nine\n");

break;

default:

printf("Not a valid digit\n");

}

}

Output:

Enter any digit = 3

Three

14. Menu Based temperature.

#include <stdio.h>

int main()

{

int choice;

float c, f;

printf("Temperature conversion menu:\n");

printf("1. Farenheit to Celcious.\n");

printf("2. Celcious to Farenheit.\n");

printf("Please enter your choice: ");

scanf("%d", &choice);

switch (choice)

{

case 1:

printf("Enter Farenheit tempeature = ");

scanf("%f", &f);

c = (f - 32) / 1.8;

printf("The temperature in Celcious is = %.2f\n", c);

break;

case 2:

printf("Enter Celcious temperature = ");

scanf("%f", &c);

f = (c \* 1.8) + 32;

printf("The temperature in Farenheit is = %.2f\n", f);

break;

default:

printf("Not a correct option\n");

}

}

Output:

Temperature conversion menu :

1. Farenheit to Celcious.

2. Celcious to Farenheit.

Please enter your choice : 1

Enter Farenheit tempeature = 100

The temperature in Celcious is = 37.78

15. Switch calculator.

#include <stdio.h>

int main()

{

double num1, num2;

char operator;

printf("Enter any operator(+, -, \*, /) = \n");

scanf("%ch", &operator);

printf("Please enter two number = \n");

scanf("%lf %lf", &num1, &num2);

switch (operator)

{

case '+':

printf("%.2lf + %.2lf = %.2lf\n", num1, num2, num1 + num2);

break;

case '-':

printf("%.2lf - %.2lf = %.2lf\n", num1, num2, num1 - num2);

break;

case '\*':

printf("%.2lf \* %.2lf = %.2lf\n", num1, num2, num1 \* num2);

break;

case '/':

printf("%.2lf / %.2lf = %.2lf\n", num1, num2, num1 / num2);

break;

default:

printf("Not a valid operator\n");

}

}

Output:

Enter any operator(+, -, \*, / ) =

+

Please enter two number =

12 8

12.00 + 8.00 = 20.00

16. Taking input from user using conditional operator.

Output:

Please enter two number = 12 8

Large number is = 12

#include <stdio.h>

int main()

{

int num1, num2, large;

printf("Please enter two number = ");

scanf("%d %d", &num1, &num2);

large = (num1 > num2) ? num1 : num2;

printf("Large number is = %d\n", large);

}

17. Bitwiseand, or , xor .

Output:

The result is = 0

The result is = 15

The result is = 15

#include <stdio.h>

int main()

{

int a = 10, b = 5, c;

c = a & b;

/\*a ও b এর বাইনারি গুণফল এর মান বের করে তা ডেসিমেল আকারে প্রিন্ট করবে\*/

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

c = a | b;

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

c = a ^ b;

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

}

19. OR operator

18. Uses of goto keyword.

#include <stdio.h>

#include <stdio.h>

int main()

{

int n = 5;

if (n >= 1 || n <= 10) printf("YES\n");

else printf("NO\n");

}

Output:

YES

#include <stdio.h>

int main()

{

int n = 5;

if (n >= 1 || n <= 10) printf("YES\n");

else printf("NO\n");

}

Output:

YES

#include <stdio.h>

int main()

{

int n = 5;

if (n >= 1 || n <= 10) printf("YES\n");

else printf("NO\n");

}

Output:

YES

int main()

{

int i = 1;

kibria:

printf("%d\n", i);

i++;

if (i < 5)

{

goto kibria;

}

}

Output:

1

2

3

4

20. Grid Traversal – গ্রিড ট্রাভার্সাল বা রোবটের ভ্রমণ

#include <stdio.h>

int main()

{

int x, y;

char ch;

printf("Please enter the enitial position = ");

scanf("%d %d", &x, &y); //5, 5

while (1)

{

scanf("%c", &ch);

if (ch == 's') break;

else if (ch == 'u') x--; //

else if (ch == 'd') x++;

else if (ch == 'r') y++;

else if (ch == 'l') y--;

}

printf("Final position of the robot is = %d, %d\n", x, y);

}

Output:

Please enter the enitial position = 2 2

d //3 2

r //3 3

d //4 3

r //4 4

s

Final position of the robot is = 4, 4

//s-stop, u-up, d-down, r-right, l-left

Loop(লুপ)

1(1). Prints ten times a letter using for loop.

1(2). Prints ten times a letter using while loop.

1(3). Prints ten times a letter using do while loop.

2(1). Even number between 1 - 100.

2(2). Odd number between 1 - 100.

3. Uses of break and continue statement.

4. Making a Multiple table.

5. Factorial Print.

6. Prime number print.

7. GCD and LCM.

8. Display sum of a digit.

9. Reverse an Integer.

10. Palindrome number.

11(1). Armstrong number or not.

11(2). Armstrong number between 1 - 1000.

12. Counting number of a digit in an integer.

13. Strong number print.

14(1). Basic Multiple Table(নামতা তৈরি করা)

14(2). Basic Multiple Table(যোগের মাধ্যমে নামতা তৈরি করা)

15. ১ - ২০ পর্যন্ত সবগুলো সংখ্যার নামতা

16(1). Uses of break statement.

16(2). Uses of continue statement.

16(3). Uses of continue statement.

17. Input - Output

1(1). Prints ten times a letter using for loop.

Output:

1. kibria

2. kibria

3. kibria

4. kibria

5. kibria

6. kibria

7. kibria

8. kibria

9. kibria

10.kibria

#include <stdio.h>

int main()

{

int i;

for (i = 1; i <= 10; i++)

{

printf("%d. kibria\n", i);

}

}

1(2). Prints ten times a letter using while loop.

Output:

1. Kibria

2. Kibria

3. Kibria

4. Kibria

5. Kibria

6. Kibria

7. Kibria

8. Kibria

9. Kibria

10.Kibria

#include <stdio.h>

int main()

{

int i = 1;

while (i <= 10)

{

printf("%2d. Kibria\n", i);

i++;

}

}

1(3). Prints ten times a letter using do while loop.

Output:

1. kibria

2. kibria

3. kibria

4. kibria

5. kibria

6. kibria

7. kibria

8. kibria

9. kibria

10.kibria

#include <stdio.h>

int main()

{

int i = 1;

do

{

printf("%d. kibria\n", i);

i++;

} while (i <= 10);

}

2(1). Even number between 1 - 100.

Output:

2

4

6

8

10

12

-

-

#include <stdio.h>

int main()

{

int i;

for (i = 2; i <= 100; i = i + 2)

{

printf("%d\n", i);

}

}

2(2). Odd number between 1 - 100.

Output:

1

3

5

7

-

-

#include <stdio.h>

int main()

{

int i;

for (i = 1; i <= 100; i = i + 2)

{

printf("%d\n", i);

}

}

3. Uses of breakand continue statement.

Output:

1

2

4

5

7

8

10

#include <stdio.h>

int main()

{

int i;

for (i = 1; i < 20; i++)

{

if (i % 3 == 0)

{

continue;

}

printf("%d\n", i);

if (i == 10)

{

break;

}

}

}

4. Making a Multiple table.

Output:

Enter any number = 5

5 X 1 = 5

5 X 2 = 10

5 X 3 = 15

5 X 4 = 20

5 X 5 = 25

5 X 6 = 30

5 X 7 = 35

5 X 8 = 40

5 X 9 = 45

5 X 10 = 50

#include <stdio.h>

int main()

{

int i, num;

printf("Enter any number = ");

scanf("%d", &num);

for (i = 1; i <= 10; i++)

{

printf("%d X %d = %d\n", num, i, num \* i);

}

return 0;

}

5. Factorial Print.

Output:

Enter any positive number = 4

The Factorial of 4 is = 24

#include <stdio.h>

int main()

{

int i, num, factorial = 1;

printf("Enter any positive number = ");

scanf("%d", &num);

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

{

factorial = factorial \* i;

}

printf("The Factorial of %d is = %d\n", num, factorial);

return 0;

}

6. Prime number print.

Output:

Enter a number = 5

This is a prime number

#include <stdio.h>

int main()

{

int num, i, count = 0;

printf("Enter a number = ");

scanf("%d", &num);

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

{

if (num % 2 == 0)

{

count++;

break;

}

}

if (count == 0)

printf("This is a prime number\n");

else

printf("This is not a prime number\n");

}

7. GCD and LCM.

Output:

Please enter two number = 12 20

The GCD is = 4

The LCM is = 60

#include <stdio.h>

int main()

{

int num1, num2, n1, n2, rem, gcd, lcm;

printf("Please enter two number = ");

scanf("%d %d", &num1, &num2);

n1 = num1;

n2 = num2;

while (n2 != 0)

{

rem = n1 % n2;

n1 = n2;

n2 = rem;

}

gcd = n1;

lcm = (num1 \* num2) / gcd;

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

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

}

8. Display sum of a digit.

Output:

Enter any number = 137

The sum is = 11

#include <stdio.h>

int main()

{

int num, temp, rem, sum = 0;

printf("Enter any number = ");

scanf("%d", &num);

temp = num;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

sum = sum + rem;

}

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

}

9. Reverse an Integer.

Output:

Enter any number = 123

Reverse of the number is = 321

#include <stdio.h>

int main()

{

int num, rem, temp, sum = 0;

printf("Enter any number = ");

scanf("%d", &num);

temp = num;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

sum = sum \* 10 + rem;

}

printf("Reverse of the number is = %d\n", sum);

}

10. Palindrome number.

Output:

Enter any number = 123

This is not a palindrome number

Enter any number = 121

This is a palindrome number

#include <stdio.h>

int main()

{

int num, rem, temp, sum = 0;

printf("Enter any number = ");

scanf("%d", &num);

temp = num;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

sum = sum \* 10 + rem;

}

if (sum == num)

printf("This is a palindrome number\n");

else

printf("This is not a palindrome number\n");

}

11(1). Armstrong number or not.

Output:

Enter the number = 128

This is not a armstrong number

Enter the number = 153

This is a armstrong number

#include <stdio.h>

int main()

{

int num, i, temp, rem, sum = 0;

printf("Enter the number = ");

scanf("%d", &num);

temp = num;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

sum = sum + rem \* rem \* rem;

}

if (sum == num)

printf("This is a armstrong number\n");

else

printf("This is not a armstrong number\n");

}

11(2) - Armstrong number between 1 - 1000.

Output:

Initial value = 1

Final value = 1000

1

153

370

371

407

#include <stdio.h>

int main()

{

int initialvalue, finalvalue, rem, i, temp, sum = 0;

printf("Initial value = ");

scanf("%d", &initialvalue);

printf("Final value = ");

scanf("%d", &finalvalue);

for (i = initialvalue; i < finalvalue; i++)

{

temp = i;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

sum = sum + rem \* rem \* rem;

}

if (sum == i)

{

printf("%d\n", i);

}

sum = 0;

}

}

12. Counting number of a digit in an integer.

Output:

Please enter the number = 134

Total number of digit = 3

#include <stdio.h>

int main()

{

int num, count = 0;

printf("Please enter the number = ");

scanf("%d", &num);

while (num != 0)

{

num = num / 10;

++count;

}

printf("Total number of digit = %d\n", count);

}

13. Strong number printf. (ফ্যাক্টরিয়াল গুলোর যোগফল ঐ সংখ্যাটির সমান)

strong number = 145 = 1! + 4! + 5! = 145.

Output:

Enter the number = 145

This is a strong number

#include <stdio.h>

int main()

{

int num, i, rem, temp, sum = 0, fact;

printf("Enter the number = ");

scanf("%d", &num);

temp = num;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

fact = 1;

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

{

fact = fact \* i;

}

sum = sum + fact;

}

if (sum == num)

printf("This is a strong number\n");

else

printf("This is not a strong number\n");

}

14(1). Basic Multiple Table(নামতা তৈরি করা)

Output:

5 X 1 = 5

5 X 2 = 10

5 X 3 = 15

5 X 4 = 20

5 X 5 = 25

5 X 6 = 30

5 X 7 = 35

5 X 8 = 40

5 X 9 = 45

5 X 10 = 50

#include <stdio.h>

int main()

{

int i, n = 5;

for (i = 1; i <= 10; i++)

{

printf("%d X %d = %d\n", n, i, n \* i);

}

}

14(2). Basic Multiple Table(যোগের মাধ্যমে নামতা তৈরি করা)

Output:

5 X 1 = 5

5 X 2 = 10

5 X 3 = 15

5 X 4 = 20

5 X 5 = 25

5 X 6 = 30

5 X 7 = 35

5 X 8 = 40

5 X 9 = 45

5 X 10 = 50

#include <stdio.h>

int main()

{

int i, sum = 0, n = 5;

for (i = 1; i <= 10; i++)

{

sum = sum + n;

printf("%d X %d = %d\n", n, i, sum);

}

}

15. ১-২০ পর্যন্ত সবগুলো সংখ্যার নামতা

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

-------------

#include <stdio.h>

int main()

{

int i, j;

for (i = 1; i <= 20; i++)

{

for (j = 1; j <= 10; j++)

{

printf("%d X %d = %d\n", i, j, i \* j);

}

printf("\n");

}

}

16(1). Uses of break statement.

Output:

1

2

3

4

5

6

7

8

9

10

#include <stdio.h>

int main()

{

int i;

for (i = 1; i <= 100; i++)

{

printf("%d\n", i);

if (i >= 10)

{

break;

}

}

}

16(2). Uses of continue statement.

\*

Output:

1

3

5

7

9

#include <stdio.h>

int main()

{

int i;

for (i = 1; i <= 10; i++)

{

if (i % 2 == 0)

{

continue;

}

printf("%d\n", i);

}

}

16(3). Uses of continue statement.

Output:

Enter the value of n = 1

Enter the value of m = 10

10

#include <stdio.h>

int main()

{

int n, m, i, j;

printf("Enter the value of n = ");

scanf("%d", &n);

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

{

printf("Enter the value of m = ");

scanf("%d", &m);

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

{

if (j % 11 == 0)

{

continue;

}

printf("%d\n", j);

}

}

}

17. Input-Output

Output:

1 2 3

1 3 2

2 1 3

2 3 1

3 1 2

3 2 1

#include <stdio.h>

int main()

{

int a, b, c;

for (a = 1; a <= 3; a++)

{

for (b = 1; b <= 3; b++)

{

for (c = 1; c <= 3; c++)

{

if (a != b && a != c && b != c)

{

printf("%d %d %d\n", a, b, c);

}

}

}

}

return 0;

}

Pattern(প্যার্টান)

Patten type – 01

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= row; col++)

{

printf("%d ", col);

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2

1 2 3

2. printf("%d ", row);

3. printf("%d ", col % 2);

4. printf("%d ", row % 2);

5. printf("%c ", col + 64);

6. printf("%c ", row + 64);

7. printf("%c ", col + 96);

8. printf("%c ", row + 96);

9. printf("\* ");

10.printf("# ");

Patten type – 02.

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

for (row = n; row >= 1; row--)

{

for (col = 1; col <= row; col++)

{

printf("%d ", col);

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1 2 3

1 2

1

2. printf("%d ", row);

3. printf("%d ", col % 2);

4. printf("%d ", row % 2);

5. printf("%c ", col + 64);

6. printf("%c ", row + 64);

7. printf("%c ", col + 96);

8. printf("%c ", row + 96);

9. printf("\* ");

10.printf("# ");

Patten type – 03.

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= row; col++)

{

printf("%d ", col);//2 space

}

printf("\n");

}

for (row = n - 1; row >= 1; row--)

{

for (col = 1; col <= row; col++)

{

printf("%d ", col);//2 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2

1 2 3

1 2

1

2. printf("%d ", row);//dui bar kory hoby.

3. printf("%d ", col % 2);

4. printf("%d ", row % 2);

5. printf("%c ", col + 64);

6. printf("%c ", row + 64);

7. printf("%c ", col + 96);

8. printf("%c ", row + 96);

9. printf("\* ");

10.printf("# ");

Patten type – 04.

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col); //1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2

1 2 3

2. printf("%d ", row);// 1 space.

3. printf("%d ", col % 2);

4. printf("%d ", row % 2);

5. printf("%c ", col + 64);

6. printf("%c ", row + 64);

7. printf("%c ", col + 96);

8. printf("%c ", row + 96);

9. printf("\* ");

10.printf("# ");

Patten type – 05.

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

for (row = n; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col); //1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1 2 3

1 2

1

2. printf("%d ", row);//1 space.

3. printf("%d ", col % 2);

4. printf("%d ", row % 2);

5. printf("%c ", col + 64);

6. printf("%c ", row + 64);

7. printf("%c ", col + 96);

8. printf("%c ", row + 96);

9. printf("\* ");

10.printf("# ");

Patten type – 06.

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

printf(" "); //2 space

for (col = 1; col <= row; col++)

printf("%d ", col); //1 space

printf("\n");

}

for (row = n - 1; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

printf(" "); //2 space

for (col = 1; col <= row; col++)

printf("%d ", col); //1 space

printf("\n");

}

}

Output:

Enter n = 3

1

1 2

1 2 3

1 2

1

2. printf("%d ", row);//1 space. dui bar kory hoby.

3. printf("%d ", col % 2);

4. printf("%d ", row % 2);

5. printf("%c ", col + 64);

6. printf("%c ", row + 64);

7. printf("%c ", col + 96);

8. printf("%c ", row + 96);

9. printf("\* ");

10.printf("# ");

Pattern type- 07

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

printf("%d ", col); //2 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1 2 3

1 2 3

1 2 3

Pattern type- 08

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2 3

1 2 3 4 5

Pattern type- 09

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

for (row = n; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1 2 3 4 5

1 2 3

1

Pattern type- 10

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

for (row = n - 1; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

return 0;

}

Enter n = 3

1

1 2 3

1 2 3 4 5

1 2 3

1

Pattern type- 11

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //1 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2

1 2 3

Pattern type- 12

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

for (row = n; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //1 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1 2 3

1 2

1

Pattern type- 13

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //1 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

for (row = n - 1; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //1 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col);//1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2

1 2 3

1 2

1

Pattern type- 14

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= row; col++)

{

printf("%d ", row \* col); //1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

2 4

3 6 9

Pattern type- 15

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

if (row == 1 || row == n || col == 1 || col == n)

printf("\* "); //1 space

else

printf(" "); //2 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 5

\* \* \* \* \*

\* \*

\* \*

\* \*

\* \* \* \* \*

Pattern type- 16

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

if (row == n || col == 1 || row == col)

printf("\* "); //1 space

else

printf(" "); //2 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 5

\*

\* \*

\* \*

\* \*

\* \* \* \* \*

Pattern type- 17

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

if (row == col || row + col == n+1)

{

printf("\* "); //1 space

}

else

{

printf(" "); //2 space

}

}

printf("\n");

}

return 0;

}

**Output:**

**Enter n = 5**

**\* \***

**\* \***

**\***

**\* \***

**\* \***

Pattern type- 18

#include <stdio.h>

int main()

{

int n, row, col, count = 0;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= row; col++)

{

printf("%d ", ++count);

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

2 3

4 5 6

Pattern type- 19

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col); //1 space

}

for (col = row - 1; col >= 1; col--)

{

printf("%d ", col); //1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1

1 2 1

1 2 3 2 1

Pattern type- 20

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

for (row = n; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col); //1 space

}

for (col = row - 1; col >= 1; col--)

{

printf("%d ", col); //1 space

}

printf("\n");

}

return 0;

}

Output:

Enter n = 3

1 2 3 2 1

1 2 1

1

Pattern type- 21

Output:

Enter n = 3

1

1 2 1

1 2 3 2 1

1 2 1

1

#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col); //1 space

}

for (col = row - 1; col >= 1; col--)

{

printf("%d ", col); //1 space

}

printf("\n");

}

for (row = n -1; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" "); //2 space

}

for (col = 1; col <= row; col++)

{

printf("%d ", col); //1 space

}

for (col = row - 1; col >= 1; col--)

{

printf("%d ", col); //1 space

}

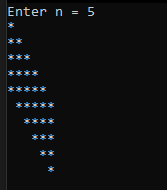
printf("\n");

}

return 0;

}

Pattern type - 22



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter the value of n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= row; col++)

{

printf("\*");

}

printf("\n");

}

for (row = n; row >=1; row--)

{

for (col = 1; col <= (n-row)+1; col++)

{

printf(" ");

}

for (col = 1; col <= row; col++)

{

printf("\*");

}

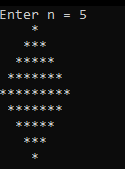
printf("\n");

}

return 0;

}

Pattern type - 23



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter the value of n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= n - row; col++)

{

printf(" ");

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("\*");

}

printf("\n");

}

for (row = n-1; row >= 1; row--)

{

for (col = 1; col <= n - row; col++)

{

printf(" ");

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("\*");

}

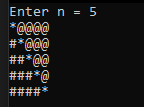
printf("\n");

}

return 0;

}

Pattern type - 24



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter the value of n = ");

scanf("%d", &n);

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

{

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

{

if (row == col)

printf("\*");

else if (row > col)

printf("#");

else if (row < col)

printf("@");

}

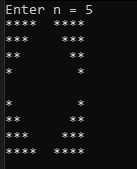
printf("\n");

}

return 0;

}

Pattern type - 25



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter the value of n = ");

scanf("%d", &n);

for (row = n-1; row >= 1; row--)

{

for (col = 1; col <= row; col++)

{

printf("\*");

}

for (col = 1; col <= (2 \* n) - (2 \* row); col++)

{

printf(" ");

}

for (col = 1; col <= row; col++)

{

printf("\*");

}

printf("\n");

}

printf("\n");

for (row = 1; row <= n-1; row++)

{

for (col = 1; col <= row; col++)

{

printf("\*");

}

for (col = 1; col <= (2 \* n) - (2 \* row); col++)

{

printf(" ");

}

for (col = 1; col <= row; col++)

{

printf("\*");

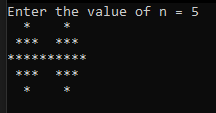
}

printf("\n");

}

}

Pattern type – 26



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter the value of n = ");

scanf("%d", &n);

for (row = 1; row <= n-2; row++)

{

for (col = 1; col <= (n-2) - row; col++)

{

printf(" ");

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("\*");

}

for (col = 1; col <= ((2 \* n-2) - (2 \* row))-2; col++)

{

printf(" ");

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("\*");

}

printf("\n");

}

for (row = (n-2)-1; row >= 1; row--)

{

for (col = 1; col <= (n - 2) - row; col++)

{

printf(" ");

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("\*");

}

for (col = 1; col <= ((2 \* n - 2) - (2 \* row)) - 2; col++)

{

printf(" ");

}

for (col = 1; col <= 2 \* row - 1; col++)

{

printf("\*");

}

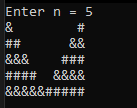
printf("\n");

}

return 0;

}

Pattern type – 27



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

for (col = 1; col <= row; col++)

{

if (row % 2 == 0)

printf("#");

else

printf("&");

}

for (col = 1; col <= (2 \* n) - (2 \* row); col++)

{

printf(" ");

}

for (col = 1; col <= row; col++)

{

if (row % 2 == 0)

printf("&");

else

printf("#");

}

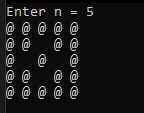
printf("\n");

}

return 0;

}

Pattern type - 28



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

if (row == 1 || row == n || col == 1 || col == n || row == col || row + col == n + 1)

printf("@ ");

else

printf(" ");

}

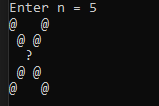
printf("\n");

}

return 0;

}

Pattern type – 29



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

if (row == col && row + col == n + 1)

printf("?");

else if (row == col || row + col == n + 1)

printf("@");

else

printf(" ");

}

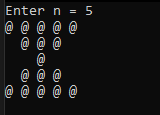
printf("\n");

}

return 0;

}

Pattern type – 30



#include <stdio.h>

int main()

{

int n, row, col;

printf("Enter n = ");

scanf("%d", &n);

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

{

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

{

if (row == 1 || row == n || row == col || row + col == n + 1 || col == n - 2)

printf("@ ");

else

printf(" ");

}

printf("\n");

}

return 0;

}

Series(সিরিজ)

1. 1 + 2 + 3 + ---------- + n.(Using for loop)

2. 1 + 3 + 5 + ---------- + n.(Using for loop)

3. 2 + 4 + 6 + ---------- + n.(Using for loop)

4. 1 + 2 + 3 + ---------- + n.(Using while loop)

5. 1 + 3 + 5 + ---------- + n.(Using while loop)

6. 2 + 4 + 6 + ---------- + n.(Using while loop)

7. 1 \* 2 + 2 \* 3 + 3 \* 4 + ---------- - +n1 \* n2.

8. 1 \* 3 + 2 \* 5 + 3 \* 7 + ---------- - +n1 \* n2.

9. 1 \* 3 \* 4 + 2 \* 5 \* 6 + 3 \* 7 \* 8 + ---------- - +n1 \* n2 \* n3.

10. 1 2 3-------- - n.

11. 1 3 5-------- - n.

12. 2 4 6-------- - n.

13. 1.5 + 2.5 + 3.5 + ------------ - +n.

14. 12 + 22 + 32 + ------------ - +n2

15. 13 + 23 + 33 + ------------ - +n3

16. 12 + 32 + 52 + ------------ - +n2

17. 1 + ½ + 1 / 3 + ---------- - +1 / n.

18. 1 × 2 × 3 ×------------ - × n.

19. 12 x 22 x 32 x------------xn2

20. 13 x 23 x 33 x------------xn3

21. 13 x 33 x 53 x------------xn3

22. 22 x 42 x 62 x------------xn2

23. 12 x 32 x 52 x------------xn2

24. 1 - 2 + 3 - 4 + 5 - 6 + ------------ - +n.//(1+3+5+----)-(2+4+6+-----)

25. Fibonacci Series(0 1 1 2 3)

1. 1 + 2 + 3 + ---------- + n.(Using for loop)

#include <stdio.h>

int main()

{

int n, i, sum = 0;

printf("Enter the last number of the series = ");

scanf("%d", &n);

printf("1+2+3+-----------+%d\n", n);

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

{

sum = sum + i;

}

printf("%d\n", sum);

return 0;

}

2. 1 + 3 + 5 + ---------- + n.(Using for loop)

#include <stdio.h>

int main()

{

int n, i, sum = 0;

printf("Enter the last number of the series = ");

scanf("%d", &n);

printf("1+3+5+-----------+%d\n", n);

for (i = 1; i <= n; i = i + 2)

{

sum = sum + i;

}

printf("%d\n", sum);

return 0;

}

3. 2 + 4 + 6 + ---------- + n.(Using for loop)

#include <stdio.h>

int main()

{

int n, i, sum = 0;

printf("Enter the last number of the series = ");

scanf("%d", &n);

printf("2+4+6+-----------+%d\n", n);

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

{

sum = sum + i;

}

printf("%d\n", sum);

return 0;

}

4. 1 + 2 + 3 + ---------- + n.(Using while loop)

#include <stdio.h>

int main()

{

int n, i = 1, sum = 0;

printf("Enter the last number of the series = ");

scanf("%d", &n);

printf("1 + 2 + 3 + --------------+%d\n", n);

while (i <= n)

{

sum = sum + i;

i = i + 1;

}

printf("%d\n", sum);

return 0;

}

5. 1 + 3 + 5 + ---------- + n.(Using while loop)

#include <stdio.h>

int main()

{

int n, i = 1, sum = 0;

printf("Enter the last number of the series = ");

scanf("%d", &n);

printf("1 + 3 + 5 + --------------+%d\n", n);

while (i <= n)

{

sum = sum + i;

i = i + 2;

}

printf("%d\n", sum);

return 0;

}

6. 2 + 4 + 6 + ---------- + n.(Using while loop)

#include <stdio.h>

int main()

{

int n, i = 2, sum = 0;

printf("Enter the last number of the series = ");

scanf("%d", &n);

printf("2 + 4 + 6 + --------------+%d\n", n);

while (i <= n)

{

sum = sum + i;

i = i + 2;

}

printf("%d\n", sum);

return 0;

}

7. 1\*2 + 2\*3 + 3\*4 +-----------+n1\*n2.

#include <stdio.h>

int main()

{

int n1, n2, i, j, sum = 0;

printf("Enter n1 and n2 = ");

scanf("%d %d", &n1, &n2);

printf("1\*2 + 2\*3 + 3\*4 + -------+%d\*%d\n", n1, n2);

for (i = 1, j = 2; i <= n1 && j <= n2; i = i + 1, j = j + 1)

{

sum = sum + i \* j;

}

printf("%d\n", sum);

return 0;

}

8. 1\*3 + 2\*5 + 3\*7 +-----------+n1\*n2.

#include <stdio.h>

int main()

{

int n1, n2, i, j, sum = 0;

printf("Enter n1 and n2 = ");

scanf("%d %d", &n1, &n2);

printf("1\*3 + 2\*5 + 3\*7 + -------+%d\*%d\n", n1, n2);

for (i = 1, j = 3; i <= n1 && j <= n2; i = i + 1, j = j + 2)

{

sum = sum + i \* j;

}

printf("%d\n", sum);

return 0;

}

9. 1\*3\*4 + 2\*5\*6 + 3\*7\*8 +-----------+n1\*n2\*n3.

#include <stdio.h>

int main()

{

int n1, n2, n3, i, j, k, sum = 0;

printf("Enter n1 and n2 = ");

scanf("%d %d %d", &n1, &n2, &n3);

printf("1\*3\*4 + 2\*5\*6 + 3\*7\*8 + -------+%d\*%d\*%d\n", n1, n2, n3);

for (i = 1, j = 3, k = 4; i <= n1 && j <= n2 && k<=n3; i = i + 1, j = j + 2, k = k+2)

{

sum = sum + i \* j \* k;

}

printf("%d\n", sum);

return 0;

}

10. 1 2 3---------n.

#include <stdio.h>

int main()

{

int n, i;

printf("Enter n = ");

scanf("%d", &n);

printf("1 2 3----------%d\n", n);

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

{

printf("%d ", i);

}

return 0;

}

11. 1 3 5---------n.

#include <stdio.h>

int main()

{

int n, i;

printf("Enter n = ");

scanf("%d", &n);

printf("1 3 5----------%d\n", n);

for (i = 1; i <= n; i = i + 2)

{

printf("%d ", i);

}

return 0;

}

12. 2 4 6---------n.

#include <stdio.h>

int main()

{

int n, i;

printf("Enter n = ");

scanf("%d", &n);

printf("2 4 6----------%d\n", n);

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

{

printf("%d ", i);

}

return 0;

}

13. 1.5 + 2.5 + 3.5 +-------------+n.

#include <stdio.h>

int main()

{

float n, i, sum = 0;

printf("Enter n = ");

scanf("%f", &n);

printf("1.5 + 2.5 + 3.5 +-----------+%f\n", n);

for (i = 1.5; i <= n; i = i+1)

{

sum = sum + i;

}

printf("%.2f\n", sum);

return 0;

}

14. 12 + 22 + 32 +-------------+n2

#include <stdio.h>

int main()

{

int n, i, sum = 0;

printf("Enter n = ");

scanf("%d", &n);

printf("1^2 + 2^2 + 3^3 +-------+%d^%d\n", n, n);

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

{

sum = sum + i \* i;

}

printf("%d\n", sum);

return 0;

}

15. 13 + 23 + 33 +-------------+n3

#include <stdio.h>

int main()

{

int n, i, sum = 0;

printf("Enter n = ");

scanf("%d", &n);

printf("1^3 + 2^3 + 3^3 +-------+%d^%d\n", n, n);

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

{

sum = sum + i \* i \* i;

}

printf("%d\n", sum);

return 0;

}

16. 12 + 32 + 52 +-------------+n2

#include <stdio.h>

int main()

{

int n, i, sum = 0;

printf("Enter n = ");

scanf("%d", &n);

printf("1^2 + 3^2 + 5^2 +-------+%d^2\n", n);

for (i = 1; i <= n; i = i + 2)

{

sum = sum + i \* i;

}

printf("%d\n", sum);

return 0;

}

17 . 1 + ½ + 1/3 +-----------+1/n.

#include <stdio.h>

int main()

{

double n, i, sum = 0;

printf("Enter the value of n = ");

scanf("%lf", &n);

printf("1 + 1/2 + 1/3 +-----------+1/%lf\n", n);

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

{

sum = sum + (1 / i);

}

printf("%.2lf\n", sum);

return 0;

}

18. 1 × 2 × 3 ×-------------× n.

#include <stdio.h>

int main()

{

int n, i, result = 1;

printf("Enter the value of n = ");

scanf("%d", &n);

printf("1 X 2 X 3 X------------X %d\n", n);

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

{

result = result \* i;

}

printf("%d\n", result);

return 0;

}

19. 12 x 22 x 32 x------------xn2

#include <stdio.h>

int main()

{

int n, i, result = 1;

printf("Enter the value of n = ");

scanf("%d", &n);

printf("1^2 X 2^2 X 3^2 X-----------X%d^2\n", n);

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

{

result = result \* i \* i;

}

printf("%d\n", result);

return 0;

}

20. 13 x 23 x 33 x------------xn3

#include <stdio.h>

int main()

{

int n, i, result = 1;

printf("Enter the value of n = ");

scanf("%d", &n);

printf("1^3 X 2^3 X 3^3 X-----------X%d^3\n", n);

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

{

result = result \* i \* i \* i;

}

printf("%d\n", result);

return 0;

}

21. 13 x 33 x 53 x------------xn3

#include <stdio.h>

int main()

{

int n, i, result = 1;

printf("Enter the value of n = ");

scanf("%d", &n);

printf("1^3 X 3^3 X 5^3 X-----------X%d^3\n", n);

for (i = 1; i <= n; i = i + 2)

{

result = result \* i \* i \* i;

}

printf("%d\n", result);

return 0;

}

22. 22 x 42 x 62 x------------xn2

#include <stdio.h>

int main()

{

int n, i, result = 1;

printf("Enter the value of n = ");

scanf("%d", &n);

printf("2^2 X 4^2 X 6^2 X-----------X%d^2\n", n);

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

{

result = result \* i \* i;

}

printf("%d\n", result);

return 0;

}

23. 12 x 32 x 52 x------------xn2

#include <stdio.h>

int main()

{

int n, i, result = 1;

printf("Enter the value of n = ");

scanf("%d", &n);

printf("1^2 X 3^2 X 5^2 X-----------X%d^2\n", n);

for (i = 1; i <= n; i = i + 2)

{

result = result \* i \* i;

}

printf("%d\n", result);

return 0;

}

24. 1-2+3-4+5-6+-------------+n.//(1+3+5+----)-(2+4+6+-----)

#include <stdio.h>

int main()

{

int n, i, even = 0, odd = 0;

printf("Enter the value of n = ");

scanf("%d", &n);

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

{

if (i % 2 == 0)

{

even = even + i;

}

else

{

odd = odd + i;

}

}

printf("Sum is = %d\n", odd - even);

return 0;

}

25. Fibonacci Series (0 1 1 2 3 )

#include <stdio.h>

int main()

{

int first = 0, second = 1, fibo, count = 0, n;

printf("Enter range = ");

scanf("%d", &n);

while (n > count)

{

if (count <= 1)

{

fibo = count;

}

else

{

fibo = first + second;

first = second;

second = fibo;

}

printf("%d ", fibo);

count++;

}

return 0;

}

Function(ফাংশন)

1(1).Function এর মাধ্যমে যোগ।

1(2).Function এর সাহায্যে যোগ।

2. Function এর মাধ্যমে একটি পূর্নসংখ্যার বর্গ।

3. Function এর মাধ্যমে একটি ত্রিভুজ এর ক্ষেত্রফল নির্নয়।

4(1) - x to the power y using define function.

4(2) - x to the power y using define function.

5(1) - x to the power y using library function.

5(2) - x to the power y without library function.

6(1) - Passing array without using function.

6(2) - Passing array using function.

7. Finding maximum value from an array using function.

8. Passing string to function.

9. Factorial using recursion.

10. Add two number using function.

11. Some function for understanding the logic.

12. Check a global variable output using function.

13. Check the value after passing a array to function.

1(1). Function এর মাধ্যমে যোগ।

Output:

Enter two numbers = 10 5

The sum is = 15

#include <stdio.h>

int main()

{

int sum(int num1, int num2);

int num1, num2;

printf("Enter two numbers = ");

scanf("%d %d", &num1, &num2);

//int result = sum(num1, num2);

printf("The sum is = %d\n", sum(num1, num2)); //result

}

int sum(int num1, int num2)

{

return num1 + num2;

}

1(2). Function এর সাহায্যে যোগ।

Output:

The sum is = 18

The sum is = 60

The sum is = 10

#include <stdio.h>

int main()

{

int sum(int a, int b, int c);

int sub(int a, int b);

printf("The sum is = %d\n", sum(5, 6, 7));

printf("The sum is = %d\n", sum(10, 20, 30));

printf("The sum is = %d\n", sub(20, 10));

}

int sum(int a, int b, int c)

{

return a + b + c;

}

int sub(int a, int b)

{

return a - b;

}

2. Function এর মাধ্যমে একটি পূর্নসংখ্যার বর্গ।

Output:

Enter a number = 5

The square is = 25

#include <stdio.h>

int main()

{

int square(int num);

int num;

printf("Enter a number = ");

scanf("%d", &num);

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

}

int square(int num)

{

return num \* num;

}

3. Function এর মাধ্যমে একটি ত্রিভুজ এর ক্ষেত্রফল নির্নয়।

#include <stdio.h>

int main()

{

float trianglearea(float base, float height);

float base, height;

printf("Enter base and height = ");

scanf("%f %f", &base, &height);

printf("The area is = %.2f\n", trianglearea(base, height));

}

float trianglearea(float base, float height)

{

return 0.5 \* base \* height;

}

Output:

Enter baseand height = 7 5

The area is = 17.50

4(1)- x to the power y using define function.

Output:

Enter baseand exponent = 2 3

The result is = 8.00

#include <stdio.h>

int main()

{

double calculatepower(double base, double exponent);

double base, exponent;

printf("Enter base and exponent = ");

scanf("%lf %lf", &base, &exponent);

printf("The result is = %.2lf\n", calculatepower(base,exponent));

}

double calculatepower(double base, double exponent)

{

double i, result = 1;

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

{

result = result \* base;

}

return result;

}

4(2)- x to the power y using define function.

Output:

The result is = 8.00

The result is = 16.00

The result is = 25.00

#include <stdio.h>

int main()

{

double calculatepower(double base, double exponent);

calculatepower(2, 3);

calculatepower(4, 2);

calculatepower(5, 2);

}

double calculatepower(double base, double exponent)

{

double i, result = 1;

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

{

result = result \* base;

}

printf("The result is = %.2lf\n", result);

}

5(1). x to the power y using library function.

Output:

Enter baseand exponent = 5 2

The result is = 25.00

#include <stdio.h>

int main()

{

double base, exponent, result = 1;

int i;

printf("Enter base and exponent = ");

scanf("%lf %lf", &base, &exponent);

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

{

result = result \* base;

}

printf("The result is = %.2lf\n", result);

}

5(2). x to the power y without library function.

Output:

Enter baseand exponent = 5 2

The result is = 25.00

#include <stdio.h>

#include <math.h>

int main()

{

double base, exponent;

printf("Enter base and exponent = ");

scanf("%lf %lf", &base, &exponent);

double result = pow(base, exponent);

printf("The result is = %.2lf\n", result);

}

6(1) - Passing array without using function.

#include <stdio.h>

int main()

{

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

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

{

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

}

}

Output:

10 20 30 40 50

6(2)- Passing array using function.

#include <stdio.h>

int main()

{

void display(int num[]);

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

display(num);

}

void display(int num[])

{

int i;

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

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

}

}

Output:

10 20 30 40 50

7. Finding maximum value from an array using function.

Output:

Maximum = 50

#include <stdio.h>

int main()

{

int maximum(int num[]);

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

printf("Maximum = %d\n", maximum(num));

}

int maximum(int num[])

{

int i;

int max = num[0];

for (i = 1; i < 5; i++)

{

if (num[i] > max)

{

max = num[i];

}

}

return max;

}

#include <stdio.h>

int main()

{

int n, i, num[100];

printf("How many numbers = ");

scanf("%d", &n);

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];

}

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

}

Output:

How many numbers = 5

12 23 34 45 56

The maximum number is = 56

#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"

"and the position is %d\n", max, position);

}

Output:

The maximum number is = 9

and the position is 6

8. Passing string to function.

Output:

k

i

b

r

i

a

#include <stdio.h>

int main()

{

void show(char ch[]);

char ch[] = "kibria";

show(ch);

}

void show(char ch[])

{

int i = 0;

while (ch[i] != '\0')

{

printf("%c\n", ch[i]);

i++;

}

}

9. Factorial using recursion.

Output:

Enter the number = 4

Factorial is = 24

#include <stdio.h>

int main()

{

int fact(int n);

int n;

printf("Enter the number = ");

scanf("%d", &n);

printf("Factorial is = %d\n", fact(n));

}

int fact(int n)

{

if (n == 1) return 1;

else return n \* fact(n - 1);

}

10. Add two number using function.

Output:

The sum is = 5.50

#include <stdio.h>

int main()

{

double add(double a, double b);

double a = 2.8, b = 2.7, c;

c = add(a, b);

printf("The sum is = %.2lf\n", c);

}

double add(double a, double b)

{

double sum = a + b;

return sum;

}

11. Some function for understanding the logic.

Output:

10 20 200

#include <stdio.h>

int main()

{

int testfunction(int x);

int x = 10, y = 20, z = 30;

z = testfunction(x);

printf("%d %d %d\n", x, y, z);

}

int testfunction(int x)

{

int y = x;

x = 2 \* y;

return (x \* y);

}

12. Check a global variable output using function.

Output:

3.140000

3.141600

#include <stdio.h>

double pi = 3.14;

int main()

{

void myfunction();

printf("%lf\n", pi);

myfunction();

printf("%lf\n", pi);

}

void myfunction()

{

pi = 3.1416;

}

13. Check the value after passing a array to function.

Output:

1

100

#include <stdio.h>

int main()

{

void testfunction(int num[]);

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

printf("%d\n", num[0]);

testfunction(num);

printf("%d\n", num[0]);

}

void testfunction(int num[])

{

num[0] = 100;

}

/\*ফাংশনের মধ্যে কোনো অ্যারে পাস করালে ওই অ্যারের আলাদা কোনো কপি তৈরি হয় না।\*/

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];

}

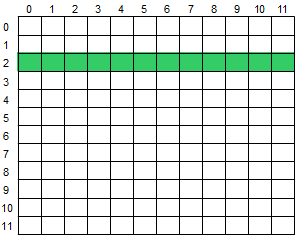
}

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);

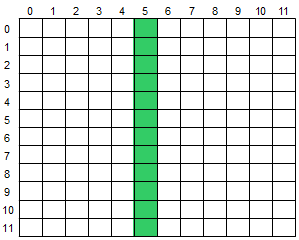
return 0;

}

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);

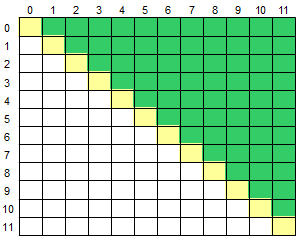
else if (ch[0] == 'M')

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);

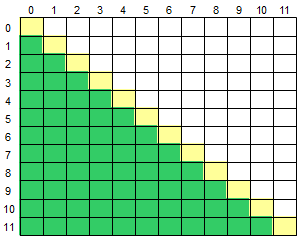
else

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);

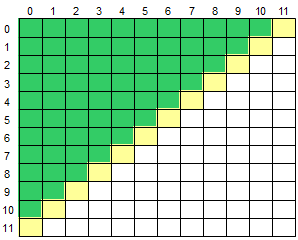
else

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);

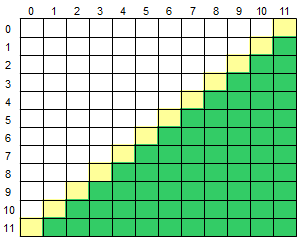
else

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);

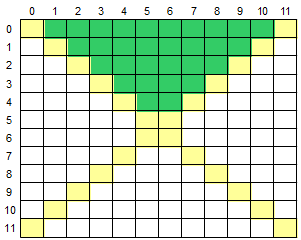
else

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

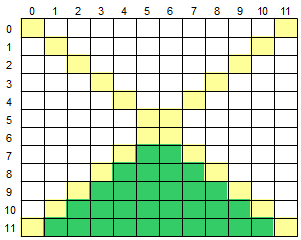
printf("%.1lf\n", sum / 30.0);

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

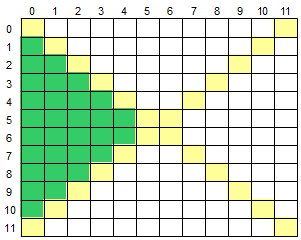
printf("%.1lf\n", sum / 30.0);

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

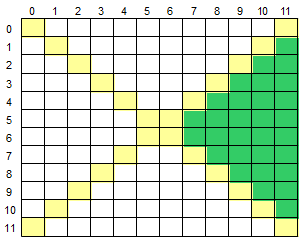
printf("%.1lf\n", sum / 30.0);

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);

String(স্টিং)

1(1) - Print using string.(স্ট্রিং এর মাধ্যমে প্রিন্ট করা)

1(2) – Print string from the user.

2(1) - Display string character wise

2(2) - Display string character wise

3(1) - Find a string length using strlen() function.

3(2) - Find a string length without using strlen() function.

4(1) - Copy a string using strcpy() function.

4(2) - Copy a string using strcpy() function from the user.

5(1) - Concat string using strcat() function.

5(2) - Concat string using strcat() function.

5(3) - Concat string without strcat() function.

6. Comparing a string using strcmp() function.

7(1).Reverse a string using strrev() function.

7(2).Reverse a string using strrev() function form the user.

7(3).Reverse a string without strrev() function.

8. String swapping(এক স্ট্রিং এর উপাদান অন্য স্ট্রিং এ কপি করা)

9. Checking a string palindrome or not.

10(1) – strupr()

10(2) – strlwr()

11(1).Determine number of vowels, cosonant, word, digitand others.

11(2).Determine number of capital and small letter

12.আউটপুট হিসাবে স্ট্রিং এর প্রতিটি শব্দ আলাদা লাইন এ প্রিন্ট হবে। বিরামচিহ্ন গুলো প্রিন্ট হবে না এবং শব্দের প্রথম অক্ষর হবে বড় হাতের।

13(1).Binary to Decimal.

13(2).Binary to Decimal from the user.

14(1).Decimal to Binary.

14(2).Decimal to Binary from the user.

15(1). Lower to Upper.

15(2). Lower to Upper from the user.

16. Concate a string without using concate function.

17. C program to encryptand decrypt a string(এনক্রিপশন এন্ড ডিক্রিপশন)

1(1) - Print using string.(স্ট্রিং এর মাধ্যমে প্রিন্ট করা)

Output:

Golam Kibria

#include <stdio.h>

int main()

{

char ch[] = "Golam Kibria";

/\*char ch[] = "Golam \

kibria"; \*/

printf("%s\n", ch);

}

1(2) – Print string from the user.

Output:

Enter your full name Golam kibria

Full name is = Golam kibria

#include <stdio.h>

int main()

{

char ch[20];

printf("Enter your full name ");

gets(ch);

printf("Full name is = %s\n", ch);

}

2(1) - Display string character wise

Output:

k

i

b

r

i

a

#include <stdio.h>

int main()

{

char ch[] = "kibria";

int i = 0;

while (ch[i] != '\0')

{

printf("%c\n", ch[i]);

i++;

}

}

2(2) - Display string character wise

Output:

k

i

b

r

i

a

#include <stdio.h>

int main()

{

char ch[] = "kibria";

int length, i;

length = strlen(ch);

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

{

printf("%c\n", ch[i]);

}

}

3(1)- Find a string length using strlen() function.

Output:

Length is = 6

#include <stdio.h>

int main()

{

char ch[10] = "kibria";

//scanf("%s", &ch);

//printf("The string is = %s\n", ch);

int length = strlen(ch);

printf("Length is = %d\n", length);

}

3(2)- Find a string length without using strlen() function.

Output:

Length is = 6

#include <stdio.h>

int main()

{

char ch[] = "kibria";

int i = 0, length = 0;

while (ch[i] != '\0')

{

i++;

length++;

}

printf("Length is = %d\n", length);

(০ ততম ইনডেক্স মানে হলো ১ তম লেন্থ)

}

4(1)- Copy a string using strcpy() function.

Output:

Main string is = kibria

Copy string is = kibria

#include <stdio.h>

int main()

{

char ch1[20] = "kibria";

char ch2[20];

strcpy(ch2, ch1);

printf("Main string is = %s\n", ch1);

printf("Copy string is = %s\n", ch2);

}

4(2)- Copy a string using strcpy() function from the user.

Output:

Main string is = Golam kibria

Copy string is = Golam kibria

#include <stdio.h>

int main()

{

char ch1[20];

char ch2[20];

printf("Main string is = ");

//scanf("%s", &ch1);

gets(ch1);

strcpy(ch2, ch1);

printf("Copy string is = %s\n", ch2);

}

5(1) - Concat string using strcat() function.

Output:

Character is = My name is Golam kibria

#include <stdio.h>

int main()

{

char ch1[20] = "My name is ";

char ch2[20] = "Golam kibria";

strcat(ch1, ch2);

printf("Character is = %s\n", ch1);

}

5(2) - Concat string using strcat() function.

Output:

Character is = My name is Golam kibria

#include <stdio.h>

int main()

{

char ch1[20] = "My name is ";

strcat(ch1, "Golam kibria");

printf("Character is = %s\n", ch1);

}

5(3)- Concat string without strcat() function.

Output:

Character is = Golam kibria

#include <stdio.h>

int main()

{

char ch1[50] = "Golam ";

char ch2[] = "kibria";

int i = 0, length = 0, j = 0;

while (ch1[i] != '\0')

{

i++;

length++;

}

while (ch2[j] != '\0')

{

ch1[length + j] = ch2[j];

j++;

}

//ch1 এর ৫তম ইনডেক্সে বসবে ch2 এর ০তম ইনডেক্স এর মান

printf("Character is = %s\n", ch1);

}

6. Comparing a string using strcmp() function.

Output:

kibria

kibria

ch1 = kibria

ch2 = kibria

String are equal

#include <stdio.h>

int main()

{

char ch1[10]; // = "kibria";

char ch2[10]; // = "kibria";

scanf("%s %s", &ch1, &ch2);

printf("ch1 = %s\nch2 = %s\n", ch1, ch2);

int d = strcmp(ch1, ch2);

if (d == 0)

printf("String are equal\n");

else

printf("Strings are not equal\n");

}

7(1). Reverse a string using strrev() function.

Output:

Character is = kibria

Reverse is = airbik

#include <stdio.h>

int main()

{

char ch[] = "kibria";

printf("Character is = %s\n", ch);

strrev(ch);

printf("Reverse is = %s\n", ch);

}

7(2). Reverse a string using strrev() function form the user.

Output:

Character is = golam kibria

Reverse is = airbik malog

#include <stdio.h>

int main()

{

char ch[10];

printf("Character is = ");

gets(ch);

strrev(ch);

printf("Reverse is = %s\n", ch);

}

7(3). Reverse a string without strrev() function.

Output:

Character is = kibria

Reverse is = airbik

#include <stdio.h>

int main()

{

char ch1[20] = "kibria";

char ch2[20];

int i = 0, length = 0, j;

//i and length for ch1 , j for ch2

while (ch1[i] != '\0')

{

i++;

length++;

}

for (j = 0, i = length - 1; i >= 0; i--, j++)

{

ch2[j] = ch1[i];

}

ch2[j] = '\0';

printf("Character is = %s\n", ch1);

printf("Reverse is = %s\n", ch2);

}

8. String swapping(এক স্ট্রিং এর উপাদান অন্য স্ট্রিং এ কপি করা)

Output:

Before swapping :

ch1 = Bangladesh

ch2 = Canada

After swapping :

ch1 = Canada

ch2 = Bangladesh

#include <stdio.h>

int main()

{

char ch1[20] = "Bangladesh";

char ch2[20] = "Canada";

char temp[20];

printf("Before swapping : \n");

printf("ch1 = %s\n", ch1);

printf("ch2 = %s\n", ch2);

strcpy(temp, ch1);

strcpy(ch1, ch2);

strcpy(ch2, temp);

printf("After swapping : \n");

printf("ch1 = %s\n", ch1);

printf("ch2 = %s\n", ch2);

}

9. Checking a string palindrome or not.

#include <stdio.h>

int main()

{

char ch1[20] = "kibria";

char ch2[20];

int i = 0, length = 0, j;

//i and length for ch1 , j for ch2

//printf("Enter a string = ");

//scanf("%s", &ch1);

while (ch1[i] != '\0')

{

i++;

length++;

}

for (j = 0, i = length - 1; i >= 0; i--, j++)

{

ch2[j] = ch1[i];

}

ch2[j] = '\0';

printf("Entered string is = %s\n", ch1);

printf("Reverse string is = %s\n", ch2);

int d = strcmp(ch1, ch2);

//সমান হলে d=0 হবে। কারন মাইনাস করলে 0 ই হবে

if (d == 0)

printf("String is palindrome\n");

else

printf("String is not palindrome\n");

}

Output:

Entered string is = kibria

Reverse string is = airbik

String is not palindrome

10(1) – strupr()

#include <stdio.h>

int main()

{

char ch[] = "kibria";

strupr(ch);

printf("Character is = %s\n", ch);

}

Output:

Character is = KIBRIA

#include <stdio.h>

int main()

{

char ch[10];

gets(ch);

strupr(ch);

printf("Character is = %s\n", ch);

}

Output:

kibria go

Character is = KIBRIA GO

10(2) – strlwr()

#include <stdio.h>

int main()

{

char ch[] = "KIBRIA";

strlwr(ch);

printf("Character is = %s\n", ch);

}

Output:

Character is = kibria

11(1). Determine number of vowels, cosonant, word, digit and others.

#include <stdio.h>

int main()

{

char str[100];

int i, vowel, consonant, digit, word, other;

i = vowel = consonant = digit = word = other = 0;

printf("Enter a string = ");

gets(str);

while ((str[i]) != '\0')

{

if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u' ||

str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' || str[i] == 'U')

vowel++;

else if ((str[i] >= 'a' && str[i] <= 'z') || (str[i] >= 'A' && str[i] <= 'Z'))

consonant++;

else if (str[i] >= '0' && str[i] <= '9')

digit++;

else if (ch == ' ')

word++;

else

other++;

i++;

}

word++;

/\*space এর আগে 1 টা word অবশ্যই থাকবে,

তাই সেই word টাকে এখানে increment করে দিলাম।\*/

printf("Number of vowels = %d\n", vowel);

printf("Number of consonants = %d\n", consonant);

printf("Number of words = %d\n", word);

printf("Number of digits = %d\n", digit);

printf("Number of others = %d\n", other);

}

Output:

Enter a string = Golam, Kibria, 34#ezaz

Number of vowels = 7

Number of consonants = 8

Number of words = 3

Number of digits = 2

Number of others = 3

11(2). Determine number of capital and small letter.

#include <stdio.h>

int main()

{

char str[50];

int i, capital, small, digit;

i = capital = small = digit = 0;

printf("Enter a string = ");

gets(str);

while (str[i] != '\0')

{

if (str[i] >= 65 && str[i] <= 90)

capital++;

else if (str[i] >= 97 && str[i] <= 122)

small++;

else if (str[i] >= 48 && str[i] <= 57)

digit++;

i++;

}

printf("Number of capital letter = %d\n", capital);

printf("Number of small letter = %d\n", small);

printf("Number of digits letter = %d\n", digit);

}

Output:

Enter a string = Golam KIbria 34

Number of capital letter = 3

Number of small letter = 8

Number of digits letter = 2

12.আউটপুট হিসাবে স্ট্রিং এর প্রতিটি শব্দ আলাদা লাইন এ প্রিন্ট হবে। বিরামচিহ্ন গুলো প্রিন্ট হবে না এবং শব্দের প্রথম অক্ষর হবে বড় হাতের।

Output:

golam kibria ezaz

Golam

Kibria

Ezaz

#include <stdio.h>

#include <string.h>

int main()

{

char s[100], ch;

int i, length, wordstarted = 0;

gets(s);

length = strlen(s);

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

{

if (s[i] >= 'a' && s[i] <= 'z')

{

if (wordstarted == 0)

{

wordstarted = 1;

ch = 'A' + s[i] - 'a';

printf("%c", ch);

}

else

{

printf("%c", s[i]);

}

}

else if ((s[i] >= 'A' && s[i] <= 'Z') || (s[i] >= '0' && s[i] <= '9'))

{

if (wordstarted == 0)

{

wordstarted = 1;

}

printf("%c", s[i]);

}

else

{

if (wordstarted == 1)

{

wordstarted = 0;

printf("\n");

}

}

}

return 0;

}

13(1). Binary to Decimal.

Output:

Decimal value is = 22

#include <stdio.h>

#include <math.h>

#include <string.h>

int main()

{

char binary[] = "10110";

int length = 5;

int position = 4;

int decimal = 0;

int i;

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

{

decimal = decimal + (binary[i] - '0') \* pow(2, position);

position--;

}

printf("Decimal value is = %d\n", decimal);

}

13(2). Binary to Decimal from the user.

Output:

Enter the binary number = 111

Decimal value is = 7

#include <stdio.h>

#include <math.h>

#include <string.h>

int main()

{

char binary[65];

int length;

int position;

int decimal = 0;

int i;

printf("Enter the binary number = ");

scanf("%s", &binary);

length = strlen(binary);

position = length - 1;

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

{

decimal = decimal + (binary[i] - '0') \* pow(2, position);

position--;

}

printf("Decimal value is = %d\n", decimal);

}

14(1). Decimal to Binary.

Output:

The binary number is = 10110

#include <stdio.h>

int main()

{

int decimalnumber = 22;

int binarynumber = 0;

int rem, temp = 1;

while (decimalnumber != 0)

{

rem = decimalnumber % 2;

decimalnumber = decimalnumber / 2;

binarynumber = binarynumber + rem \* temp;

temp = temp \* 10;

}

printf("The binary number is = %d\n", binarynumber);

}

14(2). Decimal to Binary from the user.

Output:

Enter any decimal number = 12

The binary number is = 1100

#include <stdio.h>

int main()

{

int decimalnumber;

int binarynumber = 0;

int rem, temp = 1;

printf("Enter any decimal number = ");

scanf("%d", &decimalnumber);

while (decimalnumber != 0)

{

rem = decimalnumber % 2;

decimalnumber = decimalnumber / 2;

binarynumber = binarynumber + rem \* temp;

temp = temp \* 10;

}

printf("The binary number is = %d\n", binarynumber);

}

15(1). Lower to Upper.

Output:

Bangladesh

BANGLADESH

#include <stdio.h>

int main()

{

char country[] = "Bangladesh";

int i, length = 10;

printf("%s\n", country);

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

{

if (country[i] >= 'a' && country[i] <= 'z')

{

country[i] = 'A' + (country[i] - 'a');

}

}

printf("%s\n", country);

}

15(2). Lower to Upper from the user.

Output:

bangladesh

BANGLADESH

#include <stdio.h>

int main()

{

char country[30];

int i;

gets(country);

int length = strlen(country);

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

{

if (country[i] >= 'a' && country[i] <= 'z')

{

country[i] = 'A' + (country[i] - 'a');

}

}

printf("%s\n", country);

//puts(country);

}

16. Concate a string without using concate function.

Output:

Bangladesh

#include <stdio.h>

int main()

{

char s1[] = "Bangla", s2[] = "desh", s3[12];

int i, j, length1 = 6, length2 = 4;

//i for s1 and j for s3.

for (i = 0, j = 0; i < length1; i++, j++)

{

s3[j] = s1[i];

}

for (i = 0; i < length2; i++, j++)

{

s3[j] = s2[i];

//s3 er soptom upadan hoby s2 er protom upadan.

}

s3[j] = '\0';

printf("%s\n", s3);

}

17. C program to encryptand decrypt a string(এনক্রিপশন এন্ড ডিক্রিপশন)

#include <stdio.h>

int main()

{

int i, x, n;

char str[100];

printf("\nPlease enter a string:\n");

gets(str);

n = strlen(str);

printf("\nPlease choose following options:\n");

printf("1 = Encrypt the string.\n");

printf("2 = Decrypt the string.\n");

scanf("%d", &x);

if (x == 1)

{

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

{

str[i] = str[i] + 1;

//the key for encryption is 1 that is added to ASCII value

}

printf("\nEncrypted string: %s\n", str);

}

else if (x == 2)

{

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

{

str[i] = str[i] - 1;

//the key for encryption is 1 that is subtracted to ASCII value

}

printf("\nDecrypted string: %s\n", str);

}

else

printf("\nError\n");

}

Output:

Please enter a string :

golam kibria

Please choose following options :

1 = Encrypt the string.

2 = Decrypt the string.

1

Encrypted string : hpmbn!ljcsjb

Pointer(পয়েন্টার)

1. Basic problem.

2. Finding Value and Address.

3. Finding Value.

4. Character type pointer.

5. Value of a character.

6. Using malloc function in pointer

7. Using array in pointer.

8. Void pointer.

9. Function pointer.

10. Some basic problem to better understating in pointer.

11. Pointer to a pointer.

12. Swap two number using pointer.

13. Write a program to extract a substring from a string.

পয়েন্টার হচ্ছে একটি বিশেষ ধরনের ভেরিয়েবল যেটি আরেকটি ভেরিয়েবল এর ঠিকানা রাখতে পারে।

* যে পয়েন্টার সে একটি ভেরিয়েবল এর ঠিকানা রাখবে এবং তার আগে \* বসিয়ে দিলে সেই ভেরিয়েবল এর মানও পাওয়া যাবে।
* অর্থাৎ p তে যদি আমি a নামক ভেরিয়েবল রাখি তাহলে a এর মান a ব্যবহার করেও পাওয়া যাবে আবার \*p ব্যবহার করেও পাওয়া যাবে।

ইন্টিজার এর জন্য ইন্টিজার পয়েন্টার। ডাবল এর জন্য ডাবল, ফ্লোট এর জন্য ফ্লোট, ক্যারেক্টার এর জন্য ক্যারেক্টার টাইপের পয়েন্টার। অর্থাৎ ভেরিয়েবল যে টাইপের হবে পয়েন্টার ও সেই টাইপের হবে।

* int \*p says interger pointer p
* \*p says content of p.
* called dereferencing operator

1.1 Basic problem. (2.1)

#include <stdio.h>

int main()

{

int a = 10;

printf("Value of a : %d\n", a);

printf("Address of a : %p\n", &a);

}

Output:

Value of a : 10

Address of a : 00D7FBA8

/\*Without & address of a : 0000000A\*/

1.2 Basic problem (2.2)

#include <stdio.h>

int main()

{

int a = 10;

int \*p; /\*ইন্টিজার টাইপরে পয়েন্টার ডিক্লেয়ার করলাম\*/

p = &a; /\*a নামক ইন্টিজার ভেরিয়েবল এর ঠিকানা p তে রাখলাম\*/

/\*অর্থাৎ p এর মান প্রিন্ট করলে a এর ঠিকান পাবো আবার a এর মান প্রিন্ট করার জন্য \*p বা a প্রিন্ট করবো\*/

printf("a = %d\n", \*p);

printf("Value of p is %p\n", p);

}

Output:

a = 10

Value of p is 0073F844

2.1 Finding Value and Address. (2.4)

Output:

Value of a : 10

Value of a = 20

Address of a = 00EFFD4C

Value of p = 00EFFD4C\*

#include <stdio.h>

int main()

{

int a = 10;

int \*p;

p = &a; /\*p পয়েন্টারে a এর ঠিকানা রেখেছি বা p এর মান হবে a এর ঠিকানা\*/

printf("Value of a : %d\n", a);//a or \*p

/\*p পয়েন্টার যে ঠিকানায় আছে সেই ঠিকানায় ২০ অ্যাসাইন করে দিলাম। তাই a = 20

অর্থাৎ আমি যদি কোনো ভেরিয়েবল হই তাহলে আমার ঠিকানা হচ্ছে আমার রোল নাম্বার\*/

\*p = 20;

printf("Value of a = %d\n", a); /\*a or \*p\*/

printf("Address of a = %p\n", &a);

printf("Value of p = %p\n", p); /\*a এর ঠিকানায় p এর ভেলু রেখে দিলাম\*/

}

2.2 Finding Value and Address (2.5)

Output:

Value of a : 10

Value of a : 20

Value of a : 15

Value store at location 00F9FBD4 is 15

Address of a : 00F9FBD4

Value of p : 00F9FBD4\*

#include <stdio.h>

int main()

{

int a = 10;

int \*p;

printf("Value of a : %d\n", a);

p = &a;

\*p = 20;

printf("Value of a : %d\n", a);

a = 15;

printf("Value of a : %d\n", a);

printf("Value store at location %p is %d\n", p, \*p); //\*p or a

printf("Address of a : %p\n", &a);

printf("Value of p : %p\n", p);

}

2.3 Finding Value and Address (2.6)

Output:

Value of a : 10

Value of a : 15

Value of b : 10

Value of \* p : 15

Address of x : 15726524

Address of y : 15726512

Value of p : 15726524\*

#include <stdio.h>

int main()

{

int a = 10;

int b;

int \*p;

printf("Value of a : %d\n", a);

p = &a; /\*p এর মান হবে a এর ঠিকানা। অর্থাৎ a এর যা মান হবে p এরও একই মান হবে\*/

b = \*p;

\*p = 15;

printf("Value of a : %d\n", a);

printf("Value of b : %d\n", b);

printf("Value of \*p : %d\n", \*p);

printf("Address of x : %d\n", &a);

printf("Address of y : %d\n", &b);

printf("Value of p : %d\n", p);

}

2.4 Finding Value and Address. (2.7)

Output:

Value of a : 15

Value of b : 10

Value of \*p : 15

Value of \*q : 10

#include <stdio.h>

int main()

{

int a = 10, b;

int \*p, \*q;

p = &a; /\*p এর মান হবে a এর ঠিকানা। অর্থাৎ a এর যা মান হবে p এরও একই মান হবে\*/

b = \*p;

\*p = 15;

q = &b;

printf("Value of a : %d\n", a); //a or \*p

printf("Value of b : %d\n", b);

printf("Value of \*p : %d\n", \*p);

printf("Value of \*q : %d\n", \*q);

}

3. Finding Value. (2.8)

Output:

Value of a : 15

Value of b : 20

Value of \*p : 15

Value of \*q : 20

#include <stdio.h>

int main()

{

int a = 10, b;

int \*p, \*q;

p = &a;

q = &b;

b = \*p;

\*p = 15;

\*q = 20;

printf("Value of a : %d\n", a);

printf("Value of b : %d\n", b);

printf("Value of \*p : %d\n", \*p);

printf("Value of \*q : %d\n", \*q);

}

3. Finding Value (2.10)

Output:

Value of a : 100

Value of \*p : 100

#include <stdio.h>

int main()

{

int a = 100;

int \*p = "NULL";

printf("Value of a : %d\n", a);

p = &a;

printf("Value of \*p : %d\n", \*p); //\*p or a

}

4.1 Character type pointer. (2.12)

Output:

Name of our country : Bangladesh

Address of str : 00EFFA1C

#include <stdio.h>

int main()

{

char str[] = "Bangladesh";

/\*স্টিং হচ্ছে ক্যারেক্টারের অ্যারে আর অ্যারের নামটিই তার অ্যাড্রেস বা ঠিকানা নির্দেশ করে, তাই সেটিও প্রিন্ট করেছি।\*/

printf("Name of our country : %s\n", str);

printf("Address of str : %p\n", str);

}

4.2 Character type pointer. (2.13)

Output:

Name of our country : Bangladesh

#include <stdio.h>

int main()

{

char str[] = "Bangladesh";

char \*p;

//or, char \*p = "Bangladesh;

p = str;

printf("Name of our country : %s\n", p);

}

4.3 Character type pointer. (2.14)

Output: A B C

#include <stdio.h>

int main()

{

char c1 = 'A', c2 = 'B', c3 = 'C';

char \*p1, \*p2, \*p3;

p1 = &c1; p2 = &c2; p3 = &c3;

printf("%c %c %c\n", \*p1, \*p2, \*p3);

}

5.1 Value of a character. (2.16)

Output:

Value of c : A

Value of c : A

Value of c : A

#include <stdio.h>

int main()

{

char c = 'A';

char \*p, \*\*q;

p = &c;

q = &p;

printf("Value of c : %c\n", c);

printf("Value of c : %c\n", \*p);

printf("Value of c : %c\n", \*\*q);

}

5.2 Value of a character. (2.17)

Output:

Value of c : B

Value of c : B

Value of c : B

#include <stdio.h>

int main()

{

char c = 'A';

char \*p, \*\*q;

p = &c;

q = &p;

\*\*q = 'B';

/\*c এর অ্যাড্রেস p এর মধ্যে আবার p এর অ্যাড্রেস q এর মধ্যে\*/

printf("Value of c : %c\n", c);

printf("Value of c : %c\n", \*p);

printf("Value of c : %c\n", \*\*q);

}

6.1 Using malloc function in pointer. (7.1)

#include <stdio.h>

#include <stdlib.h>

int main()

{

/\*এখানে অ্যারে ব্যবহার না করে ইন্টিজার পয়েন্টার ব্যবহার করেছি। কারন ইন্টিজার পয়েন্টার কেবল ইন্টিজারকের পয়েন্ট করতে পারে এমন নয় এটি ইন্টিজার অ্যারেকেও পয়েন্ট করতে পারে।\*/

int \*marks;

int i, n;

printf("Enter the number of sutudents : ");

scanf("%d", &n);

/\*প্রতিটি নম্বর হচ্ছে ইন্টিজার আর মোট ইন্টজার সংখ্যা n তাই আমাদের sizeof(int) x n বাইট জায়গা প্রয়োজন।\*/

marks = (int \*)malloc(sizeof(int) \* n);

/\*ঠিকঠাক মেমোরি বরাদ্দ করতে পারলে malloc ফাংশনটি একটি পয়েন্টার রির্টান করবে, যে মেমোরি বরাদ্দ করা হয়েছে তার শুরুর ঠিকানা। আর কোনো সমস্যা হলে NULL রির্টান করবে। (int \*) লিখলাম কারন \*marks হচ্ছে ইন্টজার পয়েন্টার।\*/

printf("Enter the marks for each students : \n");

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

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

}

printf("The marks is : \n");

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

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

}

}

Output:

Enter the number of sutudents : 5

Enter the marks for each students :

67 78 89 87 70

The marks is :

67 78 89 87 70

ডায়নামিক মেমোরি অ্যালোকেশন যা ব্যবহার করলে আমাদের আগে থেকে মেমোরির সাইজ নির্ধারন করতে হবে না। প্রোগ্রামটি চলার সময়ই আমরা দরকার মতো মেমোরি নিতে পারবো। এজন্য পয়েন্টার ব্যবহার করে আমাদের মেমোরি বরাদ্দ বা অ্যালোকেশন করেত হবে।

6.2 Using malloc function In pointer. (7.2)

/\*আগের প্রোগ্রমটিকেই আরেকটু সাজিয়ে লিখলাম\*/

#include <stdio.h>

#include <stdlib.h>

int main()

{

int \*marks;

int i, n;

printf("Enter the number of students : ");

scanf("%d", &n);

marks = (int \*)malloc(sizeof(int) \* n);

if (marks == NULL) {

printf("Memory allocation failed for marks\n");

return 1;

}

printf("Enter the marks of the students : \n");

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

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

}

printf("The marks is : \n");

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

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

}

/\*Now free the memory\*/

free(marks);

}

Output:

Enter the number of students : 5

Enter the marks of the students :

55 66 77 76 89

The marks is :

55 66 77 76 89

malloc এর মতো আরেকটি লাইব্রেরি ফাংশন হচ্ছে calloc. এটি malloc এর মতোই কাজ করে তবে একটু পাকনামি বেশি করে এটাই সমস্যা। এটি মেমোরির যতটুকু অংশ বরাদ্দ করে হয়েছে সেখানে সব 0 দিয়ে ইনিশিয়ালাইজ করে দেয়।

7.1 Using array in pointer. (7.3)

/\*প্রথম হতে তৃতীয় শ্রেনী পর্যন্ত সব শিক্ষার্থীর গণিত পরীক্ষার নম্বর একটি অ্যারেতে রাখতে চাচ্ছি তারমানে marks[3][1000] অর্থাৎ প্রতি শ্রেনিতে সর্বোচ্চ 1000 জন করে। কিন্ত ডায়নামিক মেমোরি অ্যালোকেশন এর মাধ্যেমে একটি পয়েন্টার অ্যারে তৈরি করে নিম্নোক্ত ভাবে করতে পারি\*/

#include <stdio.h>

#include <stdlib.h>

int main()

{

int \*array[3], num[3];

int i, j, n;

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

{

printf("Enter number of students for class %d : ", i + 1);

scanf("%d", &n);

num[i] = n;

array[i] = (int\*)malloc(sizeof(int) \* n);

if (array[i] == NULL) {

printf("Memory allocation failed\n");

return 1;

}

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

printf("Enter marks for student %d : ", j + 1);

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

}

}

printf("Result\n");

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

{

printf("Class %d : \n", i + 1);

for (j = 0; j < num[i]; j++) {

printf("%d\n", array[i][j]);

}

printf("\n");

}

//free(array);

}

Result

Class 1 :

67

78

89

98

84

Class 2 :

78

99

90

Class 3 :

98

96

Output:

Enter number of students for class 1 : 5

Enter marks for student 1 : 67

Enter marks for student 2 : 78

Enter marks for student 3 : 89

Enter marks for student 4 : 98

Enter marks for student 5 : 84

Enter number of students for class 2 : 3

Enter marks for student 1 : 78

Enter marks for student 2 : 99

Enter marks for student 3 : 90

Enter number of students for class 3 : 2

Enter marks for student 1 : 98

Enter marks for student 2 : 96

**কিছু বেসিক বিষয়ঃ**

int \*p; একটি ইন্টিজার টাইপের পয়েন্টার ডিক্লেয়ার করলাম।

* p - একটি ইন্টিজার এর মেমোরির অ্যাড্রেস রাখতে পারবো।
* \*p - সেই মেমোরি অ্যাড্রেস এ যে ভেলুটি আছে সেটি প্রিন্ট করতে পারবো।
* \*p+1 - বলতে বুঝায় p এর ঠিকানায় যে ভেরিয়েবল আছে তার মান এর সাথে এক যোগ।
* \*(p+1) - p এর ঠিকানার পরর্বতী ঠিকানায় যে ভেরিয়েবল আছে তার মান।

7.2 Using array in pointer. (7.5)

Output:

\*p = 100

\*p + 1 = 101

\*(P + 1) = 300

\*p + 2 = 102

\*(p + 2) = 500

#include <stdio.h>

int main()

{

int num[] = { 100, 300, 500, 700, 900 };

int \*p;

p = num;

printf("\*p = %d\n", \*p);

printf("\*p+1 = %d\n", \*p + 1);

printf("\*(P+1) = %d\n", \*(p + 1));

printf("\*p+2 = %d\n", \*p + 2);

printf("\*(p+2) = %d\n", \*(p + 2));

}

7.3 Using array in pointer. (7.6)

Output:

B a n g

B C D E

#include <stdio.h>

int main()

{

char \*str = "Bangladesh";

printf("%c %c %c %c\n", \*str, \*(str + 1), \*(str + 2), \*(str + 3));

printf("%c %c %c %c\n", \*str, \*str + 1, \*str + 2, \*str + 3);

}

7.4 Using array in pointer. (7.7)

#include <stdio.h>

int main()

{

char \*p, a = 10;

int \*q, b = 'F';

double \*r, c = 302.64;

p = &a; q = &b; r = &c;

printf("Size of character : %d bytes\n", sizeof(char));

printf("p = %p\n", p);

printf("p+1 = %p\n", p + 1);

printf("p+2 = %p\n", p + 2);

printf("Size of interger : %d bytes\n", sizeof(int));

printf("q = %p\n", q);

printf("q+1 = %p\n", q + 1);

printf("q+2 = %p\n", q + 2);

printf("Size of double : %d bytes\n", sizeof(double));

printf("r = %p\n", r);

printf("r+1 = %p\n", r + 1);

printf("r+2 = %p\n", r + 2);

}

Output:

Size of character : 1 bytes

p = 00EFFC9F

p + 1 = 00EFFCA0

p + 2 = 00EFFCA1

Size of interger : 4 bytes

q = 00EFFC84

q + 1 = 00EFFC88

q + 2 = 00EFFC8C

Size of double : 8 bytes

r = 00EFFC68

r + 1 = 00EFFC70

r + 2 = 00EFFC78

**Void pointer:**

void pointer ডিক্লেয়ার করার নিয়মঃ void \*ptr;

* যদি n একটি ইন্টিজার ভেরিয়েবল হয় তাহলে তার অ্যাড্রেস আমরা একটি পয়েন্টার এর

মধ্যে রাখতে পারবো এইভাবেঃ ptr = &n;

* আবার, আমরা যদি ptr কে ডিরেফারেন্স করতে চাই তাহলে শুধু \*ptr লিখলেই হবে না

আমাদের টাইপ কাষ্ট করে নিতে হবে এভাবেঃ \*((int\*)ptr)

8. Void pointer. (7.8)

#include <stdio.h>

int main()

{

void \*vp; //vp means void pointer.

int n = 10;

vp = &n;

printf("Address of n = %p\n", &n);

printf("Value of vp = %p\n", vp);

printf("Content of vp = %d\n", \*((int\*)vp));

}

Output:

Address of n = 012FFB88

Value of vp = 012FFB88

Content of vp = 10

ফাংশন পয়েন্টার এর কিছু বেসিক বিশয়ঃ

একটি নরমাল ফাংশন এর প্রোটোটাইপঃ

* int function(int, int);

কিন্তু আমরা যদি ফাংশন পয়েন্টার ব্যবহার করি তাহলে প্রোটোটাইপ হবেঃ

* int (\*function) (int, int);

অর্থাৎ function-এ আমরা একটি ফাংশন এর ঠিকানা রাখতে পারবো।

9.1 Function Pointer. (7.9)

#include <stdio.h>

int main()

{

int add(int n1, int n2);

int sub(int n1, int n2);

int (\*function)(int, int);

int n1 = 10, n2 = 5;

function = &add;

printf("Result : %d\n", function(n1, n2));

function = &sub;

printf("Result : %d\n", function(n1, n2));

}

int add(int n1, int n2)

{

return n1 + n2;

}

int sub(int n1, int n2)

{

return n1 - n2;

}

Output:

Result: 15

Result : 5

Function Pointer ব্যবহার করার একটি বড় সুবিধা হচ্ছে, এটি দিয়ে ফাংশনকে

আরেকটি ফাংশনের প্যারামিটার হিসাবে পাঠানো যায়। নিচের উদাহরণ টি দেখোঃ

9.2 Function pointer. (7.10)

#include <stdio.h>

int main()

{

int add(int n1, int n2);

int sub(int n1, int n2);

int n1 = 10, n2 = 5;

printf("Result : %d\n", operate(&add, n1, n2));

printf("Result : %d\n", operate(&sub, n1, n2));

}

int add(int n1, int n2)

{

return n1 + n2;

}

int sub(int n1, int n2)

{

return n1 - n2;

}

int operate(int (\*op)(int, int), int a, int b)

{

return op(a, b);

}

Output:

Result: 15

Result : 5

Operator ফাংশনের ভেতর একটি ফাংশনের পয়েন্টার পাঠাচ্ছি।

operator(&add, n1, n2) না লিখে operator(add, n1, n2) লিখলেও

হবে। কারন ফাংশনের নামটিই তার ঠিকানা হিসাবে ব্যবহার করা যায়। অর্থাৎ অ্যারের

বেলায় যেমন অ্যারের নামটি প্রথম ঘরের ঠিকানা হিসাবে কাজ করে।

10. Some basic problem to better understating in pointer.

#include <stdio.h>

int main()

{

int a = 5;

int \*p; /\*Pointer declaration\*/

p = &a; /\*Copying address of a to pointer p\*/

\*p = 10; /\*Use pointer to change the value of variable a\*/

printf("%d, ", a);

printf("%d, ", \*p);

printf("%d", \*&a);

}

Output: 10, 10, 10

10. Some basic problem to better understating in pointer.

#include <stdio.h>

int main()

{

int a = 5, \*p1;

float b = 2.5, \*p2;

char c = 'a', \*p3;

p1 = &a;

p2 = &b;

p3 = &c;

printf("%d, ", sizeof(p1));

printf("%d, ", sizeof(p2));

printf("%d", sizeof(p3));

}

Output: 4, 4, 4

/\*Any type of pointer gets two bytes in the memory\*/

/\*This output is depend on the computer. My compiler gives 4, 4, 4\*/

/\*Pointer store the address of a variable\*/

10. Some basic problem to better understating in pointer.

#include <stdio.h>

int main()

{

int a[5] = { 10, 20, 30, 40, 50 }, \* p;

p = &a;

/\*Address of array can be pointed into the pointer without the use of & operator like that p = a\*/

printf("%d, ", \*++p); /\*Point index number 1\*/

printf("%d, ", ++\*p);

printf("%d, ", \*p--);

printf("%d", \*p);

}

Output:

20, 21, 21, 10

11. Pointer to a pointer.

#include <stdio.h>

int main()

{

int a = 5;

int \*p1; /\*Pointer to an integer\*/

int \*\*p2; /\*Pointer to pointer to an integer\*/

int \*\*\*p3; /\*Pointer to pointer to pointer to an integer\*/

p1 = &a;

p2 = &p1;

p3 = &p2;

printf("%d, ", a);

printf("%d, ", \*p1);

printf("%d, ", \*\*p2);

printf("%d, ", \*\*\*p3);

}

Output:

5, 5, 5, 5

* Two pointer can not be multiplied divide or added.
* NULL pointer means not to point anywhere.

A NULL pointer is declare many ways which is given below:

* int \*p = NULL;
* int \*p = 0;
* if (P != NULL) printf("%d", \*p);
* if (p) printf("%d", \*p);

/\*This has the same meaning as our previous example;

if (p) is equivalent to if (p!= 0) and to if (p != NULL)\*/

/\*Swap two numbers\*/

#include <stdio.h>

int main()

{

int a, b;

printf("Before swapping a and b : ");

scanf("%d %d", &a, &b);

int temp;

temp = a;

a = b;

b = temp;

printf("After swapping a and b : %d %d\n", a, b);

}

Output:

Before swapping a and b : 5 10

After swapping a and b : 10 5

12. Swap two number using pointer.

#include <stdio.h>

int main()

{

void swap(int \*x, int \*y);

int a, b;

printf("Before swapping a and b : ");

scanf("%d %d", &a, &b);

swap(&a, &b);

printf("After swapping a and b : %d %d", a, b);

}

void swap(int \*x, int \*y)

{

int temp;

temp = \*x;

\*x = \*y;

\*y = temp;

}

Output:

Before swapping aand b : 5 10

After swapping a and b : 10 5

/\*The function prototype is declared as pointers.\*/

/\*When the function is called, the addresses are passed as actual arguments.\*/

13. Write a program to extract a substring from a string.

#include<stdio.h>

#include<string.h>

int main()

{

char str[20], news[20];

char \*s, \*t;

int position, n, i;

printf("Enter the string:");

gets(str);

printf("Enter the position and number of characters to extract:");

scanf("%d %d", &position, &n);

s = str;

t = news;

if (n == 0) n = strlen(str);

s = s + position - 1;

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

{

\*t = \*s;

s++;

t++;

}

\*t = '\0';

printf("The substring is: %s\n", news);

}

Output:

Enter the string : Dhaka bangladesh

Enter the positionand number of characters to extract : 3 8

The substring is : aka bang

File(ফাইল)

1.Introduction to file.

2. Writing a file using fputc().

3. Writing a file using fputs()

4. Writing to file using fprintf()

5. Reading a file using fgetc()

6. Reading a file using fgets()

7. Reading a file using fscanf()

8. Students details by using file.

9. Preprocessor #define

10. Preprocessor creating header file

11(1).writing to a file using fprintf()

11(2). Writing to a file using fprintf()

12. Reading from a file using fscanf()

13. নিজে নিজে হেডার ফাইল তৈরি করা

14. Sum of two number using file

15. Copy a image from input.img to output.img

16. Fingign sum and aveage using file-1

17. Finging sum and average using file-2

18. Usng of fseek and perror function.

19. Finding a file size using ftell()

20. Remove a file

21. Payroll

22. Read character from a file

23. Numbering the lines

24. A voting problem

25. A Geography Quiz Problem

26. Word frequency count

27. A Voting Problem using structure and file.

Some Important Basic.

fp = fopen("kibria.txt", "r");

fclose(fp);

fprintf(fp, "Golam Kibria %d", num);

int: ঠিক কতটি ক্যারেক্টার লিখা হলো সেই সংখ্যা রির্টান করে।

fscanf(fp, "%d", &num);

int: ঠিক কতটি ইনপুট পড়া হলো সেই সংখ্যা রির্টান করে।

sscanf(source string, "%d", &num);

int: ঠিক কতটি ইনপুট পড়া হলো সেই সংখ্যা রির্টান করে।

perror("File opening failed");

void: কোনো মান রির্টান করে না।

fgetc(fp);

int: একটি ক্যারেক্টারকে ইন্টিজার এ টাইপ কাষ্ট করে রির্টান করে।

fputc(ch, fp\_out);

int: যে ক্যারেক্টারটি লিখা হবে সেটিই রির্টান করে।

fgets(line, 80, fp\_in); //char line[80]

fgets(name);

fputs(name, fp\_in);

fseek(fp, long int offset, int whence)

long int offset : whence হতে কত বাইট পড়ে তা নির্দেশ করে।

int whence : যেই অবস্থানের সাথে offset যোগ করে হবে তার মান।

ftell(fp\_in);

return type long int. formet specifier %ld

মূলত ফাইল এর সাইজ নির্নয় করার জন্য ftell ব্যবহার হয়।

/\*Noel Kalicharan sir এর বইয়ের সমস্যা গুলোর শুধু সমাধান দেওয়া আছে। কোশ্চেন গুলো বই থেকে দেখে নিও। তাহলে প্রোগ্রাম গুলো বুঝতে সুবিধা হবে\*/

1. Inreoduction to file

#include <stdio.h>

int main()

{

Output:

compiler:

File is opened

file :

null(কিছুই না)

FILE\* file;

file = fopen("test.txt", "w");

if (file == NULL)

printf("File does not exist\n");

else

printf("File is opened\n");

fclose(file);

getch();

}

2. Writing a file using fputc()

#include <stdio.h>

int main()

{

Output:

compiler:

File is opened

File is written successfullly

file:

Golam Kibria

FILE\* file;

file = fopen("test.txt", "w");

char name[30] = "Golam Kibria";

int length = strlen(name);

int i;

if (file == NULL)

printf("File does not exist\n");

else

{

printf("File is opened\n");

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

fputc(name[i], file);

printf("File is written successfullly\n");

}

fclose(file);

getch();

}

3. Writing a file using fputs()

#include <stdio.h>

int main()

{

FILE\* file;

char name[30];

file = fopen("test.txt", "w");

if (file == NULL)

{

printf("File does not exist\n");

}

else

{

printf("File is opened\n");

printf("Enter your name : ");

gets(name);

**/\*ay 2 line upore char er nichy dileo hoto\*/**

fputs(name, file);

printf("File is written successfully\n");

}

fclose(file);

}

output:

compiler:

File is opened

Enter your name : Golam kibria

File is written successfully

File

Golam Kibria

4. Writing to file using fprintf()

#include <stdio.h>

int main()

{

FILE\* file;

char name[30];

int age;

file = fopen("test.txt", "a");

if (file == NULL)

printf("File does not exist\n");

else

{

printf("File is opened\n");

printf("Enter your name : ");

gets(name);

printf("Enter your age : ");

scanf("%d", &age);

fprintf(file, "Name = %s, Age = %d\n", name, age);

printf("File is written successfully\n");

}

fclose(file);

}

Output:

Compiler:

File is opened

Enter your name : Golam Kibria

Enter your age : 20

File is written successfully

File :

Name = Golam Kibria, Age = 20

5. Reading a file using fgetc()

#include <stdio.h>

int main()

{

FILE\* file;

char ch;

file = fopen("test.txt", "r");

if (file == NULL)

{

printf("File does not exist\n");

}

else

{

printf("File is opened\n");

while (!feof(file))

{

ch = fgetc(file);

printf("%c", ch);

}

}

fclose(file);

}

**/\*protom a file a input dity hoby then oita compilar a show korby\*/**

Output:

FIle:

Name = kibria, Age = 20

Compiler :

File is opened

Name = kibria, Age = 20

6. Reading a file using fgets()

#include <stdio.h>

int main()

{

FILE\* file;

char ch[40];

file = fopen("test.txt", "r");

if (file == NULL)

{

printf("File does not exist\n");

}

else

{

printf("File is opened\n");

while (!feof(file))

{

fgets(ch, 39, file);

printf("%s\n", ch);

}

}

fclose(file);

}

Output:

File:

Name = kibria, Age = 20

Compiler :

File is opened

Name = kibria, Age = 20

7. Reading a file using fscanf()

#include<stdio.h>

int main()

{

FILE\* file;

char fname[40];

char lname[40];

int age;

file = fopen("test.txt", "r");

if (file == NULL)

{

printf("File does not exist");

}

else

{

printf("File is opened\n");

fscanf(file, "%s\t%s\t%d", &fname, &lname, &age);

printf("%s\t%s\t%d\n", fname, lname, age);

}

fclose(file);

}

Output:

File:

Golam Kiria 20

Compiler:

Golam Kibria 20

8. Students details by using file.

#include <stdio.h>

int main()

{

FILE\* file;

char name[30];

int age, phonenumber, num, i;

file = fopen("student.txt", "a");

if (file == NULL)

{

printf("File does not exist\n");

}

else

{

printf("File is opened\n");

printf("Enter numbers of students = ");

scanf("%d", &num);

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

{

printf("Enter student name = ");

scanf("%s", &name);

printf("Enter student age = ");

scanf("%d", &age);

printf("Enter students phone number = ");

scanf("%d", &phonenumber);

fprintf(file, "\n%s\t\t%d\t%d\n", name, age, phonenumber);

}

fclose(file);

}

Output:

Compilar:

File is opened

Enter numbers of students = 2

Enter student name = kibria

Enter student age = 20

Enter students phone number = 9878

Enter student name = saim

Enter student age = 20

Enter students phone number = 7654

File:

Name Age Phonenumber

kibria 20 9878

saim 20 7654

getch();

}

9. Preprocessor #define

#include <stdio.h>

#define name "Golam Kibria"

int main()

{

printf("%s\n", name);

}

Output: Golam Kibria

Or,

#include <stdio.h>

#define du printf("Daffodil IU")

int main()

{

du;

}

Output: Daffodil IU

Or,

#include <stdio.h>

#define du printf("Daffodil IU\n")

int main()

{

du;

du;

du;

}

Output:

Daffodil IU

Daffodil IU

Daffodil IU

10. Preprocessor creating header file

#include<stdio.h>

#include "info.h"

int main ()

{

printf ("Name = %s\n", name);

printf ("Age = %d\n", age);

getch();

}

#define name "Golam Kibria"

#define age 20

info.h namok header file er moddy ay information gulo asy.

11(1).writing to a file using fprintf()

Output:

File:

Golam Kibria

Dhaka, Bangladesh

#include <stdio.h>

int main()

{

FILE\* fp;

fp = fopen("myfile.txt", "w");

fprintf(fp, "Golam Kibria\n");

fprintf(fp, "Dhaka, Bangladesh\n");

fclose(fp);

}

11(2). Writing to a file using fprintf()

#include <stdio.h>

Output:

Golam Kibria

Dhaka, Bangladesh

Ki obsta

int main()

{

FILE\* fp;

fp = fopen("myfile.txt", "w");

fprintf(fp, "Golam Kibria\n");

fprintf(fp, "Dhaka, Bangladesh\n");

fclose(fp);

fp = fopen("myfile.txt", "a");

fprintf(fp, "Ki obsta\n");

fclose(fp);

}

12. Reading from a file using fscanf()

**/\*file input -> rur code -> file output\*/**

#include <stdio.h>

int main()

{

FILE\* in, \* out;

in = fopen("input.txt", "r");

out = fopen("output.txt", "w");

int num1, num2, sum;

fscanf(in, "%d", &num1);

fscanf(in, "%d", &num2);

sum = num1 + num2;

fprintf(out, "%d\n", sum);

fclose(in);

fclose(out);

}

Output:

* **প্রথমে Notepad দিয়ে একটা input.txt নামক টেক্সট ফাইল তৈরি করে নিতে হবে।**
* **তারপর input.txt ফাইল এ দুটি value(e.g.10 20) দিয়ে ফাইল টা সেভ করে নিতে হবে।**
* **সর্বশেষ, কোডটা রান করলে output.txt নামক একটা ফাইল অটোমেটিক তৈরি হবে এবং তার মধ্যে যোগফল(30) টা দেখা যাবে।**

13. নিজে নিজে হেডার ফাইল তৈরি করা

#include <stdio.h>

#include "dimik.h"

int main()

{

int n1 = 10, n2 = 5;

printf("%d + %d = %d\n", n1, n2, add(n1, n2));

printf("%d \* %d = %d\n", n1, n2, mul(n1, n2));

return 0;

}

Output:

10 + 5 = 15

10 \* 5 = 50

**/\*dimik.h নামক হেডার ফাইল এর মধ্যে নিচের তথ্য গুলো আছে\*/**

int add(int n1, int n2)

{

return n1 + n2;

}

int mul(int n1, int n2)

{

return n1 \* n2;

}

14. Sum of two number using file

**/\*file input -> rur code -> file output\*/**

#include <stdio.h>

int main()

{

FILE\* in, \* out;

in = fopen("inp.txt", "r");

out = fopen("out.txt", "w");

int n1, n2, sum;

char line[80];

fgets(line, 80, in);

printf("%s\n", line);

sscanf(line, "%d %d", &n1, &n2);

sum = n1 + n2;

fprintf(out, "%d\n", sum);

fclose(in);

fclose(out);

}

Output:

* **প্রথমে Notepad দিয়ে একটা inp.txt নামক টেক্সট ফাইল তৈরি করে নিতে হবে।**
* **তারপর input.txt ফাইল এ দুটি value(e.g.10 25) দিয়ে ফাইল টা সেভ করে নিতে হবে।**
* **সর্বশেষ, কোডটা রান করলে out.txt নামক একটা ফাইল অটোমেটিক তৈরি হবে এবং তার মধ্যে যোগফল(35) টা দেখা যাবে।**

15. Copy a image from input.img to output.img

#include <stdio.h>

#include <stdlib.h>

int main()

{

FILE\* in, \* out;

in = fopen("kibria.png", "rb");

out = fopen("image.png", "wb");

int x;

if (in == NULL)

{

perror("File opening failed");

return EXIT\_FAILURE:

}

while (1)

{

x = fgetc(in);

if (x == EOF) {

break;

}

fputc(x, out);

}

fclose(in);

fclose(out);

}

Output:

image.png নামক একটা নতুন ফাইল তৈরি হয়ে

kibria.png এর মধ্যে যে ছবিটা ছিলো তা দেখিয়েছে।

16. Fingign sum and aveage using file. N109

#include <stdio.h>

int main()

{

FILE\* in;

in = fopen("inp.txt", "r");

int num, sum = 0, n = 0;

fscanf(in, "%d", &num);

if (num == 0) printf("No numbers were entered\n");

else

{

while (num != 0)

{

n = n + 1;

sum = sum + num;

fscanf(in, "%d", &num);

}

printf("%d numbers were entered\n", n);

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

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

}

fclose(in);

}

Output:

file: 24 13 55 32 10 0

compiler:

5 numbers were entered

The sum is = 134

The average is = 26.80

17. Finging sum and average using fileN112

**/\*file input -> run code -> file output\*/**

#include <stdio.h>

int main()

{

FILE\* in, \* out;

in = fopen("inp.txt", "r");

out = fopen("out.txt", "w");

int num, sum = 0, n = 0;

fscanf(in, "%d", &num);

if (num == 0) fprintf(out, "No numbers were entered\n");

else

{

while (num != 0)

{

n = n + 1;

sum = sum + num;

fscanf(in, "%d", &num);

}

fprintf(out, "%d numbers were entered\n", n);

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

fprintf(out, "The average is = %.2lf\n", (double)sum / n);

}

fclose(in);

fclose(out);

}

Output:

int.txt

24 13 55 32 10 0

out.txt

5 numbers were entered

The sum is = 134

The average is = 26.80

18. Usng of fseek and perror function.

#include <stdio.h>

#include <stdlib.h>

int main()

{

FILE\* fp;

fp = fopen("in.txt", "r");

int ch;

if (fp == NULL)

{

perror("Can not open file");

return EXIT\_FAILURE:

}

ch = fgetc(fp);

printf("%c\n", (char)ch);

ch = fgetc(fp);

printf("%c\n", (char)ch);

fseek(fp, 0, 0);

ch = fgetc(fp);

printf("%c\n", (char)ch);

fclose(fp);

}

File:

ABCKEFGHIJK

MNOPQRST

UNWX

YZ

**Output:**

compiler:

A

B

A

* fseek(fp, 0, SEEK\_SET) same output like A\nB\nA
* fseek(fp, 0, SEEK\_CUR) output: A\nB\nC
* fseek(fp, 0, SEEK\_END) output: A\nB
* fseek(fp, sizeof(char) \* 3, SEEK\_CUR); output: A\nB\nF/

তিনটি ক্যারেক্টার বাদ দিয়ে আউটপুট দেখাবে।

19. Finding a file size using ftell()

#include <stdio.h>

#include <stdlib.h>

int main()

{

FILE\* in;

in = fopen("kibria.png", "rb");

if (in == NULL)

{

perror("File opening failed");

return EXIT\_FAILURE;

}

fseek(in, 0, SEEK\_END);

printf("File size: %ld bytes\n", ftell(in));

printf("File size(kilobytes): %ld\n", ftell(in) / 1024);

fclose(in);

}

Output:

compiler:

**kibria.png নামক image টার সাইজ বের করে দিবে।**

20. Remove a file

#include <stdio.h>

int main()

{

int return\_value;

char\* filename = "kibria.png";

return\_value = remove(filename);

if (return\_value != 0)

{

perror("File remove Failed");

return 1;

}

printf("%s removed successfully\n", filename);

}

Output:

kibria.png নামক .png ফাইল টা পুরো সি ফোল্ডার হতে কোড টা রান করার সাথে সাথে বিমুভ হয়ে যাবে।

21. Payroll N117

#include <stdio.h>

#include <string.h>

#define maxreghour 40

#define overtimefactor 1.5

int main()

{

FILE\* inp = fopen("paydata.txt", "r");

FILE\* out = fopen("payroll.txt", "w");

char firstname[30], lastname[30], name[30], bestpaid[30];

double hours, rate, regularpay, overtimepay, netpay;

double totalbill = 0, mostpaid = 0;

int numofemploye = 0;

fprintf(out, "Name\t\t\t Hours\t Rate\t\tRegular\t\tOvertime\tNet\n\n");

//2 space & 3 space

fscanf(inp, "%s", firstname);

while (strcmp(firstname, "END") != 0)

{

numofemploye++;

fscanf(inp, "%s %lf %lf", lastname, &hours, &rate);

if (hours <= maxreghour)

{

regularpay = hours \* rate;

overtimepay = 0;

}

else

{

regularpay = maxreghour \* rate;

overtimepay = (hours - maxreghour) \* rate \* overtimefactor;

}

netpay = regularpay + overtimepay;

strcpy(name, firstname); strcat(name, " "); strcat(name, lastname);

fprintf(out, "%-15s %14.1lf %10.2lf", name, hours, rate);

fprintf(out, "%14.2lf %15.2lf %12.2lf\n", regularpay, overtimepay, netpay);

if (netpay > mostpaid)

{

mostpaid = netpay;

strcpy(bestpaid, name);

}

totalbill = totalbill + netpay;

fscanf(inp, "%s", firstname);

}

fprintf(out, "\nNumbers of employess: %d\n", numofemploye);

fprintf(out, "Total bill: %.2lf\n", totalbill);

fprintf(out, "%s earned the most pay of $%.2lf\n", bestpaid, mostpaid);

fclose(inp);

fclose(out);

}

**Output:**

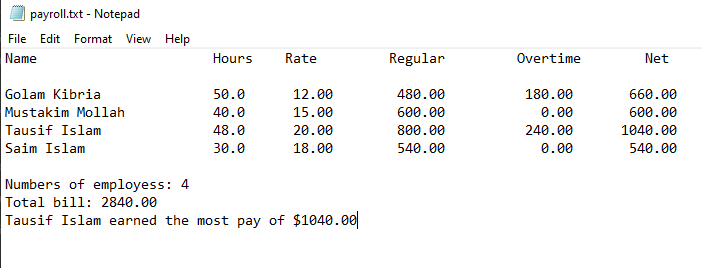
Golam Kibria 50.0 12.

Mustakim Mollah 40.0 15.00

Tausif Islam 48.0 20.00

Saim Islam 30.0 18.00

END



22. Read character from a file N156

#include <stdio.h>

int main()

{

FILE\* in;

in = fopen("input.txt", "r");

char ch;

ch = getc(in);

while (ch != '\n')

{

putchar(ch);

ch = getc(in);

}

putchar('\n');

fclose(in);

}

***/\*putchar is used for write a single character to the standard output***

***and also write the character in the next position int the output\*/***

#include <stdio.h>

int main()

{

FILE\* in;

in = fopen("input.txt", "r");

char ch;

ch = getc(in);

while (ch != '\n' && ch != EOF)

{

putchar(ch);

ch = getc(in);

}

//putchar('\n');

fclose(in);

}

***/\*putchar('\n) is not needed cuz getc(in) will***

***returned EOF when the end of file is reached\*/***

#include <stdio.h>

int main()

{

FILE\* in;

in = fopen("input.txt", "r");

char ch;

ch = getc(in);

while (ch != EOF)

Output:

File: golam Kibria

Compiler: golam kibria

{

putchar(ch);

ch = getc(in);

}

fclose(in);

}

***/\*This will also print same output\*/***

***/\*All the three problem is same but the solving system is different\*/***

23. Numbering the lines N159

***/\*This program prints the data from a file by numbering\*/***

#include <stdio.h>

int main()

{

FILE\* in;

in = fopen("input.txt", "r");

char ch;

int linenumber = 0;

int writtenlinenumber = 1;

ch = getc(in);

while (ch != EOF)

{

if (writtenlinenumber == 1)

{

printf("%2d. ", ++linenumber);

writtenlinenumber = 0;

}

putchar(ch);

if (ch == '\n')

{

writtenlinenumber = 1;

}

ch = getc(in);

}

fclose(in);

}

***/\*The expression ++linenumber means linenumber is increment***

***friest brfore print(1, 2, 3). But if we use linenumber++ it***

***will print frist then increment(0, 1, 2)\*/***

Output:

File:

There was a little girl

Who had a little curl

Right in the middle of her head

Compiler:

1. There was a little girl

2. Who had a little curl

3. Right in the middle of her head

24. A voting problem N235

#include <stdio.h>

#include <string.h>

FILE\* in, \* out;

int main()

{

void initialize(char name[][31], int vote[]);

void printresult(char name[][31], int vote[], int validvote, int spoilvote);

char name[8][31];

int vote[8];

int validvote = 0, spoilvote = 0;

int x;

in = fopen("votee.txt", "r");

out = fopen("resultt.txt", "w");

initialize(name, vote);

fscanf(in, "%d", &x);

while (x != 0)

{

if (x < 1 || x > 7) {

fprintf(out, "Invalid Vote : %d\n", x);

++spoilvote;

}

else {

++vote[x];

++validvote;

}

fscanf(in, "%d", &x);

}

printresult(name, vote, validvote, spoilvote);

fclose(in);

fclose(out);

}

void initialize(char name[][31], int vote[])

{

char lastname[30];

for (int t = 1; t <= 7; t++)

{

fscanf(in, "%s %s", name[t], lastname);

strcat(name[t], " ");

strcat(name[t], lastname);

vote[t] = 0;

}

}

int getlargest(int num[], int m, int n)

{

int big = m;

for (int i = m + 1; i <= n; i++) {

if (num[i] > num[big])

big = i;

}

return big;

}

void printresult(char name[][31], int vote[], int validvote, int spoilvote)

{

int getlargest(int num[], int m, int n);

fprintf(out, "\nNumber of voters : %d\n", validvote + spoilvote);

fprintf(out, "Number of valid vote : %d\n", validvote);

fprintf(out, "Number of spoil vote : %d\n", spoilvote);

fprintf(out, "\nCandidate Score\n\n");

for (int p = 1; p <= 7; p++) {

fprintf(out, "%-19s %3d\n", name[p], vote[p]);

}

fprintf(out, "\nThe winner\n");

int win = getlargest(vote, 1, 7);

int winningvote = vote[win];

for (int n = 1; n <= 7; n++) {

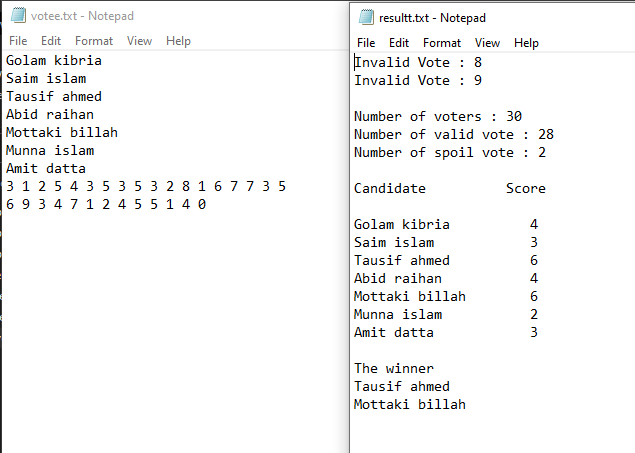
if (vote[n] == winningvote) {

fprintf(out, "%s\n", name[n]);

}

}

}



25. A Geography Quiz Problem N230

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

int main()

{

void getString(FILE\*, char[]);

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

char country[51];

char capital[51], CAPITAL[51];

char endofdata[] = "\*";

FILE\* in = fopen("quizdata.txt", "r");

if (in == NULL) {

printf("Cannot find file\n");

exit(1); //exit is used to terminate execution of a program.

}

getString(in, country);

while (strcmp(country, endofdata) != 0) {

getString(in, capital);

getString(in, CAPITAL);

askonequestion(country, capital, CAPITAL);

getString(in, country);

}

}

void askonequestion(char country[], char capital[], char CAPITAL[])

{

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

char answer[51], ANSWER[51];

printf("\nWhat is the capital of %s? ", country);

gets(answer);

lettersonlyupper(answer, ANSWER);

if (strcmp(ANSWER, CAPITAL) == 0)

printf("Correct\n");

else {

printf("Wrong. Try again\n");

printf("\nWhat is the capital of %s? ", country);

gets(answer);

lettersonlyupper(answer, ANSWER);

if (strcmp(ANSWER, CAPITAL) == 0)

printf("Correct\n");

else

printf("Wrong. Answer is : %s\n", capital);

}

}

void lettersonlyupper(char word[], char WORD[])

{

int i = 0, n = 0;

char ch;

while ((ch = word[i++]) != '\0') {

if (isalpha(ch))

WORD[n++] = toupper(ch);

}

WORD[n] = '\0';

}

void getString(FILE\* in, char str[])

{

char ch, delimiter;

int x = 0;

str[0] = '\0';

while (isspace(ch = getc(in))); //empty while body.

if (ch == EOF)

return;

delimiter = ch;

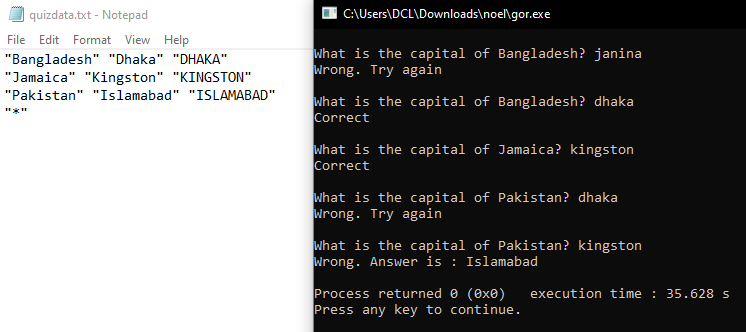
while (((ch = getc(in)) != delimiter) && (ch != EOF)) {

str[x++] = ch;

}

str[x] = '\0';

}



26. Word frequency count N-267

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

int main()

{

int getWord(FILE \* in, char str[]);

int binarysearch(int first, int last, char key[], int max, char list[][max]);

void addtolist(char item[], int max, char list[][max], int freq[], int p, int n);

void printresult(FILE \* out, int max, char list[][max], int freq[], int n);

char wordlist[50][11];

char perword[11];

int frequency[50];

int count = 0;

FILE\* in = fopen("passage.txt", "r");

if (in == NULL) {

printf("Cannot find file\n");

exit(1);

}

FILE\* out = fopen("output.txt", "w");

if (out == NULL) {

printf("Cannot create output file\n");

exit(2);

}

for (int h = 1; h <= 50; h++)

frequency[h] = 0;

while (getWord(in, perword) != 0) {

int location = binarysearch(0, count - 1, perword, 11, wordlist);

if (strcmp(perword, wordlist[location]) == 0)

++frequency[location]; /\*Word found\*/

else { /\*This is a new word\*/

if (count < 50) {

addtolist(perword, 11, wordlist, frequency, location, count - 1);

++count;

}

else

fprintf(out, "'%s' not added to table\n", perword);

}

}

printresult(out, 11, wordlist, frequency, count);

}

int getWord(FILE\* in, char str[])

{

***/\*Store the next word, if any, in str. Convert word to lowercase***

***Retrun 1 if a word is found, if not found then return 0\*/***

char ch;

int n = 0;

/\*Read until white space\*/

while (!isalpha(ch = getc(in)) && ch != EOF); ***/\*Empty while body\*/***

if (ch == EOF) return 0;

str[n++] = tolower(ch);

while (isalpha(ch = getc(in)) && ch != EOF) {

if (n < 10)

str[n++] = tolower(ch);

}

str[n] = '\0';

return 1;

}

int binarysearch(int first, int last, char key[], int max, char list[][max])

{

***/\*Search for key from list[first] to list[last] . If key is found returns keys location. If not found, return the location in which is should be inserted. The callind program will check the location to determine if found\*/***

while (first <= last) {

int mid = (first + last) / 2;

int cmp = strcmp(key, list[mid]);

if (cmp == 0)return mid;

if (cmp < 0)last = mid - 1;

else first = mid + 1;

}

return first; ***/\*Not found; should be inserted in location first\*/***

}

void addtolist(char item[], int max, char list[][max], int freq[], int p, int n)

{

***/\*Adds item in position list[p]; sets freq[p] to 1.***

***Shift list[n] down to list [p] to the right\*/***

for (int h = n; h >= p; h--) {

strcpy(list[h + 1], list[h]);

freq[h + 1] = freq[h];

}

strcpy(list[p], item);

freq[p] = 1;

}

void printresult(FILE\* out, int max, char list[][max], int freq[], int n)

{

fprintf(out, "Words\t\tFrequency\n\n");

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

fprintf(out, "%-13s %4d\n", list[h], freq[h]);

}



27. A Voting Problem using structure and file. N301

#include <stdio.h>

#include <string.h>

typedef struct {

char name[31];

int numofvotes;

}persondata;

persondata candidate[7];

typedef struct {

int valid, spoilt;

}votecount;

int main()

{

void initialize(persondata person[], int max, FILE \* in);

votecount processvotes(persondata person[], int max, FILE \* in, FILE \* out);

void printresult(persondata person[], int max, votecount c, FILE \* out);

persondata candidate[8];

votecount count;

FILE \*in = fopen("votes.txt", "r");

FILE\* out = fopen("result.txt", "w");

initialize(candidate, 7, in);

count = processvotes(candidate, 7, in, out);

printresult(candidate, 7, count, out);

fclose(in);

fclose(out);

} ***/\*End main\*/***

void initialize(persondata person[], int max, FILE\* in)

{

char lastname[31];

for (int h = 1; h <= max; h++) {

fscanf(in, "%s %s", person[h].name, lastname);

strcat(person[h].name, " ");

strcat(person[h].name, lastname);

person[h].numofvotes = 0;

}

}  ***/\*End initialize\*/***

votecount processvotes(persondata person[], int max, FILE\* in, FILE\* out)

{

votecount temp;

temp.valid = temp.spoilt = 0;

int v;

fscanf(in, "%d", &v);

while (v != 0) {

if (v<1 || v>max) {

fprintf(out, "Invalid vote: %d\n", v);

++temp.spoilt;

}

else {

++person[v].numofvotes;

++temp.valid;

}

fscanf(in, "%d", &v);

}

return temp;

} ***/\*End processvotes\*/***

int getlargest(persondata person[], int first, int last)

{

***/\*Return the index of highest votes from person[first] to person[last]\*/***

int big = first;

for (int h = first + 1; h <= last; h++) {

if (person[h].numofvotes > person[first].numofvotes)

big = h;

}

return big;

} ***/\*End getlargest\*/***

void printresult(persondata person[], int max, votecount c, FILE\* out)

{

int getlargest(persondata person[], int first, int last);

fprintf(out, "\nNumber of votes: %d\n", c.valid+c.spoilt);

fprintf(out, "Number of valid votes: %d\n", c.valid);

fprintf(out, "Number of spoilt votes: %d\n", c.spoilt);

fprintf(out, "\nCandidate score\n\n");

for (int h = 1; h <= max; h++)

fprintf(out, "%-15s %3d\n", person[h].name, person[h].numofvotes);

fprintf(out, "\nThe winner\n");

int win = getlargest(person, 1, max);

int winningvote = person[win].numofvotes;

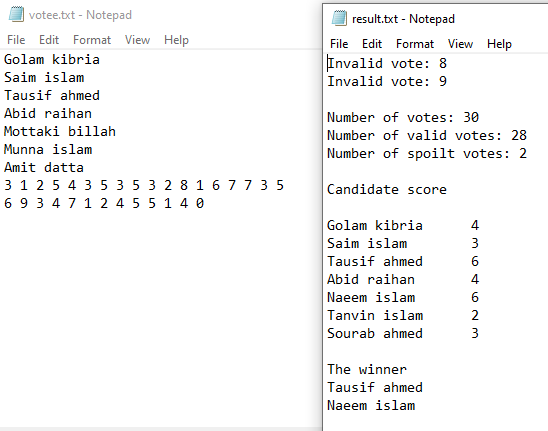
for (int h = 1; h <= max; h++) {

if (person[h].numofvotes == winningvote)

fprintf(out, "%s\n", person[h].name);

}

} ***/\*End printresult\*/***



**Structure and Union**

1(1). Determine Name, Age, Salary.

1(2). Determine Name, Age and Salary from the user.

2.Initialize structure variable.

3.Structure Comparing.

4. Array of Structure & array within structure.

5. Passing structure variable using function.

6(1). Typedef-1

6(2). Typedef-2

7(1). Structure basic problem using typedef.

7(2). Structure basic problem using typedef from the user.

8.Structure basic problem using typedef and pointer.

9.Basic problem of nested structure.

10(1). Basic rules of pointer and its address.

10(2). Student Information.

11. স্টাকচারের মেমোরি অ্যালাইনমেন্ট.

12.Introduction to union.

13.Size of Union.

14. Enum Erasion-1.

15. Enum Erasion-2.

1(1). Determine Name, Age, Salary.

Output:

Name = Golam Kibria

Age = 20

Salary = 2000

#include <stdio.h>

struct person {

char name[20];

int age;

float salary;

};

int main()

{

struct person p;

strcpy(p.name, "Golam Kibria");

p.age = 20;

p.salary = 2000;

printf("Name = %s\n", p.name);

printf("Age = %d\n", p.age);

printf("Salary = %.0f\n", p.salary);

}

1(2).Determine Name, Age and Salary from the user.

Output:

Name ? Golam kibria

Age ? 20

Salary ? 2000

Name = Golam kibria

Age = 20

Salary = 2000

#include <stdio.h>

typedef struct {

char name[20];

int age;

float salary;

}Person;

int main()

{

Person p;

printf("Name? ");

scanf(" %[^\n]", p.name);

printf("Age? ");

scanf("%d", &p.age);

printf("Salary? ");

scanf("%f", &p.salary);

printf("Name = %s\n", p.name);

printf("Age = %d\n", p.age);

printf("Salary = %.0f\n", p.salary);

}

2.Initialize structure variable.

Output:

Person1:

Age = 20

Salary = 2000

Person2 :

Age = 25

Salary = 3000

Person3 :

Age = 25

Salary = 3000

#include <stdio.h>

typedef struct {

int age;

float salray;

}Person;

int main()

{

Person person1 = { 20, 2000 };

Person person2 = { 25, 3000 };

Person person3 = person2;

printf("Person1 : \n");

printf("Age = %d\n", person1.age);

printf("Salary = %.0f\n", person1.salray);

printf("Person2 : \n");

printf("Age = %d\n", person2.age);

printf("Salary = %.0f\n", person2.salray);

printf("Person3 : \n");

printf("Age = %d\n", person3.age);

printf("Salary = %.0f\n", person3.salray);

}

3.Structure Comparing.

#include <stdio.h>

typedef struct {

int age;

float salray;

}Person;

int main()

{

Person person1 = { 20, 2000 };

Person person2 = { 25, 3000 };

Person person3 = person2;

if (person1.age == person2.age && person1.salray == person2.salray)

printf("Person one is equal to person two\n");

else

printf("Person one is not equal to person two\n");

}

Output: Person one is not equal to person two

4. Array of Structure & array within structure.

#include <stdio.h>

typedef struct {

char name[30];

int age;

float salary;

}Person;

int main()

{

Person p[10];

int n;

printf("How many? ");

scanf("%d", &n);

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

printf("Enter information for person %d\n", i + 1);

printf("Name? ");

scanf(" %[^\n]", p[i].name);

printf("Age? ");

scanf("%d", &p[i].age);

printf("Salary? ");

scanf("%f", &p[i].salary);

}

printf("\n");

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

printf("Name = %s\n", p[i].name);

printf("Age = %d\n", p[i].age);

printf("Salary = %.0f\n", p[i].salary);

}

}

Output:

How many ? 2

Enter information for person 1

Name ? Golam Kibria

Age ? 20

Salary ? 2000

Enter information for person 2

Name ? Tausif Ahmed

Age ? 25

Salary ? 3000

Name = Golam Kibria

Age = 20

Salary = 2000

Name = Tausif Ahmed

Age = 25

Salary = 3000

5.Passing structure variable using fundtion.

#include <stdio.h>

typedef struct {

char name[30];

int age;

float salary;

}Person;

int main()

{

void display(Person p);

Person person1, person2;

strcpy(person1.name, "Golam kibria");

person1.age = 20;

person1.salary = 2000;

display(person1);

strcpy(person2.name, "Tausif Ahmed");

person2.age = 25;

person2.salary = 3000;

display(person2);

}

void display(Person p)

{

printf("Name = %s\n", p.name);

printf("Age = %d\n", p.age);

printf("Salary = %.0f\n", p.salary);

}

Output:

Name = Golam kibria

Age = 20

Salary = 2000

Name = Tausif Ahmed

Age = 25

Salary = 3000

6(1). Typedef-1

#include <stdio.h>

typedef struct {

char ch;

}Letter;

int main()

{

Letter l;

l.ch = 'A';

printf("Character is = %c\n", l.ch);

}

Output:

Character is = A

6(2).Typedef-2

#include <stdio.h>

typedef struct {

int price;

char name[30];

}Book;

int main()

{

Book b;

b.price = 300;

strcpy(b.name, "Deyal");

printf("Book name : %s\n", b.name);

printf("Book price : %d\n", b.price);

}

Output:

Book name : Deyal

Book price : 300

7(1).Structure basic problem using typedef. T6.3

#include <stdio.h>

typedef struct {

int id;

char name[30];

}Student;

int main()

{

Student s;

s.id = 2281;

strcpy(s.name, "Kibria");

/\*Character array এর মধ্যে assignment operation

চালানো যায় না তাই strcpy ব্যবহার করেছি\*/

printf("Student ID : %d\n", s.id);

printf("Student Name: %s\n", s.name);

}

Output:

Student ID : 2281

Student Name : Kibria

7(2).Structure basic problem using typedef from the user.T6.4

Output:

ID ? 2281

Name ? Golam kibria

Student ID : 2281

Student Name : Golam kibria

#include <stdio.h>

typedef struct {

int id;

char name[30];

}Student;

int main()

{

Student s;

printf("ID? ");

scanf("%d", &s.id);

printf("Name? ");

scanf(" %[^\n]", s.name);

/\*এইভাবে scan করাটাই উত্তম। %s দিয়ে করলে নামের দ্বিতীয় আংশটা(kibria) বাদ পরে যেত\*/

printf("Student ID : %d\n", s.id);

printf("Student Name: %s\n", s.name);

}

8.Structure basic problem using typedef and pointer. T6.2

Output:

Student ID : 2281

Student Name : Kibria

#include <stdio.h>

typedef struct {

int id;

char \*name;

}Student;

int main()

{

Student s;

s.id = 2281;

s.name = "Kibria";

/\*Character array এর মধ্যে assignment operation

চালানো যায় না তাই ক্যারেক্টার এর বদলে পয়েন্টার ব্যবহার করেছি\*/

printf("Student ID : %d\n", s.id);

printf("Student Name: %s\n", s.name);

}

9(1).Basic problem of nested structure. T6.5

Output:

2281

Golam

Kibria

ID = 2281

Name = Golam Kibria

#include <stdio.h>

#include <string.h>

typedef struct {

char first\_name[20];

char last\_name[20];

}Name;

typedef struct {

int id;

Name;

}Id;

int main()

{

Id a;

scanf("%d", &a.id);

scanf("%s", a.first\_name);

scanf("%s", a.last\_name);

printf("ID = %d\n", a.id);

printf("Name = %s %s\n", a.first\_name, a.last\_name);

}

9(2). Basic problem of nested structure using loop. T6.6

#include <stdio.h>

#include <string.h>

typedef struct {

char first\_name[20];

char last\_name[20];

}Name;

typedef struct {

int id;

Name;

}Id;

int main()

{

Id student[5];

int n;

printf("How many students? ");

scanf("%d", &n);

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

printf("Enter the ID of student %d : ", i + 1);

scanf("%d", &student[i].id);

printf("First Name of student %d : ", i + 1);

scanf("%s", student[i].first\_name);

printf("Last Name of student %d : ", i + 1);

scanf("%s", student[i].last\_name);

}

printf("\n");

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

printf("Information of student %d\n", i + 1);

printf("ID : %d\n", student[i].id);

printf("Name : %s %s\n", student[i].first\_name, student[i].last\_name);

}

}

Output:

How many students ? 2

Enter the ID of student 1 : 2281

First Name of student 1 : Golam

Last Name of student 1 : kibria

Enter the ID of student 2 : 2282

First Name of student 2 : Tausif

Last Name of student 2 : Ahmed

Information of student 1

ID : 2281

Name : Golam kibria

Information of student 2

ID : 2282

Name : Tausif Ahmed

10(1).Basic rules of pointer and its address. T6.8

/\*10(2) পারার জন্য এই প্রোগ্রাম টা বুঝতে হবে\*/

#include <stdio.h>

int main()

{

void plus1(int a);

void plus2(int\* a);

int a = 5;

plus1(a);

printf("a = %d\n", a);

plus2(&a);

printf("a = %d\n", a);

}

void plus1(int a)

{

a = a + 10;

}

void plus2(int\* a)

{

\*a = \*a + 10;

}

Output:

a = 5

a = 15

অর্থাৎ প্রথম ফাংশন এর ভেতরে ভেরিয়েবলের সঙ্গে 10 যোগ করলেও ফাংশনটি রিটার্ন করার সঙ্গে সঙ্গে সেই ভেরিয়েবল হারিয়ে গেছে আর মেইন ফাংশন এর এর মানের কোনো পরিবর্তন হয়নি কিন্তু দ্বিতীয় ক্ষেত্রে আমরা a-এর এড্রেস পাঠিয়ে দিয়েছি। আর সেটা গ্রহণ করেছি একটি পয়েন্টার দিয়ে। এখন সেই পয়েন্টার যাকে পয়েন্ট করে, তার যেকোনো পরিবর্তন করলেও সেটি হারিয়ে যাবে না। যেহেতু মেমোরি একেবার সঠিক জায়গায় পরিবর্তন হয়ে গিয়েছ তাই মেইন ফাংশন এর ভেতরে আসার পরেও সেটি অক্ষুণ্ণ থাকবে।

10(2). Student Information. T6.9

Output:

Enter the ID of student 1 : 11

First Name of student 1 : g

Last Name of student 1 : k

Enter the ID of student 2 : 12

First Name of student 2 : t

Last Name of student 2 : a

Enter the ID of student 3 : 13

First Name of student 3 : s

Last Name of student 3 : i

Enter the ID of student 4 : 14

First Name of student 4 : a

Last Name of student 4 : r

Enter the ID of student 5 : 15

First Name of student 5 : n

Last Name of student 5 : i

Information of student 1

ID : 11

Name : g k

Grade : A

Information of student 2

ID : 12

Name : t a

Grade : A +

Information of student 3

ID : 13

Name : s i

Grade : A -

Information of student 4

ID : 14

Name : a r

Grade : F

Information of student 5

ID : 15

Name : n i

Grade : B

#include <stdio.h>

#include <string.h>

typedef struct {

char first\_name[20];

char last\_name[20];

}Name;

typedef struct {

int id;

Name;

char grade[3];

}Id;

int main()

{

void grade(Id\* s, int marks);

Id student[5];

int n = 5, i;

int marks[5] = { 72, 82, 69, 20, 50 };

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

printf("Enter the ID of student %d : ", i + 1);

scanf("%d", &student[i].id);

printf("First Name of student %d : ", i + 1);

scanf("%s", student[i].first\_name);

printf("Last Name of student %d : ", i + 1);

scanf("%s", student[i].last\_name);

strcpy(student[i].grade, "");

}

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

grade(&student[i], marks[i]);

}

printf("\n");

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

printf("Information of student %d\n", i + 1);

printf("ID : %d\n", student[i].id);

printf("Name : %s %s\n", student[i].first\_name, student[i].last\_name);

printf("Grade : %s\n", student[i].grade);

}

}

/\*ফাংশনের ভেতর সরাসরি ভ্যারিয়েবল না পাঠিয়ে, ওই ফাংশনের ঠিকানায় পাঠানো হয় তাহলে সেই ঠিকানায় ভেরিয়েবল এর কোনো পরিবর্তন করলে আমরা ফাংশনের বাইরে গেলেও সেই ভেরিয়েবল অক্ষত থাকবে। এটিকে বলে Call by reference. যা এই প্রোগ্রামটিতে ব্যবহার করা হয়েছে।\*/

void grade(Id\* s, int marks)

{

if (marks >= 80) strcpy(s->grade, "A+");

else if (marks >= 70) strcpy(s->grade, "A");

else if (marks >= 60) strcpy(s->grade, "A-");

else if (marks >= 50) strcpy(s->grade, "B");

else if (marks >= 40) strcpy(s->grade, "C");

else strcpy(s->grade, "F");

}

/\*ইউজার হতে ইনপুট নিয়ে এই প্রোগ্রামটা নিজে নিজে চেষ্টা করলেই পারা যাবে ইনশাআল্লাহ্‌\*/

11. স্টাকচারের মেমোরি অ্যালাইনমেন্ট. T6.12

#include <stdio.h>

struct one {

int n;

double d;

char c;

};

struct two {

char c;

int n;

double d;

};

int main()

{

printf("char size = %d bytes\n", sizeof(char));

printf("int size = %d bytes\n", sizeof(int));

printf("double size = %d bytes\n", sizeof(double));

printf("One size = %d bytes\n", sizeof(struct one));

printf("Two size = %d bytes\n", sizeof(struct two));

}

Output:

char size = 1 bytes /\*Character always 1 byte\*/

int size = 4 bytes /\*Int always 4 byte\*/

double size = 8 bytes /\*Double always 8 byte\*/

One size = 24 bytes /\*8+8+8=24 byte\*/

Two size = 16 bytes /\*4+4+8=16 byte\*/

12.Introduction to union.

#include <stdio.h>

union test {

int a;

int b;

};

int main()

{

union test t;

t.a = 10;

printf("%d\n", t.a);

printf("%d\n", t.b);

t.b = 20;

printf("%d\n", t.b);

printf("%d\n", t.b);

}

Output:

10

10 /\*একই মেমোরি লোকেশন শেয়ার করছে তাই t.b 10 হয়েছে\*/

20

20

13.Size of Union. T6.10

/\*যদি বিভিন্ন ডিভাইস এর জন্যে Embedded Programing করি তখন Union দরকার হবে। নইলে ভবিষ্যতে আমাদের ইউনিয়ন ব্যবহার করার সম্ভাবনা খুবই কম।\*/

#include <stdio.h>

struct hi {

char ch;

int n;

char str[16];

}Structure;

union opp {

char ch;

int n;

char str[16];

}Union;

int main()

{

int structsize, unionsize;

structsize = sizeof(Structure);

unionsize = sizeof(Union);

printf("Structure variable took %d bytes\n", structsize);

printf("Union variable took %d bytes\n", unionsize);

}

Output:

Structure variable took 24 bytes

Union variable took 16 bytes

* Structure variable 24 byte জায়গা নিয়েছে, তার কারন হচ্ছে ৩ টার জন্যে ৮ বাইট

করে ৩\*৮ = ২৪ বাইট।

* Union Variable 16 byte জায়গা নিয়েছে, তার কারন হচ্ছে এই Union এর তিনটি

সদস্য একই মেমোরি শেয়ার করে, তাই তাদের মধ্যে সবচেয়ে বড় যে 16 byte এর

স্টিং তার জন্যে 16 byte জায়গা নিয়েছে।

14. Enum Erasion-1.

#include <stdio.h>

enum daysofweak {

Sat, Sun, Mon, Tue, Wed, Thr, Fri

};

int main()

{

enum daysofweak day;

day = Tue;

printf("Day is = %d\n", day);

}

Output:

Day is = 3

/\*Returns the value of the index of the day\*/

15. Enum Erasion-2.

#include <stdio.h>

enum daysofweak {

Sat, Sun, Mon, Tue, Wed, Thr, Fri

};

int main()

{

enum daysofweak day1, day2;

day1 = Tue;

day2 = Fri;

printf("Difference is = %d\n", day2-day1);

}

Output:

Difference is = 3

Recursion(রিকার্শন)

1(0). Some basic problem for understanding recursion.

1(1). Local and Global Variable.

1(2). Local and Global Variable.

2(1). Static variable.

2(2). Static variable.

3(1). Recursion basic problem.

3(2). Recursion basic problem.

3(3). Recursion basic problem.

3(4). Recursion basic problem.

3(5). Recursion basic problem.

4(1). Factorial using recursion.

4(2). Factorial using recursion.

5(1). Fibonacci number.

5(2). Fibonacci number.

1(0)

* Recursion is a process in which a function calls itself directly or indirectly.

#include <stdio.h>

int fun(int n) //caller

{

if (n == 1)

return 1;

else

return 1 + fun(n - 1); //calling

}

int main()

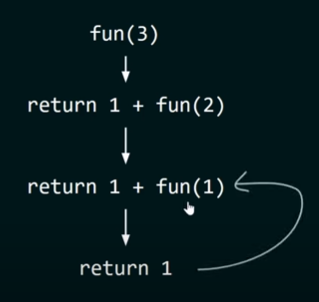
{

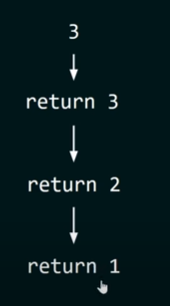
int n = 3;

printf("%d", fun(n));

}

Output: 3





* Determine the factorial of a number.

#include <stdio.h>

int fact(int n)

{

if (n == 1)

return 1;

else

return n \* fact(n - 1);

}

int main()

{

int n;

printf("Enter the number : ");

scanf("%d", &n);

printf("Factorial of %d is %d\n", n, fact(n));

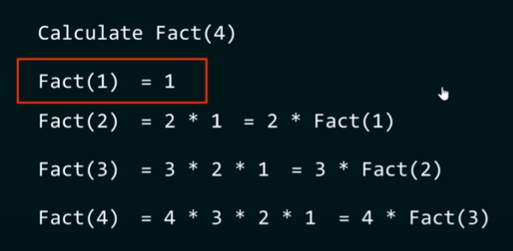
}

Output:

Enter the number : 4

Factorial of 4 is 24

**Fact(n) = n \* fact(n-1)**



* WAP to print numbers from 1 to 10 in such a way that when number is odd, add 1 and when number is even, subtract 1.

#include <stdio.h>

void odd();

void even();

int n = 1;

void odd()

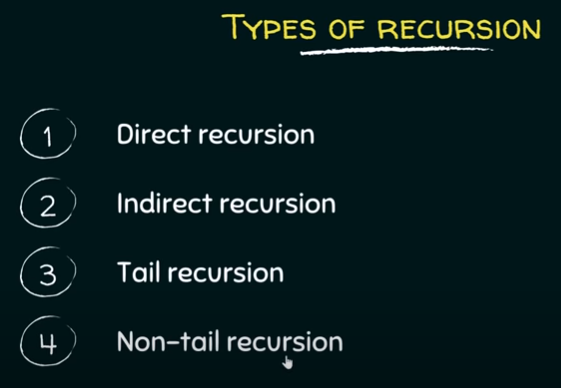
{

if (n <= 10) {

printf("%d ", n + 1);

n++;

even();



}

//return;

}

void even()

{

if (n <= 10) {

printf("%d ", n - 1);

n++;

odd();

}

//return;

}

int main()

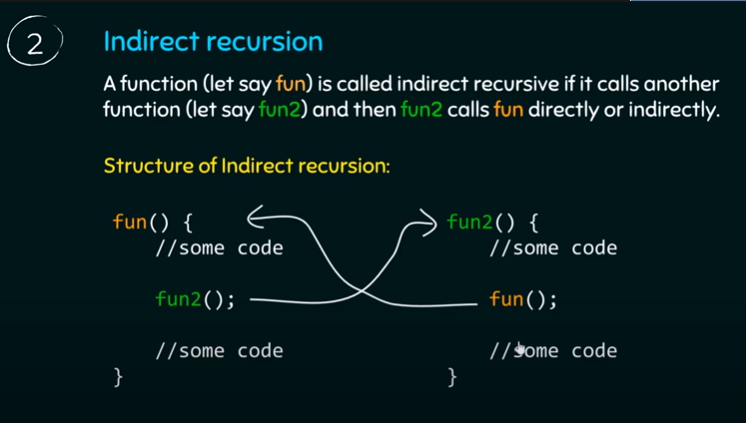
{

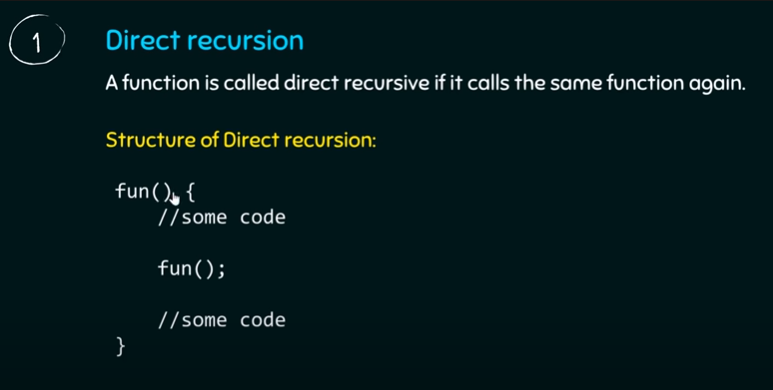
odd();

}

Output:

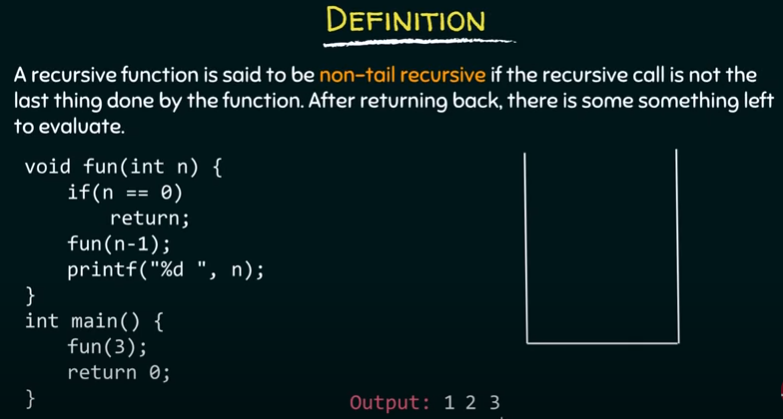
2 1 4 3 6 5 8 7 10 9

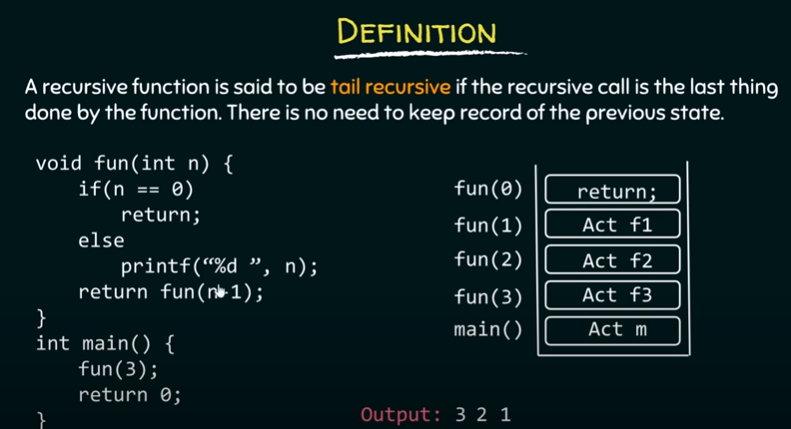




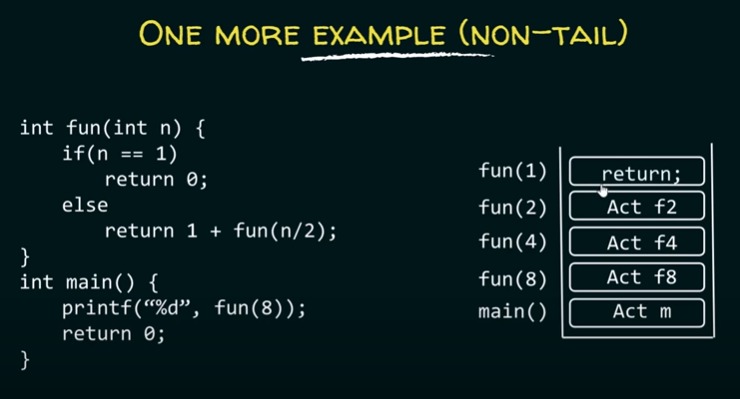
Tail recursive :

Non-tail recursive :





One more example of non-tail function.



**The output of this program is 3**

1 + 0 = 1

1 + 1 = 2

1 + 2 = 3

This 3 return to the main. That’s why the output is 3

return 1 + fun(n/2)

After returning back we need to evaluate this add that is why it is a non-tail recursion.

!(1). Local and Global Variable.

Output:

x = 0, y = 0

#include <stdio.h>

int x; /\*Global Variable\*/

int main()

{

int y; /\*Local Variable\*/

printf("x = %d, y = %d\n", x, y);

}

Remember:

* ফাংশনের প্যারামিটার হিসাবে যেসব ভেরিয়েবল পাঠানো হয় সেগুলো লোকাল ভেরিয়েবল।
* Global Variable declare করার সময় কোনো মান অ্যাসাইন না করলে অটোমেটিক ০ প্রিন্ট হয়।

1(2). Local and Global Variable.

Output:

Void function

x = 20, y = 10

Main function

x = 20, y = 5

#include <stdio.h>

int x = 1;

void myfunction(int y)

{

y = y \* 2;

x = x + 10;

printf("Void function\n");

printf("x = %d, y = %d\n", x, y);

}

int main()

{

int y = 5;

x = 10;

myfunction(y);

printf("Main function\n");

printf("x = %d, y = %d\n", x, y);

}

প্রোগ্রাম টা একটু মনোযোগ দিয়ে দেখো তাহলেই বুঝতে পারবা।

Remember:

* এখানে 2টি y-ই আলাদা, তারা দুটি আলাদা ফাংশনের লোকাল ভেরিয়েবল।

একটি ফাংশনের লোকাল ভেরিয়েবল আরেকটি ফাংশনের ভেরত থেকে এক্সেস করা যায় না।

* x যেহেতু গ্লোবাল ভেরিয়েবল তাই মেইন ফাংশনে x এর মান 20 প্রিন্ট হবে।

2(1). Static variable

Output:

a = 1

b = 1

#include <stdio.h>

int a;

static int b;

void function()

{

a = a + 1;

b = b + 1;

}

int main()

{

function();

printf("a = %d\n", a);

printf("b = %d\n", b);

}

Remember:

a and b কে গ্লোবাল স্পেস এ ডিক্লেয়ার করা হয়েছে। তাই এদেরকে ইনিশিয়ালাইজড

না করা সত্ত্বেও মান ০ হয়ে গেছে। এরপর ফাংশনের মধ্যে এদের মান ১ করে বাড়িয়ে দিলাম।

Global variable: যেকোনো ফাইলের যেকোনো ফাংশন থেকে এক্সেস করা যাবে।

Static global variable: শুধু মাত্র যে ফাইলে ডিক্লেয়ার করা হয়েছে ওই ফাইলের

ফাংশন গুলোর মধ্যে এক্সেস করা যাবে।

2(2). Static variable

Output:

a = 12, b = 12

a = 12, b = 14

a = 12, b = 16

#include <stdio.h>

void function()

{

int a = 10;

static int b = 10;

a = a + 2;

b = b + 2;

printf("a = %d, b = %d\n", a, b);

}

int main()

{

function();

function();

function();

}

b is static int variable.

a always print 12 but b print 12, 14, 16

This the rule of static variable.

3(1). Recursion basic problem.

Output:

I am learning recursion

I am learning recursion

I am learning recursion

I am learning recursion

#include <stdio.h>

void recursion(int count)

{

if (count == 5)

{

return;

}

printf("I am learning recursion\n");

recursion(count + 1);

return;

}

int main()

{

recursion(1);

}

Recursion হল একটি ফাংশন নিজেই নিজেকে কল করা। তবে কল করার কাজটি বুঝে-শুনে করতে হবে।

কখন recursion বন্ধ করতে হবে সেটি প্রোগ্রাম এর ভিতর বলে দিতে হবে।

3(2). Recursion basic problem.

Output:

Count = 1

Count = 2

Count = 3

Count = 4

Count = 5

#include <stdio.h>

void recursion(int count)

{

if (count > 5)

{

return;

}

printf("Count = %d\n", count);

recursion(count + 1);

}

int main()

{

recursion(1);

}

3(3). Recursion basic problem.

Output:

Count = 5

Count = 4

Count = 3

Count = 2

Count = 1

#include <stdio.h>

void recursion(int count)

{

if (count > 5)

{

return;

}

recursion(count + 1);

printf("Count = %d\n", count);

}

int main()

{

recursion(1);

}

কোনো ফাংশনের return type যদি void হয় তাহলে ফাংশনের শেষ লাইনে

return; না লিখলেও চলে। আপনা আপনি সেটি রির্টার করে। void মানে

কম্পাইলার তার কাছ থেকে কোনো কিছু আশা করে না। সে ঠিক ঠাক ফেরত আসলেই খুশি।

3(4). Recursion basic problem.

Output:

Count = 1

Count = 2

Count = 3

Count = 4

Count = 5

Count = 5

Count = 4

Count = 3

Count = 2

Count = 1

#include <stdio.h>

void recursion(int count)

{

if (count > 5)

{

return;

}

printf("Count = %d\n", count);

recursion(count + 1);

printf("Count = %d\n", count);

}

int main()

{

recursion(1);

}

3(5). Recursion basic problem.

Output:

Count = 1

Count = 2

Count = 3

Count = 4

Count = 5

#include <stdio.h>

void recursion()

{

static int count = 1;

if (count > 5)

{

return;

}

printf("Count = %d\n", count);

count = count + 1;

recursion();

}

int main()

{

recursion();

}

Static Variable ব্যবহার করেই ফাংশনটি কতবার চলবে তা নিয়ন্ত্রন করতে পারি।

সেক্ষেত্রে আর count নামক ভেরিয়েবল আমাদের ফাংশনের প্যারামিটার হিসাবে বারবার পাঠাতে হবে না।

4(1). Factorial using recursion

Output:

5

Factorial of 5 is 120

#include <stdio.h>

int factorial(int n)

{

if (n == 0)

return 1;

else

return n \* factorial(n - 1);

}

int main()

{

int n;

scanf("%d", &n);

if (n < 0)

{

printf("Undefined\n");

//return 0;

}

printf("Factorial of %d is %d\n", n, factorial(n));

}

4(2). Factorial using recursion

#include <stdio.h>

#include <string.h>

int call = 0;

int factorial(int n)

{

call = call + 1;

if (n == 0)

return 1;

else

return n \* factorial(n - 1);

}

int main()

{

int n;

scanf("%d", &n);

if (n < 0)

{

printf("Undefined\n");

//return 0;

}

printf("Factorial of %d is %d\n", n, factorial(n));

printf("Number of fucntion calls : %d\n", call);

}

Output:

5

Factorial of 5 is 120

Number of fucntion calls : 6

Factorial function-টি কতবার কল করা হয়েছে সেটি বের করার প্রোগ্রাম।

এটা বের করার জন্য একটি গ্লোবাল ভেরিয়েবল প্রয়োজন।

5(1). Fibonacci number

#include <stdio.h>

#include <string.h>

int call = 0;

int fibo(int n)

{

call = call + 1;

if (n == 1 || n == 2)

return 1;

else

return fibo(n - 1) + fibo(n - 2);

}

int main()

{

int n;

char str[3];

scanf("%d", &n);

if (n == 1) strcpy(str, "st");

else if (n == 2) strcpy(str, "nd");

else if (n == 3) strcpy(str, "rd");

else strcpy(str, "th");

printf("%d %s fibonacci number is = %d\n", n, str, fibo(n));

printf("Number of function calls = %d\n", call);

}

Output:

6

6 th fibonacci number is = 8

Number of function calls = 15

5(2). Fibonacci number

Output:

5

5 th fibonacci number is = 5

Number of function calls = 7

1 : 1

2 : 1

3 : 2

4 : 3

5 : 5

6 : 0

7 : 0

8 : 0

9 : 0

10 : 0

11 : 0

#include <stdio.h>

#include <string.h>

int call = 0;

int fib[50];

int fibo(int n)

{

call = call + 1;

if (fib[n] != 0)

return fib[n];

if (n == 1 || n == 2)

return fib[n];

else

return fib[n] = fibo(n - 1) + fibo(n - 2);

return fib[n];

}

int main()

{

int n;

char str[3];

fib[1] = 1;

fib[2] = 1;

scanf("%d", &n);

if (n == 1) strcpy(str, "st");

else if (n == 2) strcpy(str, "nd");

else if (n == 3) strcpy(str, "rd");

else strcpy(str, "th");

printf("%d %s fibonacci number is = %d\n", n, str, fibo(n));

printf("Number of function calls = %d\n", call);

for (n = 1; n < 12; n++) {

printf("%2d : %2d\n", n, fib[n]);

}

}

Bitwise operation and some extra problem

1. Time count.

2. Random Number.

3. Constant

4. Creating macro.

5. Enum basic problem

6. Bitwise not

7. Shifting right or left

8. Bitwise and

9. Bitwise or

10. Bitwise x-or (exclusive or )

11. Determine even and odd using bitwise operation

12. Uppercase and lowercase using bitwise operator.

13. কোনো একটি সংখ্যা ২ এর পাওয়ার কিনা তা নির্নয়।

14. কোনো ধনাত্নক পূনসংখ্যার বাইনারি রুপে মোট কয়টি বিট 1

15. দুটি সংখ্যা সমান কিনা তা নির্নয়।

1. Time count.

#include <stdio.h>

#include <time.h>

void fun(int x, int n)

{

x = n \* 2;

}

int main()

{

int i, j, x, n;

clock\_t starttime, endtime;

double timeelapsed;

starttime = clock();

n = 12345678;

for (i = 0; i < 1000000000; i++) {

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

x = n \* 2;

}

}

endtime = clock();

timeelapsed = (double)(endtime - starttime) / CLOCKS\_PER\_SEC;

printf("Time : %lf seconds\n", timeelapsed);

starttime = clock();

n = 12345678;

for (i = 0; i < 1000000000; i++) {

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

fun(x, n);

}

}

endtime = clock();

timeelapsed = (double)(endtime - starttime) / CLOCKS\_PER\_SEC;

printf("Time : %lf seconds\n", timeelapsed);

return 0;

}

Output:

Time : 12.627000 seconds

Time : 21.584000 seconds

2. Random Number.

Output:

2233

20874

28795

11503

23403

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int main()

{

time\_t t;

srand((unsigned)time(&t));

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

printf("%d\n", rand());

}

return 0;

}

* কোডটি রান করলে প্রতেকবার ভিন্ন ভিন্ন আউটপুট আসবে। Random number এর জন্যে stdlib.h হেডার ফাইলের ভেতরে rand() ফাংশনটি ব্যবহার করতে হবে।
* srand function-এ এমন সংখ্যা পাঠাতে হবে যেটি একেকবার একেকরকম হবে। আমরা যদি কম্পিউটার এর বর্তমান সময়(time(&t)) পাঠাই তাহলে কিন্তু একেকবার একেক সংখ্যা পাঠানোর কাজটি হয়ে যাবে।

3(1). Constant

Output: a = 50, b = 60

#include <stdio.h>

#define p 50

#define q 60;

int main()

{

int a = p;

int b = q

printf("a = %d, b = %d\n", a, b);

}

q - কে ডিফাইন করেছি 60; (সেমিকোলন সহ) তাই কম্পাইল হওয়ার সময় q এর বদলে 60; বসে যাবে। ফলে কোনো এরর হবে না।

3(2). Constant

Output:

Value of n = 1

Value of PI = 3.141593

#include <stdio.h>

#include <math.h>

#define n 1

#define PI (2 \* acos(0))

int main()

{

printf("Value of n = %d\n", n);

printf("Value of PI = %lf\n", PI);

}

PI-এর মান বের করার জন্য সবচেয়ে ভালো পদ্বতি এটা।

4(1). Creating macro.

#include <stdio.h>

#define MAX(a, b) (a>b? a:b)

int main()

{

int a = 10, b = 20;

double x = 5.543, y = 3.1245;

printf("Maximum of %d and %d is = %d\n", a, b, MAX(a, b));

printf("Maximum of %lf and %lf is = %lf\n", x, y, MAX(x, y));

}

Output:

Maximum of 10 and 20 is = 20

Maximum of 5.543000 and 3.124500 is = 5.543000

* Here we use ternary operator.
* Note: This macro is worked both int and double data type.
* It is good programing practice to use capital letter for constant(like MAX).

4(2). Creating macro.

Output:

Before swapping : a = 10, b = 20

After swapping : a = 20, b = 10

#include <stdio.h>

#define SWAP(a, b) { \

a ^= b; \

b ^= a; \

a ^= b; \

}

int main()

{

int a = 10, b = 20;

printf("Before swapping : ");

printf("a = %d, b = %d\n", a, b);

SWAP(a, b);

printf("After swapping : ");

printf("a = %d, b = %d\n", a, b);

}

If we want to make multiple line macro then we must follow this above procedure. And must use backslash sign(\) at the end of every statement.

5(1). Enum basic problem

#include <stdio.h>

enum color {red, green, blue};

int main()

{

enum color selectedcolor;

int num;

printf("Enter num 1 for red, 2 for green and 3 for blue : ");

scanf("%d", &num);

if (num == 1) selectedcolor = red;

else if (num == 2) selectedcolor = green;

else if (num == 3) selectedcolor = blue;

printf("Selected color : %d", selectedcolor);

}

Output:

Enter num 1 for red, 2 for green and 3 for blue : 3

Selected color : 2

This output looks like array index which is start from 0. If we enter 3 then the index of 3 is (3-1) = 2 and this is the output of selected color.

return 0 কথাটি চাইলে বাদ দেয়া যাবে। যদি এই লাইনটি না লিখা হয়

ডিফল্ট হিসাবে main() ফাংশন এক্সিকিউশনের শেষে 0 রির্টান করবে।

5(2). Enum basic problem

Output:

Value : 0

Value : 1

Value : 2

Value : 3

#include <stdio.h>

enum color {No\_color, red, green, blue};

int main()

{

printf("Value : %d\n", No\_color);

printf("Value : %d\n", red);

printf("Value : %d\n", green);

printf("Value : %d\n", blue);

}

5(3). Enum basic problem

#include <stdio.h>

enum color { No\_color = 0, red = 40, green = 44, blue = 80 };

int main()

{

printf("Value : %d\n", No\_color);

printf("Value : %d\n", red);

printf("Value : %d\n", green);

printf("Value : %d\n", blue);

}

Output:

Value : 0

Value : 40

Value : 44

Value : 80

5(4). Enum basic problem

#include <stdio.h>

enum day { saturday, sunday, monday, tuesday, wednesday, thrusday, friday };

int main()

{

enum day today;

today = saturday;

switch (today)

{

case saturday:

printf("Today is saturday\n");

break;

case sunday:

printf("Today is sunday\n");

break;

case monday:

printf("Today is monday\n");

break;

case tuesday:

printf("Today is tuesday\n");

break;

case wednesday:

printf("Today is wednesday\n");

break;

case thrusday:

printf("Today is thrusday\n");

break;

case friday:

printf("Today is friday\n");

break;

default:

printf("Enter a valid day\n");

break;

}

}

Output:

Today is Saturday

6. Bitwise not

#include <stdio.h>

int main()

{

char a, b;

a = 0;

b = ~a;

printf("a = %d, b = %d\n", a, b);

a = 1;

b = ~a;

printf("a = %d, b = %d\n", a, b);

}

Output:

a = 0, b = -1

a = 1, b = -2

NOTE:

When a = 0 then b = -1 ,cause

The binary value of 0 is = 00000000

If we reverse it then the value is = 11111111 (This is the value of -1)

So, that’s why when a = 0 then b = -1.

Similarly, when a = 1 then b = -2, cause

The binary value of 1 is = 00000001

If we reverse if then the value is = 11111110 (This is the value of -2)

So, that’s why when a = 1 then b = -2

The main work of bitwise not (~) operator is reverse the number.

(Like that: 0 to 1 and 1 to 0)

Remember: This is not the same as logical not(!) operator.

7. Shifting right or left

#include <stdio.h>

int main()

{

int n, x, m;

while (1)

{

/\*The program will be closed when you enter 0\*/

printf("Please enter your number : ");

scanf("%d", &n);

if (n == 0)

break;

printf("How many bits you want to shift left? ");

scanf("%d", &x);

m = n << x;

printf("Result : %d\n", m);

}

}

Output:

Please enter your number : 9

How many bits you want to shift left ? 1

Result : 18

Please enter your number : 2

How many bits you want to shift left ? 1

Result : 4

Please enter your number : 2

How many bits you want to shift left ? 2

Result : 8

Please enter your number : 0

NOTE:

<< : shift left

>> : shift right

8. Bitwise and

Output: 5 & 6 = 4

#include <stdio.h>

NOTE:

5 = 00000101

6 = 00000110

--------------

5&6 = 00000100

int main()

{

int n1 = 5, n2 = 6, n3;

n3 = n1 & n2;

printf("%d & %d = %d\n", n1, n2, n3);

}

5 এর একদম ডানদিকের বিটের সাথে 6 এর একদম ডানদিকের বিটের মধ্যে অপারেশন চালাবে।

9. Bitwise or

Output: 5 | 6 = 7

#include <stdio.h>

int main()

{

int n1 = 5, n2 = 6, n3;

n3 = n1 | n2;

printf("%d | %d = %d\n", n1, n2, n3);

}

আগেরটার মতো চিন্তা করো।

10. Bitwise xor(exclusive or)

Output: 5 ^ 6 = 3

#include <stdio.h>

**1 0 output 1**

**1 1 output 0**

**0 0 output 0**

int main()

{

int n1 = 5, n2 = 6, n3;

n3 = n1 ^ n2;

printf("%d ^ %d = %d\n", n1, n2, n3);

}

11. Determine even and odd using bitwise operation

Output:

4

This is even number

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

if (n & 1) printf("This is odd number\n");

else printf("This is even number\n");

}

NOTE:

4 & 1 = 00000100 & 00000001 = 00000000 = 0 (even)

5 & 1 = 00000101 & 00000001 = 00000001 = 1 (odd)

12. Uppercase and lowercase using bitwise operator.

#include <stdio.h>

char to\_upper(char ch)

{

return ch & 95;

}

char to\_lower(char ch)

{

return ch | 32;

}

int main()

{

char \*str = "aBcdefghijklmNOPQrstuvwxYZ";

for (int i = 0; i < 26; i++)

{

printf("Uppercase : %c\n", to\_upper(str[i]));

printf("Lowercase : %c\n", to\_lower(str[i]));

}

}

Output:

Uppercase: A

Lowercase : a

Uppercase : B

Lowercase : b

Uppercase : C

Lowercase : c

Uppercase : D

Lowercase : d

-------------

-------------

NOTE:

return ch & 95; Why we use this?

Look, first character of the str is a

a = 01100001

95 = 01011111

-------------------

a&95 = 01000001 (This is the value of A or 65)

13. কোনো একটি সংখ্যা ২ এর পাওয়ার কিনা তা নির্নয়।

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

if (n > 0 && (n & (n - 1)) == 0)

printf("Yes, %d is a power of 2\n", n);

else

printf("No, %d is not a power of 2\n", n);

}

Output:

16

Yes, 16 is a power of 2

NOTE:

Look 1, 2, 4, 8, 16 this is the number which is the power of 2.

The binary value of this number are:

1 = 1, 2 = 10, 4 = 100

মাত্র একটি বিট হলো ১

এখন যেই বিট ১ সেটাকে ০ করে দিয়ে তার ডান পাশের সবগুলো বিটকে যদি ১

করে দেই তাহলে কি আসে?

যেমনঃ 4 = 100 --> 4 = 011 (এটা হলো ৩ এর বাইনারি মান)

এই ৪ এবং ৩ এর মধ্যে bitwise and operation চালালে ফলাফল হবে ০

তাহলে n(4) এর মান যদি ০ না হয় এবং n&(n-1) এর মান যদি ০ হয় তাহলে

n(4) হবে ২ এর পাওয়ার।

14. কোনো ধনাত্নক পূনসংখ্যার বাইনারি রুপে মোট কয়টি বিট 1

Output:

4

Number of 1 : 1

#include <stdio.h>

int main()

{

int n, i;

scanf("%d", &n);

int count = 0;

while (n)

{

if (n & 1) {

count++;

}

n = n >> 1;

}

printf("Number of 1 : %d\n", count);

}

NOTE: The binary value of 4 is 100

So, the number of 1 is : 1

আমরা যদি ডানদিক থেকে ২য় বিটটি কি ০ নাকি ১ সেটি নির্নয় করতে যাই

তাহলে ১ কে এক ঘর left shift করে তার সঙ্গে সংখ্যাটির& করলেই হবে।

#include <stdio.h>

int main()

Output:

7

Number of 1 : 3

{

int n = 32;

int num, i;

int count = 0;

scanf("%d", &num);

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

{

if (num & (1 << i))

count++

}

printf("Number of 1 : %d\n", count);

}

int n = sizeof(int) \* CHAR\_BIT; #include <limits.h>

15. দুটি সংখ্যা সমান কিনা তা নির্নয়।

Output:

4 4

Numbers are equal

#include <stdio.h>

int main()

{

int n, m;

scanf("%d %d", &n, &m);

if (n ^ m)

printf("Numbers are not equal\n");

else

printf("Numbers are equal\n");

}

NOTE:

x-or এর ক্ষেত্রে নিয়ম হচ্ছে যদি সংখ্যা দুটি সমান হয় তাহলে আউটপুট ০

অ্যারেতে একটি সংখ্যা কেবল একবারই আছে।

বাকি সংখ্যা গুলো দুই বার করে আছে। যে সংখ্যাটি একবার আছে সেটি নির্নয় করতে হবে।

#include <stdio.h>

int main()

{

int array[] = { 1, 2, 3, 4, 1, 2, 3 };

int i, n = 7;

int result = array[0];

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

{

result = result ^ array[i];

}

printf("Result is : %d\n", result);

}

Output: Result is : 4

আমরা যদি এক্সর করি তাহলে কিন্তু ফলাফল হবে যে সংখ্যাটি একবার আছে সেটি।

কারন যেগুলো দুই বার করে আছে সেগুলোতে এক্সর চালালে ফলাফল হবে ০

৫২ টি প্রোগ্রামিং সমস্যা ও সমাধান

সমস্যা ২৭ – আর্মস্ট্রং সংখ্যা

সমস্যা ২৮ – এলোমেলো সংখ্যা

সমস্যা ২৯ – চিহ্ন পরিচয়

সমস্যা ৩০ – যোগ্য সংখ্যা-১

সমস্যা ৩১ – যোগ্য সংখ্যা-২

সমস্যা ৩২ – x এর গুনিতক

সমস্যা ৩৩ – বিভাজনসাধ্য ১

সমস্যা ৩৪ – বিভাজনসাধ্য ২

সমস্যা ৩৫ – বৃত্তের বাইরে

সমস্যা ৩৬ – শব্দ সাজানো

সমস্যা ৩৭ – সংখ্যা বিপর্যয়

সমস্যা ৩৮ – হীরক রাজ্য

সমস্যা ৩৯ - প্যালিনড্রোম

সমস্যা ৪০ – ধারার যোগফল-১

সমস্যা ৪১ – ধারার যোগফল-২

সমস্যা ৪২ – ধারার যোগফল-৩

সমস্যা ৪৩ – হিসাবকিতাব

সমস্যা ৪৪ – প্যাসকেলের ত্রিভুজ-১

সমস্যা ৪৫ – প্যাসকেলের ত্রিভুজ-২

সমস্যা ৪৬ – ত্রিভূজের ক্ষেত্রফল

সমস্যা ৪৭ – অ্যারের জোট

সমস্যা ৪৮ – নিখোঁজ সংখ্যা

সমস্যা ৪৯ – মৌলিক কি না

সমস্যা ৫০ – লেফট-রাইট

সমস্যা ৫১ – খোঁজ দ্যা সার্চ-১

সমস্যা ৫২ – খোঁজ দ্যা সার্চ-২

সমস্যা ১ – জোড়-বিজোর ১

সমস্যা ২ – জোড়-বিজোর ২

সমস্যা ৩ – অধোগামী সংখ্যা

সমস্যা ৪ - ভাজক

সমস্যা ৫ – বাক্স ১

সমস্যা ৬ – যোগফল নির্নয়

সমস্যা ৭ – সংখ্যা গণনা

সমস্যা ৮ – ছোট থেকে বড়

সমস্যা ৯ – পূর্ণবর্গ সংখ্যা

সমস্যা ১০ – রান রেট-১

সমস্যা ১১ – গৌণিক বা ফ্যাক্টরিয়াল

সমস্যা ১২ – ফ্যাক্টরিয়াল ১০০

সমস্যা ১৩ – টমি মিয়ার প্রোবাবিলিটি

সমস্যা ১৪ – অক্ষরের ঘনঘটা

সমস্যা ১৫ – অক্ষর গণনা

সমস্যা ১৬ – শব্দ বিপর্যয়

সমস্যা ১৭ – স্বরবর্ণ গণনা

সমস্যা ১৮ – স্বরবর্ণ – ব্যঞ্জনবর্ণ

সমস্যা ১৯ – শব্দ গণনা-১

সমস্যা ২০ - শব্দ গণনা-২

সমস্যা ২১ – উলটে দেখা

সমস্যা ২২ – মৌলিক সংখ্যা

সমস্যা ২৩ – বর্ণমালা থেকে সংখ্যা

সমস্যা ২৪ – একান্তর উপাদান

সমস্যা ২৫ – লগিষ্ট সাধারন গুন্নীয়ক(ল.সা.গু)

সমস্যা ২৬ – এলিয়েন গুপি

**সমস্যা ১ – জোড়-বিজোর ১**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int num;

scanf("%d", &num);

if (num % 2 == 0) printf("even\n");

else printf("odd\n");

}

}

Output:

3

100

even

0

even

1111

odd

এখানে num এর মান হতে পারবে 0-2147483647 পর্যন্ত।

তারমানে int টাইপের ভেরিয়েবল ব্যবহার করলেই হবে,

কারন int টাইপের ভেরিয়েবল এর সবোচ্চ মান 2147483647.

**সমস্যা ২ – জোড়-বিজোর ২**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

char num[101];

scanf("%s", num);

int length = strlen(num);

char lastcharacter = num[length - 1];

int lastnumber = lastcharacter - '0';

if (lastnumber % 2 == 0) printf("even\n");

else printf("odd\n");

}

}

Output:

3

1000

even

0

even

1111

Odd

* স্টিং এর ক্ষেত্রে & ব্যবহার না করাই উওম। scanf("%s", num);

তবে ব্যবহার করলেও সমস্যা নাই।

* এখানে স্টিং ব্যবহার করা হয়েছে কারন একটি সংখ্যায় সবোচ্চ 100 টি digit থাকতে পারে।

যা অনেক অনেক বড় সংখ্যা।

* int/sign int/unsign int- variable হচ্ছে 32 বিট বা 232 বা 4294967296.
* long long int- variable হচ্ছে 64 বিট। এতে 264 বা 19 টি digit রাখা যাবে।

**সমস্যা ৩ – অধোগামী সংখ্যা**

#include <stdio.h>

int main()

{

int count = 0;

for (int i = 1000; i >= 1; i--)

{

printf("%d\t", i);

count = count + 1;

if (count == 5) {

printf("\n");

count = 0;

}

}

}

Output:

1000 999 998 997 996

995 994 993 992 991

990 989 988 987 986

- - - - -

10 9 8 7 6

5 4 3 2 1

if (count == 5) বা, if (count % 5 == 0) এভাবে করলেও হতো।

**সমস্যা ৪ - ভাজক**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int x = 1; x <= t; x++)

{

int n;

scanf("%d", &n);

printf("Case %d:", x);

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

{

if (n % i == 0)

printf(" %d", i);

}

printf("\n");

}

}

Output:

3

6

Case 1 : 1 2 3 6

15

Case 2 : 1 3 5 15

23

Case 3 : 1 23

**সমস্যা ৫ – বাক্স ১**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int x = 1; x <= t; x++)

{

int n;

scanf("%d", &n);

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

{

for (int j = 1; j <= n; j++)

{

printf("\*");

}

printf("\n");

}

printf("\n");

}

}

Output:

3

1

\*

3

\*\*\*

\*\*\*

\*\*\*

5

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

\*\*\*\*\*

Second printf("\n");

এটাকে এভাবে লিখলেও হবে,

if (x != t)

printf("\n");

**সমস্যা ৬ – যোগফল নির্নয়**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int x = 1; x <= t; x++)

{

int n;

**বিকল্প পদ্বতিঃ**

int n;

scanf("%d", &n);

int sum = 0;

sum = sum + n % 10;

sum = sum + n / 10000;

printf("%d", sum);

scanf("%d", &n);

int temp = n;

int sum = 0, rem;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10000;

sum = sum + rem;

}

printf("%d\n", sum);

}

}

Output:

3

12345

6

56789

14

14310

1

যেহেতু ৫ টা সংখ্যা নিয়ে প্রোগ্রাম টা করা হয়েছে তাই 10000 দিয়েছি।

৩ টা সংখ্যা হলে 100 দিতাম। ৪ টা হলে 1000 দিতাম।

**সমস্যা ৭ – সংখ্যা গণনা**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int x = 1; x <= t; x++)

{

char str[100001];

gets(str);

***/\*scanf(" %[^\n]", str);\*/***

int i = 0;

int count = 0, flag = 0;

***/\*This loop will run, untill it reach at the end of the string\*/***

for (int i = 0; str[i] != '\0'; i++)

{

if (((str[i] >= '0' && str[i] <= '9') || str[i] == '-') && flag == 0) {

count = count + 1;

flag = 1;

}

else if (!((str[i] >= '0' && str[i] <= '9') || str[i] == '-') && flag == 1)

flag = 0;

}

printf("%d\n", count);

}

}

শব্দ গণনা

#include <stdio.h>

int main()

{

char str[151];

gets(str);

int word = 1;

for (int i = 0; str[i] != '\0'; i++)

{

if (str[i] == ' ' && str[i + 1] != ' ')

word++;

}

printf("%d", word);

}

Output:

Golam Kibria Ezaz

3

Output:

3

1 - 2 10000 - 50 20 7 455

7

9

1

- 98 876 65

3

getchar() is used for

flushing unwanted \n.

gets() use করলে getchar()

use করতে হবে।

**সমস্যা ৮ – ছোট থেকে বড়**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int n1, n2, n3;

scanf("%d %d %d", &n1, &n2, &n3);

int larger, middle, smaller;

if (n1 > n2 && n1 > n3) larger = n1;

else if (n2 > n1 && n2 > n3)larger = n2;

else larger = n3;

if (n1 < n2 && n1 < n3)smaller = n1;

else if (n2 < n1 && n2 < n3)smaller = n2;

else smaller = n3;

middle = n1 + n2 + n3 - larger - smaller;

printf("Case %d: %d %d %d\n", i, smaller, middle, larger);

}

}

Output:

3

3 2 1

Case 1 : 1 2 3

1 2 3

Case 2 : 1 2 3

10 5 6

Case 3 : 5 6 10

**সমস্যা – ছোট থেকে বড় (বিকল্প সমাধান)**

**/\*অ্যারের সাহায্যে\*/**

#include <stdio.h>

int main()

{

int array[10];

int n;

printf("How many number : ");

scanf("%d", &n);

printf("Enter %d number : \n", n);

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

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

}

int temp;

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

{

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

{

if (array[i] > array[j])

{

temp = array[i];

array[i] = array[j];

array[j] = temp;

}

}

}

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

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

}

}

Output:

How many number : 5

Enter 5 number :

34 12 67 99 35

12 34 35 67 99

যদি বড় থেকে ছোট ক্রমানুসারে সাজাতে বলতো তখন শুধু লাষ্টের লুপটা এইভাবে পরিবর্তন করলেই হবে

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

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

}

**সমস্যা ৯ – পূর্ণবর্গ সংখ্যা**

#include <stdio.h>

#include <math.h>

Output:

3

16

Yes

18

No

196

Yes

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int n, a;

scanf("%d", &n);

a = sqrt(n);

if (a \* a == n)

printf("Yes\n");

else printf("No\n");

}

}

#include <stdio.h>

#include <math.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

double n;

scanf("%lf", &n);

double a = sqrt(n);

if (ceil(a) == floor(a))

printf("Yes\n");

else printf("No\n");

}

}

* পূর্নবর্গ সংখ্যা হচ্ছে সেই সব সংখ্যা যাদের বর্গমূল একটি পূর্নসংখ্যা।
* অবশ্যই অবশ্যই math.h লিখতে হবে, না হলে wrong answer দেখাবে।
* ceil(2.1) return value 3.0
* ceil(2.8) return value 3.0
* floor(2.1) return value 2
* floor(2.8) return value 2

**সমস্যা ১০ – রান রেট-১**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int orakorcy, amrakorci, ballbaki;

scanf("%d %d %d", &orakorcy, &amrakorci, &ballbaki);

int ballkhelci = 300 - ballbaki;

double crr, rrr;

crr = ((double)amrakorci / (double)ballkhelci) \* 6;

rrr = ((double)(orakorcy - amrakorci + 1) / (double)ballbaki) \* 6;

if (amrakorci > orakorcy)

rrr = 0.0;

printf("%.2lf %.2lf\n", crr, rrr);

}

}

Output:

2

300 294 6

6.00 7.00

200 100 100

3.00 6.06

***crr এবং rrr বের করার সূত্র দুইটা অবশ্যই মনে রাখতে হবে।***

**সমস্যা ১১ – গৌণিক বা ফ্যাক্টরিয়াল**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int x = 1; x <= t; x++)

{

int n;

scanf("%d", &n);

long long int factorial = 1;

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

{

factorial = factorial \* i;

}

printf("%lld\n", factorial);

}

}

Output:

3

6

720

10

3628800

15

1307674368000

15 এর factorial অনেক বড় সংখ্যা তাই long long int ডাটা টাইপ ব্যবহার করা হয়েছে।

long long int- variable হচ্ছে 64 বিট। এতে 264 বা 19 টি digit রাখা যাবে।

**সমস্যা ১২ – ফ্যাক্টরিয়াল ১০০**

**/\*একটি সংখ্যার ফ্যাক্টরিয়াল এর শেষে কতটি ০ আছে তা বের করতে হবে\*/**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int n;

scanf("%d", &n);

int count = 0;

int x = 5, a;

while (n >= x)

{

a = n / x;

x = x \* 5;

count = count + 1;

}

printf("%d\n", count);

}

}

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

int count = 0, rem;

while (n > 0) {

rem = n % 10;

n = n / 10;

if (rem == 0) count++;

else break;

}

printf("%d\n", count);

}

Output:

বিকল্প সমাধান

3

6

1

15

1

100

2

এই প্রোগ্রামটিতে ৫ দিয়ে কম্পেয়ার করা হয়েছে।

কারন ৫০০ = ৫X১০X১০

তারমানে দুইটি ১০ আছে তাই ০ দুটি।

**সমস্যা ১৪ – অক্ষরের ঘনঘটা**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

char firstline[10001];

char secondline[2];

*/\*যেহেতু স্পেস থাকবে তাই এইভাবে স্কেন করাই উত্তম*

*অবশ্যই আগে একটা স্পেস দিতে হবে।\*/*

scanf(" %[^\n]", firstline);

scanf("%s", secondline);

int count = 0;

int length = strlen(firstline);

for (int i = 0; i < length; i++)

{

if (secondline[0] == firstline[i])

count++;

}

if (count > 0)

printf("Occurance of %c in %s = %d\n", secondline[0], firstline, count);

else

printf("%c is not found\n", secondline[0]);

}

}

scanf(" %[^\n]", s);

% এর আগে স্পেস দেয়ার কারন হলো আগের লাইনের শেষে যে

নিউলাইন ক্যারেক্টার(\n) আছে সেটি যেন এর ভেতর ঠুকে যায়।

Output:

2

hello world

l

Occurance of l in hello world = 3

golam kibria

a

Occurance of a in golam kibria = 2

**সমস্যা ১৫ – অক্ষর গণনা**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

char s[1001];

int count[26];

scanf(" %[^\n]", s);

int length = strlen(s);

memset(count, 0, sizeof count);

***/\*Syntax of memset (N, o, sizeof N)\*/***

for (int i = 0; i < length; i++)

{

if (s[i] >= 'a' && s[i] <= 'z')

count[s[i] - 'a']++;

}

for (int i = 0; i < 26; i++)

{

if (count[i] != 0)

printf("%c = %d\n", 'a' + i, count[i]);

}

printf("\n");

}

}

When i = 0

'a'+i = 97+0 = 97

So, %d = 97 but %c = 'a'

Output:

2

hello

e = 1

h = 1

l = 2

o = 1

baby

a = 1

b = 2

y = 1

Function memset() is a library function of string.h. It is used to fill a block of memory with given / particular value. It is used when you want to fill all or some of the blocks of the memory with a particular value.

**সমস্যা ১৬ – শব্দ বিপর্যয়**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int h = 0; h <= t; h++)

{

char str[1001];

gets(str);

***/\*scanf(" %[^\n]", str);\*/***

int length = strlen(str);

int temp = 0;

for (int i = 0; i <= length; i++)

{

if (str[i] == ' ' || str[i] == '\0')

{

if (temp != 0)

{

printf(" ");

}

int last = i - 1;

while (last >= temp)

{

printf("%c", str[last]);

last--;

}

temp = i;

}

}

printf("\n");

}

}

Output:

2

This is a test

sihT si a tset

Hello world

olleH dlrow

**সমস্যা ১৭ – স্বরবর্ণ গণনা**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int h = 1; h <= t; h++)

{

char str[1001];

gets(str);

int count = 0;

int length = strlen(str);

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

if (str[i] == 'a' || str[i] == 'e' || str[i] == 'i' || str[i] == 'o' || str[i] == 'u' || str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' || str[i] == 'U') {

count++;

}

}

printf("Number of vowels : %d\n", count);

}

}

Output:

2

i am a programmer

Number of vowels : 6

hello world

Number of vowels : 3

for (int i = 0; i < length; i++)

এই লাইন টা এইভাবে লিখলেও হতো,

for (int i = 0; str[i]!='\0'; i++)

**সমস্যা ১৮ – স্বরবর্ণ – ব্যঞ্জনবর্ণ**

#include <stdio.h>

#include <string.h>

int main()

{

int n;

scanf("%d", &n);

getchar();

for (int h = 1; h <= n; h++)

{

char str[1001];

gets(str);

char vowel[1001];

char consonent[1001];

int v = 0, c = 0;

for (int i = 0; str[i] != '\0'; i++)

{

if ((str[i] >= 'a' && str[i] <= 'z') || (str[i] >= 'A' && str[i] <= 'Z'))

{

if (str[i] == 'a' || str[i] == 'A' || str[i] == 'e' || str[i] == 'E' || str[i] == 'i' || str[i] == 'I' || str[i] == 'o' || str[i] == 'O' || str[i] == 'u' || str[i] == 'U') {

vowel[v] = str[i];

v++;

}

else {

consonent[c] = str[i];

c++;

}

}

}

***/\*Terminating string by NULL charecter\*/***

vowel[v] = '\0';

consonent[c] = '\0';

printf("%s\n", vowel);

printf("%s\n", consonent);

}

}

for (int i = 0; str[i] != '\0'; i++)

{ এর নিচে এই লাইন টা দিলেও হতো

if (str[i] == ' ') {

i++;

continue;

}

Output:

2

this is very easy

iieea

thssvrysy

it is a rainy sunday

iiaaiua

tsrnysndy

**সমস্যা ১৯ – শব্দ গণনা-১**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int i = 1; i <= t; i++)

{

char str[10001];

gets(str);

int length = strlen(str);

int word = 1;

for (int i = 0; i < length; i++)

{

if (str[i] == ' ' && str[i + 1] != ' ')

word++;

}

printf("Total word : %d\n", word);

}

}

Output:

2

Hello world

Total word : 2

Dhaka is the capital of bangladesh

Total word : 6

**সমস্যা ২০ - শব্দ গণনা-২**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int i = 1; i <= t; i++)

{

char str[10001];

gets(str);

int length = strlen(str);

int word = 1;

for (int i = 0; i < length; i++)

{

if (str[i] == ' ' && str[i + 1] != ' ')

word++;

}

printf("Total word : %d\n", word);

}

}

Output:

2

Hurrah!was heard on all sides

Total word : 6

Hello, I'm a broker

Total word : 4

**সমস্যা ২১ – উলটে দেখা**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int h = 1; h <= t; h++)

{

char str[1001];

gets(str);

int length = strlen(str);

for (int i = length - 1; i >= 0; i--)

{

printf("%c", str[i]);

}

printf("\n");

}

}

Output:

3

string

gnirts

programming

gnimmargorp

aruna

anura

**সমস্যা ২২ – মৌলিক সংখ্যা**

#include<stdio.h>

#include<math.h>

**/\*this function will check wheather it's prime or not.**

**if the number is prime then this function will simply return 1**

**else it will return 0 and we will add the return value to count\*/**

int primeornot(int x) **/\*x = a X b ; a ও b এর যেকোনো একটি সংখ্যা অবশ্যই**

**x এর বর্গমূলের সমান বা তার চেয়ে ছোট হবে\*/**

{

int i;

for (i = 2; i <= sqrt(x); i++)

{

if (x % i == 0) return 0;

}

return 1;

}

int main()

{

int t, i;

scanf("%d", &t);

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

{

int a, b;

scanf("%d %d", &a, &b);

int count = 0;

**//creating a loop from a to b**

while (a <= b)

{

**/\*1 and 0 is not prime number so**

**if a=1 or a=0 we won't check prime or not\*/**

if (a == 1 || a == 0)

{

a++;

continue;

}

**//adding returning value to countt**

count = count + primeornot(a);

a++;

}

printf("%d\n", count);

}

return 0;

}

**সমস্যা ২২ – মৌলিক সংখ্যা**

/\*আগের টার তুলনায এই পদ্বতিটা সহ্য\*/

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int low, high, i, flag;

printf("Enter two numbers : ");

scanf("%d %d", &low, &high);

printf("prime numbers : ");

int count = 0;

while (low <= high)

{

flag = 0;

if (low == 1 || low == 0) {

low++;

continue;

}

***/\*If low is a non-prime number, flag will be 1\*/***

for (i = 2; i <= low / 2; i++)

{

if (low % i == 0) {

flag = 1;

break;

}

}

if (flag == 0)

{

count++;

printf("%d ", low);

}

low++;

}

printf("\nSo, total prime number %d\n", count);

}

}

Output:

2

Enter two numbers : 1 5

prime numbers : 2 3 5

So, total prime number 3

Enter two numbers : 25 50

prime numbers : 29 31 37 41 43 47

So, total prime number 6

**সমস্যা ২৩ – বর্ণমালা থেকে সংখ্যা**

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int h = 1; h <= t; h++)

{

char str[101];

gets(str);

int length = strlen(str);

/\*int count = 0;\*/

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

if (str[i] >= 'A' && str[i] <= 'Z') {

printf("%d", str[i] - 64);

}

/\*if (str[i] == 'A' || str[i] == 'E' || str[i] == 'I' || str[i] == 'O' || str[i] == 'U')

count++;\*/

}

/\*printf("\nNumber of vowels : %d\n", count);\*/

printf("\n");

}

}

Output:

3

ABZ

1226

ZYB

26252

CODING

31549147

যদি বলতো কতগুলো vowel আছে তা নির্নয় করো

তখন কমেন্ট এর গুলো দিয়ে দিলেই হবে।

**সমস্যা ২৪ – একান্তর উপাদান**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int num[101];

int n;

scanf("%d", &n);

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

{

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

}

for (int i = 0; i < n; i = i + 2)

{

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

}

printf("\n");

}

}

Output:

2

5 1 2 3 4 5

1 3 5

10 1 4 55 66 22 0 76 11 23 78

1 55 22 76 23

**সমস্যা ২৫ – লগিষ্ট সাধারন গুন্নীয়ক(ল.সা.গু)**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int num1, num2, n1, n2, rem;

int lcm, gcd;

***/\*printf("Please enter two number : ");\*/***

scanf("%d %d", &num1, &num2);

n1 = num1;

n2 = num2;

while (n2 != 0)

{

rem = n1 % n2;

n1 = n2;

n2 = rem;

}

gcd = n1;

lcm = (num1 \* num2) / gcd;

***/\*printf("GCD = %d\n", gcd);\n\*/***

printf("LCM = %d\n", lcm);

}

}

Output:

2

30 15

LCM = 30

12 16

LCM = 48

**n1 % n2 = rem**

30 % 15 = 0

15 % 0(n2 is o, so the loop can not execute)

So, gcd = n1 = 15;

And, lcm = (30 \* 15) / 15 = 30;

**সমস্যা ২৬ – এলিয়েন গুপি**

/\*একদিনে তার সরবরাহকৃত খাবারের অর্ধেক খেয়ে ফেলে।

অথাৎ, যদি x পরিমাণ খাবার নিয়ে আসে তাহলে ১ দিনে খায় x/2.

প্রশ্ন হচ্ছে তাহলে তার খাবার শেষ করতে কতদিন লাগবে সেটা বের করা।\*/

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

double x;

scanf("%lf", &x);

int count = 0;

while (x > 1.0)

{

x = x / 2;

count++;

}

printf("%d days\n", count);

}

}

Output:

3

40

6 days

200

8 days

300

9 days

**সমস্যা ২৭ – আর্মস্ট্রং সংখ্যা**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int n;

scanf("%d", &n);

int rem, sum = 0;

int temp = n;

while (temp != 0)

{

rem = temp % 10;

temp = temp / 10;

sum = sum + rem \* rem \* rem;

}

if (sum == n)

printf("%d is an armstrong numebr\n", n);

else

printf("%d is not an armstrong number\n", n);

}

}

Output:

3

100

100 is not an armstrong number

153

153 is an armstrong numebr

371

371 is an armstrong numebr

Example of Armstrong number:

153 = 13+53+33

এখানে 3 নিয়েছি কারন 153 এর মধ্যে 3 টি অঙ্ক রয়েছে।

**সমস্যা ২৮ – এলোমেলো সংখ্যা**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int n;

scanf("%d", &n);

int max = -100;

int flag = 0;

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

{

int num;

scanf("%d", &num);

if (num >= max) max = num;

else {

flag = 1;

break;

}

}

if (flag == 0) printf("YES\n");

else printf("NO\n");

}

}

Output:

2

5

1 2 3 4 5

YES

10

1 2 3 4 5 6 7 9 10 8

NO

**সমস্যা ২৯ – চিহ্ন পরিচয়**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

char ch;

getchar();

ch = getchar();

if (ch >= 'a' && ch <= 'z')

printf("Lowercase Character");

else if (ch >= 'A' && ch <= 'Z')

printf("Uppercase Character");

else if (ch >= '0' && ch <= '9')

printf("Numerical Digit");

else

printf("Special Character");

printf("\n");

}

}

Output;

4

a

Lowercase Character

A

Uppercase Character

5

Numerical Digit

;

Special Character

**সমস্যা ৩০ – যোগ্য সংখ্যা-১**

* Perfect number/যোগ্য সংখ্যা?
* কোনো একটি সংখ্যা এবং ঐ সংখ্যাটির ভাজকগুলোর যোগফল যদি পরস্পর সমান হয় তাহলেই তাকে perfect number বলা যাবে।

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int num;

scanf("%d", &num);

int sum = 0;

for (int i = 1; i <= num / 2; i++)

{

if (num % i == 0)

sum = sum + i;

}

if (sum == num)

printf("YES, %d is perfect number\n", num);

else

printf("NO, %d is not a perfct number\n", num);

}

}

Output:

3

6

YES, 6 is perfect number

28

YES, 28 is perfect number

30

NO, 30 is not a perfct number

যদি (num<=264-1) বলে তখন unsign long long ব্যবহার করতে হবে।

এর formet specifier হলো %llu

An unsigned long long occupies 8 bytes of memory;

It stores an integer from 0 to 2 ^ 64 - 1,

**সমস্যা ৩১ – যোগ্য সংখ্যা-২**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int num;

scanf("%d", &num);

if (num >= 6)printf("6\n");

if (num >= 28)printf("28\n");

if (num >= 496)printf("496\n");

if (num >= 8128)printf("8128\n");

if (num >= 33550336)printf("33550336\n");

printf("\n");

}

}

Output:

2

100

6

28

500

6

28

496

* int num এর পরিবর্তে long long int num ব্যবহার করা যেতে পারে।

তখন, formet specifier হবে %lld

* 1-40000000 এর মধ্যে মাত্র পাঁচটি যোগ্য সংখ্যা আছে

সেগুলো হলো 6, 28, 496, 8128, 33550336

**সমস্যা ৩২ – x এর গুনিতক**

/\*x থেকে n পর্যন্ত x এর গুনিতক গুলো প্রিন্ট করতে হবে। যদি x>n হয় তাহলে invalid প্রিন্ট করতে হবে।\*/

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int x, n;

scanf("%d %d", &x, &n);

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

{

if (i % x == 0)

printf("%d ", i);

}

if (x > n) printf("Invalid\n");

printf("\n");

}

}

Output:

3

2 10

2 4 6 8 10

99 1000

99 198 297 396 495 594 693 792 891 990

10 2

Invalid

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

এই লুপ টা এইভাবেও লিখতে পারতাম

for (int i = x; i <= n; i = i + x)

**সমস্যা ৩৩ – বিভাজনসাধ্য ১**

/\*প্রোগ্রামটির আউটপুটে a থেকে b পর্যন্ত যতগুলো সংখ্যা c দ্বারা নিঃশেষে বিভাজ্য সেই সংখ্যা গুলো প্রিন্ট করতে হবে।\*/

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int a, b, c;

scanf("%d %d %d", &a, &b, &c);

for (int i = a; i <= b; i++)

{

if (i % c == 0)

printf("%d ", i);

}

printf("\n");

}

}

Output:

3

2 20 3

3 6 9 12 15 18

50 60 5

50 55 60

55 100 6

60 66 72 78 84 90 96

for (int i = a; i <= b; i++)

এই লুপটাকে এইভাবেও লিখা যেতো

for (int i = 1; i <= b; i = i + 1)

* Int দিয়ে করলেও কোড হয়ে যাবে কিন্ত যদি রেঞ্জ বেশি হয় সেক্ষেত্রে long দিয়ে করাটাই উত্তম।
* long এর formet specifier হলো %ld

**সমস্যা ৩৪ – বিভাজনসাধ্য ২**

/\*প্রোগ্রামটির আউটপুটে 1 থেকে c পর্যন্ত যতগুলো সংখ্যা a ও c দ্বারানিঃশেষে বিভাজ্য সেই সংখ্যা গুলো প্রিন্ট করতে হবে।\*/

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int a, b, c;

scanf("%d %d %d", &a, &b, &c);

for (int i = 1; i <= c; i++)

{

if (i % a == 0 && i % b == 0)

printf("%d ", i);

}

printf("\n");

}

}

Output:

3

2 3 50

6 12 18 24 30 36 42 48

3 5 50

15 30 45

5 6 100

30 60 90

**সমস্যা ৩৫ – বৃত্তের বাইরে**

#include <stdio.h>

#include <math.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

double x1, y1;

double r;

double x2, y2;

scanf("%lf %lf", &x1, &y1);

scanf("%lf", &r);

scanf("%lf %lf", &x2, &y2);

double distance = sqrt((x2 - x1) \* (x2 - x1) + (y2 - y1) \* (y2 - y1));

if (r >= distance)

printf("The point is inside the circle\n");

else

printf("The point is not inside the circle\n");

printf("\n");

}

}

Output:

2

1 1

4

10 - 14

The point is not inside the circle

1 1

8

5 6

The point is inside the circle

**/\*যেহেতু sqrt ফাংশনটি ব্যবহার করেছি তাই math.h নামক হেডার ফাইলটি ব্যবহার করা হয়েছে\*/**

**সমস্যা ৩৬ – শব্দ সাজানো**

/\*Alphabetical order\*/

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int n;

scanf("%d", &n);

char name[10][10], cname[10][10], temp[10];

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

{

scanf("%s", name[i]);

strcpy(cname[i], name[i]);

}

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

{

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

{

if (strcmp(name[i], name[j]) > 0)

{

strcpy(temp, name[i]);

strcpy(name[i], name[j]);

strcpy(name[j], temp);

}

}

}

printf("\n");

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

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

}

}

}

Output:

1

5

xray

apple

cat

bat

house

apple

bat

cat

house

xray

**সমস্যা ৩৭ – সংখ্যা বিপর্যয়**

**/\*একটি সংখ্যাকে উল্টে প্রিন্ট করা\*/**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int n;

scanf("%d", &n); //12

int sum = 0;

while (n != 0)

{

sum = sum \* 10; //0, 20

sum = sum + n % 10; //0+2=2, 20+1=21

n = n / 10; //1, 0

}

printf("%d\n", sum);

}

}

Output:

3

612

216

1431056

6501341

10041992

29914001

**সমস্যা ৩৮ – হীরক রাজ্য**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int n, a;

scanf("%d %d", &n, &a);

for (int row = 1; row <= n; row++)

{

for (int col = 1; col <= row; col++)

{

printf("%d ", a);

}

printf("\n");

}

for (int row = n - 1; row >= 1; row--)

{

for (int col = 1; col <= row; col++)

{

printf("%d ", a);

}

printf("\n");

}

printf("\n");

}

}

5 2

2

2 2

2 2 2

2 2 2 2

2 2 2 2 2

2 2 2 2

2 2 2

2 2

2

Output:

2

3 1

1

1 1

1 1 1

1 1

1

**/\*পুরোটাই প্যার্টান টাইপ-৩ এর মতো সেম\*/**

**সমস্যা ৩৯ - প্যালিনড্রোম**

/\*কোনো একটি সংখ্যা প্যালিনড্রোম কিনা তা নির্ণয়\*/

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n); //111

int temp, rem, sum = 0;

temp = n;

while (temp != 0)

{

rem = temp % 10; //1, 1, 1

temp = temp / 10; //11, 1, 0

sum = sum \* 10 + rem; //1, 11, 111

}

if (sum == n)

printf("Yes! It is palindrome\n");

else

printf("Sorry! It is not palindrome\n");

}

Output:

121

Yes!It is palindrome

123

Sorry!It is not palindrome

আবার শুধু যদি sum print করতাম তখন reverse হয়ে যেত।

**সমস্যা ৩৯ – প্যালিনড্রোম**

/\*কোনো একটি স্টিং প্যালিনড্রোম কিনা তা নির্ণয়\*/

#include <stdio.h>

#include <string.h>

int main()

{

int t;

scanf("%d", &t);

getchar();

for (int h = 1; h <= t; h++)

{

char str[1001];

gets(str);

int length = strlen(str);

char cstr[1001];

cstr[length] = '\0';

int i = 0;

*/\*সুবিধার জন্য আলাদা একটা ভেরিয়েবল এ রেখে দিলাম\*/*

int a = length - 1;

while (a >= 0)

{

cstr[a] = str[i];

a--; i++;

}

***/\*Checking palindrome or not\*/***

if (strcmp(str, cstr) == 0)

printf("Yes. It is palindrome\n");

else

printf("No. It is not palindrome\n");

}

}

Output:

3

wow

Yes.It is palindrome

string

No.It is not palindrome

civic

Yes.It is palindrome

**সমস্যা ৪০ – ধারার যোগফল-১**

/\*x0+x1+x2+x3+x4+---+xk.x এবং k এর মান দেয়া থাকবে। অর্থাৎ লুপটা চলবে 0 থেকে k পর্যন্ত।\*/

#include <stdio.h>

#include <math.h>

int main()

{

int t;

scanf("%d", &t);

for (int i = 1; i <= t; i++)

{

int x, k;

scanf("%d %d", &x, &k);

int sum = 0;

for (int i = 0; i <= k; i++)

{

sum = sum + pow(x, i);

}

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

}

}

Output:

3

2 10

Result is = 2047

5 10

Result is = 12207031

10 5

Result is = 111111

/\*result ভেরিয়েবল এর ডাটা টাইপ long long ব্যবহার করলেও হবে। long long ডাটা টাইপের ফরমেট স্পেসিফায়ার হলো %lld \*/

**সমস্যা ৪১ – ধারার যোগফল-২**

+ + -------+

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int n;

scanf("%d", &n);

double sum = 0;

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

{

int factorial = 1;

for (int j = 1; j <= i; j++) {

factorial = factorial \* j;

}

sum = sum + (double)i / factorial;

}

printf("%.4lf\n", sum);

}

}

Output:

3

5

2.7083

8

2.7183

10

2.7183

/\*এইভাবে করলেও হবে তবে ফাংশন ইউজ করে করাটা বেশি যুক্তিপূর্ন এবং ফেক্সিবল।\*/

**সমস্যা ৪১ – ধারার যোগফল-২**

#include <stdio.h>

long long fact(int n)

{

long long sum = 1;

int i;

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

{

sum \*= i;

}

return sum;

}

int main()

{

int t, i;

scanf("%d", &t);

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

{

int number;

scanf("%d", &number);

int j = 1;

double sum = 0;

while (j <= number)

{

sum += (double)j / fact(j);

j++;

}

printf("%0.4lf\n", sum);

}

return 0;

}

15! বেশ বড় সংখ্যা তাই long long int ব্যবহার করা হয়েছে।

এর formet specifier হলো %lld

Output:

3

5

2.7083

8

2.7183

10

2.7183

**সমস্যা ৪২ – ধারার যোগফল-৩**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int n;

scanf("%d", &n);

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

{

if (i == 1) printf("2 + ");

else if (i == 0)printf("1\n");

else printf("2^%d + ", i);

}

}

}

Output:

3

5

2 ^ 5 + 2 ^ 4 + 2 ^ 3 + 2 ^ 2 + 2 + 1

8

2 ^ 8 + 2 ^ 7 + 2 ^ 6 + 2 ^ 5 + 2 ^ 4 + 2 ^ 3 + 2 ^ 2 + 2 + 1

10

2 ^ 10 + 2 ^ 9 + 2 ^ 8 + 2 ^ 7 + 2 ^ 6 + 2 ^ 5 + 2 ^ 4 + 2 ^ 3 + 2 ^ 2 + 2 + 1

**সমস্যা ৪৩ – হিসাবকিতাব**

/\*pq mod c এর মান নির্নয় করার প্রোগ্রাম লিখতে হবে।\*/

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int p, q, c;

scanf("%d %d %d", &p, &q, &c);

int sum = 1;

for (int i = 1; i <= q; i++)

{

sum = (sum \* p) % c; /\*ডিরেক্ট করে ফেললাম\*/

}

***/\*int result = sum % c;\*/***

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

}

}

Output:

3

2 3 3

Result = 2

2 10 5

Result = 4

50 2 3

Result = 1

int power = pow(p, q);

int a = power % c;

printf("%d", a);

এইভাবে করলেও হবে। উপরে অবশ্যই math.h লিখতে হবে।

**সমস্যা ৪৪ – প্যাসকেলের ত্রিভুজ-১**

#include <stdio.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

int row;

scanf("%d", &row);

for (int i = 1; i <= row; i++)

{

int a = 1;

for (int j = 1; j <= i; j++)

{

printf("%d ", a);

a = a \* (i - j) / j;

}

printf("\n");

}

printf("\n");

}

}

Output:

2

3

1

1 1

1 2 1

4

1

1 1

1 2 1

1 3 3 1

3 এর জন্য 3 টা রো এর পরিবর্তে যদি 4 টা রো

দিতে চাই তখন শুধু i<=row+1 করে দিলেই হবে।

**সমস্যা ৪৫ – প্যাসকেলের ত্রিভুজ-২**

#include <stdio.h>

int main()

{

int row;

scanf("%d", &row);

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

{

for (int j = 0; j <= row - i; j++)

{

printf(" "); /\*2 space\*/

}

int a = 1;

for (int k = 0; k <= i; k++)

{

if (i == 0 || k == 0)a = 1;

else a = a \* (i - k + 1) / k;

printf("%4d", a);

}

printf("\n");

}

}

Output:

8

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1 6 15 20 15 6 1

1 7 21 35 35 21 7 1

**সমস্যা ৪৬ – ত্রিভূজের ক্ষেত্রফল**

#include <stdio.h>

#include <math.h>

int main()

{

int t;

scanf("%d", &t);

for (int h = 1; h <= t; h++)

{

double a, b, c, s, area;

scanf("%lf %lf %lf", &a, &b, &c);

s = (a + b + c) / 2;

area = sqrt(s \* (s - a) \* (s - b) \* (s - c));

printf("Area = %.3lf\n", area);

}

}

Output:

3

24 30 18

Area = 216.000

13 18 15

Area = 95.917

20 20 20

Area = 173.205

ত্রিভূজের তিন বাহুর দৈর্ঘ্য দেওয়া থাকলে ক্ষেত্রফল নির্নয় করার সূত্র হলোঃ

অর্ধপরিসীমা, s =

Area = sqrt(s x (s-a) x (s-b) x (s-c)

**সমস্যা ৪৭ – অ্যারের জোট**

#include <stdio.h>

int main()

{

int f, i;

scanf("%d", &f);

int array1[101];

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

{

scanf("%d", &array1[i]);

}

int s, j;

scanf("%d", &s);

int array2[101];

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

{

scanf("%d", &array2[j]);

}

int l;

int add[101];

for (i = 0, j = 0, l = 0; i < f && j < s; l++)

{

if (array1[i] > array2[j])

{

add[l] = array2[j];

j++;

}

else

{

add[l] = array1[i];

i++;

}

}

while (i < f)

{

add[l] = array1[i];

l++; i++;

}

while (j < s)

{

add[l] = array2[j];

l++; j++;

}

for (int a = 0; a < f + s; a++) {

printf("%d ", add[a]);

}

}

/\*টেষ্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজিবি হয়ে যাবে বলে আর দিলাম না\*/

Output:

7

10 20 30 40 50 60 70

5

11 22 33 44 55

10 11 20 22 30 33 40 44 50 55 60 70

**সমস্যা ৪৮ – নিখোঁজ সংখ্যা**

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

int array[101];

for (int i = 0; i < n - 1; i++)

{

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

}

for (int a = 1; a <= n; a++)

{

int count = 0;

for (int i = 0; i < n - 1; i++) {

if (a == array[i])

count = 1;

}

if (count == 0) {

printf("Missing number is = %d\n", a);

break;

}

}

}

Output:

7

2 1 4 6 5 3

Missing number is = 7

/\*টেষ্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজিবি হয়ে যাবে বলে আর দিলাম না\*/

**সমস্যা ৪৯ – মৌলিক কি না**

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

int count = 0;

if (n == 2)

printf("It's a prime number\n");

else

{

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

if (n % 2 == 0) {

count++;

break;

}

}

if (count == 0)

printf("This is prime number\n");

else

printf("This is not a prime number\n");

}

}

Output:

7

This is prime number

/\*টেষ্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজিবি হয়ে যাবে বলে আর দিলাম না\*/

**সমস্যা ৫০ – লেফট-রাইট**

#include <stdio.h>

int main()

{

char str[51];

gets(str);

int length = strlen(str);

for (int i = 0; i < length; i++)

{

if (str[i] == 'L')

str[i] = str[i - 1];

else if (str[i] == 'R')

str[i] = str[i + 1];

}

printf("%s\n", str);

}

Output:

oL7

oo7

/\*টেষ্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজিবি হয়ে যাবে বলে আর দিলাম না অবশ্যই getchar() ইউজ করতে হবে \*/

**সমস্যা ৫১ – খোঁজ দ্যা সার্চ-১**

/\*দুটি স্টিং দেওয়া থাকবে যার দ্বিতীয়টি প্রথমটির সাবস্টিং। খুঁজে বের করতে হবে প্রথমটিতে সাবস্টিংটি প্রথম কোথা হতে শুরু হয়েছে।\*/

#include <stdio.h>

#include <string.h>

int main()

{

char str1[131];

char str2[131];

scanf("%s", &str1);

scanf("%s", &str2);

int length1 = strlen(str1);

int length2 = strlen(str2);

for (int i = 0; i <= length1 - length2; i++)

{

if (strncmp(str1 + i, str2, length2) == 0) {

printf("%d\n", i);

break;

}

}

}

Output:

banana ana

1

foorbar foorbar

0

/\*টেষ্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজিবি হয়ে যাবে বলে আর দিলাম না।

অবশ্যই getchar() ইউজ করতে হবে\*/

**সমস্যা ৫২ – খোঁজ দ্যা সার্চ-২**

/\*দুটি স্টিং দেওয়া থাকবে যার দ্বিতীয়টি প্রথমটির সাবস্টিং। খুঁজে বের করতে হবে প্রথমটিতে সাবস্টিংটি মোট কত বার আছে।\*/

#include <stdio.h>

#include <string.h>

int main()

{

char str1[131];

char str2[131];

scanf("%s", &str1);

scanf("%s", &str2);

int length1 = strlen(str1);

int length2 = strlen(str2);

int count = 0;

for (int i = 0; i <= length1-length2; i++)

{

if (strncmp(str1 + i, str2, length2) == 0) {

count++;

}

}

printf("%d\n", count);

}

Output:

banana ana

2

foorbar foorbar

1

/\*টেষ্ট কেস টা ইচ্ছে মতো ধরে নিও, বেশি হাবিজিবি হয়ে যাবে বলে আর দিলাম না।

অবশ্যই getchar() ইউজ করতে হবে\*/

**সমস্যা-৫১ এবং ৫২ বিকল্প নিয়ম**

#include <stdio.h>

#include <string.h>

int main()

{

char str1[131];

char str2[131];

scanf("%s", &str1);

scanf("%s", &str2);

int length1 = strlen(str1);

int length2 = strlen(str2);

int i, j, count=0;

for (i = 0; i <= length1 - length2; i++)

{

if (str2[0] == str1[i])

{

for (j = 1; j < length2; j++)

{

if (str2[j] != str1[i + j])

break;

}

if (j == length2)

{

printf("%d\n", i);

break;

***/\*count++;\*/***

}

}

}

***/\*printf("%d\n", count);\*/***

}

৫১ এর ক্ষেত্রে printf হবে আর ৫২ এর ক্ষেত্রে count হবে।

Noel Kalicharan

1. WAP to prints the area of the square.

2.1 Print a message.

3. Escape sequence.

4. WAP to print the sum of two numbers.

5. Sum, sub, multiple, dividend, modulus of two number.

6. Print an integer using a field width.

7. Print a floating - point number.

8. Print a double number.

9. Expression with integer and floating - point values.

10. Assigning double / float to int.

11.1 Print a simple string.

12. Copy a string.

13. Joining two string or string concatenation.

14. Print some number.

15. Print some number using field width.

16. Read a string and print it.

17. WAP to take three integer and print their average.

18. WAP to take a whole number and print its square.

19. WAP to read the data from the customer.

20. Tickets are sold in three categories.

21. Print job charge based on hours worked and cost of parts

22. Find the sum of two length.

23. Request three marks. Print their average and pass / fail.

24. Request a score and prints letter grade.

25. Calculate pay.

26. Request three sides and determine types of triangle.

27. Print the sum of several numbers entered by user.

28. Print the sum and count of several numbers entered by user.

29. Print the sum and average entered by user.

30. Find the HCF of two numbers entered by user.

31. Increment and Decrement operator.

32. Find the largest of a set of numbers entered by user.

33. Find the smallest of a set of numbers entered by user.

34. Print five times a letter using loop.

35. Print many times of a letter entered by the user.

36. Multiplication table.

37. Multiplication table (From 🡪 To).

38. Temperature convention menu.

39. Expressive power of for.

40. Determine HCF of two number using do…while loop.

41. Interest at the bank.

42. Read and print character.

43. Read a character and print its code and the value of EOF.

44. Print a character and its code.

45. Character in arithmetic expression.

46. Uppercase to lowercase.

47. Lowercase to uppercase.

48. Read and print the first three character in the data.

49. Read and print the first non - blank character in the data.

50. Print all character before the first blank in the data.

51. Print the first non - blank character and count number of non - blank.

52. Count the number of characters of the input line.

53. Count the number of character and blank in the input line.

54. Read a line of data and find the largest character.

55. Read character and digit and print only digit.

56. WAP to show how skip lines fits into a complete program.

57. Finding largest number between two number. Using function.

58. Finding largest number between two number. Using function.

59. Largest character between the two character. Using function.

60. Print the day.

61. Print the day using function.

62. Determine HCF of two number using function.

63. Using HCF to find LCM.

64. Determine factorial using function.

65. Factorial from 0 to 7.

66. Combinations.

67. Print job charge based on hours worked and cost of parts.

68. Calculate pay using function.

69. Sum of exact divisors.

70. Sum of exact divisor using function.

71. Perfect number between 1 to 10000.

72. Some character function.

73. Position of a letter in the alphabet.

74. Take some digit and non - digit number and print only digit number.

75. Find the sum of two length given in meters and centimeters.

76. Find average and difference from average.

77. Letter frequency count.

78. Sum of number using array.

79. Find first alphabetical word before first blank space.

80. Counting number of spaces.

81. Reverse the character in a string.

82. Check a string palindrome or not.

83. Check a string palindrome or not using function.

84. A better palindrome function.

85. Print a day using string.

86. Find maximum number and its position using array.

87. Find maximum number by array using function.

88. Find small number by array using function.

89. Print a desire number from an array index.

90. Print a desire number from an array index using function.

91. Ascending and Descending order.

92. Print some string in alphabetical order.

93. Reverse a string.

94. Reverse a string by alphabetical order.

95. Sorted some number.

96. Sort parallel array

97. Binary search.

98. Binary search using function.

99. Frequency of a character.

100. Merge Sorted List.

101. Merge Sorted List (From the user)

102. Merge Sorted List (Using Function)

103. Printing date using structure.

104. Printing date using structure from the user.

105. printing date using structure by typedef

106. Printing month, date and year by structure using character array.

107. Array of structure.

108. Search an array of structure.

109. Sort an array of structure.

110. Read, Search and Sort a Structure.

111. Nested Structure.

112. Structure to represent a Fraction.

113. Manipulate Fraction by structure.

114. Pass Structures to Function.

1. WAP to prints the area of the square. p7

Output:

Enter length of side = 12

Area of square is = 144

#include <stdio.h>

int main()

{

int a, s;

printf("Enter length of side = ");

scanf("%d", &s); ***//storage length in s***

a = s \* s;

printf("\nArea of square is = %d\n", a);

return 0;

}

2.1 Print a message. P12

Output:

Welcome to Bangladesh

#include <stdio.h>

int main()

{

printf("Welcome to Bangladesh");

}

2.2 Print a message. P15

Output:

My name is Kibria

I live in Dhaka – Bangladesh

#include <stdio.h>

int main()

{

printf("My name is Kibria\n");

printf("\nI live in Dhaka-Bangladesh\n");

}

3. Escape sequence. P16

#include <stdio.h>

int main()

{

//escape sequence

printf("\"Dhaka is the capital of Bangladesh\"\n");

printf("Golam \\Kibria\n");

}

Output:

"Dhaka is the capital of Bangladesh"

Golam \Kibria

4. WAP to print the sum of two numbers. P19

Output:

14 + 26 = 40

#include <stdio.h>

int main()

{

int a, b, sum;

a = 14;

b = 26;

sum = a + b;

printf("%d + %d = %d\n", a, b, sum);

return 0;

}

5. Sum, sub, multiple, dividend, modulus of two number. P31

#include <stdio.h>

int main()

{

int a = 15;

int b = 24;

printf("%d %d\n", b - a + 7, b - (a + 7));

printf("%d %d\n", b - a - 4, b - (a - 4));

printf("%d %d\n", b % a / 2, b % (a / 2));

printf("%d %d\n", b \* a + 2, b \* (a + 2));

printf("%d %d\n", b / 2 \* a, b / (2 \* a));

return 0;

}

Output:

16 2

5 13

4 3

362 408

180 0

6. Print an integer using a field width. P33

Output:

9876

- 3

501

9876

-3

501

#include <stdio.h>

int main()

{

int a = 9876;

int b = -3;

int c = 501;

printf("%d\n", a);

printf("%d\n", b);

printf("%d\n", c);

printf("\n\n");

printf("%5d\n", a);

printf("%5d\n", b);

printf("%5d\n", c);

}

7. Print a floating-point number. P35

#include <stdio.h>

int main()

{

float a = 419.56;

***/\*যদি .5678 থাকতো এবং নিচে .2f থাকতো তখন .57 প্রিন্ট হতো\*/***

float b = -8.70;

float c = 3.25;

printf("%6.2f\n", a);

printf("%6.2f\n", b);

printf("%6.2f\n", c);

***//printf("6.2f \n6.2f \n6.2f \n", a, b, c);***

return 0;

}

Output:

419.56

-8.70

3.25

8. Print a double number. P36

#include <stdio.h>

int main()

{

***/\*we can store a double value in a double variable***

***and a float value in a float variable***

***if we assign a double to a float some precision may be lost\*/***

double d = 987.654321;

double x = d;

printf("%lf\n", x);

return 0;

}

Output:

987.654321

9. Expression with integer and floating-point values. P37

Output:

3.800000

42.059162

3.800000

#include <stdio.h>

int main()

{

double n = 19;

int a = 5;

double result = n / a;

double c = 4.68;

float d = 8.987;

double result2 = c \* d;

double x = 19;

double y = 5;

double result3 = x / y;

printf("%lf\n", result);

printf("\n%lf\n", result2);

printf("\n%lf\n", result3);

return 0;

}

10.1 Assigning double/float to int. p37

Output:

1.400000

#include <stdio.h>

int main()

{

int a = 7, b = 5;

float x;

x = (float)a / b;

***/\*or, a / (float)b***

***or, (float)a/ (float)b\*/***

printf("%f\n", x);

}

10.2 Assigning double/float to int. p38

Output:

987

#include <stdio.h>

int main()

{

***/\*when we assign a double value to an int***

***the factorial part is dropped\*/***

double d = 987.654321;

int n = d;

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

return 0;

}

11.1 Print a simple string. P38

#include <stdio.h>

int main()

{

***/\*a long string can be broke up into pieces.***

***when the program is compiled, c will join***

***the pieces and making one string\*/***

printf("My name is "

"Golam Kibria. "

"I live in Dhaka\n");

}

Output:

My name is Golam Kibria.I live in Dhaka

11.2 Print a simple string. P39

Output:

Hello, Golam Kibria

#include <stdio.h>

int main()

{

char name[50] = "Golam Kibria";

printf("Hello, %s\n", name);

}

12. Copy a string. P40

Output: Hello, Golam Kibria

#include <stdio.h> ***//needed for printf***

#include <string.h> ***//needed for strcpy***

int main()

{

char name[50];

strcpy(name, "Golam Kibria");

printf("Hello, %s\n", name);

}

13. Joining two string or string concatenation. P40

Output: Alice in Bangladesh

#include <stdio.h>

#include <string.h>

int main()

{

char name[30] = "Alice";

char last[30] = "Bangladesh";

strcat(name, " in ");

strcat(name, last);

printf("%s\n", name);

}

14. Print some number. P42

#include <stdio.h>

int main()

{

int a = 13;

int b = a + 12;

printf("%d %d\n", a, b);

int c = a + b;

a = a + 11;

printf("a = %d b = %d c = %d\n", a, b, c);

}

Output:

13 25

a = 24 b = 25 c = 38

15. Print some number using field width. P43

Output:

75

75 //3 space

75

00075

#include <stdio.h>

int main()

{

int num = 75;

printf("%d\n", num);

printf("%5d\n", num);

printf("%-5d\n", num);

printf("%05d\n", num);

}

16. Read a string and print it. P54

#include <stdio.h>

int main()

{

char name[50];

printf("Hi, what's your name? ");

gets(name);

printf("Delighted to meet you, %s\n", name);

}

Output:

Hi, what's your name? Kibria

Delighted to meet you, Kibria

17. WAP to take three integer and print their average. P55

#include <stdio.h>

int main()

{

int a, b, c;

double avg;

printf("Enter 3 integer = ");

scanf("%d %d %d", &a, &b, &c);

avg = (a + b + c) / 3.0;

//or, avg = (double)(a+b+c)/3;

printf("The average is = %.2lf\n", avg);

}

Output:

Enter 3 integer = 23 7 10

The average is = 13.33

18. WAP to take a whole number and print its square. P56

#include <stdio.h>

int main()

{

int num;

scanf("%d", &num);

printf("\nSquare of %d is %d\n", num, num \* num);

}

Output:

4

Square of 4 is 16

19. WAP to read the data from the customer, calculate the interest and service charge and print the customer’s name, average balance, interest and service charge.

Interest = 6% of average balance.

Service charge = 50% per transaction. P57

#include <stdio.h>

int main()

{

char name[50], aNum[50];

double avg;

int tran;

printf("Name? ");

gets(name);

printf("Accounr number? ");

gets(aNum);

printf("Average banance? ");

scanf("%lf", &avg);

printf("Num of transaction? ");

scanf("%d", &tran);

double interest = avg \* 0.06;

double sCharge = tran / 2.0; ***//or, (double)tran***

printf("Name: %s\n", name);

printf("Average balance: $%.2lf\n", avg);

printf("Interest: $%.2lf\n", interest);

printf("Service charge: $%.2lf\n", sCharge);

}

Output:

Name ? Golam Kibria

Accounr number ? 354545634256

Average banance ? 2500

Num of transaction ? 13

Name : Golam Kibria

Average balance : $2500.00

Interest : $150.00

Service charge : $6.50

20. Tickets are sold in three categories: reserve, stands and grounds. WAP for these values and print the amount of money collected from each category of tickets. Also print the total amount of tickets sold and the total amount of money collected. P60

#include <stdio.h>

int main()

{

//r=reserve || s=stands || g = grounds

double rPrice, sPrice, gPrice;

int rSold, sSold, gSold;

printf("Reserve price and tickets sold? ");

scanf("%lf %d", &rPrice, &rSold);

printf("Stands price and tickets sold? ");

scanf("%lf %d", &sPrice, &sSold);

printf("Grounds price and tickets sold? ");

scanf("%lf %d", &gPrice, &gSold);

double rSeals, sSales, gSales;

rSeals = rPrice \* rSold;

sSales = sPrice \* sSold;

gSales = gPrice \* gSold;

int tticketsold = rSold + sSold + gSold;

double tmoneycollected = rSeals + sSales + gSales;

printf("Reserve sales: $%.2lf\n", rSeals);

printf("Stands sales: $%.2lf\n", sSales);

printf("Grounds sales: $%.2lf\n", gSales);

printf("%d tickets were sold\n", tticketsold);

printf("Total money collected: $%.2lf\n", tmoneycollected);

}

Output:

Reserve priceand tickets sold ? 100 500

Stands price and tickets sold ? 75 4000

Grounds price and tickets sold ? 40 8000

Reserve sales : $50000.00

Stands sales : $300000.00

Grounds sales : $320000.00

12500 tickets were sold

Total money collected : $670000.00

21. Print job charge based on hours worked and cost of parts.

NOTE: $100 per hour and minimum charge for any job is $150. P70

Output:

Hours workd ? 2.5

Cost of part ? 20

Charge of the job = $270.00

Hours workd ? 1

Cost of part ? 25

Charge of the job = $150.00

#include <stdio.h>

int main()

{

double hours, parts, jobCharge;

printf("Hours workd? ");

scanf("%lf", &hours);

printf("Cost of part? ");

scanf("%lf", &parts);

jobCharge = hours \* 100 + parts;

if (jobCharge < 150)

jobCharge = 150;

printf("Charge of the job = $%.2lf\n", jobCharge);

}

This program is same as above (21) but only difference is this program illustrates the use of symbolic constants.

#include <stdio.h>

#define chargePerhour 100

#define conditionValue 150

int main()

{

double hours, parts, jobCharge;

printf("Hours workd? ");

scanf("%lf", &hours);

printf("Cost of part? ");

scanf("%lf", &parts);

jobCharge = hours \* chargePerhour + parts;

if (jobCharge < conditionValue)

jobCharge = conditionValue;

printf("Charge of the job = $%.2lf\n", jobCharge);

}

/\*উপরের টার মতো সেম আউটপুট\*/

22. Find the sum of two length. P73

For example: 3m 25cm and 2m 15cm is 5m 40cm but the sum of 3m 75cm and 5m 50cm is 9m 25cm.

Output:

Enter value for mand cm = 5 75

Enter value for mand cm = 3 45

Sum is 9m 20cm

#include <stdio.h>

int main()

{

int m1, cm1, m2, cm2, mSum, cmSum;

printf("Enter value for m and cm = ");

scanf("%d %d", &m1, &cm1);

printf("Enter value for m and cm = ");

scanf("%d %d", &m2, &cm2);

mSum = m1 + m2;

cmSum = cm1 + cm2;

if (cmSum >= 100)

{

cmSum = cmSum - 100;

mSum = mSum + 1;

}

printf("\nSum is %dm %dcm\n", mSum, cmSum);

}

This program is same as above (22).

Output:

Enter value for mand cm = 3 150

Enter value for mand cm = 2 200

Sum is 8m 50cm

#include <stdio.h>

int main()

{

int m1, cm1, m2, cm2, mSum, cmSum;

printf("Enter value for m and cm = ");

scanf("%d %d", &m1, &cm1);

printf("Enter value for m and cm = ");

scanf("%d %d", &m2, &cm2);

mSum = m1 + m2;

cmSum = cm1 + cm2;

if (cmSum >= 100)

{

mSum = mSum + cmSum / 100;

cmSum = cmSum % 100;

}

printf("\nSum is %dm %dcm\n", mSum, cmSum);

}

23. Request three marks. Print their average and pass/fail. P76

Output:

Enter 3 marks = 60 40 56

Average is = 52.00 Pass

#include <stdio.h>

int main()

{

int mark1, mark2, mark3;

double avg;

printf("Enter 3 marks = ");

scanf("%d %d %d", &mark1, &mark2, &mark3);

avg = (mark1 + mark2 + mark3) / 3.0;

printf("Average is = %.2lf ", avg);

if (avg >= 50)

printf("Pass\n");

else

printf("Fail\n");

}

24. Request a score and prints letter grade. P84

Output:

Enter a score = 70

Grade B

#include <stdio.h>

int main()

{

int score;

printf("Enter a score = ");

scanf("%d", &score);

printf("\nGrade ");

if (score < 50) printf("F\n");

else if (score < 75) printf("B\n");

else printf("A\n");

***/\*else {***

***if (score < 75) printf("B\n");***

***else printf("A\n");***

***}\*/***

}

25. Calculate pay. P77

Suppose we have values for hours worked and rate of pay (the amount paid per hour) and wish to calculate a person’s regular pay, overtime pay and gross pay based on the following.

If hours worked is less than or equal to 40, regular pay is calculated by multiplying hours worked by rate of pay and overtime pay is 0. If greater than 40 then regular pay is calculated by multiplying the hours in excess of 40 by the rate of pay by 1.5. Gross pay is calculated by adding regular pay and overtime pay.

#include <stdio.h>

int main()

{

double hours, rateperhour, regularPay, overtimePay, grossPay;

printf("Hours worked? ");

scanf("%lf", &hours);

printf("Rate of pay? ");

scanf("%lf", &rateperhour);

if (hours <= 40)

{

regularPay = hours \* rateperhour;

overtimePay = 0;

}

else

{

regularPay = 40 \* rateperhour;

overtimePay = (hours - 40) \* rateperhour \* 1.5;

}

grossPay = regularPay + overtimePay;

printf("\nRegular pay: $%.2lf\n", regularPay);

printf("Overtime pay: $%.2lf\n", overtimePay);

printf("Gross pay: $%.2lf\n", grossPay);

}

Output:

Hours worked ? 50

Rate of pay ? 12

Regular pay : $480.00

Overtime pay : $180.00

Gross pay : $660.00

This program is same as 25. We just solve this problem using symbolic constant.

#include <stdio.h>

#define maxregularhour 40

#define overtimefactor 1.5

int main()

{

double hours, rateperhour, regularPay, overtimePay, grossPay;

printf("Hours worked? ");

scanf("%lf", &hours);

printf("Rate of pay? ");

scanf("%lf", &rateperhour);

if (hours <= maxregularhour)

{

regularPay = hours \* rateperhour;

overtimePay = 0;

}

else

{

regularPay = maxregularhour \* rateperhour;

overtimePay = (hours - maxregularhour) \* rateperhour \* overtimefactor;

}

grossPay = regularPay + overtimePay;

printf("\nRegular pay: $%.2lf\n", regularPay);

printf("Overtime pay: $%.2lf\n", overtimePay);

printf("Gross pay: $%.2lf\n", grossPay);

}

Output:

Hours worked ? 50

Rate of pay ? 12

Regular pay : $480.00

Overtime pay : $180.00

Gross pay : $660.00

26. Request three sides and determine types of triangle. P86

Output:

Enter 3 sides of a triangle = 7 4 7

Isosceles

#include <stdio.h>

int main()

{

int a, b, c;

printf("Enter 3 sides of a triangle = ");

scanf("%d %d %d", &a, &b, &c);

if (a <= 0 || b <= 0 || c <= 0)

printf("\nNot a triangle\n");

else if (a + b <= c || b + c <= a || c + a <= b)

printf("\nNot a triangle\n");

else if (a == b && b == c)

printf("\nEqualateral\n");

else if (a == b || b == c || c == a)

printf("\nIsosceles\n");

else if (a \* a + b \* b == c \* c)

printf("\nRight angle\n");

else

printf("\nScalene\n");

}

27. Print the sum of several numbers entered by user. P95

Output:

Enter a number(0 to end) : 24

Enter a number(0 to end) : 13

Enter a number(0 to end) : 55

Enter a number(0 to end) : 32

Enter a number(0 to end) : 19

Enter a number(0 to end) : 0

The sum is = 143

#include <stdio.h>

int main()

{

int num, sum = 0;

printf("Enter a number (0 to end): ");

scanf("%d", &num);

while (num != 0)

{

sum = sum + num;

printf("Enter a number (0 to end): ");

scanf("%d", &num);

}

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

}

28. Print the sum and count of several numbers entered by user. P98

Output:

Enter a number(0 to end) : 24

Enter a number(0 to end) : 13

Enter a number(0 to end) : 55

Enter a number(0 to end) : 32

Enter a number(0 to end) : 19

Enter a number(0 to end) : 0

5 numbers were entered

The sum is = 143

#include <stdio.h>

int main()

{

int num, sum = 0, c = 0;

printf("Enter a number (0 to end): ");

scanf("%d", &num);

while (num != 0)

{

c = c + 1;

sum = sum + num;

printf("Enter a number (0 to end): ");

scanf("%d", &num);

}

printf("\n%d numbers were entered\n", c);

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

}

29. Print the sum and average entered by user. P98

Output:

Enter a number(0 to end) : 24

Enter a number(0 to end) : 13

Enter a number(0 to end) : 55

Enter a number(0 to end) : 32

Enter a number(0 to end) : 19

Enter a number(0 to end) : 0

5 numbers were entered

The sum is = 143

The average is = 28.60

#include <stdio.h>

int main()

{

int num, sum = 0, c = 0;

double avg;

printf("Enter a number (0 to end): ");

scanf("%d", &num);

/\*If user entered 0 then, there is a problem thst's why we use this system.\*/

if (num == 0) printf("No numbers entered\n");

else

{

while (num != 0)

{

c = c + 1;

sum = sum + num;

printf("Enter a number (0 to end): ");

scanf("%d", &num);

}

printf("\n%d numbers were entered\n", c);

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

printf("The average is = %.2lf\n", (double)sum / c);

}

}

30. Find the HCF of two numbers entered by user. P97

Output:

Enter two numbers = 42 24

Their HCF is = 6

#include <stdio.h>

int main()

{

int m, n, rem;

printf("Enter two numbers = ");

scanf("%d %d", &m, &n);

while (n != 0)

{

rem = m % n;

m = n;

n = rem;

}

printf("\nTheir HCF is = %d\n", m);

***/\*m is always greater than n. If n is greater than m then***

***the compiler automatically store the big value into m.\*/***

}

31. Increment and Decrement operator. P100

Output:

Suffix: 7

Prefix : 8

Suffix : 8

Prefix : 7

#include <stdio.h>

int main()

{

int n = 7, m = 3;

printf("Suffix: %d\n", m = n++);

printf("Prefix: %d\n", n = ++m);

printf("\n");

printf("Suffix: %d\n", m = n--);

printf("Prefix: %d\n", n = --m);

***/\*++n increment n befoe using its value,***

***whereas n++ increment n after using its value.\*/***

}

32. Find the largest of a set of numbers entered by user. P103

Output:

Enter a number(0 to end) : 36

Enter a number(0 to end) : 17

Enter a number(0 to end) : 43

Enter a number(0 to end) : 52

Enter a number(0 to end) : 50

Enter a number(0 to end) : 0

The large is = 52\*/

#include <stdio.h>

int main()

{

int num, bigNum;

printf("Enter a number(0 to end): ");

scanf("%d", &num);

if (num == 0)return;

/\*In c, the keyword return can be used in main to

halt he program by returning to the operating system.\*/

bigNum = num;

while (num != 0)

{

if (num > bigNum) bigNum = num;

printf("Enter a number(0 to end): ");

scanf("%d", &num);

}

printf("\nThe large is = %d\n", bigNum);

}

33. Find the smallest of a set of numbers entered by user. P106

Output:

Enter a number(0 to end) : 36

Enter a number(0 to end) : -17

Enter a number(0 to end) : 43

Enter a number(0 to end) : -52

Enter a number(0 to end) : 50

Enter a number(0 to end) : 0

The smallest is = -52

#include <stdio.h>

int main()

{

int num, smallNum;

printf("Enter a number(0 to end): ");

scanf("%d", &num);

if (num == 0)return;

smallNum = num;

while (num != 0)

{

if (num < smallNum) smallNum = num;

printf("Enter a number(0 to end): ");

scanf("%d", &num);

}

printf("\nThe smallest is = %d\n", smallNum);

}

34. Print five times a letter using loop. P123

output:

1. Golam Kibria

2. Golam Kibria

3. Golam Kibria

4. Golam Kibria

5. Golam Kibria

#include <stdio.h>

int main()

{

for (int i = 1; i <= 5; i++)

{

printf("%d. Golam Kibria\n", i);

}

}

35. Print many times of a letter entered by the user. P124

Output:

How many line ? 12

1. Golam Kibria

2. Golam Kibria

3. Golam Kibria

4. Golam Kibria

5. Golam Kibria

6. Golam Kibria

7. Golam Kibria

8. Golam Kibria

9. Golam Kibria

10. Golam Kibria

11. Golam Kibria

12. Golam Kibria

#include <stdio.h>

int main()

{

int n;

printf("How many line you want to print? ");

scanf("%d", &n);

printf("\n");

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

{

printf("%2d. Golam Kibria\n", i);

}

}

36.1 Multiplication table. P126

Output:

2 X 1 = 2

2 X 2 = 4

2 X 3 = 6

2 X 4 = 8

2 X 5 = 10

2 X 6 = 12

2 X 7 = 14

2 X 8 = 16

2 X 9 = 18

2 X 10 = 20

#include <stdio.h>

int main()

{

for (int i = 1; i <= 10; i++)

{

printf("2 X %2d = %2d\n", i, 2 \* i);

}

}

36.2 Multiplication of a table. P127

output:

Which table ? 5

5 X 1 = 5

5 X 2 = 10

5 X 3 = 15

5 X 4 = 20

5 X 5 = 25

5 X 6 = 30

5 X 7 = 35

5 X 8 = 40

5 X 9 = 45

5 X 10 = 50

#include <stdio.h>

int main()

{

int n;

printf("Which table? ");

scanf("%d", &n);

for (int i = 1; i <= 10; i++)

{

printf("%d X %2d = %2d\n", n, i, n \* i);

}

}

37. Multiplication table (From 🡪 To). P128

Output:

Which table ? 6

From ? 10

To ? 16

6 X 10 = 60

6 X 11 = 66

6 X 12 = 72

6 X 13 = 78

6 X 14 = 84

6 X 15 = 90

6 X 16 = 96

#include <stdio.h>

int main()

{

int n;

printf("Which table? ");

scanf("%d", &n);

int f, t;

printf("From? ");

scanf("%d", &f);

printf("To? ");

scanf("%d", &t);

if (f > t) {

printf("Invalid data!\n");

***/\*cuz from value is bigger than to to value\*/***

}

else

{

for (int i = f; i <= t; i++)

{

printf("%2d X %2d = %2d\n", n, i, n \* i);

}

}

}

38. Temperature convention menu. P130

Output:

Celcius Farenheit

0 32

10 50

20 68

30 86

40 104

50 122

60 140

70 158

80 176

90 194

100 212

#include <stdio.h>

int main()

{

double c, f;

printf("Celcius Farenheit\n\n");

for (c = 0; c <= 100; c = c + 10)

{

f = (c \* 1.8) + 32;

printf("%5.0lf%10.0lf\n", c, f);

}

}

39. Expressive power of for. P132

Output:

Enter the end ? 10

1 9

2 8

3 7

4 6

5 5

#include <stdio.h>

int main()

{

int start, end;

printf("Enter the end? ");

scanf("%d", &end);

for (start = 1, end = end - 1; start <= end; start++, end--)

{

printf("%d %d\n", start, end);

}

}

40. Determine HCF of two number using do…while loop. P134

Output:

Enter the value of aand b : 40 - 9

Enter the value of a and b : 30 0

Enter the value of a and b : 200 16

The HCF of 200 and 16 is = 8

#include <stdio.h>

int main()

{

int a, b, rem;

do {

printf("Enter the value of a and b : ");

scanf("%d %d", &a, &b);

} while (a <= 0 || b <= 0);

***/\*both a and b must be positive\*/***

printf("\nThe HCF of %d and %d is = ", a, b);

do {

rem = a % b;

a = b;

b = rem;

} while (b != 0);

printf("%d\n", a);

}

41. Interest at the bank.

A man deposit $1000 in a bank at an interest rate of 10% per year. At the end of each year, the interest earned is added to the amount on deposit and this become the new deposit for the next year. WAP to determine the year in which the amount accumulated first exceeds $2000. For each year, print the deposit at the beginning of the year and the interest year for that year until the target is reached. P135

#include <stdio.h>

int main()

{

int year = 0;

double initialdeposit, interestrate, target, deposit, interest;

printf("Intial deposit? ");

scanf("%lf", &initialdeposit);

printf("Rate of interest? ");

scanf("%lf", &interestrate);

printf("Target deposit? ");

scanf("%lf", &target);

printf("\nYear Deposit Interest\n\n");

deposit = initialdeposit;

do {

year++;

interest = deposit \* interestrate / 100;

deposit = deposit + interest;

printf("%3d %8.2lf %8.2lf\n", year, deposit, interest);

} while (deposit <= target);

printf("\nDeposit exceed $%7.2lf at the end of year %d\n", target, year);

}

Output:

Intial deposit ? 1000

Rate of interest ? 10

Target deposit ? 2000

Year Deposit Interest

1 1100.00 100.00

2 1210.00 110.00

3 1331.00 121.00

4 1464.10 133.10

5 1610.51 146.41

6 1771.56 161.05

7 1948.72 177.16

8 2143.59 194.87

Deposit exceed $2000.00 at the end of year 8

42. Read and print character. P145

Output:

Hello

Value of EOF is -1

#include <stdio.h>

int main()

{

char ch = getchar();

***//int ch = getchar();***

printf("Value of EOF is %d \n", EOF);

***/\*if we call getchar when there***

***is no more data it simplay returns -1\*/***

}

43. Read a character and print its code and the value of EOF. P46

#include <stdio.h>

int main()

{

printf("Type some data and press Enter \n");

char ch = getchar();

***/\*if we use only getch() then its only take h***

***so, we use getchar() cuz it take the full character like hello\*/***

printf("\nThe first character is : %c\n", ch);

printf("It's code is : %d\n", ch);

printf("Value of EOF is : %d\n", EOF);

Output:

Type some dataand press Enter

Hello

The first character is : H

It's code is : 72

Value of EOF is : -1

***/\*we can also use : int ch = getchar();\*/***

}

44. Print a character and its code. P143

Output:

The character is : K

The code is : 75

#include <stdio.h>

int main()

{

char ch = 'K'; ***//must be single quote***

printf("The character is : %c\n", ch);

printf("The code is : %d\n", ch);

}

45. Character in arithmetic expression. P144

Output:

The character is : k

The code is : 107

#include <stdio.h>

int main()

{

char ch = 'K' + 32; ***//must be single quote***

printf("The character is : %c\n", ch);

printf("The code is : %d\n", ch);

}

46. Uppercase to lowercase. P144

Output:

Enter any uppercase letter : A

The lower case letter is : a

The code is = 97

#include <stdio.h>

int main()

{

char ch;

printf("Enter any uppercase letter : ");

ch = getchar();

/\*or, ch = getchar()+32 if we use this then

ch+32(next line) is no needed\*/

printf("The lower case letter is : %c\n", ch+32);

printf("The code is = %d\n", ch + 32);

}

47. Lowercase to uppercase. P144

Output:

Enter any lowercase letter : a

The uppercase letter is : A

The code is = 65

#include <stdio.h>

int main()

{

char ch;

printf("Enter any lowercase letter : ");

ch = getchar();

/\*or, ch = getchar()-32 if we use this then

ch-32(next line) is no needed\*/

printf("The uppercase letter is : %c\n", ch-32);

printf("The code is = %d\n", ch - 32);

}

48. Read and print the first three character in the data. P148

#include <stdio.h>

int main()

{

printf("Type some data and press Enter \n");

int i;

for (i = 1; i <= 3; i++){

char ch = getchar();

printf("Character %d is %c\n", i, ch);

}

}

Output:

Type some data and press Enter

Hi, Kibria

Character 1 is H

Character 2 is i

Character 3 is,

49. Read and print the first non-blank character in the data. P149

#include <stdio.h>

int main()

{

printf("Press some data and press Enter\n");

char ch = getchar();

/\*as long as ch is a blank it will take input again and again. But when getchar find a non-blank character then it directly go to the printf statement\*/

while (ch == ' ')

{

ch = getchar();

}

printf("The first non-blank is %c\n", ch);

}

Output:

Press some dataand press Enter

---Kibria // its indicate space(-)

The first non - blank is K\*

50. Print all character before the first blank in the data. P150

#include <stdio.h>

int main()

{

printf("Type some data and press Enter\n");

char ch = getchar();

/\*As long as getchar() does not find a non-blank character the while loop is running. But when it's find a blank character then it print all the character before the black. If user enter 'space' as the first character then its print nothing\*/

Output:

Type some dataand press Enter

Golam kibria

G

o

l

a

m

while (ch != ' ')

{

printf("%c\n", ch);

ch = getchar();

}

}

51. Print the first non-blank character and count number of non-blank. P152

#include <stdio.h>

int main()

{

printf("Type some data and press Enter\n");

char ch = getchar();

int count = 0;

/\*It's count number of space\*/

while (ch == ' ')

{

count++;

ch = getchar();

}

printf("The number of leading brackets %d\n", count);

printf("The forst non-blank is %c\n", ch);

}

Output:

Type some dataand press Enter

----kibria //- means space(e.g.)

The number of leading brackets 4

The forst non - blank is k

52. Count the number of characters of the input line. P153

#include <stdio.h>

int main()

{

printf("Type some data and press Enter\n");

char ch = getchar();

int count = 0;

/\*repeat as long as \n. When a user press enter

the new line character(\n) returned by getchar\*/

while (ch != '\n')

{

ch = getchar();

count++;

}

printf("The number of character : %d\n", count);

}

Output:

Type some data and press Enter

Hi kibria.How are you.

The number of character : 23

53. Count the number of character and blank in the input line. P154

Output:

Type some dataand press Enter

golam kibria ezaz

Number of character is = 17

Number of blanks = 2

#include <stdio.h>

int main()

{

char ch;

int numberofcharacter = 0;

int numberofblanks = 0;

printf("Type some data and press Enter\n");

ch = getchar();

while (ch != '\n')

{

numberofcharacter++;

ch = getchar();

/\*If character is blank then numberofblank is increment\*/

if (ch == ' '){

numberofblanks++;

}

}

printf("Number of character is = %d\n", numberofcharacter);

printf("Number of blanks = %d\n", numberofblanks);

}

54. Read a line of data and find the largest character. P155

Output:

Type some dataand press Enter

Where The Mind Is Without Fear

The largest character is = u

#include <stdio.h>

int main()

{

char ch;

int bigchar = '\0';

/\*The value of null character('\0') is 0(zero)\*/

printf("Type some data and press Enter\n");

ch = getchar();

while (ch != '\n')

{

if (ch > bigchar){

bigchar = ch;

}

ch = getchar();

}

printf("The largest character is = %c\n", bigchar);

}

55. Read character and digit and print only digit. P163

#include <stdio.h>

int main()

{

char ch;

int num = 0;

ch = getchar();

printf("Type some data including number\n");

/\*As long as the character is

not a digit keep reading\*/

while (ch < '0' || ch > '9')

{

ch = getchar();

}

/\*This loop is executed when

character is a digit\*/

while (ch >= '0' && ch <= '9')

{

num = num \* 10 + ch - '0';

ch = getchar();

}

printf("Number is = %d\n", num);

}

/\*integer value of digit = code for digit character - code for character '0'\*/

Output:

Type some data including number

Golam kibria 3435 diu

Number is = 3435

56. WAP to show how skiplines fits into a complete program. P167

Output:

My name is

Golam kibria

#include <stdio.h>

int main()

{

void skiplines(int n);

printf("My name is \n");

skiplines(2);

printf("Golam kibria\n");

}

void skiplines(int n)

{

int i;

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

printf("\n");

}

}

57.1 Finding largest number between two number. Using function. P171

Output:

Enter two whole number : 4 6

The largest number is : 6

#include <stdio.h>

int main()

{

int max(int a, int b);

int n1, n2;

printf("Enter two whole number: ");

scanf("%d %d", &n1, &n2);

printf("The largest number is: %d\n", max(n1, n2));

}

int max(int a, int b)

{

if (a > b)

return a;

else

return b;

}

57.2 Finding largest number between two number. Using function. P171

#include <stdio.h>

int main()

{

int max(int a, int b);

int n1, n2;

printf("Enter two whole number: ");

scanf("%d %d", &n1, &n2);

while (n1 != 0 || n2 != 0) {

printf("The largest number is : %d\n", max(n1, n2));

printf("Enter two whole number: ");

scanf("%d %d", &n1, &n2);

}

Output:

Enter two whole number : 4 6

The largest number is : 6

Enter two whole number : 18 88

The largest number is : 88

Enter two whole number : 33 56

The largest number is : 56

Enter two whole number : 0 9

The largest number is : 9

Enter two whole number : 0 0

}

int max(int a, int b)

{

if (a > b)

return a;

else

return b;

}

58. Finding largest number between two number. Using function. P173

Output:

Enter two whole number : 4 7

The smallest number is : 4

Enter two whole number : 44 55

The smallest number is : 44

Enter two whole number : 0 0

#include <stdio.h>

int main()

{

int min(int a, int b);

int n1, n2;

printf("Enter two whole number: ");

scanf("%d %d", &n1, &n2);

while (n1 != 0 || n2 != 0) {

printf("The smallest number is: %d\n", min(n1, n2));

printf("Enter two whole number: ");

scanf("%d %d", &n1, &n2);

}

}

int min(int a, int b)

{

if (a < b)

return a;

else

return b;

}

59. Largest character between the two character. Using function. P173

Output:

Enter two character : A C

The largest number is : 67

#include <stdio.h>

int main()

{

int max(int a, int b);

char ch1, ch2;

printf("Enter two character: ");

scanf("%c %c", &ch1, &ch2);

printf("The largest number is: %d\n", max(ch1, ch2));

}

int max(int a, int b)

{

if (a > b)

return a;

else

return b;

}

60. Print the day. P173

Output:

Enter a day from 1 to 7 : 3

Monday

#include <stdio.h>

int main()

{

int n;

printf("Enter a day from 1 to 7: ");

scanf("%d", &n);

if (n == 1) printf("Saturday\n");

else if (n == 2) printf("Sunday\n");

else if (n == 3) printf("Monday\n");

else if (n == 4) printf("Tuesday\n");

else if (n == 5) printf("Wednesday\n");

else if (n == 6) printf("Thirsday\n");

else if (n == 7) printf("Friday\n");

else printf("Invalid day\n");

}

61. Print the day using function. P174

Output:

Enter a day from 1 to 7 : 4

Tuesday

#include <stdio.h>

int main()

{

void printday(int n);

int n;

printf("Enter a day from 1 to 7: ");

scanf("%d", &n);

printday(n);

}

void printday(int n)

{

if (n == 1) printf("Saturday\n");

else if (n == 2) printf("Sunday\n");

else if (n == 3) printf("Monday\n");

else if (n == 4) printf("Tuesday\n");

else if (n == 5) printf("Wednesday\n");

else if (n == 6) printf("Thirsday\n");

else if (n == 7) printf("Friday\n");

else printf("Invalid day\n");

}

62. Determine HCF of two number using function. P176

#include <stdio.h>

int main()

{

int hcf(int m, int n);

int a, b;

printf("Enter two positive number = ");

scanf("%d %d", &a, &b);

while (a > 0 || b > 0) {

printf("The HCF is = %d\n", hcf(a, b));

printf("Enter two positive number = ");

scanf("%d %d", &a, &b);

}

}

int hcf(int m, int n)

{

Output:

Enter two positive number = 24 42

The HCF is = 6

Enter two positive number = 32 512

The HCF is = 32

Enter two positive number = 100 31

The HCF is = 1

Enter two positive number = 0 0

int rem;

while (n != 0) {

rem = m % n;

m = n;

n = rem;

}

return m;

}

63. Using HCF to find LCM. P177

#include <stdio.h>

int main()

{

int hcf(int m, int n);

int a, b;

printf("Enter two positive number = ");

scanf("%d %d", &a, &b);

while (a > 0 || b > 0) {

printf("The HCF is = %d\n", hcf(a, b));

printf("The LCM is = %d\n", (a \* b) / hcf(a, b));

printf("Enter two positive number = ");

scanf("%d %d", &a, &b);

}

}

int hcf(int m, int n)

{

int rem;

while (n != 0) {

Output:

Enter two positive number = 8 6

The HCF is = 2

The LCM is = 24

Enter two positive number = 42 24

The HCF is = 6

The LCM is = 168

Enter two positive number = 0 0

rem = m % n;

m = n;

n = rem;

}

return m;

}

63. Determine factorial. P178

Output:

Enter a positive number = 4

4! = 24

Enter a positive number = 0

0! = 1

#include <stdio.h>

int main()

{

int n, fact = 1;

printf("Enter a positive number = ");

scanf("%d", &n);

int i;

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

fact = fact \* i;

}

printf("%d! = %d\n", n, fact);

}

64. Determine factorial using function. P178

Output:

Enter a positive number = 4

The factorial of 4 is 24

Enter a positive number = 5

The factorial of 5 is 120

Enter a positive number = 8

The factorial of 8 is 40320

Enter a positive number = 0

#include <stdio.h>

int main()

{

int factorial(int x);

int n;

printf("Enter a positive number = ");

scanf("%d", &n);

while (n > 0) {

printf("The factorial of %d is %d\n", n, factorial(n));

printf("Enter a positive number = ");

scanf("%d", &n);

}

}

int factorial(int x)

{

int fact = 1;

int i;

for (i = 2; i <= x; i++) {

fact = fact \* i;

}

return fact;

}

65. Factorial from 0 to 7. P181

Output:

z z!

--------

0 1

1 1

2 2

3 6

4 24

5 120

6 720

7 5040

#include <stdio.h>

int main()

{

int factorial(int x);

int n;

printf("z z!\n"); //5 space

printf("--------\n\n");

for (n = 0; n <= 7; n++) {

printf("%1d %5d\n", n, factorial(n));

}

}

int factorial(int x)

{

int fact = 1;

int i;

for (i = 2; i <= x; i++) {

fact = fact \* i;

}

return fact;

}

66. Combinations.

Suppose there are 7 people on a committee. How many subcommittees of 3 people can be formed? The answer is denote by 7C3 or 7!/4!\*3!

#include <stdio.h>

int main()

{

int factorial(int n);

int combination(int n, int r);

int n, r, ncr;

printf("Enter value for n and r: ");

scanf("%d %d", &n, &r);

while (n != 0) {

ncr = combination(n, r);

if (ncr == 1) {

printf("There is 1 combination of %d object "

"taken %d at a time\n", n, r);

}

else {

printf("There are %d combination of %d object "

"taken %d at a time\n", ncr, n, r);

}

printf("Enter value for n and r: ");

scanf("%d %d", &n, &r);

}

}

int factorial(int n)

{

int fact = 1, i;

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

fact = fact \* i;

}

return fact;

}

***/\*we can also use a and b. a for n and b for r\*/***

int combination(int n, int r)

{

int factorial(int n);

return factorial(n) / (factorial(n - r) \* factorial(r));

}

Output:

Enter value for nand r : 7 3

There are 35 combination of 7 object taken 3 at a time

Enter value for nand r : 6 6

There is 1 combination of 6 object taken 6 at a time

Enter value for nand r : 0 0

67. Print job charge based on hours worked and cost of parts. P184

NOTE: $100 per hour and minimum charge for any job is $150. P70

Output:

Hours worked ? 2.5

Cost of parts ? 20

Charge for the job : $270.00

#include <stdio.h>

#define chargeperhour 100

#define minjobcost 150

int main()

{

double total(double hours, double parts);

double hours, parts;

printf("Hours worked? ");

scanf("%lf", &hours);

printf("Cost of parts? ");

scanf("%lf", &parts);

printf("Charge for the job: $%3.2lf\n", total(hours, parts));

}

double total(double hours, double parts)

{

double jobcharge;

jobcharge = hours \* chargeperhour + parts;

if (jobcharge < minjobcost)

return minjobcost;

return jobcharge;

}

68. Calculate pay using function. P185

Suppose we have values for hours worked and rate of pay (the amount paid per hour) and wish to calculate a person’s regular pay, overtime pay and gross pay based on the following.

If hours worked is less than or equal to 40, regular pay is calculated by multiplying hours worked by rate of pay and overtime pay is 0. If greater than 40 then regular pay is calculated by multiplying the hours in excess of 40 by the rate of pay by 1.5. Gross pay is calculated by adding regular pay and overtime pay.

Output:

Hours worked ? 50

Rate of pay ? 12

Total pay : $660.00

#include <stdio.h>

#define maxregularhour 40

#define overtimefactor 1.5

int main()

{

double totalpay(double hours, double rate);

double hours, rate;

printf("Hours worked? ");

scanf("%lf", &hours);

printf("Rate of pay? ");

scanf("%lf", &rate);

printf("Total pay: $%3.2lf\n", totalpay(hours, rate));

}

double totalpay(double hours, double rate)

{

double regularpay, overtimepay;

if (hours <= maxregularhour)

return hours \* rate;

return maxregularhour \* rate +

(hours - maxregularhour) \* rate \* overtimefactor;

}

69. Sum of exact divisors. P186

Output:

Enter a number: 15

9

#include <stdio.h>

int main()

{

int n;

printf("Enter a number: ");

scanf("%d", &n);

int i;

int sum = 1;

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

if (n % i == 0) {

sum = sum + i;

}

}

printf("%d\n", sum);

}

70.1 Sum of exact divisor using function. P187

Output:

Enter a number : 50

Sum of exact divisor is : 43

#include <stdio.h>

int main()

{

int sumofed(int n);

int n;

printf("Enter a number: ");

scanf("%d", &n);

printf("Sum of exact divisor is : %d\n", sumofed(n));

}

int sumofed(int n)

{

int i;

int sum = 1;

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

if (n % i == 0) {

sum = sum + i;

}

}

return sum;

}

70.2 Sum of exact divisor using function. P187

#include <stdio.h>

int main()

{

int sumofed(int n);

int n;

printf("Enter a number: ");

scanf("%d", &n);

while (n != 0) {

int x = sumofed(n);

if (x > n) printf("Abundant\n");

else if (x < n) printf("Deficient\n");

else printf("Perfect\n");

printf("Enter a number: ");

scanf("%d", &n);

}

}

int sumofed(int n)

{

int i;

int sum = 1;

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

if (n % i == 0) {

sum = sum + i;

}

}

return sum;

}

Output:

Enter a number : 15

Deficient

Enter a number : 12

Abundant

Enter a number : 0

71. Perfect number between 1 to 10000. P187

#include <stdio.h>

int main()

{

int i, n = 10000;

int sum = 0;

printf("Perfect number between 1 to 10000\n");

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

{

int p = 1;

while (p <= (i / 2)) {

if (i % p == 0) {

sum = sum + p;

}

p++;

}

if (sum == i) {

Output:

Perfect number between 1 to 10000

6

28

496

8128

printf("%d\n", i);

}

sum = 0;

}

}

72. Some character function. P188

Output:

Enter a character : A

The return value is : 1

#include <stdio.h>

int main()

{

int uppercase(char ch);

char ch;

printf("Enter a characte : ");

scanf("%c", &ch);

printf("The return value is : %d\n", uppercase(ch));

}

int uppercase(char ch)

{

if (ch >= 'Z' && ch <= 'Z')

return 1;

return 0;

/\*we can also write this 3 line into 1 line like this return ch>='A && ch<='Z'. This will be return 1 if the statement is true, and return 0 when it is false\*/

}

/\*The lowercase is also same as uppercase just change this statement like this if(ch>='a' && ch<='z')\*/

/\*If we want to find the digit it is also same as uppercase, just change this statement like this if(ch>='0' && ch<='9')\*/

73. Position of a letter in the alphabet. P189

#include <stdio.h>

int main()

{

int position(char ch);

char ch;

printf("Type some letter and non-letter and press Enter\n");

ch = getchar();

while (ch != '\n') {

printf("%c %3d\n", ch, position(ch));

ch = getchar();

}

}

int uppercase(char ch){

return ch >= 'A' && ch <= 'Z';

}

int lowercase(char ch){

return ch >= 'a' && ch <= 'z';

}

int position(char ch){

int uppercase(char ch);

int lowercase(char ch);

if (uppercase(ch)) return ch - 'A' + 1;

if (lowercase(ch)) return ch - 'a' + 1;

return 0;

}

Output:

Type some letter and non - letter and press Enter

Fa$&n

F 6

a 1

$ 0

& 0

n 14

ch = getchar();

while (ch != '\n') {

printf("%c %3d\n", ch, position(ch));

ch = getchar();

}

we can write this four line into two line like this:

while ((ch = getchar()) != '\n') {

printf("%c %3d\n", ch, position(ch));

}

74. Take some digit and non-digit number and print only digit number. P191

#include <stdio.h>

int main()

{

int getInt();

printf("Type some digit and non-digit number and press Enter\n");

printf("Number is = %d\n", getInt());

}

int getInt()

{

char ch;

ch = getchar();

***/\*As long as the character is not a digit keep reading\*/***

while (ch < '0' || ch>'9') {

ch = getchar();

}

int num = 0;

***/\*At this point character find first digit\*/***

while (ch >= '0' && ch <= '9') {

num = num \* 10 + ch - '0';

ch = getchar();

}

return num;

}

Output:

Type some digit and non-digit and press Enter

Kibrai3435kdh

Number is = 3435

we can write this two line like this:

while (ch < '0' || ch>'9') or, while (!digit(ch))

* As long as the character is not a digit keep reading.

while (ch >= '0' && ch <= '9') or, while (digit(ch))

* At this point character find first digit.

getInt reads data character by character and returns the next

integer found. The function does not take any argument but the bracket

must be written after the name.

75. Find the sum of two length given in meters and centimeters. P192

#include <stdio.h>

**//#include <ctype.h>**

int main()

{

int m1, cm1, m2, cm2, msum, cmsum;

int getInt();

printf("Enter first length: ");

m1 = getInt();

cm1 = getInt();

printf("Enter second length: ");

m2 = getInt();

cm2 = getInt();

msum = m1 + m2;

cmsum = cm1 + cm2;

if (cmsum >= 100) {

msum = msum + 1;

cmsum = cmsum - 100;

}

printf("\nSum is %dm %dcm\n", msum, cmsum);

}

int getInt()

{

char ch;

ch = getchar();

***/\*As long as the character is not a digit keep reading\*/***

while (ch < '0' || ch>'9') {

ch = getchar();

}

int num = 0;

***/\*At this point character find first digit\*/***

while (ch >= '0' && ch <= '9') {

num = num \* 10 + ch - '0';

ch = getchar();

}

return num;

}

Output:

Enter first length : 3m 75cm

Enter second length : 5m 50cm

Sum is 9m 25cm

76. Find average and difference from average. P204

#include <stdio.h>

#define maxnumber 100

int main()

{

int n;

printf("Enter up to 100 numbers (end with 0)\n");

scanf("%d", &n);

int x = 0;

double sum = 0;

int num[maxnumber];

while (n != 0) {

sum = sum + n;

num[x++] = n;

scanf("%d", &n);

}

if (x == 0) printf("No numbers entered\n");

else {

printf("\nNumbers entered: %d\n", x);

printf("Sum of numbers: %1.0lf\n", sum);

double average = sum / x;

printf("The average is: %3.2lf\n", average);

printf("Numbers are different from average\n");

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

printf("%4d %7.2lf\n", num[i], num[i] - average);

}

}

}

Output:

Enter up to 100 numbers(end with 0)

2 7 5 3 0

Numbers entered : 4

Sum of numbers : 17

The average is : 4.25

Numbers are different from average

2 - 2.25

7 2.75

5 0.75

3 - 1.25

77. Letter frequency count. P207

#include <stdio.h>

Golam kibria

Letter Frequency

a 2

b 1

c 0

d 0

e 0

f 0

g 1

h 0

i 2

j 0

k 1

l 1

m 1

n 0

o 1

p 0

q 0

r 1

s 0

t 0

u 0

v 0

w 0

x 0

y 0

z 0

int main()

{

int position(char ch);

char ch;

int n;

int lettercount[27];

for (n = 1; n <= 26; n++) {

lettercount[n] = 0;

}

ch = getchar();

while (ch != '\n') {

n = position(ch);

if (n > 0) {

++lettercount[n];

}

ch = getchar();

}

printf("Letter Frequency\n");

for (n = 1; n <= 26; n++) {

printf("%5c %5d\n", 'a' + n - 1, lettercount[n]);

}

}

int position(char ch)

{

if (isupper(ch)) return ch - 'A' + 1;

if (islower(ch)) return ch - 'a' + 1;

return 0;

}

78. Sum of number using array. P212

Output:

How many numbers : 5

Enters numbers :

3 8 1 5 7

Sum is : 24

#include <stdio.h>

int main()

{

int score[10];

int n;

printf("How many numbers: ");

scanf("%d", &n);

printf("Enters numbers: \n");

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

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

}

int sum = 0;

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

sum = sum + score[j];

}

printf("Sum is : %d\n", sum);

}

**/\*The same program using function\*/**

#include <stdio.h>

int main()

{

int calculatesum(int score[], int n);

int score[10];

int n;

***//printf("How many numbers: ");***

scanf("%d", &n);

printf("Enters numbers: \n");

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

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

}

printf("Sum is : %d\n", calculatesum(score, 5));

}

int calculatesum(int score[], int n)

{

Output: Same as above.

int sum = 0;

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

sum = sum + score[j];

}

return sum;

}

79. Find first alphabetical word before first blank space. P215

Output:

12$kibria dhaka35

kibria

#include <stdio.h>

int main()

{

char word[10];

int n = 0;

char ch = getchar();

***//read data untill it find alphabet.***

while (!isalpha(ch)) {

ch = getchar();

}

***//when find alphabet it execute this loop.***

while (isalpha(ch)) {

word[n++] = ch;

ch = getchar();

}

word[n] = '\0';

printf("%s\n", word);

}

80. Counting number of spaces. P216

#include <stdio.h>

int main()

{

char ch[] = "How we live and how we die";

int i = 0;

int space = 0;

while (ch[i] != 0) {

if (ch[i] == ' ') {

space++;

}

i++;

}

printf("Number of space: %d\n", space);

}

Output:

Number of space : 6

If we want to take it from the user we just

write the program like this:

char ch[40];

gets(ch);

Then the whole program is same as above

81. Reverse the character in a string. P218

#include <stdio.h>

#include <string.h>

int main()

{

char ch[30];

printf("Type some data : ");

gets(ch);

int length;

length = strlen(ch)-1;

int m = 0;

while (m < length) {

char c = ch[m];

ch[m] = ch[length];

ch[length] = c;

Output:

Type some data : golam kibria

airbik malog

/\*if we type comma in input it will also printed in output.\*/

we can also write the while loop like this

for(m = 0, length; m<length; m++, length--)\*

m++; length--;

}

printf("%s\n", ch);

}

82. Check a string palindrome or not. P222

Output:

Enter a string

civic

The string is a palindrome.

#include <stdio.h>

#include <string.h>

int main()

{

char a[100], b[100];

printf("Enter a string\n");

gets(a);

strcpy(b, a);

strrev(b);

if (strcmp(a, b) == 0)

printf("The string is a palindrome.\n");

else

printf("The string isn't a palindrome.\n");

return 0;

}

83. Check a string palindrome or not using function. P222

#include <stdio.h>

#include <string.h>

int main()

{

int palindrome(char ch[]);

char ch[30];

printf("Type some data : ");

gets(ch);

***/\*Two double quotes denote the empty string\*/***

while (strcmp(ch, "") != 0) {

if (palindrome(ch))

printf("Palindrome\n");

else

printf("Not palindrome\n");

printf("Type some data : ");

gets(ch);

}

}

int palindrome(char ch[])

{

int m = 0;

int length;

length = strlen(ch) - 1;

while (m < length)

if (ch[m++] != ch[length--])

return 0;

return 1;

}

Output:

Type some data : kibria

Not palindrome

Type some data : civic

Palindrome

Type some data :

**//press enter to end the output.**

84. A better palindrome function. P224

#include <stdio.h>

#include <string.h>

int main()

{

void onlylowerletter(char phase[], char ch[]);

int palindrome(char ch[]);

char ch[30], phase[30];

printf("Type some data : ");

gets(phase);

***/\*Two double quotes denote the empty string\*/***

while (strcmp(phase, "") != 0) {

onlylowerletter(phase, ch);

printf("Converted to: %s\n", ch);

if (palindrome(ch))

printf("Palindrome\n");

else

printf("Not palindrome\n");

printf("Type some data : ");

gets(phase);

}

}

void onlylowerletter(char phase[], char ch[])

{

int j = 0, n = 0;

char q;

while ((q = phase[j++]) != '\0')

if (isalpha(q)) ch[n++] = tolower(q);

ch[n] = '\0';

}

int palindrome(char ch[])

{

int m = 0;

int length;

length = strlen(ch) - 1;

while (m < length)

if (ch[m++] != ch[length--])

return 0;

return 1;

Output:

Type some data : Madam i'm adam

Converted to : madamimadam

Palindrome

Type some data : Golam, kibria

Converted to : golamkibria

Not palindrome

Type some data :

}

85. Print a day using string. P227

Output:

Enter a day number = 5

wednesday

#include <stdio.h>

int main()

{

int n;

printf("Enter a day number = ");

scanf("%d", &n);

char day[10][10] = { "", "saturday", "sunday", "monday", "tuesday",

"wednesday", "thrisday", "friday" };

if (n < 1 || n > 7)

printf("Invalid Day\n");

else

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

}

86. Find maximum number and its position using array. P233

Output:

How many numbers : 5

12 13 43 55 17

Maximum number is : 55

#include <stdio.h>

int main()

{

int num[100];

int n, i;

printf("How many numbers : ");

scanf("%d", &n);

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

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

}

int max = num[0];

int position;

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

if (max < num[i]) {

max = num[i];

position = i;

}

}

printf("Maximum number is : %d\n", max);

}

87. Find maximum number by array using function. P233

#include <stdio.h>

int main()

{

int maxnumber(int num[], int n);

int num[100];

int n, i;

printf("How many numbers : ");

scanf("%d", &n);

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

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

}

printf("Maximum number is : %d\n", maxnumber(num, n));

}

int maxnumber(int num[], int n)

{

int i;

int max = num[0];

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

if (max < num[i]) {

max = num[i];

}

}

return max;

}

Output:

How many numbers : 5

12 13 43 55 17

Maximum number is : 55

88. Find small number by array using function. P233

#include <stdio.h>

int main()

{

int minnumber(int num[], int n);

int num[100];

int n, i;

printf("How many numbers : ");

scanf("%d", &n);

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

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

}

printf("Small number is : %d\n", minnumber(num, n));

}

int minnumber(int num[], int n)

{

int i;

int min = num[0];

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

if (min > num[i]) {

min = num[i];

}

}

return min;

}

Output:

How many numbers : 5

12 13 22 11 45

Small number is : 11

89. Print a desire number from an array index. P245

Output:

61

#include <stdio.h>

int main()

{

int num[7] = { 35, 17, 48, 25, 61, 12, 42 };

printf("%d\n", num[4]);

}

90. Print a desire number from an array index using function. P245

#include <stdio.h>

int main()

{

int num[10];

int n;

printf("How many numbers : ");

scanf("%d", &n);

int i;

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

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

{

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

}

int x;

printf("Enter the index number you like to print : ");

scanf("%d", &x);

if (x > n) {

printf("Please inter correct index. This is too big\n");

}

else {

printf("The %dth number is : %d\n", x, num[x]);

}

}

Output:

How many numbers : 5

Enter 5 numbers : 12 23 33 45 67

Enter the index number you like to print : 2

The 2th number is : 33

91. Ascending and Descending order. P249

#include <stdio.h>

int main()

{

int i, j, temp, n, number[30];

printf("Enter the value of n \n");

scanf("%d", &n);

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

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

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

}

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

{

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

{

if (number[i] > number[j])

{

temp = number[i];

number[i] = number[j];

number[j] = temp;

}

}

}

printf("Ascending order : \n");

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

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

printf("\nDescending order : \n");

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

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

}

Output:

Enter the value of n

7

Enter 7 numbers

57 48 49 65 15 33 52

Ascending order :

15 33 48 49 52 57 65

Descending order :

65 57 52 49 48 33 15

92. Print some string in alphabetical order. P259

#include <stdio.h>

#include <string.h>

int main()

{

char str[25][25];

int n;

printf("How many string : ");

scanf("%d", &n);

printf("Enter string one by one : \n");

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

{

gets(str[i]);

}

char temp[25];

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

{

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

{

if (strcmp(str[i], str[j]) > 0)

{

strcpy(temp, str[i]);

strcpy(str[i], str[j]);

strcpy(str[j], temp);

}

}

}

printf("\nThe sorted string is : ");

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

{

***/\*printf("%s\n", str[i]);***\*/

puts(str[i]);

Output:

How many string : 5

Enter string one by one :

kibria

saim

tausif

abid

naeem

The sorted string is :

abid

kibria

naeem

saim

tausif

}

}

93. Reverse a string.

Output:

Enter the string : kibria

Reverse string is : airbik

#include<stdio.h>

#include<string.h>

int main()

{

char str[100], temp;

int i, j;

printf("Enter the string :");

gets(str);

i = 0;

j = strlen(str);

while (i < j)

{

temp = str[i];

str[i] = str[j-1];

str[j-1] = temp;

i++; j--;

}

printf("\nReverse string is :%s", str);

}

94. Reverse a string by alphabetical order.

Enter the string : kibria

abiikr

#include <stdio.h>

#include <string.h>

int main()

{

char string[100];

printf("Enter the string : ");

***//scanf(“ % s”, string);***

gets(string);

char temp;

int i, j;

int n = strlen(string);

for (i = 0; i < n - 1; i++) {

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

if (string[i] > string[j]) {

temp = string[i];

string[i] = string[j];

string[j] = temp;

}

}

}

***/\*printf(“The sorted string is : % s”, string);\*/***

puts(string);

}

95. Sorted some number. P257

#include <stdio.h>

int main()

{

void sort(int num[], int n);

int num[10];

int n;

printf("Type 10 numbers which is end by 0\n");

scanf("%d", &n);

int i = 0;

while (n != 0)

{

num[i++] = n;

scanf("%d", &n);

}

sort(num, i);

printf("\nThe sorted number are\n");

for (int h = 0; h < i; h++)

{

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

}

printf("\n");

}

void sort(int num[], int n)

{

for (int h = 1; h < n; h++)

{

int key = num[h];

int a = h - 1;

while (a >= 0 && num[a] > key)

{

num[a + 1] = num[a];

a--;

}

num[a + 1] = key;

}

}

Output:

Type 10 numbers which is end by 0

57 48 79 65 15 33 52 0

The sorted number are

15 33 48 52 57 65 79

96. Sort parallel array

#include <stdio.h>

int main()

{

void parallelsort(int initial, int final, int max, char name[][15], int id[]);

char name[5][15] = { "Kibria", "Saim", "Tausif", "Abid", "Naeem" };

int id[5] = { 3456, 6543, 7865, 1278, 9834 };

parallelsort(0, 4, 15, name, id);

printf("\nThe sorted name and id's are \n");

for (int h = 0; h < 5; h++)

{

printf("%s\t\t%d\n", name[h], id[h]);

}

}

void parallelsort(int initial, int final, int max, char name[][15], int id[])

{

char key[15];

for (int h = initial; h <= final; h++)

{

strcpy(key, name[h]);

int m = id[h];

int k = h - 1;

while (k >= initial && strcmp(key, name[k]) < 0)

{

strcpy(name[k + 1], name[k]);

id[k + 1] = id[k];

--k;

}

strcpy(name[k + 1], key);

id[k + 1] = m;

}

}

Output:

The sorted nameand id's are

Abid 1278

Kibria 3456

Naeem 9834

Saim 6543

Tausif 7865

when we take it from the user just change this

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

scanf("%s", &name[i]);

}

int id[5];

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

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

}

97. Binary search.

#include <stdio.h>

int main()

{

int first, last, middle, n, search, array[100];

printf("How many numbers? ");

scanf("%d", &n);

printf("Enter %d integers:\n", n);

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

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

printf("Enter the value to find : ");

scanf("%d", &search);

first = 0;

last = n - 1;

middle = (first + last) / 2;

while (first <= last) {

if (array[middle] == search) {

printf("%d is present at index %d.\n", search, middle);

break;

}

else if (array[middle] < search)

first = middle + 1;

else

last = middle - 1;

middle = (first + last) / 2;

}

if (first > last)

printf("Not found! %d is not present in the list.\n", search);

}

Output:

How many numbers ? 5

Enter 5 integers :

12 23 34 45 56

Enter the value to find : 56

56 is present at index 4.

98. Binary search using function.

#include <stdio.h>

int main()

{

int binarysearch(int array[], int search, int first, int last);

int first, last, middle, n, search, array[100];

printf("How many numbers? ");

scanf("%d", &n);

printf("Enter %d integers:\n", n);

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

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

printf("Enter the value to find : ");

scanf("%d", &search);

first = 0;

last = n - 1;

middle = (first + last) / 2;

if (first > last)

printf("Not found! %d is not present in the list.\n", search);

int ans = binarysearch(array, search, 0, n - 1);

if (ans == -1) printf("%d not found\n", search);

else printf("%d found is positon %d\n", search, ans);

}

int binarysearch(int array[], int search, int first, int last)

{

int middle = (first + last) / 2;

while (first <= last) {

if (array[middle] == search) return middle;

else if (array[middle] < search) first = middle + 1;

else last = middle - 1;

middle = (first + last) / 2;

}

}

Output:

How many numbers ? 5

Enter 5 integers :

12 23 34 67 78

Enter the value to find : 34

34 found is positon 2

99. Frequency of a character. P266

#include <stdio.h>

int main()

{

char str[1000], ch;

int count = 0;

printf("Enter a string: ");

***/\*fgets(str, sizeof(str), stdin);\*/***

gets(str);

printf("Enter a character to find its frequency: ");

scanf("%c", &ch);

for (int i = 0; str[i] != '\0'; ++i) {

if (ch == str[i])

++count;

}

printf("Frequency of %c = %d", ch, count);

return 0;

}

Output:

Enter a string : kitao golam kibria ki obsta tomar

Enter a character to find its frequency : i

Frequency of i = 4

100. Merge Sorted List. P273

#include <stdio.h>

int main()

{

int A[] = { 21, 28, 35, 40, 61, 75 };

int B[] = { 16, 25, 47, 54 };

int C[20];

int i = 0; int j = 0; int k = -1;

int m = 6; int n = 4;

while (i < m || j < n) {

if (i == m)

C[++k] = B[j++];

else if (j == n)

C[++k] = A[i++];

else if (A[i] < B[j])

C[++k] = A[i++];

else

C[++k] = B[j++];

}

int x = m + n;

for (int h = 0; h < x; h++)

printf("%d ", C[h]);

}

Output:

16 21 25 28 35 40 47 54 61 75

আরো অনেক ইজি ভাবে এই সমস্যাটার সমাধান করে দেয়া আছে ।

সমস্যা ৪৭ – অ্যারের জোট এই সমাধানটা দেখো। (৫২ টি প্রোগ্রামিং সমস্যা)

101. Merge Sorted List(From the user)

#include <stdio.h>

int main()

{

int A[10];

int B[8];

int C[20];

int p, q;

printf("How many value for A? ");

scanf("%d", &p);

printf("Enter %d value for A : \n", p);

for (int z = 0; z < p; z++) {

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

}

printf("How many value for B? ");

scanf("%d", &q);

printf("Enter %d value for B : \n", q);

for (int y = 0; y < q; y++) {

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

}

int i = 0; int j = 0; int k = -1;

int m = p; int n = q;

while (i < m || j < n) {

if (i == m)

C[++k] = B[j++];

else if (j == n)

C[++k] = A[i++];

else if (A[i] < B[j])

C[++k] = A[i++];

else

C[++k] = B[j++];

}

int x = m + n;

for (int h = 0; h < x; h++)

printf("%d ", C[h]);

}

Output:

How many value for A ? 6

Enter 6 value for A :

21 28 35 40 61 75

How many value for B ? 4

Enter 4 value for B :

15 25 47 54

15 21 25 28 35 40 47 54 61 75

102. Merge Sorted List(Using Function)

#include <stdio.h>

int main()

{

int merge(int A[], int m, int B[], int n, int C[]);

int A[] = { 21, 28, 35, 40, 61, 75 };

int B[] = { 16, 25, 47, 54 };

int C[20];

int n = merge(A, 6, B, 4, C);

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

printf("%d ", C[h]);

}

int merge(int A[], int m, int B[], int n, int C[])

{

int i = 0; int j = 0; int k = -1;

while (i < m || j < n) {

if (i == m)

C[++k] = B[j++];

else if (j == n)

C[++k] = A[i++];

else if (A[i] < B[j])

C[++k] = A[i++];

else

C[++k] = B[j++];

}

return m + n;

}

Output:

16 21 25 28 35 40 47 54 61 75

103. Printing date using structure. P282

#include <stdio.h>

struct date {

int day;

int month;

int year;

};

int main()

{

struct date date\_of\_birth;

date\_of\_birth.day = 15;

date\_of\_birth.month = 11;

date\_of\_birth.year = 2000;

printf("%d/%d/%d\n", date\_of\_birth.day, date\_of\_birth.month, date\_of\_birth.year);

}

Output:

15/11/2000

104. Printing date using structure from the user. P282

#include <stdio.h>

struct date {

int day;

int month;

int year;

};

int main()

{

void printday(struct date d);

struct date date\_of\_birth;

date\_of\_birth.day;

date\_of\_birth.month;

date\_of\_birth.year;

scanf("%d %d %d", &date\_of\_birth.day, &date\_of\_birth.month, &date\_of\_birth.year);

printday(date\_of\_birth);

}

void printday(struct date d)

{

printf("%d/%d/%d\n", d.day, d.month, d.year);

}

Output:

15 11 2000

15/11/2000

105. printing date using structure by typedef

#include <stdio.h>

typedef struct {

int day;

int month;

int year;

}Date;

int main()

{

Date date\_of\_birth;

date\_of\_birth.day = 15;

date\_of\_birth.month = 11;

date\_of\_birth.year = 2000;

printf("%d/%d/%d\n", date\_of\_birth.day, date\_of\_birth.month, date\_of\_birth.year);

}

Output:

15 / 11 / 2000

106. Printing month, date and year by structure using character array.

#include <stdio.h>

typedef struct {

int day;

char month[4]; ***/\*Goes 0 to 3. 0,1,2 index is for Nov and 3 index for \0\*/***

int year;

}Date;

int main()

{

Date dob; ***/\*Here dob means date of birth\*/***

dob.day = 15;

strcpy(dob.month, "Nov");

dob.year = 2000;

printf("%s %d, %d\n", dob.month, dob.day, dob.year);

}

Output:

Nov 15, 2000

107. Array of structure. P285

#include <stdio.h>

struct person {

char name[30];

int age;

float salary;

char gender[2];

};

int main()

{

struct person man[2];

int i;

for (i = 0; i < 2; i++)

{

printf("Enter information for person %d : \n", i + 1);

printf("Name? ");

scanf("%s", &man[i].name);

printf("Age? ");

scanf("%d", &man[i].age);

printf("Salary? ");

scanf("%f", &man[i].salary);

printf("Gender? ");

scanf("%s", &man[i].gender);

}

for (i = 0; i < 2; i++)

{

printf("\nInformaton for person %d : \n", i + 1);

printf("Name : %s\n", man[i].name);

printf("Age : %d\n", man[i].age);

printf("Salary : %.2f\n", man[i].salary);

#include <stdio.h>

struct person {

char gender;

};

int main()

{

struct person man;

printf("Gender? ");

scanf("%c", &man.gender);

printf("%c", man.gender);

}

Output:

Gender ? M

M

printf("Gender : %s\n", man[i].gender);

}

}

Output:

Enter information for person 1 :

Name? kibria

Age? 20

Salary? 123.4

Gender? M

Enter information for person 2 :

Name? nina

Age? 20

Salary? 345.67

Gender? F

Informaton for person 1 :

Name : kibria

Age : 20

Salary : 123.40

Gender : M

Informaton for person 2 :

Name : nina

Age : 20

Salary : 345.67

Gender : F

#include <stdio.h>

typedef struct {

char gender;

}person;

int main()

{

person man;

printf("Gender? ");

scanf("%c", &man.gender);

printf("%c", man.gender);

}

Output:

Gender ? M

M

108. Search an array of structure. P286

#include <stdio.h>

struct employee {

int num;

char name[30];

int salary;

};

int main()

{

struct employee a[50];

int i; int n; int m;

printf("How many employee? ");

scanf("%d", &n);

printf("Enter employee information.\n"

"Number Name Salary\n");

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

scanf("%d %s %d", &a[i].num, &a[i].name, &a[i].salary);

}

printf("Which employee you want to search? ");

scanf("%d", &m);

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

if (m == a[i].num)

break;

}

if (i < n) printf("Enployee found at index %d\n", i);

else printf("Not found\n");

}

Output:

How many employee ? 4

Enter employee information.

Number Name Salary

10 saim 2500

20 gkez 2400

30 tausf 2600

40 motta 3000

Which employee you want to search ? 30

Enployee found at index 2

109. Sort an array of structure. P287

#include<stdio.h>

struct cricket {

char pname[20];

char tname[20];

float avg;

};

int main()

{

struct cricket player[10], temp;

int i, j, n;

printf("Enter the value of n? ");

scanf("%d", &n);

printf("Enter %d value.\n", n);

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

printf("\nEnter Player Name : ");

scanf("%s", player[i].pname);

printf("Enter Team Name : ");

scanf("%s", player[i].tname);

printf("Enter Average : ");

scanf("%f", &player[i].avg);

}

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

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

if (strcmp(player[j].tname, player[j + 1].tname) > 0) {

temp = player[j];

player[j] = player[j + 1];

player[j + 1] = temp;

}

}

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

printf("\n%s\t%s\t%.2f", player[i].pname, player[i].tname, player[i].avg);

}

}

Output:

Enter the value of n ? 3

Enter 3 value.

Enter Player Name : saim

Enter Team Name : bd

Enter Average : 45

Enter Player Name : tausif

Enter Team Name : sa

Enter Average : 56

Enter Player Name : kibria

Enter Team Name : pk

Enter Average : 50

saim bd 45.00

kibria pk 50.00

tausif sa 56.00

110. Read, Search and Sort a Structure.

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

typedef struct {

char name[31];

int age;

char gender;

}student;

int main()

{

void getString(FILE \* in, char str[]);

int getData(FILE \* in, student list[]);

int search(char key[], student list[], int n);

void sort(student list[], int n);

student pupil[100];

char name[31];

FILE\* in = fopen("inputfile.txt", "r");

if (in == NULL) {

printf("Error opening inputfile.\n");

exit(1);

}

int numofstudents = getData(in, pupil);

if (numofstudents == 0) {

printf("No data supplied for students.\n");

}

printf("\n");

for (int h = 0; h < numofstudents; h++)

printf("Name: %-15s Age: %2d\tGender: %1c\t\n", pupil[h].name, pupil[h].age, pupil[h].gender);

printf("\n");

getString(in, name);

while (strcmp(name, "END") != 0) {

int ans = search(name, pupil, numofstudents);

if (ans == -1) printf("%s not found\n", name);

else printf("%s found at location %d\n", name, ans);

getString(in, name);

}

sort(pupil, numofstudents);

printf("\n");

for (int h = 0; h < numofstudents; h++)

printf("Name: %-15s Age: %2d\tGender: %1c\t\n", pupil[h].name, pupil[h].age, pupil[h].gender);

} ***/\*End main\*/***

int getData(FILE\* in, student list[])

{

char temp[31];

void getString(FILE \* in, char str[]);

char readChar(FILE \* in);

int n = 0;

getString(in, temp);

while (n < 100 && strcmp(temp, "END") != 0) {

strcpy(list[n].name, temp);

fscanf(in, "%d", &list[n].age);

list[n].gender = readChar(in);

n++;

getString(in, temp);

}

return n;

} ***/\*End getData\*/***

int search(char key[], student list[], int n)

{

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

if (strcmp(key, list[h].name) == 0)return h;

return -1;

***/\*Search for key in list[0] to list[n-1].***

***If found return the location; if not found return -1.\*/***

} ***/\*End search\*/***

void sort(student list[], int n)

{

student temp;

int k;

for (int h = 1; h < n; h++) {

temp = list[h];

k = h - 1;

while (k >= 0 && strcmp(temp.name, list[k].name) < 0) {

list[k + 1] = list[k];

k = k - 1;

}

list[k + 1] = temp;

}

} ***/\*End sort\*/***

void getString(FILE\* in, char str[])

{

***/\*The string is read from the file in.***

***The first non-whitespace character is the delimiter.***

***Store in str, the next string within delimiter.\*/***

char ch, delimiter;

int n = 0;

str[0] = '\0';

***/\*Read over whitespace\*/***

while (isspace(ch = getc(in))); ***/\*Empty while body\*/***

if (ch == EOF) return;

delimiter = ch;

while (((ch = getc(in)) != delimiter) && (ch != EOF))

str[n++] = ch;

str[n] = '\0';

} ***/\*End getString\*/***

char readChar(FILE\* in)

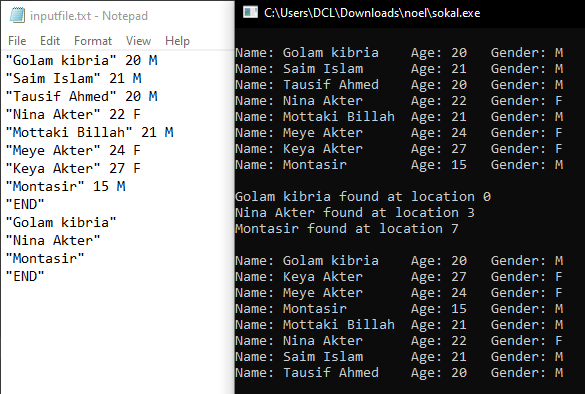
{

char ch;

while (isspace(ch = getc(in))); ***/\*Empty while body\*/***

return ch;

} ***/\*End readChar\*/***



111. Nested Structure. P292

#include <stdio.h>

struct address

{

char city[20];

int pin;

char phone[14];

};

struct employee

{

char name[20];

struct address add;

};

int main()

{

struct employee emp;

printf("Enter employee information?\n");

scanf("%s %s %d %s", emp.name, emp.add.city, &emp.add.pin,emp.add.phone);

printf("Printing the employee information....\n");

printf("name: %s\nCity: %s\nPincode: %d\nPhone: %s", emp.name, emp.add.city, emp.add.pin, emp.add.phone);

}

Output:

Enter employee information ?

kibria

dhaka

1212

01790037447

Printing the employee information....

name: kibria

City : dhaka

Pincode : 1212

Phone : 01790037447

112. Structure to represent a Fraction.

#include <stdio.h>

typedef struct {

int num;

int den; ***/\*den means denominator\*/***

}Fraction;

int main()

{

Fraction f;

f.num;

f.den;

scanf("%d %d", &f.num, &f.den);

printf("%d/%d", f.num, f.den);

}

Output:

5 9

5 / 9

#include <stdio.h>

typedef struct {

int num;

int den; ***/\*den means denominator\*/***

}Fraction;

int main()

{

Fraction f;

f.num = 5;

f.den = 9;

printf("%d/%d", f.num, f.den);

}

Output: 5 / 9

113. Manipulate Fraction by structure. P294

#include <stdio.h>

typedef struct {

int num1;

int den1;

int num2;

int den2;

}Fraction;

int main()

{

Fraction a, b;

a.num1 = 3; a.den1 = 7;

b.num2 = 5; b.den2 = 8;

int x = ((a.num1 \* b.den2) + (b.num2 \* a.den1));

int y = (a.den1 \* b.den2);

printf("%d/%d+%d/%d = %d/%d", a.num1, a.den1, b.num2, b.den2, x, y);

}

Output:

Output: 3 / 7 + 5 / 8 = 59 / 56

114. Pass Structures to Function. P304

#include <stdio.h>

struct student {

char firstname[64];

char lastname[64];

char id[64];

int score;

};

int main(void)

{

void displayDetail(struct student std);

struct student stdArr[3];

int i, n;

printf("How many students? "); scanf("%d", &n);

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

printf("Enter detail of %d student.\n", (i + 1));

printf("Enter First Name: ");

scanf("%s", stdArr[i].firstname);

printf("Enter Last Name: ");

scanf("%s", stdArr[i].lastname);

printf("Enter ID: ");

scanf("%s", stdArr[i].id);

printf("Enter Score: ");

scanf("%d", &stdArr[i].score);

}

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

printf("\nStudent %d Detail:\n", (i + 1));

displayDetail(stdArr[i]);

}

}

void displayDetail(struct student std)

{

printf("Firstname: %s\n", std.firstname);

printf("Lastname: %s\n", std.lastname);

printf("ID: %s\n", std.id);

printf("Score: %d\n", std.score);

}

Output:

How many students ? 1

Enter detail of 1 student.

Enter First Name : Golam

Enter Last Name : kibria

Enter ID : 2281

Enter Score : 10

Student 1 Detail:

Firstname: Golam

Lastname : kibria

ID : 2281

Score : 10