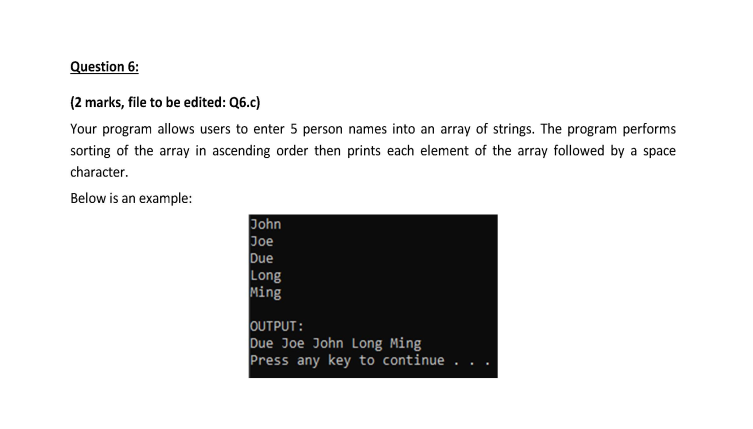
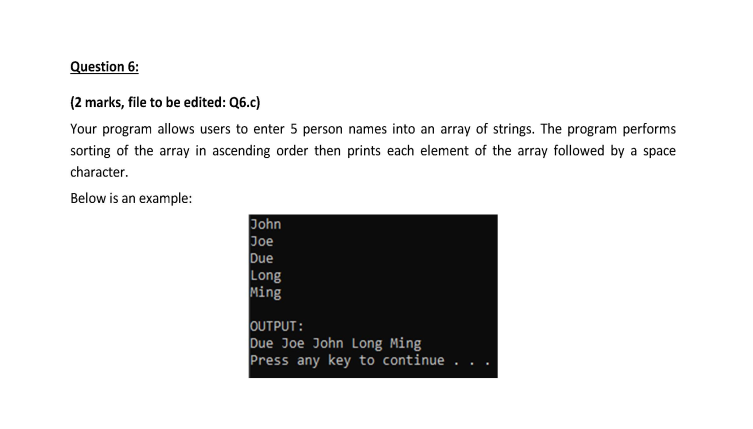
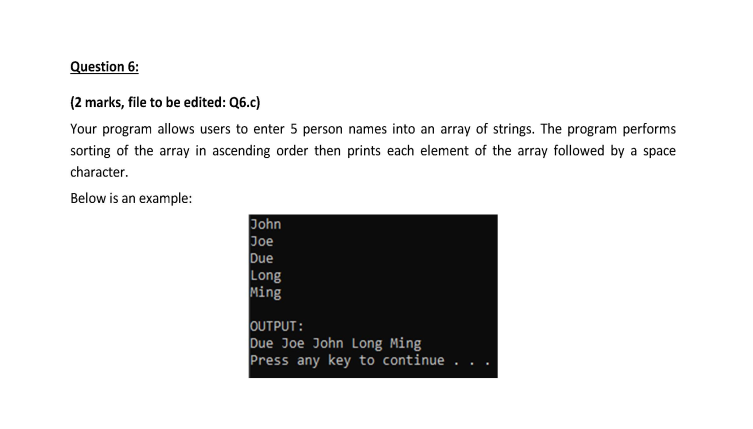
x 

#include <stdio.h>

char str[5][80], tmp[80];#include <stdlib.h>

#include <string.h>

#include <math.h>

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int i, j;

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

scanf("%s", str[i]);

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

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

{

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

{

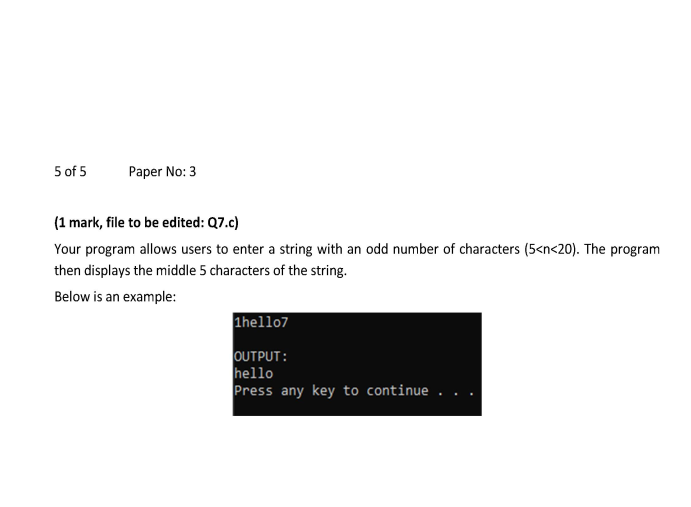
if (strcmp(str[i], str[j]) > 0)

{

// ham cmp so sanh 2 chuoi

// uu tien alphabt

strcpy(tmp, str[i]);

strcpy(str[i], str[j]); strcpy(str[j], tmp);

}

}

}

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

printf("%s ", str[i]);

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

#include <stdio.h>

1. #include <stdlib.h>
2. #include <string.h>
3. #include <math.h>
5. int main()
6. {
7. system("cls");
8. // INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:
   1. char str[100];
   2. gets(str);
   3. int j = strlen(str);
   4. int i;
10. // Fixed Do not edit anything here.
11. printf("\nOUTPUT:\n");
12. //@STUDENT: WRITE YOUR OUTPUT HERE:
13. for (i = j / 2 - 2; i <= j / 2 + 2; i++)
14. {
15. printf("%c", str[i]);=r3
16. }
18. //--FIXED PART - DO NOT EDIT ANY THINGS HERE
19. printf("\n");
20. system("pause");
21. return (0);
22. }
23. #include <stdio.h>
24. #include <stdlib.h>
25. #include <string.h>
26. #include <math.h>
28. int main()
29. {
30. system("cls");
31. // INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:
32. char [100];
33. gets(ststr[100];
34. gets(str);
35. //@STUDENT: WRITE YOUR OUTPUT HERE:
36. for (i = j / 2 - 2; i <= j / 2 + 2; i++)
37. {
38. printf("%c", str[i]);
39. }
41. //--FIXED PART - DO NOT EDIT ANY THINGS HERE
42. printf("\n");
43. system("pause");
44. return (0);

Đảo ngược số

//đảo ngược số

include <stdio.h>

int reverse(int n)

{

int reNum = n % 10; // b1 láº¥y chá»¯ sá»‘ cuá»‘i cÃ¹ng

n /= 10; // bá» chá»¯ sá»‘ cuá»‘i cÃ¹ng

int last;

while (n > 0)

{

last = n % 10; // láº¥y chá»¯ sá»‘ cuá»‘i cÃ¹ng

n /= 10; // bá» chá»¯ sá»‘ cuá»‘i cÃ¹ng

reNum = reNum \* 10 + last; // vÃ²ng láº·p Ä‘á»ƒ thá»±c hiá»‡n bÆ°á»›c 2 3 4

}

return reNum;#

}

int main()

{

int n;

printf("INPUT NUMBER: ");

scanf("%d", &n);

printf("REVERSE NUMBER OF %d IS %d #incldue", n, reverse(n));

return 0;

}

// armstrong

#include <stdio.h>

#include <math.h>

int countDigits(int num)

{

int count = 0;

while (num > 0)

{

num /= 10;

count++;

}

return count;

}

bool isArmstrong(int num)

{

int numDigit = countDigits(num);

int tmp = num;

int sum = 0;

int last;

while (tmp > 0)

{

last = tmp % 10;

tmp /= 10;

sum += pow(last, numDigit);

}

if (sum == num)

return true;

return false;

}

int main()

{

int num;

printf("input number: ");

scanf("%d", &num);

if (isArmstrong(num) == true)

1. int j = strlen(str);
2. int i;
4. // Fixed Do not edit anything here.
5. printf("\nOUTPUT:\n");

{

printf("%d is Armstrong number.", num);

}

else

{

printf("%d is not Armstrong number.", num);

}

//getch();

}

Text

Description automatically generated

#include <stdio.h>

int main()

{

    int n, reversed = 0, remainder, original;

    printf("Enter an integer: ");

    scanf("%d", &n);

    original = n;

    // reversed integer is stored in reversed variable

    while (n != 0 && n >= 0)

    {

        remainder = n % 10;

        reversed = reversed \* 10 + remainder;

        n /= 10;

    }

    // palindrome if orignal and reversed are equal

// ve hinh tam giac can chieu dai 2 canh = n nhu vi du

//Vi du n=4

/\*

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

\*/

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

int main() {

system("cls");

//INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int n;

scanf("%d", &n);

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

int i,j;

for(i = 0; i < 2\*n - 1;i++)//In cot dung de in 2\*n so cot chua sao nhu n=4 thi co 8 cot chua sao

{

//Trong vong for nay dung de in hang

for(j = 0; j < n;j++)//Trong 1 hang toi da la n sao

{

if(i >= n -1 - j && i <= n -1 +j)

/\*i>= n -1 - j: Dieu kien in ra so sao hang phia duoi\*/

/\*i <= n -1 +j: Dieu kien in ra so sao o hang phia tren\*/

Qq printf("\*");

else printf(" ");

}

printf("\n");

}

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system ("pause");

return(0);

}

//Nhap vao so nguyen duong n va n so nguyen. Tim so xuat hien nhieu nhat

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

int main() {

system("cls");

//INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int n;

int arr[1000], b[1000] = {0}, c[1000] = {0}; //Khoi tao 3 mang so nguyen

int i;

scanf("%d", &n); //Nhap so phan tu cua mang arr

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

{

scanf("%d", &arr[i]); // Nhap n phan tu mang so nguyen arr

}

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

{

if(arr[i] > 0) //Mang so nguyen b[] ghi lai so lan xuat hien cua

b[arr[i]]++; //cac phan tu lon hon 0 trong day arr

if(arr[i] < 0) //Mang so nguyen c[] ghi lai so lan xuat hien cua

c[-arr[i]]++; //cac phan tu nho hon 0 trong day arr

}

int max = 0;

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

{

if(arr[i] > 0)

{

if(b[arr[i]] > max)

max = b[arr[i]];

}

else

{

if(c[-arr[i]] > max)

max = c[-arr[i]];

}

}// Vong lap tra ra so lan xuat hien nhieu nhat cua mot phan tu trong arr

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

for(i = 0; i <1000 ;i++)

{

if(b[i] == max)

{

printf("%d", i);

}

if(c[i] == max)

printf("%d", -i);

} //In ra phan tu co lan xuat hien nhieu nhat trong day

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system ("pause");

return(0);

}

//Tinh tong sum = 1/x + 1/x^2 + 1/x^3 +... + 1/x^n

//Voi x va n la so nguyen nhap tu ban phim

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

int main() {

system("cls");

//INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int x,n;

scanf("%d%d", &x, &n); //Nhap vao hai so nguyen x va n

double sum = 0; //Khoi tao gia tri cua tong bang khong

int i;

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

{

sum+= 1 / (pow(x,i)); //pow(x,i) la ham tinh luy thua bac i cua x

}

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

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

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system ("pause");

return(0);

}

// nhap vao mot chuoi bat ki, xoa het cac ki tu và so, chi giu lai cac chu cai

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

#include <stdbool.h>

int main() {

system("cls");

//INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

char str[100];

scanf("%[^\n]", str);

int i,j;

bool all\_al = false;

while(all\_al == false) // lap de xoa ki tu khong phai chu cai

{

all\_al = true;

for(i = 0; i < strlen(str);i++)

{

if(!isalpha(str[i])) // kiem tra xem str[i] co phai la chu cai khong

{

for(j = i; j < strlen(str) ; j++) // xoa str[i] neu str[i] khong phai chu cai

{

str[j] = str[ j + 1];

}

}

}

for( i = 0; i < strlen(str);i++) // kiem tra trong str co ki tu nao khong phai chu cai khong

{

if(!isalpha(str[i]))

all\_al = false;

}

}

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

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

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system ("pause");

return(0);

}

//nhap vao mot so tu nhien n, in ra 4 so nguyen to gan nhat lon hon n

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

#include <stdbool.h>

bool isPrime(int n) // ham kiem tra so nguyen to

{

if( n < 2)

return false;

int i;

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

{

if(n % i == 0)

return false;

}

return true;

}

int main() {

system("cls");

//INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int count = 4;

int n;

scanf("%d", &n);

int i = 1;

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

while(count > 0) // in ra 4 so nguyen to lon hon n

{

if(isPrime(n + i))

{

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

count--;

}

i++;

}

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

system ("pause");

return(0);

}

Graphical user interface, text, application

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

int sochan\_max(int a[], int n)

{

int max;

int i = 0;

while (a[i] % 2 != 0)

i++;

max = a[i];

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

if (a[j] % 2 == 0)

if (max < a[j])

max = a[j];

return max;

}

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int arr[5];

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

{

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

}

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

printf("%d", sochan\_max(arr, 5));

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

Graphical user interface

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

int main()

{

// system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int arr[5];

int i, j;

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

{

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

}

// Fixed Do not edit anything here.

printf("OUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

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

{

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

{

if (arr[j] < arr[j - 1])

{

int tmp = arr[j];

arr[j] = arr[j - 1];

arr[j - 1] = tmp;

}

}

}

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

{

if (arr[i] % 2 == 0)

{

printf("%d\n", arr[i]);

}

}

}

Text

Description automatically generated with medium confidence

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int n, arr[10], i, check = 0, j, k;

scanf("%d", &n);

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

{

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

}

for (i = 0; i < n - 1; i++)

{

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

{

if (arr[i] == arr[j] && arr[i] % 2 != 0)

{

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

{

arr[k] = arr[k + 1];

}

n--;

j--;

}

}

}

// Fixed Do not edit anything here.

printf("OUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

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

{

if (check == 0)

check = 1;

else

printf("\n");

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

}

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

Graphical user interface, text, application

Description automatically generated

#include <stdio.h>

#include <string.h>

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

char s[100];

gets(s);

int i, j, count = 0;

char tmp[50][50];

for (i = 0; i < strlen(s); i++)

{

j = 0;

while (s[i] != ' ' && s[i] != '\0')

{

tmp[count][j] = s[i];

j++;

i++;

}

tmp[count][j] = '\0';

if (tmp[count][0] == 'h' && tmp[count][j - 1] == 'g')

count++;

}

// Fixed Do not edit anything here.

printf("\nOUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

printf("%d", count);

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

Graphical user interface, text

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

int main() ,

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int n, i, str[21], j, k, count = 0;

char c[21];

scanf("%d ", &n);

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

{

scanf("%c", &c[i]);

getchar();

}

// Fixed Do not edit anything here.

printf("OUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

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

{

str[i] = 1;

for (j = 0; j < n + 1; j++)

{

if (i != j)

{

if (c[i] == c[j])

{

str[i]++;

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

c[k] = c[k + 1];

n--;

j--;

}

}

}

}

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

{

if (str[i] >= 2)

{

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

if (count != 1)

printf("\n");

count++;

if (count == 2)

break;

}

}

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

Graphical user interface, text

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int a, tmp, flag = 0;

scanf("%d", &a);

tmp = a;

while (a % 2 == 0)

{

a /= 2;

flag++;

}

// Fixed Do not edit anything here.

printf("OUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

if (a == 1)

{

printf("%d", flag);

}

else

{

printf("%d is not a power of 2", tmp);

}

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

Graphical user interface, text

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

char s[100];

int i, count = 0, tmp = 0;

gets(s);

// Fixed Do not edit anything here.

printf("OUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

for (i = 0; i < strlen(s); i++)

{

if (s[i] != ' ' && tmp < 3)

{

while (s[i] != ' ')

{

count++;

++i;

;

}

tmp++;

}

}

printf("%d", count);

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

Graphical user interface, text

Description automatically generated

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <math.h>

#include <ctype.h>

int checkprime(int n)

{

int i;

if (n < 2)

return 0;

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

if (n % i == 0)

return 0;

return 1;

}

int main()

{

system("cls");

// INPUT - @STUDENT:ADD YOUR CODE FOR INPUT HERE:

int n;

scanf("%d", &n);

// Fixed Do not edit anything here.

printf("OUTPUT:\n");

//@STUDENT: WRITE YOUR OUTPUT HERE:

if (checkprime(n))

{

printf("0x%X", n);

}

else

{

printf("%d is not a prime number", n);

}

//--FIXED PART - DO NOT EDIT ANY THINGS HERE

printf("\n");

system("pause");

return (0);

}

String:

Đếm số kí tự số trong xâu nhập từ bàn phím.

#include <stdio.h>

#include <conio.h>

#include <string.h>

int demktso(char s[])

{

int i,dem=0;

for (i=0;i<strlen(s);i++)

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

dem++;

return dem;

}

void main()

{

char s[100];

gets(s);

printf("so ki tu so la : %d",demktso(s));

getch();

}

Đếm số kí tự hoa trong xâu nhập từ bàn phím.

#include <stdio.h>

#include <conio.h>

#include <string.h>

int demkthoa(char s[])

{

int i,dem=0;

for (i=0;i<strlen(s);i++)

if ( s[i]>='A' && s[i]<='Z' )

dem++;

return dem;

}

void main()

{

char s[100];

gets(s);

printf("so ki tu hoa la : %d",demkthoa(s));

getch();

}

Đếm số kí tự thường trong xâu nhập từ bàn phím.

#include <stdio.h>

#include <conio.h>

#include <string.h>

int demktthuong(char s[])

{

int i,dem=0;

for (i=0;i<strlen(s);i++)

if ( s[i]>='a' && s[i]<='z' )

dem++;

return dem;

}

void main()

{

char s[100];

gets(s);

printf("so ki tu thuong la : %d",demktthuong(s));

getch();

}

Đếm các số tự nhiên trong xâu nhập từ bàn phím. Các kí tự số gần nhau ghép thành 1 số tự nhiên.

Ví dụ: a123bc4d56ef

cho ra đáp án là : 3

#include <stdio.h>

#include <conio.h>

#include <string.h>

void main()

{

char s[100];

int i=0,j,dem=0;

gets(s);

while (i<=strlen(s))

{

j=0;

while ( s[i]>='0' && s[i]<='9' )

{

i++;

j++;

}

i++;

if (j!=0) dem++;

}

printf("%d",dem);

getch();

}

## Tính độ dài của một chuỗi C

#include <stdio.h> int main() {  
    char s[] = "Chương trinh đang chạy";  
    int i;  
    for (i = 0; s[i] != '\0'; ++i);  
      printf("Độ dài chuỗi: %d", i);  
    return 0;  
}

 nối hai chuỗi trong C

#include <stdio.h> int main() {  
  char s1[100] = "Lập trình ", s2[] = "c cơ bản";  
  int length, j;  
  // chiều dài cửa hàng của s1 trong biến chiều dài  
  length = 0;  
  while (s1[length] != '\0') {  
    ++length;  
  }  
  // nối s2 thành s1  
  for (j = 0; s2[j] != '\0'; ++j, ++length) {  
    s1[length] = s2[j];  
  }  
  // chấm dứt chuỗi s1  
  s1[length] = '\0';  
  printf("Chuỗi sau cùng: ");  
  puts(s1);  
  return 0;  
}

1) Nhập vào 1 chuỗi và xuất chuỗi đó ra theo chiều ngược lại:  
  
VD Nhập vào tran van thoa xuất ra aoht nav nart

#include <conio.h> #include <stdio.h> #include <string.h>  //thư viện chuỗi  
  
int main()  
{  
    char xau[30];  
    printf("Nhap vao 1 chuoi: ");  
    gets(xau);  
    for(int i=strlen(xau)-1;i>=0;i--)  //strlen trả về độ dài của chuỗi  
    {  
        printf("%c",xau[i]);  
    }  
    getch();  
}

2) Nhập vào 1 chuỗi và xuất chuỗi đó ra theo chiều ngược lại:  
  
VD Nhập vào tran van thoa xuất ra thoa van tran

#include <stdio.h> #include <conio.h> #include <string.h>  
  
int main()  
{  
   char xau[50];  
   printf("Nhap vao mot chuoi: ");  
   gets(xau);  
   int a=strlen(xau)-1;  
   for(int i=strlen(xau)-1;i>=0;i--)  
   {  
      if(xau[i]==32 || i==0)  
      {  
         if(i==0)  
         {  
            printf(" ");  
         }  
         for(int j=i;j<=a;j++)  
         {  
            printf("%c",xau[j]);  
         }  
         a=i-1;  
      }  
   }  
   getch();  
}

3) Nhập vào họ và tên tách ra họ, tên;  
  
VD Nhập vào tran van thoa xuất ra tran thoa

#include <conio.h> #include <stdio.h> #include <string.h>  
  
int main()  
{  
    char xau[30];  
    printf("Nhap vao mot chuoi: ");  
    gets(xau);  
    for(int i=0;i<strlen(xau);i++)   
    {  
        if(xau[i]!=32)  
        {  
            printf("%c",xau[i]);  
        }  
        else         {  
         for(int j=strlen(xau)-1;j>=i;j--)  
         {  
            if(xau[j]==32)  
            {  
               for(int k=j;k<=strlen(xau)-1;k++)  
               printf("%c",xau[k]);  
               break;  
            }  
         }  
         break;  
      }  
    }  
    getch();  
}

4) Nhập vào họ và tên xuất ra họ, tên đệm, tên mỗi từ 1 dòng;  
  
VD Nhập vào tran van thoa xuất ra  
tran  
van  
thoa  
#include <conio.h> #include <stdio.h> #include <string.h>  
  
int main()  
{  
    char xau[30];  
    printf("Nhap vao mot chuoi: ");  
    gets(xau);  
    for(int i=0;i<=strlen(xau)-1;i++)  
    {  
      if(xau[i]!=32)  
      {  
          printf("%c",xau[i]);  
      }  
        else       {  
            printf("\n");  
      }  
    }  
    getch();  
}

5) Nhập vào 1 dãy số và đọc dãy số đó.  
  
VD: 123 đọc là một trăm hai mươi ba

#include <conio.h> #include <stdio.h> #include <string.h>  
  
char doc\_so[50];  
char \*docso(int n)  
{  
   char doc[10][5]={"","Mot","Hai","Ba","Bon","Nam","Sau","Bay","Tam","Chin"};  
   doc\_so[0]=0;  
   int donvi=n%10;  
   n=n/10;  
   int chuc=n%10;  
   int tram=n/10;  
   if(tram>0)  
   {  
      strcat(doc\_so,doc[tram]);  
      strcat(doc\_so," Tram ");  
   }  
   if(chuc>0)  
   {  
      if(chuc==1)  
      strcat(doc\_so," Muoi ");  
      else       {  
         strcat(doc\_so,doc[chuc]);  
         strcat(doc\_so," Muoi ");  
      }  
   }  
   if(donvi>0)  
   strcat(doc\_so,doc[donvi]);  
   return doc\_so;  
}  
int main()  
{  
   int n;  
    printf("Nhap vao mot day so: ");  
    scanf("%d",&n);  
    if(n==0)  
    {  
      printf("Khong");  
   }  
   else    {  
      int tram=n%1000;  
      n=n/1000;  
      int ngan=n%1000;  
      n=n/1000;  
      int trieu=n%1000;  
      int ty=n/1000;  
      if(ty>0)  
      {  
         printf("%s Ty",docso(ty));  
      }  
      if(trieu>0)  
      {  
         printf(" %s Trieu ",docso(trieu));  
      }  
      if(ngan>0)  
      {  
         printf(" %s Ngan ",docso(ngan));  
      }  
      if(tram>0)  
      {  
         printf(" %s ",docso(tram));  
      }  
   }  
   getch();  
}

6) Nhập vào 1 chuỗi sau đó nhập vào váo 1 từ và kiểm tra xem từ đó có xuất hiện trong chuỗi trên hay không, nếu có thì xuất hiện bao nhiêu lần.  
  
VD Nhập vào tran van thoa. Nhập kí tự t --> có 2 lần

#include <stdio.h> #include <conio.h> #include <string.h>  
  
int main()  
{  
    char xau[50];  
    char kitukiemtra;  
    int dem;  
    printf("Nhap vao mot chuoi: ");  
    gets(xau);  
    printf("Nhap vao ki tu muon kiem tra: ");  
    scanf("%c",&kitukiemtra);  
    for(int i=0;i<strlen(xau)-1;i++)  
    {  
        if(xau[i]==kitukiemtra)  
        dem++;  
    }  
    if(dem==0)  
    printf("Ki tu %c khong co trong chuoi",kitukiemtra);  
    else     printf("Ki tu %c xuat hien %d lan trong chuoi",kitukiemtra,dem);  
    getch();  
}

Hình

\*

\* \*

\* \* \*

\* \* \* \*

\* \* \* \* \*

1. #include <stdio.h>
2. int main() {

    if (original == reversed)

        printf("%d is a palindrome.", original);

    else

        printf("%d is not a palindrome.", original);

    return 0;

}

1. int i, j, rows;
2. printf("Enter the number of rows: ");
3. scanf("%d", &rows);
4. for (i = 1; i <= rows; ++i) {
5. for (j = 1; j <= i; ++j) {
6. printf("\* ");
7. }
8. printf("\n");
9. }
10. return 0;
11. }

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

1. #include <stdio.h>
2. int main() {
3. int i, j, rows;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = 1; i <= rows; ++i) {
7. for (j = 1; j <= i; ++j) {
8. printf("%d ", j);
9. }
10. printf("\n");
11. }
12. return 0;
13. }

A

B B

C C C

D D D D

E E E E E

1. #include <stdio.h>
2. int main() {
3. int i, j;
4. char input, alphabet = 'A';
5. printf("Enter an uppercase character you want to print in the last row: ");
6. scanf("%c", &input);
7. for (i = 1; i <= (input - 'A' + 1); ++i) {
8. for (j = 1; j <= i; ++j) {
9. printf("%c ", alphabet);
10. }
11. ++alphabet;
12. printf("\n");
13. }
14. return 0;
15. }

\* \* \* \* \*

\* \* \* \*

\* \* \*

\* \*

\*

1. #include <stdio.h>
2. int main() {
3. int i, j, rows;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = rows; i >= 1; --i) {
7. for (j = 1; j <= i; ++j) {
8. printf("\* ");
9. }
10. printf("\n");
11. }
12. return 0;
13. }

1 2 3 4 5

1 2 3 4

1 2 3

1 2

1

1. #include <stdio.h>
2. int main() {
3. int i, j, rows;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = rows; i >= 1; --i) {
7. for (j = 1; j <= i; ++j) {
8. printf("%d ", j);
9. }
10. printf("\n");
11. }
12. return 0;
13. }

\*

\* \* \*

\* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

1. #include <stdio.h>
2. int main() {
3. int i, space, rows, k = 0;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = 1; i <= rows; ++i, k = 0) {
7. for (space = 1; space <= rows - i; ++space) {
8. printf(" ");
9. }
10. while (k != 2 \* i - 1) {
11. printf("\* ");
12. ++k;
13. }
14. printf("\n");
15. }
16. return 0;
17. }

1

1

2 3 2

3 4 5 4 3

4 5 6 7 6 5 4

5 6 7 8 9 8 7 6 5

1. #include <stdio.h>
2. int main() {
3. int i, space, rows, k = 0, count = 0, count1 = 0;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = 1; i <= rows; ++i) {
7. for (space = 1; space <= rows - i; ++space) {
8. printf(" ");
9. ++count;
10. }
11. while (k != 2 \* i - 1) {
12. if (count <= rows - 1) {
13. printf("%d ", i + k);
14. ++count;
15. } else {
16. ++count1;
17. printf("%d ", (i + k - 2 \* count1));
18. }
19. ++k;
20. }
21. count1 = count = k = 0;
22. printf("\n");
23. }
24. return 0;
25. }

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*

1. #include <stdio.h>
2. int main() {
3. int rows, i, j, space;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = rows; i >= 1; --i) {
7. for (space = 0; space < rows - i; ++space)
8. printf(" ");
9. for (j = i; j <= 2 \* i - 1; ++j)
10. printf("\* ");
11. for (j = 0; j < i - 1; ++j)
12. printf("\* ");
13. printf("\n");
14. }
15. return 0;
16. }

1

1 1

1 2 1

1 3 3 1

1 4 6 4 1

1 5 10 10 5 1

1. //Pascal's Triangle
2. #include <stdio.h>
3. int main() {
4. int rows, coef = 1, space, i, j;
5. printf("Enter the number of rows: ");
6. scanf("%d", &rows);
7. for (i = 0; i < rows; i++) {
8. for (space = 1; space <= rows - i; space++)
9. printf(" ");
10. for (j = 0; j <= i; j++) {
11. if (j == 0 || i == 0)
12. coef = 1;
13. else
14. coef = coef \* (i - j + 1) / j;
15. printf("%4d", coef);
16. }
17. printf("\n");
18. }
19. return 0;
20. }

1

2 3

4 5 6

7 8 9 10

1. #include <stdio.h>
2. int main() {
3. int rows, i, j, number = 1;
4. printf("Enter the number of rows: ");
5. scanf("%d", &rows);
6. for (i = 1; i <= rows; i++) {
7. for (j = 1; j <= i; ++j) {
8. printf("%d ", number);
9. ++number;
10. }
11. printf("\n");
12. }
13. return 0;
14. }

A picture containing night sky

Description automatically generated

#include <stdio.h>

int main()

{

int i, j, n;

scanf("%d", &n);

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

{

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

{

if (i == 1 || i == n || i == j)

{

printf("\* ");

}

else

printf(" ");

}

printf("\n");

}

}

Chart, scatter chart

Description automatically generated

#include <stdio.h>

int main()

{

int i, j, n;

scanf("%d", &n);

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

{

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

{

if (i == 1 || i == n || i == j || i + j == n + 1)

{

printf("\* ");

}

else

printf(" ");

}

printf("\n");

}

}

/\*

9

\* \* \* \* \* \* \* \* \*

\* \* \* \*

\* \* \* \*

\* \* \* \*

\* \* \*

\* \* \* \*

\* \* \* \*

\* \* \* \*

\* \* \* \* \* \* \* \* \*

\*/

#include <stdio.h>

int main()

{

int i, j, n;

scanf("%d", &n);

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

{

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

{

if (i == 1 || i == j || j == 1 || j == n || i == n || i + j == n + 1)

printf("# ");

else

printf(" ");

}

printf("\n");

}

}

/\*

5

\*\*\*\*\*

\* \*

\* \*

\* \*

\*\*\*\*\*

\*/

#include <stdio.h>

int main()

{

int i, j, n;

scanf("%d", &n);

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

{

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

{

printf(" ");

}

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

{

if (i == 1 || j == 1 || i == n || j == n)

{

printf("\*");

}

else

printf(" ");

}

printf("\n");

}

}

/\*

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

\*\*\* \*\*\*

\*\*\* \*\*\*

\*\*\* \*\*\*

\*/

#include <stdio.h>

int main()

{

int i, j, space, rows = 8, star = 0;

/\* Printing upper triangle \*/

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

{

if (i < 5)

{

/\* Printing upper triangle \*/

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

{

printf(" ");

}

/\* Printing stars \*/

while (star != (2 \* i + 1))

{

printf("\*");

star++;

;

}

star = 0;

/\* move to next row \*/

printf("\n");

}

else

{

/\* Printing bottom walls of huts \*/

for (j = 0; j < 9; j++)

{

if ((int)(j / 3) == 1)

printf(" ");

else

printf("\*");

}

printf("\n");

}

}

return 0;

}

input n: 5

\*

\* \* \*

\* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*

#include <stdio.h>

int main()

{

int n;

printf("input n: ");

scanf("%d", &n);

//thoi2

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

{

for (int j = 1; j <= n - i; j++)

{

printf(" ");

}

for (int j = 1; j <= 2 \* i - 1; j++)

{

printf(" \* ");

}

printf("\n");

}

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

{

for (int j = 1; j <= n - i; j++)

{

printf(" ");

}

for (int j = 1; j <= 2 \* i - 1; j++)

{

printf(" \* ");

}

printf("\n");

}

}

5

\* \* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \*

\* \* \* \*

\* \*

\* \* \* \*

\* \* \* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \* \*

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

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

{

for (int j = 1; j <= 2 \* n; j++)

{

if (j <= n - i + 1 || j >= n + i)

{

printf(" \* ");

}

else

{

printf(" ");

}

}

printf("\n");

}

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

{

for (int j = 1; j <= 2 \* n; j++)

{

if (j <= n - i + 1 || j >= n + i)

{

printf(" \* ");

}

else

{

printf(" ");

}

}

printf("\n");

}

}

/\*

Enter the number of columns5

\*\*\*\*\*

\*\*\*\*

\*\*\*

\*\*

\*

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

\*/

#include <stdio.h>

int main(void)

{

int n;

printf("Enter the number of columns");

scanf("%d", &n);

// printing the upper part of the pattern..

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

{

for (int j = 1; j <= n - i; j++)

{

printf(" ");

}

for (int k = 0; k <= n - i; k++)

{

printf(" \* ");

}

printf("\n");

}

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

{

for (int j = 1; j < i + 1; j++)

{

printf(" ");

}

for (int k = 1; k <= i + 1; k++)

{

printf(" \* ");

}

printf("\n");

}

return 0;

}

/\*

Enter the odd number only5

+

+

+ + + + +

+

+

\*/

#include <stdio.h>

int main(void)

{

int n;

printf("Enter the odd number only");

scanf("%d", &n);

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

{

if (i == ((n / 2) + 1))

{

for (int j = 1; j <= n; j++)

{

printf(" + ");

}

}

else

{

for (int j = 1; j <= n / 2; j++)

{

printf(" ");

}

printf(" + ");

}

printf("\n");

}

return 0;

}

5

1

1 2 1

1 2 3 2 1

1 2 3 4 3 2 1

1 2 3 4 5 4 3 2 1

#include <stdio.h>

#include <math.h>

void tamGiacThuong(int h)

{

for (int i = 1; i <= h; i++)

{

for (int j = 1; j < 2 \* h; j++)

{

if (abs(h - j) <= (i - 1))

{

printf("%3d", i - abs(h - j));

}

else

{

printf(" ");

}

}

printf("\n");

}

}

int main()

{

int h;

scanf("%d", &h);

tamGiacThuong(h);

return 0;

}

Enter the number of rows5

1 2 3 4 5 4 3 2 1

1 2 3 4 3 2 1

1 2 3 2 1

1 2 1

1

#include <stdio.h>

#include <stdlib.h>

int main() {

int i,j,rows,space=0;

printf("Enter the number of rows");

scanf("%d",&rows);//taking numer of rows from user

for(i=rows; i>=1; i--){

//outer for loop

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

printf(" ");

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

printf("%d ",j);

for(j=i-1; j>=1; j--)

printf("%d ",j);

printf("\n");

space++;

}

getch();

return 0;

}

/\*

8

\*

\*\*

\* \*

\* \*

\* \*

\* \*

\* \*

\*\*\*\*\*\*\*\*

\*/

#include <stdio.h>

int main()

{

int i, j, n;

scanf("%d", &n);

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

{

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

{

if (j == 1 || i == n || i == j)

printf("\*");

else

printf(" ");

}

printf("\n");

}

}

/\*

Enter the number of rows: 5

\* \* \* \*

\* \* \* \* \* \* \* \*

\* \* \* \* \* \* \* \* \*

\* \* \* \* \* \* \*

\* \* \* \* \*

\* \* \*

\*

\*/

#include <stdio.h>

int main()

{

int i, j, rows;

printf("Enter the number of rows: ");

scanf("%d", &rows);

/\* printing top semi circular shapes of heart \*/

for (i = rows / 2; i <= rows; i += 2)

{

/\* Printing Spaces \*/

for (j = 1; j < rows - i; j += 2)

{

printf(" ");

}

/\* printing stars for left semi circle \*/

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

{

printf(" \* ");

}

/\* Printing Spaces \*/

for (j = 1; j <= rows - i; j++)

{

printf(" ");

}

/\* printing stars for right semi circle \*/

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

{

printf(" \* ");

}

/\* move to next row \*/

printf("\n");

}

/\* printing inverted start pyramid \*/

for (i = rows; i >= 1; i--)

{

for (j = i; j < rows; j++)

{

printf(" ");

}

for (j = 1; j <= (i \* 2) - 1; j++)

{

printf(" \* ");

}

/\* move to next row \*/

printf("\n");

}

return 0;

}