1. Viết chương trình hiện ra màn hình hai câu thông báo sau:

S1[]=“Chao ban! ”

S2[]=“Chuc mung ban den voi mon hoc Nhap Mon Lap Trinh.”

Yêu cầu: nối 2 chuỗi trên thành 1, và hiển thị in hoa.

Kết quả: “CHAO BAN! CHUC MUNG BAN DEN VOI MON HOC NHAP MON LAP TRINH.”

Đáp án:

#include <stdio.h>

#include <string.h>

void main()

{

char S1[] = "Chao ban!";

char S2[] = "Chuc mung ban den voi nhap mon lap trinh";

strupr(strcat(S1, S2));

printf("chuoi: %s", S1);

}

1. Viết chương trình xuất chuỗi viết hoa ở câu 1. ra màn hình 10 dòng cùng nội dung.

#include <stdio.h>

#include <string.h>

void main()

{

char S1[100] = "Chao ban!";

char S2[] = "Chuc mung ban den voi nhap mon lap trinh";

strcat(S1, S2);

strupr(S1);

int i=0;

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

{

printf("%s\n", S1);

}

}

1. Viết chương trình ghi chuỗi viết hoa ở câu 1. Thành tập tin kiểu text tên ontapchuoi.txt lưu vào ổ c. Sau đó đọc nội dung từ tập tin này ra màn hình.

#include <stdio.h>

#include <conio.h>

#include <string.h>

void main()

{

//3.a

file \*fw = fopen("c:\\ontap.txt","w");

//3.b

file \*fr = fopen("c:\\ontap.txt","r");

char s1[100]="chao ti ";

char s2[100]="dai hoc hoa sen !";

strupr(strcat(s1,s2));

int i;

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

{

// printf("%s\n",s1);

fprintf(fw,"%s\n",s1);

}

fclose(fw);

if (fr== null)

{

printf("error: could not open file");

return 1;

}

// read one character at a time and

// display it to the output

printf("content file is \n");

char ch;

while ((ch = fgetc(fr)) != eof)

putchar(ch);

// close the file

fclose(fr);

return 0;

1. Viết chương trình đếm số kí tự sau khi thực hiện nối chuỗi ở câu 1.

    for(i = 0; i <= strlen(S1); i++)

    {

        if(isalpha(S1[i]))

            letter++;

    }

    printf("Letter: %d\n\n",letter);

1. Viết chương trình cho phép người dùng nhập vào chuỗi bất kỳ, hiển thị chuỗi đó ra màn hành 5 dòng.

  char c[50];

  printf("Enter word: ");

  gets(c);

    int i;

    for ( i = 0; i < 5; i++)

    {

      printf("%s\n",c);

    }

1. Nhập vào 1 số từ 1-7, hiển thị thứ tiếng anh tương ứng. Nếu người dùng không nhập trong các số này chương trình nên hiển thị thông điệp “enter again, pls”.

    int ch;

    printf("Choose number[1-7]: ");

    scanf("%d",&ch);

    switch (ch)

    {

    case 1: printf("Monday"); break;

    case 2: printf("Tuesday"); break;

    case 3: printf("wednesday"); break;

    case 4: printf("Thursday"); break;

    case 5: printf("Friday"); break;

    case 6: printf("Saturday"); break;

    case 7: printf("Sunday"); break;

    default: printf("Enter again, pls!"); break;

    }

1. Nhập vào 1 số từ 1-12, hiển thị tháng tiếng anh tương ứng. Nếu người dùng không nhập trong các số này chương trình nên hiển thị thông điệp “enter again, pls”.

    int ch;

    printf("Choose number[1-12]: ");

    scanf("%d",&ch);

    switch (ch)

    {

    case 1: printf("Jenuary"); break;

    case 2: printf("February"); break;

    case 3: printf("March"); break;

    case 4: printf("April"); break;

    case 5: printf("May"); break;

    case 6: printf("June"); break;

    case 7: printf("July"); break;

    case 8: printf("August"); break;

    case 9: printf("September"); break;

    case 10: printf("October"); break;

    case 11: printf("November"); break;

    case 12: printf("December"); break;

    default: printf("Enter again, pls!"); break;

    }

1. Nhập vào dấu +-\*/% thì chương trình gọi hàm thực hiện hiện tính toán theo dấu tương ứng. Nếu người dùng không nhập trong các kí hiệu này thì chương trình nên hiển thị thông điệp “no known, pls, enter again: +-\*/%”.

int cong(int a, int b);

int tru(int a, int b);

int nhan(int a, int b);

int chia(int a, int b);

int chiadu(int a, int b);

int main()

{

    int a,b;

    char ch;

    printf("Enter value a: ");

    scanf("%d",&a);

    printf("Enter value b: ");

    scanf("%d",&b);

    printf("Choose number[+,-,\*,/,%]: ");fflush(stdin);

    scanf("%c",&ch);

    switch (ch)

    {

        case '+': printf("phep cong: %d",cong(a, b)); break;

        case '-': printf("phep cong: %d",tru(a, b)); break;

        case '\*': printf("phep cong: %d",nhan(a, b)); break;

        case '/': printf("phep cong: %d",chia(a, b)); break;

        case '%': printf("phep cong: %d",chiadu(a, b)); break;

        default: printf("No known, pls! enter again: +-\*/%"); break;

    }

    getch();

}

int cong(int a, int b)

{

    int s = 0;

    s = a + b;

        return s;

}

int tru(int a, int b)

{

    int s = 0;

    s = a - b;

        return s;

}

int chia(int a, int b)

{

    int s = 0;

    s = a / b;

        return s;

}

int nhan(int a, int b)

{

    int s = 0;

    s = a \* b;

        return s;

}

int chiadu(int a, int b)

{

    int s = 0;

    s = a % b;

        return s;

}

1. Nhập vào 1 mảng tối đa 10 phần tử nguyên, tính tổng các số nguyên đã nhập vào mảng.

#include <stdio.h>

#include <conio.h>

#include <string.h>

void main()

{

int n;

do

{

printf("enter n: ");

scanf("%d",&n);

}while(n<=0||n>10);

int a[n];

int sum=0;

int i;

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

{

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

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

sum+=a[i];

}

printf("sum is %d",sum);

}

1. Nhập vào 1 mảng tối đa 10 phần tử nguyên, giá trị của các phần tử nhập trong khoảng [-10,10]. Chương trình tính tổng các số nguyên dương đã nhập vào mảng.

#include <stdio.h>

#include <conio.h>

#include <string.h>

void main()

{

int n;

do

{

printf("enter n: ");

scanf("%d",&n);

}while(n<=0||n>10);

int a[n];

int sum=0;

int i;

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

{

do

{

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

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

}while(a[i]<-10 || a[i]>10);

if(a[i]>0)

{

sum+=a[i];

}

}

printf("sum is %d",sum);

}

1. Nhập vào 1 mảng tối đa 10 phần tử nguyên, giá trị của các phần tử nhập trong khoảng [-10,10]. Chương trình tính tổng các số nguyên chẵn dương đã nhập vào mảng.

    int n,arr[10];

    do

    {

        printf("Enter number of array[maximun 10]: ");

        scanf("%d",&n);

    }while(n <= 0 || n > 10);

    int i;

    int sum = 0;

    for ( i = 0; i < n; i++)

    {

        do

        {

          printf("arr[%d] = ",i);

          scanf("%d",&arr[i]);

        } while (arr[i] < -10 || arr[i] > 10);

        if (arr[i]%2 == 0)

        {

            sum += arr[i];

        }

    }

    printf("The result is %d",sum);

1. Nhập vào 1 mảng tối đa 10 phần tử nguyên, giá trị của các phần tử nhập trong khoảng [-10,10]. Chương trình tính tổng các số nguyên lẻ dương đã nhập vào mảng.

    int n,arr[10];

    do

    {

        printf("Enter number of array[maximun 10]: ");

        scanf("%d",&n);

    }while(n <= 0 || n > 10);

    int i;

    int sum = 0;

    for ( i = 0; i < n; i++)

    {

        do

        {

          printf("arr[%d] = ",i);

          scanf("%d",&arr[i]);

        } while (arr[i] < -10 || arr[i] > 10);

        if (arr[i]%2 != 0)

        {

            sum += arr[i];

        }

    }

    printf("The result is %d",sum);

1. Viết lại câu 10, 11, 12 cho mảng số thực (thập phân).

float arrtong(int n, float arr[])

{

    int i;

    float sum = 0;

    for ( i = 0; i < n; i++)

    {

        sum += arr[i];

    }

    return sum;

}

float arrtongchan(int n, float arr[])

{

    int i;

    float sum = 0;

    for ( i = 0; i < n; i++)

    {

        if (arr[i] >= 0 && arr[i] - (int)arr[i] == 0)

        {

            if ((int)arr[i]%2 == 0)

            {

                sum += arr[i];

            }

        }

    }

    return sum;

}

float arrtongle(int n, float arr[])

{

    int i;

    float sum = 0;

    for ( i = 0; i < n; i++)

    {

        if (arr[i] >= 0 && arr[i] - (int)arr[i] == 0)

        {

            if ((int)arr[i]%2 == 1)

            {

                sum += arr[i];

            }

        }

    }

    return sum;

}

int main()

{

    int n,i=0;

    float arr[10];

    do

    {

        printf("Enter number of array[maximum 10]: ");

        scanf("%d",&n);

    } while (n <= 0 || n >= 10);

    for(i = 0; i < n; i++)

        {

            printf("arr[%d] = ",i);

            scanf("%f",&arr[i]);

            while(arr[i] < -10 || arr[i] > 10)

            {

                printf("Pls! Enter again.^^\n");

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

                scanf("%f",&arr[i]);

            }

        }

    printf("The array sum %.1f\n",arrtong(n,arr));

    printf("The array even sum %.1f\n",arrtongchan(n,arr));

    printf("The array odd sum %.1f",arrtongle(n,arr));

}

1. Viết lại câu 10. Tuy nhiên, mảng có n phần tử và các phần tử nhập giá trị nguyên (>0) tùy ý. Chương trình tính tổng số nguyên liên tiếp, tổng số nguyên lẻ, tổng số nguyên chẵn. Chương trình có sử dụng hàm đệ qui.
2. Tính giai thừa của số n sử dụng hàm đệ qui.

{

    int n;

    printf("Enter value: ");

    scanf("%d",&n);

    printf("The result: %d",factorial(n));

    getch();

}

int factorial(int n)

{

    if (n == 0 || n == 1)

        return 1;

    else

        return n \* factorial(n - 1);

}

1. Tính fibonaci của số n sử dụng hàm đệ qui.

{

    int n,i;

    printf("Enter vallue: ");

    scanf("%d",&n);

    printf("%d", fibonaci(n));

}

int fibonaci(int n)

{

    if (n == 1 || n == 2)

        return 1;

    else

        return fibonaci(n - 1) + fibonaci(n - 2);

}

1. Hiển thị số lớn nhất, nhỏ nhất, trung bình của mảng số nguyên n phần tử.

{

    int n,i;

    int arr[50];

    printf("Enter number of array: ");

    scanf("%d",&n);

    for ( i = 0; i < n; i++)

    {

        printf("arr[%d] = ",i);

        scanf("%d",&arr[i]);

    }

    printf("Max array: %d\n", maxarr(n,arr));

    printf("Min array: %d\n", minarr(arr,n));

    printf("Avg array: %.1f\n", averagearr(arr,n));

}

int maxarr(int n, int arr[])

{

    int i;

    int max = arr[0];

    for ( i = 0; i < n; i++)

    {

        if (max < arr[i])

        {

            max = arr[i];

        }

    }

    return max;

}

int minarr(int arr[], int n)

{

    int i;

    int min = arr[0];

    for ( i = 0; i < n; i++)

    {

        if (min > arr[i])

        {

            min = arr[i];

        }

    }

    return min;

}

float averagearr(int arr[], int n)

{

    int i;

    int s = 0;

    float avg = 0;

    for (i = 0; i < n; i++)

    {

        s += arr[i];

    }

    return (float)s/n;

}

1. Viết chương trình nhập vào mảng 2 chiều có a[3][4]. Giá trị của các phần tử này là số nguyên [0,10]. Viết chương trình tính tổng giá trị của mảng.

int main()

{

    int m,n,i,j;

    int arr[3][4];

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 4; j++)

        {

            do

            {

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

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

            } while (arr[i][j] < 0 || arr[i][j] > 10);

        }

    }

    for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 4; j++)

        {

            printf("%d\t",arr[i][j]);

        }

        printf("\n\n");

    }

    int s = 0;

     for ( i = 0; i < 3; i++)

    {

        for ( j = 0; j < 4; j++)

        {

            s += arr[i][j];

        }

    }

    printf("Sum array is %d",s);

}

1. Viết chương trình nhập vào mảng 2 chiều có a[m][n]. Với m,n là số nguyên được nhập từ bàn phím. Giá trị của các phần tử này là số nguyên [0,10]. Viết chương trình tính tổng giá trị của mảng.

{

    int m,n,i,j;

    int arr[50][50];

    printf("Enter number of colums: ");

    scanf("%d",&m);

    printf("Enter number of rows: ");

    scanf("%d",&n);

    for ( i = 0; i < m; i++)

    {

        for ( j = 0; j < n; j++)

        {

            do

            {

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

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

            } while (arr[i][j] < 0 || arr[i][j] > 10);

        }

    }

    for ( i = 0; i < m; i++)

    {

        for ( j = 0; j < n; j++)

        {

            printf("%d\t",arr[i][j]);

        }

        printf("\n\n");

    }

    int s = 0;

    printf("Even array is: ");

    for ( i = 0; i < m; i++)

    {

        for ( j = 0; j < n; j++)

        {

            if (arr[i][j]%2 == 1)

            {

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

                s += arr[i][j];

            }

        }

    }

    printf("\nThe even sum array: %d",s);

}