\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

int main()

{

foo();

foo();

foo();

return 0;

}

foo()

{

int y=0;

static int z;

y++;

z++;

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

}

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Welcome to GDB Online.

GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby, Perl,

C#, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS, SQLite, Prolog.

Code, Compile, Run and Debug online from anywhere in world.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

int \* calc(int \*p, int n);

int main()

{

int a[100],n,i;

int \*p;

p=a;

int \*result;

printf("enter number");

scanf("%d",&n);

printf("enter elements");

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

{

scanf("%d",(p+i));

}

result = calc(a,n);

printf("%d",\*(result+0));

printf("%d",\*(result+1));

printf("%d",\*(result+2));

return 0;

}

int \*calc(int \*p, int n)

{

int max,min,sum=0,i;

static int result[3];

max=\*(p+0);

min=\*(p+0);

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

{

if(\*(p+i)>max)

max=\*(p+i);

if(\*(p+i)<min)

min=\*(p+i);

sum=sum+(\*(p+i));

}

\*(result+0)=sum;

\*(result+1)=min;

\*(result+2)=max;

return(result);

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

int \* calc(int \*p, int n);

int main()

{

int a[100],n,i;

int \*p;

p=a;

int \*result;

printf("enter number");

scanf("%d",&n);

printf("enter elements");

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

{

scanf("%d",(p+i));

}

result = calc(a,n);

printf("%d",\*(result+0));

printf("%d",\*(result+1));

printf("%d",\*(result+2));

free(result);

return 0;

}

int \*calc(int \*p, int n)

{

int max,min,sum=0,i;

int \*result;

result=(int \*)malloc(3\*sizeof(int));

max=\*(p+0);

min=\*(p+0);

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

{

if(\*(p+i)>max)

max=\*(p+i);

if(\*(p+i)<min)

min=\*(p+i);

sum=sum+(\*(p+i));

}

\*(result+0)=sum;

\*(result+1)=min;

\*(result+2)=max;

return(result);

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

int factorial(int number)

{

if (number<=1)

{

return 1;

}

else

{

return number\*factorial(number-1);

}

}

int main()

{

int result;

result=factorial(3);

printf("factorial is %d",result);

return 0;

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

#include <string.h>

int main()

{

char str[10],i,n;

gets(str);

n=strlen(str);

int flag=0;

for(i=0;i<(n/2);i++)

{

if(str[i]!=str[(n-1)-i])

{

flag=1;

break;

}

}

if(flag==0)

{

printf("pallindrome");

}

else

{

printf("not pallindrome");

}

return 0;

}

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <stdio.h>

#include <string.h>

int main()

{

char str[10],a[10],i,j,n,m;

gets(str);

n=strlen(str);

int flag=0;

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

{

if(isalpha(str[i]))

{

if(islower(str[i]))

{

a[j]=str[i];

}

else

{

a[j]= tolower(str[i]);

}

}

}

m=strlen(a);

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

for(j=0;j<(m/2);j++)

{

if(str[j]!=str[(m-1)-j])

{

flag=1;

break;

}

}

if(flag==0)

{

printf("pallindrome");

}

else

{

printf("not pallindrome");

}

return 0;

}