Û

 \Diamond

Coding C

Tab

&

Auto saved at 20:03:44

RUN MENU

```
#include <stdio.h>
int main() {
   int num, reversed = 0, remainder, original;
    printf("Enter an integer: ");
   scanf("%d", &num);
   original = num;
   while (num != 0) {
        remainder = num % 10;
        reversed = reversed * 10 + remainder;
       num /= 10;
    if (original == reversed) {
       printf("%d is a palindrome.\n", original);
    } else {
       printf("%d is not a palindrome.\n", original);
    return 0;
```

Tab

{}

&

 \leftarrow

 \Box

①

Auto saved at 11:20:28

RUN MENU

Auto saved at 11:53:48

RUN MENU

```
2 main()
4 int a,b,c,d;
5 printf("enter the values of a,b,c,d");
6 scanf("%d %d%d%d",&a,&b,&c,&d);
7 if(d>c)
8 {
    if(d>b,d>a)
    printf("d is biggest no");
     if(a>b)
     printf("a is biggest no");
     printf("b is biggest no");
   if(c>b,c>a)
   printf("c is biggest no");
     if(a>b)
     printf("a is biggest");
     printf("b is biggest no");
```

 Tab
 {}
 ""
 ;
 5
 ☆
 ☆

 =
 \
 &
 ,
 ☆
 ↓
 ⇒

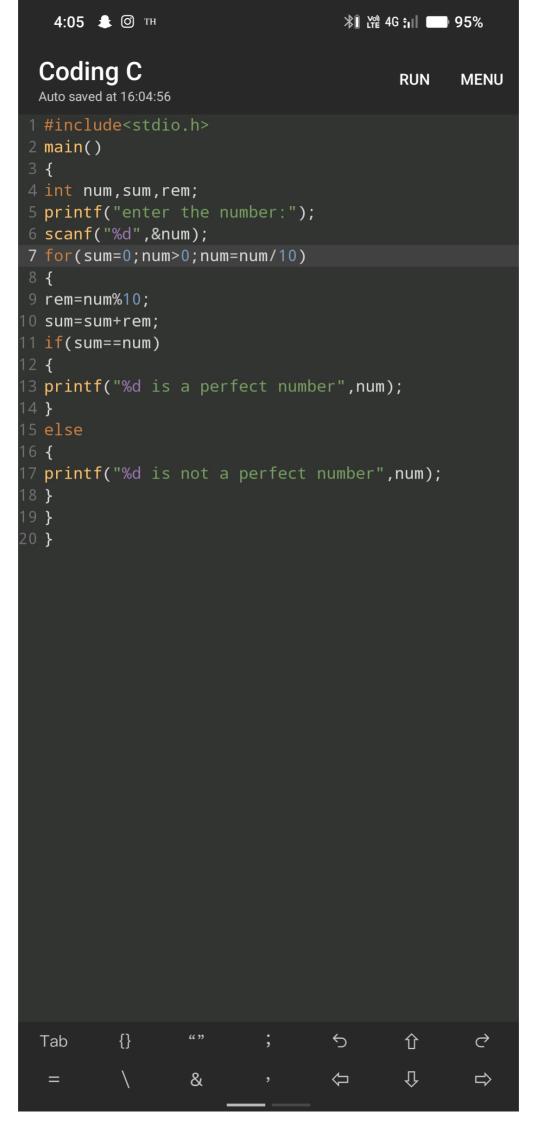
Auto saved at 12:59:35

RUN MENU

```
1 #include<stdio.h>
2 | void main()
3 {
4 int i,num,count=0;
5 printf("enter the number value:");
6 scanf("%d",num);
7 i=1;
8 while(i<=num)
9 {
0 | if(num%i==0)
1 count++;
2 i++;
3 | }
4 if(count==2)
5 {
6 printf("%d is prime number",num);
7 }
8 else
9 {
0 printf("%d is not a prime number");
1 }
2 }</pre>
```

 Tab
 {}
 ""
 ;
 5
 û
 ♂

 =
 \
 &
 ,
 ⇔
 ↓
 ⇒



Auto saved at 17:07:28

RUN MENU

```
1 #include<stdio.h>
2 main()
3 {
4 int X,Y,a,b;
5 printf("enter the values of X,Y,a,b");
6 scanf("%d %d %d %d ",&X,&Y,&a,&b);
7 printf("numbers that are divisible by%d and%d from%d and %d are:",a,b,X,Y);
8 for(int i=X;i<=Y;i++)
9 {
0 if(i%a==0 && i%b==0)
11 printf("%d",i);
12 }
13 ]</pre>
```

 Tab
 {}
 ""
 ;
 5
 û
 ♂

 =
 \
 &
 ,
 ↓
 ↓
 ↓

Auto saved at 17:42:25

RUN MENU

```
1 #include<stdio.h>
2 #include<math.h>
3 main()
4 {
5 int a,b,c;
6 float x1,x2;
7 printf("enter the equation a*x*x+b*x+c=0");
8 scanf("%d %d %d",&a,&b,&c);
9 | x1==(-b+√(b*b-4*a*c))/(2*a);
10 | x2==(-b-√(b*b-4*a*c))/(2*a);
11 printf("the roots of the %dx*x+%dx+%d=0 are %f and%f :",a,b,c,x1,x2);
12 }
```

 Tab
 {}
 ""
 ;
 5
 ☆
 ċ

 =
 \
 &
 ,
 ☆
 ↓
 ⇒