

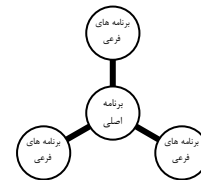
روند اجرای مسئله :

۱. حل مسئله
۲. طراحی الگوریتم
۳. رسم فلوچارت
۴. ترجمه برنامه به یک زبان برنامه نویسی
۵. دگ برنامه
۶. نوشتن برنامه با مترجم یا مفسر و تهیه فایل سورس (*.cpp یا *.c)
۷. جمع آوری و ساخت فایل آبجکت (*.obj)
۸. link کردن و ساخت فایل اجرایی (*.exe , *.com , *.bat)

سورس : فایلی که برنامه اصلی را در آن ذخیره میکنیم .

دگ : برنامه ای که روی کاغذ نوشته میشود .

الگوریتم : تقسیم برنامه به چند برنامه کوچکتر .



زبان محاوره : مترادف و متشابه - اصطلاحات - تجزیه و ترکیب و ...

زبان برنامه نویسی : یک زبان اختراعی است (متعلق به هیچ قومی نیست) و مشکلات زبان محاوره ای را ندارد .

سطوح زبان برنامه نویسی :

۱. سطح پایین : زبان ماشین - دستیابی به bit و byte و آدرس دهی (اسمبلی و ...)
۲. سطح میانی : نزدیک به زبان ماشین همراه با زبان سطح بالا (C و ...)
۳. سطح بالا : (cobol , pascal , basic و ...)

اولین زبان علمی fortran

اولین زبان تجاری RPG

زبان C بعد از زبان B اختراع شد

زبان C در سال ۱۹۷۰ ساخته شد

خصوصیات زبان C :

۱. زبانی است سطح میانی (هم خصوصیات زبان سطح بالا و هم خصوصیات زبان سطح پایین را دارد) ، به همین دلیل به مفاهیم bit و byte و مفاهیمی که زبان سطح پایین داراست دسترسی دارد .
۲. قابل حمل است .
۳. ساختیافته است : الف) به ستون حساس است - ب) سعی می کند از دستوراتی که سبب گنگی می شود (مانند goto) استفاده نکند - ج) سعی میکند حلقه های خود را مستقل و ساده مطرح کند - د) استفاده از زیر برنامه ها و بلاگها - ه) خوانایی برنامه بالا رفته .
۴. ساختیافته بلاکی نیست (یعنی امکان تعیین یک زیر برنامه در یک برنامه دیگر وجود ندارد) .
۵. انواع داده ها در آن قابل تعریف است .
۶. برای برنامه نویسی سیستم مناسب است (یعنی : سیستم عامل ، ویرایشگرها و ... را میتوان با آن نوشت) .
۷. میتواند از سخت افزار و نرم افزارها استفاده کند .
۸. سرعتش نسبت به زبانهای دیگر بالاتر است و سبب صرفه جویی در وقت میشود .
۹. بسیار قابل انعطاف میباشد (هر ایده ای را میتوان در برنامه پیاده کرد) .
۱۰. تعداد کلمات کلیدی در آن بسیار کم است .
۱۱. نسبت به بزرگی و کوچکی حروف حساس است .
۱۲. شیء گرا است .

دستورات پیش پردازنده :

تابع اصلی :

```
main ()
{
    شروع
    دستورات اجرایی و تعاریف
}
    پایان
```

تفاوت مترجم و مفسر :

۱. مفسر تا یک حجم خاص را پشتیبانی می کند ولی مترجم همه حجمها را قبول میکند .
۲. مترجم تمام برنامه را گرفته و تمامی خطها را ترجمه میکند تا آخر ، در صورت ایجاد مشکل ترجمه نکرده و اجرا نمیکند اما مفسر خط به خط ترجمه و اجرا میکند .
۳. مترجم در صورت اجرای زیاد نیازی به ترجمه و لینک دوباره ندارد ولی مفسر در ازای هر بار اجرا هم ترجمه میکند و هم لینک انجام میدهد .

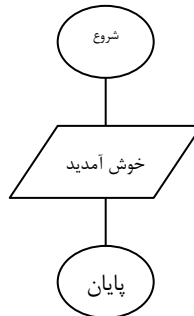
نمونه برنامه :

```
#include<iostream.h>
int main()
{
    -----
    return 0;
}
```

در صورت ننوشتن نوع تابع قبل از main() به صورت پیشفرض void انتخاب میشود .
main() ، نام برنامه ، دستورات اجرایی و دستورات پیش پردازنده جزء فلوجارت نیستند .

۱ _ برنامه ای بنویسید که یک نوشته را نمایش دهد :

```
#include<iostream.h>
main()
{
    cout<<"welcome";
}
یا
#include<stdio.h>
main()
{
    printf("welcome");
}
```



دستور چاپ :

```
#include<iostream.h>
    cout<<" " ;
#include<stdio.h>
    printf(" " );
```

_ ۲

```
#include<iostream.h>
#include<stdio.h>
main()
{
    clrscr();
    gotoxy(30,20);
    cout<<"welcome";
    // printf("welcome");
}
```

_ *

```
#include<iostream.h>
#include<stdio.h>
main()
```

```

{
clrscr();
gotoxy(35,20);
printf("welcome");
gotoxy(35,21);
printf("1388-04-27");
}

```

\t مانند tab عمل کرده و ۸ خانه به جلو حرکت میکند

\a تک بوق

\r کار دکمه اینتر را انجام میدهد

متغیر : اسمی است که به یک محل از حافظه گفته میشود که با صدا کردن آن اسم میتوان به محتوای آن آدرس دسترسی پیدا کرد ، با یک حرف انگلیسی شروع میشود ، کوچک و بزرگی حروف اهمیتی ندارد ولی تا آخر باید به همان صورت نوشته شود . به آن قسمت از حافظه که اسم نسبت میدهیم ، ساختار هم باید بدهیم . برای هر متغیر یک نوع نیاز هست .

انواع متغیر :

int صحیح

float اعشاری

char کاراکتری

_**

```

#include<stdio.h>
main()
{
Int a=15,b=235,c=1245;
clrscr();
printf("%20d\n",a);
printf("%20d\n",b);
printf("%20d\n",c);
}

```

۳ _ برنامه ای که در وسط صفحه متن زیر را چاپ کند :

بسمه تعالی

نام : ...

۱

۲۵

۳۴۲

۱

۲۵

۳۴۲

```

#include<conio.h>
#include<stdio.h>
main()
{
int a=1,b=25,c=342;
clrscr();
gotoxy(33,5);
printf("besme taala\n");
gotoxy(33,7);
printf("name\t: emad attar\a");
gotoxy(33,27);
printf("%-10d",a);
gotoxy(33,29);
printf("%-10d",b);
gotoxy(33,31);
printf("%-10d",c);
gotoxy(40,36);
}

```

```
printf("%10d",a);
gotoxy(40,38);
printf("%10d",b);
gotoxy(40,40);
printf("%10d",c);
}
```

۴_ برنامه ای که متغیری را گرفته و به صورت فرم در وسط صفحه چاپ کند :

```
#include<conio.h>
#include<stdio.h>
main()
{
float a=1.5,b=2.5,c=13.75;
clrscr();
gotoxy(33,27);
printf("a=%-10.1f",a);
gotoxy(33,29);
printf("b=%-10.1f",b);
gotoxy(33,31);
printf("c=%-10.2f",c);
}
```

۵_ برنامه ای که یک عدد را خوانده ، ده برابر کند و در خروجی چاپ کند :

```
#include<conio.h>
#include<stdio.h>
main()
{
float a;
clrscr();
gotoxy(33,22);
scanf("%f",&a);
gotoxy(33,29);
a=a*10;
printf("%10.2f",a);
}
```

Printf(" "); چاپ
Scanf("%d",&a); خواندن

۶_ برنامه ای که سه عدد را خوانده ، معدل را بدست آورده و چاپ کند .

```
#include<iostream.h>
#include<conio.h>
main()
{
int sum=0,a,b,c;
float avg;
cin>>a>>b>>c;
sum=a+b+c;
avg=(float)sum/3;
cout<<"avg:"<<avg;
}
```

۷_ برنامه ای که داده هایی را گرفته و به صورت لیستی مانند لیست زیر چاپ کند .

| |
|---------------|
| لیست کارمندان |
| حسین ۲۲۰۰۳۴۵ |
| حسن ۴۷۵۹۶۱۱ |

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
//list hoghugh
clrscr();
char na1[15],na2[15];
float n1,n2;
cout<<"plase enter data :"<<endl;
cout<<" "<<endl;
cout<<"name :"<<" ";
cin>>na1;
cout<<"hoghugh:"<<" ";
cin>>n1;
cout<<"name :"<<" ";
cin>>na2;
cout<<"hoghugh:"<<" ";
cin>>n2;
gotoxy(35,12);
cout<<"list hoghugh :"<<endl;
cout<<" "<<endl;
gotoxy(37,15);
cout<<na1<<"\t"<<n1;
gotoxy(37,17);
cout<<na2<<"\t"<<n2;
return 0;
}

```

| | | | |
|-------|-----------------|---------------|--------|
| Int | - ۳۲۷۶۷ | ۳۲۷۶۷ | ۱۶ بیت |
| Char | - ۱۲۷ | ۱۲۷ | ۸ بیت |
| float | - ۱۰ به توان ۳۸ | ۱۰ به توان ۳۸ | ۳۲ بیت |

Short = کوتاه

Long = طولانی

Unsigned = بدون علامت

Double = دوبرابر

Unsigned int a ; هر عدد منفی را با مثبت جمع میکند

Long int b ; طول عدد ۵ رقمی به ۱۰ تبدیل میشود

Int a=b=c=5 ;

یا

Int a=5,b=5,c=5;

Float a=15 ;

یا

Float a ;

a=15 ;

یا

Float a ;

Cin>>a ;

ثابت :

```
#define a 15
```

```
Const int a=15 ;
```

دلایل استفاده :

۱. اگر چیزی که میخواهیم مقدار بدهیم دارای مقدار طولانی باشد .

۲. اگر بخواهیم یک مقدار ثابت را مقدار دهی کنیم .

Project => Make all => *.exe

۸_ برنامه ای که یک کاراکتر را خوانده ، سپس کد کاراکتر را در خروجی نمایش دهد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
char a;
int b;
cout<<"enter a char : ";
cin>>a;
b=a;
cout<<"char code : "<<b;
}
```

۹_ برنامه ای که یک عدد را خوانده ، سپس کاراکتر مربوط به آن را نمایش دهد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
char a;
int b;
cout<<"enter a number : ";
cin>>b;
a=b;
cout<<"char name : "<<a;
}
```

۱۰_ برنامه ای که عدد دورقمی را خوانده ، معکوس را چاپ کند .

{ محاسبه باقیمانده : $b=a/10$;
 { محاسبه خارج قسمت : $c=a\%10$;

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a,b,c;
cout<<"plase enter data : "<<endl;
cout<<" "<<endl;
cin>>a;
b=a/10;
c=a%10;
cout<<c<<b;
}
```

یک واحد به a اضافه میکند .

```
a++;
++a;
```

یک واحد از a کم میکند .

```
a--;
--a;
{
Int a,b,c;
Cin>>a>>b>>c;
C=++a + b++;
```

به a یکواحد اضافه میکند با b جمع میکند و دیگر کاری با b ندارد و در C میریزد .

```
C=++a + ++b
```

به a یکواحد اضافه میکند بعد به b یکواحد اضافه میکند ، هر دو را با هم جمع میکند و در C میریزد .

C=a++ + ++b

به b یکواحد اضافه میکند با a جمع میکند و دیگر کاری با a ندارد و در C میریزد .

```
Cout<<c<<a<<b;
}
```

_ ۱۱

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
int a,b,c;
cout<<"plase enter data :"<<endl;
cout<<" "<<endl;
cin>>a>>b>>c;
//c=++a + b++;
//c=a++ + b++;
c=++a + ++b;
cout<<c<<" "<<a<<" "<<b;
return 0;
}
```

_ ۱۲

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a=12,b=7,c;
c=a&b;
cout<<c;
}
```

تقدم عملگره :

۱. داخلی ترین پرانتز
۲. ++ --
۳. - (تعووض علامت)
۴. ضرب ، تقسیم ، جمع
۵. جمع و تفریق

C=++a - b*c / -d + 4*a ;

از چپ به راست

شیفت به چپ << ۰۰۰۱۱۰۰۰

شیفت به راست >> ۰۰۰۰۰۱۱۰

۱۳ _ برنامه ای که داده ها را گرفته و به صورت زیر در خروجی چاپ کند .

| |
|-------------------------------|
| نام : حسین |
| حقوق : ۲۵۰۰۰۰ |
| حقوق حسین برابر با ۲۵۰۰۰۰ است |

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
```

```

clrscr();
char name[15];
float num;
gotoxy(35,10);
cout<<"plase enter data :"<<endl;
cout<<" "<<endl;
gotoxy(35,12);
cout<<"name :"<<" ";
cin>>name;
gotoxy(35,13);
cout<<"hoghugh:"<<" ";
cin>>num;
gotoxy(35,15);
cout<<"hoghogh:"<<name<<" "<<num;
return 0;
}

```

۱۴ _ انجام عمل تقسیم .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int a=17,b;
b=a>>1;
cout<<b;
}

```

۱۵ _ انجام عمل ضرب .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
int a,b;
cin>>a;
b=a<<2;
cout<<b;
return 0;
}

```

```

a=17;
b=a>>1;

```

به تعداد شیفتهای ۲ تقسیم میکند (تعداد شیفتهای ۱ است پس یکبار به ۲ تقسیم میشود)

جواب => b=8

>> تقسیم میشود _ << ضرب میشود .

۱۷ در مبنای ۱۰ به مبنای ۲ باید به ۲ تقسیم شود تا باقیمانده صفر بشود . جواب میشود : ۱۰۰۰۱
چون باید در ۸ خانه گنجانده شود کمبود آن را صفر پر میکنیم . جواب میشود : ۰۰۰۱۰۰۰۱
چون تعداد شیفت ۱ است به راست ، پس یک خانه به راست شیفت میکند . جواب میشود : ۰۰۰۰۱۰۰۰

۱۶ _ (و) یا (&) :

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();

```



```
int a,b,c;
cin>>a>>b;
c=a&b;
cout<<c;
}
```

١٧_ (يا) يا (|) :

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int a,b,c;
cin>>a>>b;
c=a|b;
cout<<c;
}
```

١٨_ (نقيض) يا (~) :

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int a,b,c;
cin>>a>>b;
c=a~b;
cout<<c;
}
```

١٩_ (ياى مانعت الجمع) يا (^) :

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
int a,b,c;
cin>>a>>b;
c=a^b;
cout<<c;
return 0;
}
```

_ ٢٠

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
int a,b,c;
cin>>a>>b;
c=~a & b>>1;
cout<<c;
return 0;
}
```

بالاترین ها :

~ .۱
>> << .۲
& .۳
^ | .۴

تقدم عملگر ها :

-
+ - * / %
++ --
+= -= *= /= /=

a=a+b; = a+=b;
a=a-b; = a-=b;
a=a*b; = a*=b;

به b به اندازه a { -, +, *, } میشود و در a ذخیره میگردد .

علامت سؤال (؟) :

فرم کلی :

< عبارت ۳ > : < عبارت ۲ > ? < عبارت ۱ > = متغیر ;

عبارت ۱ : یک گناره که یا درست است یا نا درست

عبارت ۲ : در صورت درستی عبارت ۱ اجرا میشود

عبارت ۳ : در صورت نادرستی عبارت ۱ اجرا میشود .

۲۱ _

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
int a,b,c;
cin>>b;
a=(b<7)?b+2:b-3;
cout<<a;
return 0;
}
```

۲۲ _ برنامه ای که دو عدد را خوانده ، عدد کوچکتر را بنویسد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
int a,b,c;
cin>>a>>b;
c=(a<b)?a:b;
cout<<c;
return 0;
}
```

علامت (,) یا (کاما) :

فرم کلی :

(عبارت ۲ و عبارت ۱) = متغیر ;

(مثال) :

```
Cin>>c;  
a=(b=4*c-1,b+3);
```

_ ۲۳

```
#include<iostream.h>  
#include<conio.h>  
#include<stdio.h>  
main()  
{  
clrscr();  
int a,b,c;  
cin>>c;  
a=(b=4*c-1,b+3);  
cout<<a;  
}
```

| عبارت | عبارت رابطه ای | گزاره |
|-------|----------------|-------|
| | < | |
| | <= | |
| | > | |
| | >= | |
| | != | |
| | == | |

a+2==7

گزاره ۱ همیشه درست

گزاره ۰ همیشه نادرست

علامت (&&) :

(3>a) && (b<2)

هر دو باید درست باشند

اگر اولی غلط ، دومی درست : جواب غلط

اگر اولی درست ، دومی درست : جواب درست

اگر اولی غلط ، دومی غلط : جواب غلط

علامت (||) :

(7==2) || (3!=a)

تقدم عملگرها :

هرکدام اول بود از چپ به راست

! ~ ++ -- sizeof

-

* / %

+ -

<< >>

<<= >>=

= = !=

^ |

&&

||

?

= += -= *= /= %=

,

_ ۲۴ نشان دادن عملکرد دستور (sizeof) :

```
#include<iostream.h>  
#include<conio.h>
```

```
#include<stdio.h>
main()
{
clrscr();
cout<<sizeof(int);
}
```

cout<<sizeof(int);

_ مقدار عدد اشغال شده توسط متغیر داخل پرانتز در حافظه را بر میگرداند .

* _ دستوری بنویسید که $(c = a (b+2) / 3b-2a)$ را محاسبه کند .

$C = a * (b+2) / (3 * b - 2 * a) ;$

* _ دستور $c = 2 * a * (b+3) / (4+2*a) / b-c$

یا $c = 2*a*(b+3) / ((4+2*a)*b)-c$

چه معادله ای را در سیستم محاسبه میکند ؟

$C=2a(b+3)b / (4 + 2a) -c$

۲۵ _ برنامه ای که اعداد طبیعی ۱ تا ۵۰۰ را تولید کرده و چاپ کند .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
int i;
for(i=1;i<=500;i++)
{
cout.width(10);
cout<<i<<"\n ";
}
return 0;
}
```

نکته :

Cout.width(10);

از سمت راست مرتب میکند

حلقهٔ for :

(تغییرات متغیر ؛ شرط پایانی ؛ مقادیر اولیه) for : شکل کلی

۲۶ _ برنامه ای که شروع و پایان اعداد طبیعی را خوانده ، از شروع تا پایان را تولید کرده و بنویسد . (نکته : اگر عددی خارج از محدوده نوع متغیر وارد شود مقدار منفی برمیگرداند)

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int i,a,b;
cin>>a>>b;
for(i=a;i<=b;i++)
{
cout.width(10);
cout<<i<<"\n ";
}
}
```

۲۷ _ برنامه ای که اعداد فرد از a تا b را تولید کرده و نمایش دهد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
```

```

int i,a,b;
cin>>a>>b;
for(i=a;i<=b;i+=5)
{
cout.width(10);
cout<<i<<"\n ";
}
}

```

۲۸ _ برنامه ای که اعداد طبیعی از a تا b را جمع کرده و حاصلجمع را چاپ کند .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int i,a,b,sum=0;
cin>>a>>b;
for(i=a;i<=b;i++)
sum=sum+i;
cout<<endl<<sum;
}

```

۲۹ _ برنامه ای که اعداد از a تا b را به صورت زیر نوشته و حاصلجمع را برگرداند .
 $(1/a + 1/a+1 + 1/a+2 + \dots + 1/b) << (1/1 + 1/2 + 1/3 + \dots + 1/500)$

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
int main()
{
clrscr();
float i,a,b,sum=0;
cin>>a>>b;
for(i=a;i<=b;i++)
sum=sum+(1/i);
cout<<endl<<sum;
return 0;
}

```

۳۰ _ برنامه ای که اعداد از a تا b را به صورت زیر نوشته و حاصل را برگرداند .
 $(1/1 - 1/2 + 1/3 - \dots - 1/500)$

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
float i,j,a,b,sum=0;
cin>>a>>b;
for(i=a,j=1;i<=b;i++,j=j*-1)
sum=sum+(1/i)*j;
cout<<endl<<sum;
}

```

۳۱ _ برنامه ای که حاصلجمع زیر را تولید کرده و بنویسد . (n را بخواند)
 $(1/1! + 1/2! + 1/3! + \dots + 1/n!)$

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
float i,j,a,b,sum=0;
cin>>a>>b;

```

```

for(i=a,j=1;i<=b;i++,j=j*i)
    sum=sum+(1/j);
cout<<endl<<sum;
}

```

۳۲ _ برنامه ای که سری زیر را محاسبه و چاپ کند .
 $(1/1! + 1/2! - 1/3! + \dots 1/n!)$

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
float a,b,i,j,c,s=0;
cout<<"Number One :"<<"\t";
cin>>a;
cout<<"Number Two :"<<"\t";
cin>>b;
for(i=a,j=1,c=1;i<=b;i++,j=j*i,c=c*-1)
    s=s+(1/j)*c;
cout<<endl<<"End :"<<"\t"<<"\t"<<s;
}

```

۳۳ _ برنامه ای که تعداد جملات را خوانده و حاصل را بدست آورده و چاپ کند .

```

S= 1/1!+1 - 3/2!+4 + 5/3!+6 - 7/4!+8 + ...
for(i=1,a=1,b=1,c=2,k=1;i<=n;i++,a+=2,b*=i,c+=2,k*=-1)
s=s+a/(b+c)*k;

```

a : صورت
b : فاکتوریل
c : عدد جمع شده
k : علامت

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
float a,b,i,s,c,k,n;
cin>>n;
for(i=1,a=1,b=1,c=2,k=1;i<=n;i++,a+=2,b*=i,c+=2,k*=-1)
    s=s+a/(b+c)*k;
cout<<endl<<s;
}

```

۳۴ _ برنامه ای که تعداد اعداد و اعداد را خوانده ، معدل را پیدا کرده و چاپ کند .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
float n,s=0,i,a;
cout<<"tedad :"<<"\t";
cin>>n;
for(i=1;i<=n;i++)
{
cout<<"\t";
cin>>a;
s=s+a;
}
s/=n;
cout<<"\t"<<s;
}

```

۳۵ _ برنامه ای که تعداد کاراکتر را خوانده و با خواندن هر کاراکتر کد آن را بنویسد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
float n,s=0,i,a;
int b;
char j;
cout<<"tedad :"<<"\t";
cin>>n;
for(i=1;i<=n;i++)
{
cout<<"\t";
cout<<endl<<"enter a char "<<j<<" "<<".:"<<" ";
cin>>j;
b=j;
cout<<"char code  "<<j<<" "<<".:"<<" "<<b<<endl;
}
j/=n;
}
```

۳۶ _ برنامه ای که اعداد زیر را تولید کرده و چاپ کند . (۰/۱ و ۰/۲ و ۰/۳)

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
float n,i;
cin>>n;
for(i=0.1;i<=n;i+=0.1)
cout<<i<<" ";
}
```

۳۷ _ برنامه ای که کاراکتر های روی مانیتور را بخواند تا به علامت (.) برسد ، سپس تعداد این کاراکتر ها را بنویسد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i;
char a;
cin>>a;
for(i=1;a=cin.get() != '!';i++);
cout<<i<<" ";
}
```

۳۸ _ برنامه ای که هربار کاراکتری را خوانده و کد آن را نوشته تا به علامت (*) رسیده و خارج شود .

Cin>> : میخواند و به خط بعد می رود

Cin.get() : پشت سر هم میخواند

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int s,i;
char a;
for(i=1;a=cin.get() != '*';i++);
cout<<i;
}
```

۳۹ _ برنامه ای که جدول ضرب ۱۰*۱۰ را تولید کرده و نمایش دهد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int i,j;
for(i=1;i<=10;i++)
{
for(j=1;j<=10;j++)
{
//gotoxy(5*j,i*2);
//cout<<i*j<<" ";

printf("%5d",i*j);

//cout<<"\t"<<i*j;
}
cout<<endl;
}
}
```

۴۰ _ برنامه ای عدد را خوانده ، فکتوریل را محاسبه کند .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,n,s;
cin>>n;
s=1;
for(i=1;i<=n;i++)
s=s*i;
cout<<s;
}
```

۴۱ _ برنامه ای که دو عدد را خوانده ، عدد اول را به توان دومی برساند و بنویسد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,a,b,s;
cin>>a>>b;
s=1;
for(i=1;i<=b;i++)
s=s*a;
cout<<s;
}
```

۴۲ _ برنامه ای که تعداد n سطر و m ستون را خوانده و به شکل زیر چاپ کند .

```
****
****
****
```

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,m,n;
cin>>n>>m;
for(i=1;i<=n;i++)
{
for(j=1;j<=m;j++)
```



```

cout<<"*";
cout<<endl;
}
}

```

۴۳_ برنامه ای که n تعداد سطر را بخواند و به صورت زیر چاپ کند .

```

*
**
***
****
*****

```

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,n;
cin>>n;
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
cout<<"*";
cout<<endl;
}
}

```

۴۴_ برنامه ای که n تعداد سطر را بخواند و به صورت زیر چاپ کند .

```

*****
****
***
**
*

```

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,n;
cin>>n;
for(i=n;i>=1;i--)
{
for(j=1;j<=i;j++)
cout<<"*";
cout<<endl;
}
}

```

۴۵_ برنامه ای که n تعداد سطر را بخواند و به صورت زیر چاپ کند .

```

1
12
123

```

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,n;
cin>>n;
cout<<endl;
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
cout<<j;
cout<<endl;
}
}

```

۴۶ _ برنامه ای که n تعداد سطر را بخواند و به صورت زیر چاپ کند .

۱
۲۲
۳۳۳

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
int i,j,n;
cin>>n;
cout<<endl;
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
cout<<i;
cout<<endl;
}
}
```

۴۷ _ برنامه ای که n تعداد سطر را بخواند و به صورت زیر چاپ کند .

۱۱۱
۲۲۲

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,m,n;
cin>>n>>m;
cout<<endl;
for(i=1;i<=n;i++)
{
for(j=1;j<=m;j++)
cout<<i;
cout<<endl;
}
}
```

۴۸ _ برنامه ای که n تعداد سطر را بخواند و به صورت زیر چاپ کند .

۱۲۳
۱۲۳

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,m,n;
cin>>n>>m;
cout<<endl;
for(i=1;i<=n;i++)
{
for(j=1;j<=m;j++)
cout<<j;
cout<<endl;
}
}
```

۴۹_ n را بخواند و به شکل زیر چاپ کند .

```
*
**
***
****
*****
```

```
#include<iostream.h>
main()
{
int i,j,k,n;
cin>>n;
for(i=1;i<=n;i++)
{
for(j=1;j<=n-i;j++)
cout<<" ";
for(k=1;k<=2*i-1;k++)
cout<<"*";
cout<<endl;
}
}
```

شکل کلی حلقه while :

While (شرط)

```
{
-----
}
```

حلقه بی نهایت :

For (; ;)

```
{
-----
}
```

While (1)

```
{
-----
}
```

۵۰_ با حلقه while برنامه ای بنویسید که هر بار کاراکتری را خوانده و چاپ کند تا زمانی که به (*) رسید متوقف شود .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
char a;
while(a!='*')
{
cout<<endl;
cin>>a;
cout<<a;
}
}
```

۵۱_ با حلقه while برنامه ای بنویسید که تعداد کاراکترهای مانیتور را خوانده و بنویسد ، زمانی که به (*) رسید بایستد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a;
a=0;
while(cin.get()!='.')
{
a++;
}
}
```

```
cout<<a;
}
```

شکل کلی حلقه `while ~ do` :

```
Do
{
----
}
While ( شرط ) ;
```

فرق حلقه `while` با `do ~ while` :

در حلقه `while` برنامه در صورت درستی شرط اجرا میشود اما در حلقه `do ~ while` برنامه یکبار حداقل اجرا میشود حتی اگر شرط نادرست باشد .
۵۲ _ برنامه ای که عددی را در مبنای ۱۰ گرفته و در مبنای ۴ بنویسد .

۱۰ در مبنای ۱۰ = ۰۱۰ در مبنای ۲

۳۷ در مبنای ۱۰ = ۲۱۱ در مبنای ۴

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a,j=1,r,s=0;
cin>>a;
while(a>0)
{
r=a%4;
s=s+r*j;
a=a/4;
j=j*10;
}
cout<<s;
}
```

۵۳ _ برنامه ای که عدد را در مبنای ۱۰ گرفته و به مبنای ۷ ببرد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a,j=1,r,s=0;
cin>>a;
while(a>0)
{
r=a%7;
s=s+r*j;
a=a/7;
j=j*10;
}
cout<<s;
}
```

۵۴ _ برنامه ای که عددی را در مبنای ۵ گرفته و به مبنای ۱۰ ببرد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a,j=1,r,s=0;
cin>>a;
while(a>0)
{
r=a%10;
s=s+r*j;
```

```

a=a/10;
j=j*5;
}
cout<<s;
}

```

۵۵ _ برنامه ای که عددی را خوانده و به عکس چاپ کند .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int n,a,s=0;
cin>>n;
while(n>0)
{
a=n%10;
s=s*10+a;
n=n/10;
}
cout<<s;
}

```

۵۶ _ برنامه ای که ۱۰ عدد را بخواند ، بزرگترین آن را پیدا کرده و چاپ نماید .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
float a,i,b;
cin>>a;
for(i=2;i<=10;i++){
cin>>b;
while(a<b)
a=b;
}
cout<<a;
}

```

۵۷ _ برنامه ای که تعداد سطر را خوانده ، به شکل زیر چاپ نماید .

```

*
**
***
**
*
*****

```

```

#include<iostream.h>
main()
{
int i,j,k,n,m;
cin>>n;
// n=tedad satr
for(i=1;i<=n-1;i++)
{
for(j=1;j<=n-i;j++)
cout<<" ";
if(2*i-1 != 1) cout<<"*";
for(k=1;k<=2*i-3;k++)
cout<<" ";
cout<<"*"<<endl;
}
for(m=1;m<=n;m++)
cout<<"*"<<" ";
}

```

If (شرط)

جزای درستی ;

If (شرط)

جزای درستی ;

else

جزای نادرستی ;

If (a==b)

Cout<<"barabarand";

If (a<=b)

C=2*a+b;

else

K=2*b-1;

۵۸ _ با استفاده از دستور If برنامه ای بنویسید که ۱۰ عدد را گرفته ، بزرگترین و کوچکترین آنها را نمایش دهد .

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
#include<stdio.h>
```

```
main()
```

```
{
```

```
float l,a,s,i;
```

```
cin>>l>>s;
```

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

```
{
```

```
cin>>a;
```

```
if(a>l) l=a;
```

```
else if(a<s) s=a;
```

```
}
```

```
cout<<endl<<"Min :"<<" "<<s<<"\t"<<"Max :"<<" "<<l;
```

```
} //l=max , s=min
```

۵۹ _ برنامه ای که از ۵ تا ۵۰ را روی مانیتور نوشته بطوریکه روی هر سطر ۵ عدد نوشته شود .

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
#include<stdio.h>
```

```
main()
```

```
{
```

```
int i;
```

```
for(i=5;i<=500;i++)
```

```
{
```

```
if(i%5==0)
```

```
cout<<endl;
```

```
cout<<i<<" ";
```

```
}
```

```
}
```

۶۰ _ برنامه ای که تعداد علامت هایی که روی یک سطر وارد میشود را خوانده و چاپ کند .

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
#include<stdio.h>
```

```
main()
```

```
{
```

```
int i=0;
```

```
char a;
```

```
while(a=getche() != '\r')
```

```
{
```

```
a=getche();
```

```

if(a<=57 && a>=48)i++;
}
cout<<endl;
cout<<i<<" ";
}

```

۶۱_ برنامه ای که عددی را خوانده و طبق جدول زیر معادل حرفی آن را چاپ کند .

| | | | |
|----|----|----|-------|
| ۱ | تا | ۱۰ | عالی |
| ۱۱ | تا | ۲۰ | خوب |
| ۲۱ | تا | ۳۰ | متوسط |
| ۳۱ | تا | ۴۰ | ضعیف |
| ۴۱ | تا | ۵۰ | بد |

```

#include<iostream.h>
main()
{
int a;
cin>>a;
if(a>=1&&a<=10)cout<<"a'li";
else
if(a>=11&&a<=20)cout<<"khoub";
else
if(a>=21&&a<=30)cout<<"motevaset";
else
if(a>=31&&a<=40)cout<<"zaeif";
else
if(a>=41&&a<=50)cout<<"bad";
}

```

۶۲_ برنامه ای که حقوق کارمندان را گرفته و مالیات را برحسب لیست زیر محاسبه کند .

| | | | |
|-------|-----|-----|-----|
| ۰ | تا | ۱۰۰ | %۰ |
| ۱۰۰ | تا | ۲۰۰ | %۱۰ |
| ۲۰۰ | تا | ۳۰۰ | %۱۵ |
| ۳۰۰ | تا | ۴۰۰ | %۲۰ |
| ۴۰۰ | تا | ۵۰۰ | %۳۰ |
| بالای | ۵۰۰ | | %۴۰ |

```

#include<iostream.h>
main()
{
int p,t;
cin>>p;
if(p<100)t=0;
else
if(p<200)t=0+(p-100)*0.10;
else
if(p<300)t=0+10+(p-200)*0.15;
else
if(p<400)t=0+10+15+(p-300)*0.20;
else
if(p<500)t=0+10+15+20+(p-400)*0.30;
else
t=0+10+15+20+30+(p-500)*0.40;
cout<<t;
}

```

۶۳_ برنامه ای که نمرات دانشجو را گرفته و بر اساس جدول زیر معادل حرفی آن را چاپ کند .

۰ تا ۱۲ d

c ۱۵ تا ۱۲

b ۱۷ تا ۱۵

a ۲۰ تا ۱۷

```
#include<iostream.h>
main()
{
int a;
cin>>a;
if(a>=0&&a<=12)cout<<"d";
else
if(a>=12&&a<=15)cout<<"c";
else
if(a>=15&&a<=17)cout<<"b";
else
if(a>=17&&a<=20)cout<<"a";
}
```

دستور : break;

break ; از یک حلقه خارج میشود و به حلقه بعد میرود

exit; از کل برنامه خارج میشود

continue; خط زیر دستور اجرا نمیشود و حلقه ادامه پیدا میکند

۶۴ __ برنامه ای که یک حلقه بی پایان تشکیل دهد ، سپس در داخل حلقه یک عدد پیدا کند ، اگر عدد کمتر از ۲۰ بود ۲ واحد به آن اضافه کرده و ادامه دهد ، اگر عدد بیشتر از ۲۰ بود برنامه خاتمه یابد ، اگر عدد برابر با خود ۲۰ بود عدد را بنویسد و ادامه یابد .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
float a;
for(;;)
{
cin>>a;
if(a<20)
{
a=a+2;
cout<<a;
continue;
}
if (a==20)
{
cout<<a;
continue;
}
break;
}
}
```

۶۵ __ برنامه ای که تعدادی عدد مثبت را خوانده ، بنویسد و معدل را محاسبه کند ، تا زمانی که عدد منفی وارد شد ، معدل را چاپ کند .

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
float a,n=0,s=0;
gotoxy(30,1);
cout<<"copyright by : emad attar"<<endl;
while(1)
```



```

{
cin>>a;
if(a<0) break;
s=s+a;
n++;
}
s=s/n;
cout<<"moadel :"<<" "<<s;
}

```

ترکیب گزاره ها : && (ضرب) ، || (جمع دو تا) (یا) ، ! (نقیض) .

۶۶ _ برنامه ای که نام درس ، نمره و تعداد واحد درسهای دانشجو را گرفته و بنویسد . نام آخرین درس دانشجو C است که با وارد کردن آن برنامه خاتمه می یابد .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
float n,v,a=0,av=0;
char na[20];
while(1)
{
cin>>na>>n>>v;
a+=n*v;
av+=v;
if(na=='c')break;
}
a/=av;
cout<<"moadel :"<<" "<<a;
}

```

دستور :

```

goto < برچسب >;
=> نمونه
Cin>>a;
S+=a;
If (a<0) goto 10;

```

دستور :

```

Switch ( عبارت )
{
Case دستورات : تعداد
}

```

۶۷ _ برنامه ای که یک عدد را که برابر روز هفته است را از ورودی گرفته ، در خروجی نام روز را چاپ کند .

```

#include<iostream.h>
main()
{
int a;
cin>>a;
switch(a)
{
case 0:
{
cout<<"shanbe";break;
}
}
}

```



```

case 'a':
cout<<(a+b+5);
break;
case 'b':
cout<<(a-b+5);
break;
case 'c':
cout<<(a*b+5);
break;
case 'd':
cout<<(a*5/b);
break;
}
}

```

آرایه _ بردار :

۶۹_ برنامه ای که ۱۰ عدد را خوانده ، معدل را پیدا کند ، عددهایی که از معدل کمتر هستند را چاپ کند .

```

#include<iostream.h>
main()
{
float a[10],m=0;
int i;
for(i=0;i<10;i++)
cin>>a[i];
for(i=0;i<10;i++)
m+=a[i];
m=m/10;
for(i=0;i<10;i++)
if(a[i]<m)
cout<<endl<<a[i]<<endl;
}

```

۷۰_ برنامه ای که نمرات ۱۰ دانش آموز را خوانده ، تمام دانش آموزانی که نمراتشان بیشتر از معدل است را بنویسد .

```

#include<iostream.h>
main()
{
float a[10],m=0;
int i;
for(i=0;i<10;i++)
cin>>a[i];
for(i=0;i<10;i++)
m+=a[i];
m=m/10;
for(i=0;i<10;i++)
if(a[i]>m)
cout<<endl<<a[i]<<endl;
}

```

۷۱_ برنامه ای که عدد را خوانده ، واریانس و میانگین را محاسبه و چاپ کند .

```

#include<iostream.h>
main()
{
float a[5],m=0,s=0;
int i;
for(i=0;i<5;i++)
{
cin>>a[i];
m+=a[i];
}
m/=5;

```

```

cout<<"avg:"<<m;
for(i=0;i<5;i++)
s+=(a[i]-m)*(a[i]-m);
s/=5;
cout<<endl<<"varians:"<<s;
}

```

۷۲ _ برنامه ای که ۱۰ عدد را خوانده ، سپس کوچکترین ، بزرگترین و تعداد تکرار هر دو را چاپ کند .

```

#include<iostream.h>
main()
{
int i,a[10],v=0,d=0;
float k=10,b=0;
for(i=0;i<10;i++)
{
cin>>a[i];
if(b<a[i])
b=a[i];
if(k>a[i])
k=a[i];
}
for(i=0;i<10;i++)
{
if(a[i]==b)
{
d=d+1;
}
if(a[i]==k)
{
v=v+1;
}
}
cout<<"Min:"<<k<<endl;
cout<<"Max:"<<b<<endl;
cout<<"Min Tekrar:"<<v<<" "<<"Max Tekrar:"<<d;
}

```

۷۳ _ برنامه ای که اسم و شماره تلفن داخلی افراد را گرفته و اسامی افراد را بر حسب شماره تلفن مرتب کند .

* شماره تلفن ها از ۱۰۱ تا ۱۲۰

تعداد افراد ۲۰ نفر *

```

#include<iostream.h>
main()
{
char na[20];
int i,j,t;
for(i=0;i<20;i++)
{
cout<<"plase enter phone number:"<<" ";
cin>>t;
j=t-101;
cout<<"plase enter name:"<<" ";
cin>>na[j];
}
cout<<endl;
for(i=0;i<20;i++)
{
cout<<"name is "<<na[i]<<" & ";
cout<<"phone number is "<<i+101<<endl;
}
}

```

مرتب سازی حبابی :

۷۴ _ برنامه ای که ۲۰ عدد را گرفته ، به روش حبابی مرتب کند و در خروجی چاپ نماید .

```
#include<iostream.h>
main()
{
float a[20],b;
int i,j;
for(i=0;i<20;i++)
cin>>a[i];
for(i=0;i<19;i++)
for(j=i+1;j<20;j++)
if(a[i]>a[j]){
b=a[i];
a[i]=a[j];
a[j]=b;
}
for(i=0;i<20;i++)
cout<<a[i]<<" ";
}
```

روش جستجو (ترتیبی) داده های نامرتب و داده های پیایی) _ تصادفی :

۷۵ _ برنامه ای که ۲۰ عدد را در یک تابلو قرار داده ، سپس عددی را گرفته ، سپس مشخص کند که عدد در تابلو وجود دارد یا خیر ، اگر وجود دارد شماره خانه را برگرداند در غیر اینصورت بنویسد نیست .

```
#include<iostream.h>
#include<conio.h>
main()
{
int j=0,a[20],i,c;
for(i=0;i<20;i++)
cin>>a[i];
cout<<"enter number : ";
cin>>c;
for(i=0;i<20;i++)
if(a[i]==c)
{
j=i+1;
break;
}
if(j!=0)cout<<"find"<<" "<<j;
else cout<<"not find";
}
```

۷۶ _ برنامه ای که داده ها را خوانده و به صورت صعودی یا نزولی مرتب کند ، سپس یک عدد را گرفته و به روش دودویی مشخص کند وجود دارد یا نه .

```
#include<iostream.h>
#include<conio.h>
main()
{
int a[10],i,s,j,b,m,l,h;
for(i=0;i<10;i++)
cin>>a[i];
for(i=0;i<10;i++)
for(j=i+1;j<10;j++)
if (a[i]>a[j])
{b=a[i];
a[i]=a[j];
a[j]=b;
}
}
```

```

cout<<"number for search\n";
cin>>s;
l=0;
h=9;
while(l<=h)
{
m=(l+h)/2;
if (a[m]==s)
{
cout<<"number is in list"<<" ";
cout<<m+1<<"\n";
break;
}
else if(a[m]>s)h=m-1;
else l=m+1;
}
if (l>h)cout<<"not found";
for (i=0;i<10;i++)
cout<<a[i]<<" ";
}

```

۷۷ _ برنامه ای که ۵۰۰ ریال را به مقادیر ۲۵۰، ۱۰۰، ۱۰، ۵، ۲ و ۱ ریالی تقسیم کند .

```

#include<iostream.h>
#include<conio.h>
main()
{
float i,j,k,m,n,t,b;
for(i=0;i<=500;i++)
for(j=0;j<=250;j++)
for(k=0;k<=100;k++)
for(m=0;m<=20;m++)
for(n=0;n<=2;n++)
for(t=0;t<=2;t++)
{if(i+2*j+5*j+10*m+100*n+250*t==500)
cout<<i<<j<<k<<m<<n<<t<<endl;
b++;
}
cout<<"taadad maadelat: " <<b;
}

```

۷۸ _ برنامه ای که یک ماتریس ۳*۲ را به صورت سطری خوانده و به صورت ستونی چاپ کند .

Float a[2][3]

ستون * سطر

```

-----
#include<iostream.h>
main()
{
float a[2][3];
int i,j;
//khandan

// satri be sotooni
for(i=0;i<2;i++)
for(j=0;j<3;j++)
/* sotooni be satri
for(j=0;j<3;j++)
for(i=0;i<2;i++)*/
{
cout<<"a["<<i<<"]["<<j<<"]=";
cin>>a[i][j];
}
//neveshtane transpose

```

```

cout<<"transpose = "<<endl;
for(j=0;j<3;j++)
{
cout.width(10);
for(i=0;i<2;i++)
cout<<a[i][j]<<" ";
cout<<endl;
}
}

```

۷۹ _ برنامه ای که دو تا ماتریس 2×3 سطری را خوانده سپس حاصلجمع را بنویسد .

```

#include<iostream.h>
main()
{
float a[2][3],b[2][3],c[2][3]; // a=matris 1 _ b=matris 2 _ c=hasel
int i,j;
//khandan matris 1
for(i=0;i<2;i++)
for(j=0;j<3;j++)
cin>>a[i][j];
//khandan matris 2
for(i=0;i<2;i++)
for(j=0;j<3;j++)
cin>>b[i][j];
//hasel jam
for(i=0;i<2;i++)
for(j=0;j<3;j++)
c[i][j]=a[i][j]+b[i][j];
//neveshtane hasel jam
for(i=0;i<2;i++)
{
cout.width(10);
for(j=0;j<3;j++)
cout<<c[i][j]<<" ";
cout<<endl;
}
}

```

۸۰ _ برنامه ای که دو تا ماتریس 3×2 و 2×4 را خوانده ، حاصلضرب را بدست آورده و بنویسد .

* برای ضرب a را از اولی میگیریم و j را از دومی *

```

#include<iostream.h>
main()
{
float a[2][3],b[2][4],c[3][4]; // a=matris 1 _ b=matris 2 _ c=hasel
int i,j,k;
//khandan matris 1
cout<<"enter matris 1 : "<<endl;
for(i=0;i<3;i++)
for(j=0;j<2;j++)
cin>>a[i][j];
//khandan matris 2
cout<<"enter matris 2 : "<<endl;
for(i=0;i<2;i++)
for(j=0;j<4;j++)
cin>>b[i][j];
//neveshtan matris 1
cout<<"matris 1 : "<<endl;
for(i=0;i<3;i++)
{
for(j=0;j<2;j++)
cout<<a[i][j]<<" ";
}
}

```

```

        cout<<endl;
    }
//neveshtan matrix 2
cout<<"matrix 2 : "<<endl;
for(i=0;i<2;i++)
    {
        for(j=0;j<4;j++)
            cout<<b[i][j]<<" ";
        cout<<endl;
    }
//hasel zarb
for(i=0;i<3;i++)
    for(j=0;j<4;j++)
        {
            c[i][j]=0;
            for(k=0;k<2;k++)
                c[i][j]+=a[i][k]*b[k][j];
        }
//neveshtane hasel zarb
cout<<"hasel zarb : "<<endl;
for(i=0;i<3;i++)
    {
        cout.width(10);
        for(j=0;j<4;j++)
            cout<<c[i][j]<<" ";
        cout<<endl;
    }
}

```

۸۱ _ برنامه ای که یک ماتریس توافقی ۵*۵ را کامل کند .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
.
.
.
}

```

۸۲ _ برنامه ای که یک عدد را خوانده ، مقسوم علیه های آن را پیدا کرده و بنویسد .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,a;
cin>>a;
for(i=1;i<=a/2;i++)
if(a%i==0)cout<<i<<" ";
}

```

۸۳ _ برنامه ای که دو عدد را خوانده ، مقسوم علیه های مشترک را پیدا و چاپ کند .

* ا از ۱ تا عدد کوچکتر جلو میرود *

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int i,j,a,b;
cin>>a>>b;
for(i=1;i<=a/2;i++)

```



```

if(a%i==0)if(b%i==0)
cout<<i<<" ";
}

```

۸۴ _ برنامه ای که عددی را خوانده ، مشخص کند که عدد اول است یا خیر .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
int a,i,j=0;
cin>>a;
for(i=2;i<=a/2;)
    if(a%i==0)j++;
    else
    if(j==0)cout<<"aval hast";
else cout<<"aval nist";
}

```

تابع (زیر برنامه) :

خواص :

۱. چون هر برنامه کوچک است راحت تر نوشته میشود
۲. اشکال زدایی آسان است
۳. برنامه ها مستقل هستند
۴. امکان استفاده مجدد از آنها وجود دارد
۵. متغیرها بر روی هم تأثیر نمیگذارند

_ هر برنامه دارای یک تابع و یک main است .

_ در C تابع های تودرتو وجود ندارد .

شکل کلی تابع :

```

Void
( لیست پارامترها ) نام
{
.
.
.
}

```

انواع تابع :

۱. از main چیزی نمیگیرد و به main هم چیزی برنمیگرداند
۲. از main یک مقدار میگیرد ولی به main چیزی برنمیگرداند
۳. از main تعدادی مقدار میگیرد و یک مقدار به main برمیگرداند
۴. هر تعداد که خواهد از main میگیرد و هر تعداد هم که خواهد به main برمیگرداند

مثال برای نمونه یک :

۸۵ _ برنامه ای که عددی را گرفته به مقدار عدد (*) چاپ کند .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
star()
{
int i;
for(i=0;i<=10;i++)
    cout<<"*";
}

```

```

        cout<<endl;
    }
    main ()
    {
    int i,n;
    cin>>n;
    for(i=0;i<n;i++)
    star();
    }
    // or
    /*#include<iostream.h>
    #include<conio.h>
    #include<stdio.h>
    star();
    main ()
    {
    int i,n;
    cin>>n;
    for(i=0;i<n;i++)
    star();
    }
    star()
    {
    int i;
    for(i=0;i<=10;i++)
        cout<<"*";
        cout<<endl;
    }
    }

```

مثال برای نمونه دوم تابع :

۸۶ _ برنامه ای که یک مقدار را گرفته و به صورت مثلثی (*) چاپ کند .

```

#include<iostream.h>
#include<conio.h>
#include<stdio.h>
star(int p)
{
int i;
for(i=0;i<p;i++)
    cout<<"*";
    cout<<endl;
}
main ()
{
int i,n;
cin>>n;
for(i=1;i<=n;i++)
star(i);
}

```

۸۷ _ برنامه ای که عددی را خوانده ، به کمک توابع به توان ۲ برساند ، جذرش را بگیرد ، لگاریتمش را حساب کند ، سینوس را محاسبه کند و

e را به توان عدد برساند و حاصل را در خروجی چاپ کند .

```

#include<iostream.h>
#include<math.h>
main ()
{
float a;
cin>>a;
cout<<cos(a); // cos peyda mikonad
cout<<exp(a); // e be tavane adad (neper)
cout<<log(a); // log
cout<<log10(a); //log bar mabnaye 10

```

```

cout<<eqr(a); // jazr
cout<<sqrt(a); //tavan
char b;
cin>>b;
cout<<tolower(b); // bozorg be kochik
//cout<<toupper(b); // kochik be bozorg
cout<<pow(a,3);
}

```

توابع کامل و مستقل هستند .

توابع می‌توانند در برنامه اصلی صدا شوند و همچنین در هر تابع میتوان تابع دیگری را فراخوانی کرد.

مثال برای نمونه ۲ تابع :

۸۸ _ برنامه تابعی که طول و عرض و ارتفاع را گرفته ، حجم را محاسبه کرده و در خروجی چاپ کند . (مقدار اولیه نقش آفرین نمیباشد)

```

#include<iostream.h>
aa(float a,float b,float c)
{
cout<<"hajm="<<a*b*c;
}
main()
{
float m,n,p;
cin>>m>>n>>p;
aa(m,n,p);
}

```

مثال برای حالت کلی توابع :

_ ۸۹

```

#include<iostream.h>
aa(float a=1,float b=2,float c=5)
{
cout<<"hajm="<<a*b*c<<endl;
}
main()
{
float m,n,p;
cin>>m>>n>>p;
aa(m,n,p);
//aaaaaaaa
aa(7,9,6);
//aaaaaaaa
aa(7,9);
//aaaaaaaa
aa(m,n);
//aaaaaaaa
aa(7);
aa();
}

```

۹۰ _ برنامه ای که سه عدد C,b,a را گرفته به توان یکدیگر برساند و حاصل را برگرداند ، اگر هر کدام از C,b را ندهیم ۱ حساب میشود .

```

#include<iostream.h>
aa(int a,int b=1,int c=1)
{
long int s=1,w=1,i;
for(i=1;i<=c;i++)
s*=b;
for(i=1;i<=s;i++)
w=w*a;
cout<<"tavan :"<<w<<endl;
}
main()
{

```

```

long int m,n,p;
cin>>m>>n>>p;
aa(m,n,p);
//aaaaaaaa
aa(m,n);
//aaaaaaaa
aa(m);
}

```

مثال برای نمونهٔ ۳ تابع :

```
float aa(int a,int b)
```

۹۱ _ برنامهٔ تابعی که دو عدد را خوانده ، به کمک برنامه فرعی اولی را به توان دومی رسانده و در برنامهٔ اصلی حاصل را چاپ کند .

```

#include<iostream.h>
int aa(int a,int b)
{
int s=1,i;
for(i=1;i<=b;i++)
s*=a;
return s;
}
main()
{
int m,n;
cin>>m>>n;
cout<<aa(m,n);
}

```

۹۲ _ برنامه ای که a و b را خوانده ، حاصل زیر را محاسبه کرده و بنویسد .

$$\frac{(a+b)^a (2a+b)^b}{(3a+2b)^2}$$

```

#include<iostream.h>
float aa(float a,float b)
{
float s=1,i;
for(i=1;i<=b;i++)
s*=a;
return s;
}
main()
{
float a,b;
cin>>a>>b;
cout<<aa(a+b,a)*aa(2*a+b,b)/aa(3*a+2*b,2);
}

```

۹۳ _ برنامه ای که توان دو جمله ای را خوانده ، ضرایب بسط دوجمله ای را محاسبه کرده و بنویسد .

```

1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1

```

```

#include<iostream.h>
int ff(int a)
{
int s=1,i;
for(i=1;i<=a;i++)
s*=i;
return s;
}

```

```

}
main()
{
int n,i;
cout<<"enter tavan :"<<" ";
cin>>n;
for(i=0;i<=n;i++)
cout<<ff(n)/(ff(n-i)*ff(i))<<" ";
}

```

۹۴_ برنامه ای که سه عدد m و n و p را خوانده ، حاصل زیر را محاسبه و چاپ کند .

$$(m^n \cdot n! \cdot n^p) / ((m+p)^m \cdot p!)$$

```

#include<iostream.h>
int ff(int m)
{
int s=1,i;
for(i=1;i<=m;i++)
s*=i;
return s;
}
float aa(float a,float b)
{
float s=1,i;
for(i=1;i<=b;i++)
s*=a;
return s;
}
main ()
{
int m,n,p;
cin>>m>>n>>p;
cout<<aa(m,n)*ff(n)*aa(n,p)/(aa(m+p,m)*ff(p));
}

```

کاربرد متغیر محلی :

۹۵_

```

#include<iostream.h>
aa(float a,float b)
{
float s;
s=a*b;
cout<<s;
}
bb(float a,float c)
{
float s;
s=a+c;
cout<<s;
}
main()
{
float a,l;
cin>>a;
aa(a,l);
bb(a,l);
}

```

(تغییر S (متغیر) در کل برنامه تأثیر نمیگذارد)

```
#include<iostream.h>
float a;
aa(float b)
{
    cout<<a*b;
    a++;
}
bb(float c)
{
    cout<<a+c<<endl;
    a=a+10;
}
main()
{
    cin>>a;
    aa(7);
    cout<<endl;
    bb(5);
    cout<<a;
}
```

```
#include<iostream.h>
float a=3;
aa(float b)
{
    a*=b;
}
bb(float c)
{
    a+=c;
}
cc(float c)
{
    a-=c;
}
main()
{
    float m,n;
    cin>>m>>n;
    aa(m);
    bb(n);
    cc(m);
    cout<<a;
}
```

```
#include<iostream.h>
float a=3; // motaghayere omoomi
aa(float b)
{
    float a=10;// motaghayere mahali
    a=b+::a;
    cout<<a<<endl;
    //:: 'a' mahali taghir mikonad va baghiye ba 'a' omoomi kar mikonand
}
bb(float c)
```

```

    {
    a+=c;
    }
cc(float c)
{
a-=c;
}
main()
{
float m,n;
cin>>m>>n;
aa(m);
cout<<a;
bb(n);
cc(m);
cout<<endl<<a;
}

```

نقش متغیر محلی و عمومی در یک برنامه :

— ۹۹

```

#include<iostream.h>
float a=3; // motaghayere omoomi
aa(float b) // b=20
{
float a=10;// motaghayere mahali
a=a*b+::a; // a=10*20+3=203
cout<<a<<endl; // a=203
//'::' yani 'a' mahali taghir mikonad va baghiye ba 'a' omoomi kar mikonand
}
bb(float c) // c=21
{
a+=c; // a=21+3=24 // 'a' omoomi = 24
}
cc(float c)
{
a-=c; // a-2=24-2=22
}
main()
{
float a; // 'a' mahali = 21
cin>>a; // a=20
aa(a); // aa(20)
a++; // a=21
bb(a); // bb(21)
cout<<a*::a; // 21*24=504 // 'a' mahali * 'a' omoomi
cc(2); // a=24-2=22 // cc(2) ba 'a' omoomi ejra mishavad
cout<<endl<<a; // a=21
}

```

توابع بازگشتی :

```

4! = 4 * 3! =24
3! = 3 * 2! = 6
2! = 2 * 1! = 2
1! = 1 * 0! = 1
0! = 1

```

۱۰۰ _ برنامه ای که با استفاده از تابع بازگشتی فاکتوریل را محاسبه و چاپ نماید .

```

#include<iostream.h>
int ff(int a)
{
if(a==0) return 1;
else return a*ff(a-1);
}
main()

```

```
{
int m;
cin>>m;
cout<<ff(m);
}
```

نمونه برنامه :

```
a=13;
```

a را برابر ۱۳ قرار میدهد

```
b=~a<<1;
```

۱۳ را به مبنای ۲ برده آن را نقیض کرده ، یک واحد به راست شیفت میدهد و در b میگذارد

```
cout<<(b^a);
```

b را با a با رعایت علامت یای مانع الجمع ترکیب میکند

```
00001101
00011010
```

```
-----
11100100
00011101
```

```
-----
11101001
```

۱۰۱ _ برنامه ای که صفحه مانیتور را پاک کرده ، اعداد از ۱۰ تا ۱۰۰ را پیدا کرده ، وسط صفحه ۱۰ تا ۱۰ چاپ کند .

```
#include<iostream.h>
#include<conio.h>
main()
{
int i,j;
clrscr();
gotoxy(17,12);
for(i=10;i<=100;i+=10)
cout<<i<<" ";
}
```

۱۰۲ _ برنامه ای که ۱۰ عدد مرتب را خوانده ، به صورت دودویی در خروجی چاپ کند ، سپس یک عدد را خوانده و اگر در لیست بود چاپ کند هست و اگر در لیست نبود چاپ کند نیست .

```
#include<iostream.h>
#include<conio.h>
main()
{
int j=0,a[10],i,c;
for(i=0;i<10;i++)
cin>>a[i];
cout<<"enter number : ";
cin>>c;
for(i=0;i<10;i++)
if(a[i]==c)
{
j=i+1;
break;
}
if(j!=0)cout<<"find is :"<<" "<<j;
else cout<<"not find";
}
```


کلاسهای حافظه :

در زبان C چهار نوع کلاس داریم :

۱ _ کلاس اتوماتیک :

فرم کلی :

```
نام      نوع      کلاس ;
a      int     Auto  ;
```

اگر کلاس را ننویسیم بصورت پیش فرض **auto** در نظر گرفته میشود .

auto : به متغیرهایی گفته میشود که در یک تابع و یا به صورت عمومی تعریف میشوند .

در این نوع تابع به محض تمام شدن از متغیر خارج میشود .

نمونه برنامه :

```
aa(float a)
{
  Int b;
  .
  .
  .
}
main()
{
  Int c;
  .
  .
  .
  aa(c);
  .
  .
  .
}
```

۲ _ ثابت register :

برای تعریف متغیر به شکل **register** به فرم زیر عمل میکنیم :

```
register int a ;
```

_ به تمامی متغیرهایی گفته میشود که در یک تابع نوشته میشوند ، با شروع نطبع ایجاد شده و با پایان تابع میشوند .

_ متغیرهای ثابت در ثباتها بوجود می آیند .

_ ثباتها سریعترین تغییرها هستند .

_ استفاده از این نوع متغیرها محدودیت دارد چون تعداد ثباتها محدود است .

_ در صورت استفاده از این نوع متغیرها باید به کامپایلر مراجعه کرد .

_ زمانی که از **register** استفاده کنیم در صورتی که کامپایلر نتواند این نوع را به برنامه اختصاص دهد نوع را به **auto** تغییر میدهد .

۳ _ استاتیک :

فرم کلی :

```
static int b ;
```

_ هم میتواند محلی و هم میتواند عمومی باشد .

_ در صورت اختصاص ندادن مقدار اولیه به این متغیر پیشفرض صفر قرار داده میشود .

_ وقتی وارد یک تابع میشویم از این متغیر استفاده میشود ، وقتی از تابع خارج میشویم مقدار تابع حفظ میشود ، اگر بار دوم

به این تابع مراجعه کنیم مقدار اولیه مقداری است که قبلاً حفظ شده بود .

_ وقتی برنامه را اجرا کنیم مقدار متغیر آخر تابع استاتیک مقدار متغیر اول تابع بعدی قرار داده میشود .

_ متغیرهای این نوع تابع در هیچ کجا بغير از تابع موجود قابل استفاده نیستند .

۴ _ کلاس خارجی :

فرم کلی :

```
extream      int      a ;
```

_ متغیرهایی که بصورت **extream** تعریف میشوند ، در تمام فایل‌های برنامه که در یک محل ذخیره شده اند قابل استفاده میباشد .

۱۰۳ _ برنامه ای که هر دو نوع تابع استاتیک و اتوماتیک را داشته و تفاوتشان را نشان دهد.

```
#include<iostream.h>
aa()
{
int a=3;
static int b=2;
b=a*2+b; // b=8 , a=21 _ 'a' hazf mishavad vali 'b' hazfnemishavad
a=a*7;
cout.width(30);
cout<<"auto :"<<" "<<a<<endl;
cout.width(30);
cout<<"static :"<<" "<<b<<endl;
} // vaghti az tabe kharej mishavim 'auto' hazf , vali 'static' sabet ast
main()
{
int i;
for(i=0;i<5;i++)
aa(); // adres negahdari mikonad
}
```

کلبرد تابع باعث مفهومی تر شدن و تسریع در غلط گیری میشود ولی سبب اجرای کند برنامه شده زیرا برنامه در زمان اجرا نیاز به آدرس تابع ، ذخیره آن و برگشت تابع دارد .

اگر تابع کوتاه باشد میتوانیم از لغت inline قبل از نام تابع استفاده کنیم ، هرکجا که نام تابع صدا شود خود برنامه در آنجا کپی میشود که باعث طولانی تر شدن برنامه میشود ولی در عوض اجرای برنامه سریعتر میشود .

۱۰۴ _ برنامه تابعی بنویسید که یک آرایه ۱۰ عضوی را گرفته ، بزرگترین عضو آن را نوشته و برگرداند . این برنامه به دو صورت نوشته میشود که در زیر هر دو برنامه با کلمه OR از هم جدا شده اند .

```
#include<iostream.h>
float aa(float a[10])
{
int i;
float s=a[0];
for(i=0;i<10;i++)
if(a[i]>s) s=a[i];
return s;
}
main()
{
float b[10];
int i;
cout<<"enter number :"<<endl;
for(i=0;i<10;i++)
cin>>b[i];
aa(b);
cout<<endl<<"max is:"<<" "<<aa(b);
}
```

// or

```
/*#include<iostream.h>
float a[10];
float aa()
{
int i;
float s=a[0];
for(i=0;i<10;i++)
if(a[i]>s) s=a[i];
return s;
}
main()
```

```

{
int i;
cout<<"enter number :"<<endl;
for(i=0;i<10;i++)
cin>>a[i];
cout<<endl<<"max is:"<<" "<<aa();
}*/

```

برای نوشتن تابع دو روش پیش رو داریم :

روش اول : وقتی حدس بزنیم که مقدار داده هایمان از یک مقدار بیشتر نمیشود .

روش دوم : وقتی مقدار داده هایمان را ندانیم .

۱۰۵ _ برنامه ای که ابتدا تعداد متغیرها را خوانده ، سپس به آن تعداد عدد بخواند ، آنگاه معدل را پیدا کرده بعد اعدادی که از معدل بیشتر هستند را پیدا کرده و چاپ نماید . (نمونه برای روش اول)

```

#include<iostream.h>
main()
{
float a[50],s=0;
int n,i;
cout<<"tedad :"<<" ";
cin>>n;
// khandan
for(i=0;i<n;i++)
    cin>>a[i];
// moadel
for(i=0;i<n;i++)
    s+=a[i];
    s/=n;
// peyda kardan bozorgtarin
cout<<"adad bozorgtar az moadel :"<<" ";
for(i=0;i<n;i++)
    if(a[i]>s) cout<<a[i]<<" ";
}

```

۱۰۶ _ برنامه‌ی تابعی که سه آرایه را گرفته ، بزرگترین آن را پیدا کرده و چاپ کند .

(این تابع به تعداد وابسته نیست یعنی با هر تعداد کار میکند)

```

#include<iostream.h>
float aa(float a[],int n)
{
int i;
float s=a[0];
for(i=0;i<n;i++)
    if(a[i]>s) s=a[i];
    return s;
}
main()
{
float b[5],c[10],d[15];
int i;
// khandane b
for(i=0;i<5;i++)
    cin>>b[i];
    cout<<"max b is :"<<aa(b,5);
// khandane c
for(i=0;i<10;i++)
    cin>>c[i];
    cout<<"max c is :"<<aa(c,10);
// khandane d
for(i=0;i<15;i++)
    cin>>d[i];
}

```

```

    cout<<"max d is :"<<aa(d,15);
}

```

۱۰۷ _ برنامه ای با سه تابع بنویسید که دو ماتریس ۳*۳ را خوانده ، تابع اول حاصل جمع ، تابع دوم حاصل تفریق و تابع سوم حاصلضرب را محاسبه کرده و چاپ کند .

```

#include<iostream.h>
float a[3][3],b[3][3],c[3][3];
void ad()
{
int i,j;
for(i=0;i<3;i++)
{
for(j=0;j<3;j++)
c[i][j]=a[i][j]+b[i][j];
}
}

void mi()
{
int i,j;
for(i=0;i<3;i++)
for(j=0;j<3;j++)
c[i][j]=a[i][j]-b[i][j];
}

void ml()
{
int i,j,k;
for(i=0;i<3;i++)
for(j=0;j<3;j++)
c[i][j]=0;
for(k=0;k<3;k++)
c[i][j]+=a[i][k]*b[k][j];
}

void wr()
{
int i,j;
for(i=0;i<3;i++)
{
for(j=0;j<3;j++)
cout<<c[i][j]<<" ";
cout<<endl;
}
}

main()
{
int i,j;
for(i=0;i<3;i++)
for(j=0;j<3;j++)
cin>>a[i][j];

for(i=0;i<3;i++)
for(j=0;j<3;j++)
cin>>b[i][j];

ad();
wr();
mi();
wr();
ml();
wr();
}

```

۱۰۸ _ برنامه ای که یک رشته ۱۰ تایی را خوانده سپس مشخص کند که حرف f در آن وجود دارد یا نه .

```

#include<iostream.h>
#include<conio.h>
main()

```

```

{
char a[10];
int i;
cin>>a;
for(i=0;i<10;i++)
    if(a[i]=='f')
        cout<<"hast";
getch();
}

```

طریقه خواندن یک رشته :

```

char a[10];
cin>>a ;
cout<<a;

```

۱۰۹ _ برنامه ای که یک رشته ۱۰ تایی را گرفته سپس به ترتیب عکس چاپ کند .

```

#include<iostream.h>
#include<conio.h>
main()
{
char a[10];
int i;
cin>>a;
for(i=8;i>=0;i--)
    cout<<a[i];
getch();
}

```

اشاره گر : متغیری است که میتواند آدرس یک خانه از آرایه را در خود ذخیره کند و برای خواندن و نوشتن هم مفید نیست .

شکل کلی :

```

    * اسم ;
    نوع
    * p ;
    Int

```

۱۱۰ _ برنامه ای که متغیری را خوانده سپس آدرس آن متغیر را در خروجی چاپ کند .

```

#include<iostream.h>
main()
{
int *p;
int a;
cin>>a;
p=&a;
cout<<*p;
}

```

کاربرد "\0" :

۱۱۱ _ در صورتی که از این دستور در پایان برنامه استفاده شود خروجی به صورت یک رشته چاپ میشود .

```

#include<iostream.h>
main()
{
char a[10];
cin>>a;
a[0]='a';
a[1]='l';
a[2]='i';
a[3]='\0';
cout<<a;
} خروجی = ali

```

شکل کلی اول :

```
Cin.get ( نام رشته , طول رشته );
```

نمونه :

```
Char a[20];
```

```
Cin.get(a,15);
```

به a ۲۰ خانه اختصاص می‌دهیم

میخواهیم فقط ۱۵ خانه از a پر شود

شکل کلی دوم :

```
Cin.get( جداکننده , طول رشته , نام رشته );
```

نمونه :

_ ۱۱۲

```
#include<iostream.h>
main()
{
char a[20];
cin.get(a,15,'z');
cout<<a;
}
```

متغیرهای پویا :

همانطور که گفتیم ، اشاره گر به آدرس جایی اشاره میکند که ما به کمک ای آدرس میتوانیم به محل مورد نظر دسترسی پیدا کنیم ، بنابر این نیاز نیست که به آن محل اسم معینی بدهیم، این نحوه کار با حافظه را حافظه پویا مینامند، در حقیقت با متغیرهای بی نام میخواهیم کار کنیم .
_ نحوه ایجاد و حذف متغیرهای بینام :

نوع = new اشاره گر → نحوه ایجاد
اشاره گر delete → نحوه حذف

_ ۱۱۳ برنامه ای که به کمک یک متغیر پویا ، عددی را گرفته و در ۵ ضرب کند و در خروجی چاپ کند .

```
#include<iostream.h>
main()
{
int *p;
p=new int;
cin>>*p;
cout<<5**p;
delete p;
}
```

تذکر :

اگر به هر دلیلی دستور new نتواند آدرس تخصیص دهد ، محتوای اشاره گر تهی میماند (nul میماند) در اینصورت اشاره گر به جایی اشاره نمیکند ، بنابر این لازم است پس از استفاده از هر new از تخصیص حافظه اطمینان پیدا کنیم ، برای این کار از هدر فایل "#include<stdlib.h>" استفاده میکنیم ، اگر این هدر فایل را فعال کنیم میتوانیم از دستور زیر برای اطمینان استفاده کنیم .

شکل کلی :

```
#include<stdlib.h>
.
.
.
if (!p) exit (1) ;
```

_ ۱۱۴ برنامه ای که به کمک یک متغیر پویا عددی را گرفته و به توان ۲ برساند و در خروجی چاپ کند .

```
#include<iostream.h>
#include<conio.h>
#include<stdlib.h>
main()
{
int *p;
p=new int;
if(!p)
{
```

```

    cout<<"hafeze takhsis nayaft :";
    getch();
    exit(1);
}
cout<<"enter number :";
cin>>*p;
cout<<*p**p;
delete p;
}

```

مثال برای نوع چهارم تابع :

۱۱۵ _ برنامه ای که ۲ عدد را خوانده ، به کمک یک تابع اولی را به توان ۲ برساند ، دومی را به توان ۳ برساند و در برنامه اصلی حاصل را محاسبه کرده و چاپ نماید .

```

#include<iostream.h>
aa(int *p,int *q)
{
*p=*p**p;
*q=*q**q**q;
}
main()
{
int a,b;
cout<<"enter 2 number :"<<endl;
cin>>a>>b;
aa(&a,&b);
cout<<"moraba avali ="<<a<<endl;
cout<<"mokaab dovomi ="<<b;
}

```

تذکر :

همه آرایه ها در زبان C مانند اشاره گرها هستند ، یعنی هر دو به آدرس شروع یک حافظه اشاره میکنند . همانطور که با اشاره گر میتوانیم کار کنیم با آرایه هم میتوانیم کار کنیم . در اینبار به مثال زیر توجه فرمایید .

_ ۱۱۶

```

#include<iostream.h>
main()
{
int a[5],i;
int *p;
p=a;
a[0]=12;
a[1]=15;
a[2]=10;
a[3]=4;
a[4]=7;
for(i=0;i<5;i++)
cout<<a[i]<<" ";
}
یا

```

```

#include<iostream.h>
main()
{
int a[5],i;
int *p;
p=a;
*p=12;
*(p+1)=15;
*(p+2)=10;
*(p+3)=4;
*(p+4)=7;
for(i=0;i<5;i++)
cout<<*(p+i)<<" ";
}

```

نمونه اول :

```
Char a[]="abcdef";  
Cout<<a;
```

نمونه دوم :

```
Char *a;  
a="abcdef";  
cout<<a;
```

نمونه اول و دوم هر دو یک کار را انجام می‌دهند . برای درک بهتر به مثال زیر توجه کنید .

_ ۱۱۷

```
#include<iostream.h>  
main()  
{  
char a[]="abcdef";  
cout<<a;  
}
```

یا

```
#include<iostream.h>  
main()  
{  
char *a;  
a="abcdef";  
cout<<a;  
}
```

۱۱۸ _ برنامه ای که تعدادی عدد را خوانده ، معدل را پیدا کند ، اعداد بیشتر از معدل را مشخص کرده و در خروجی چاپ کند .

(ابتدا تعداد اعداد خوانده شود و در ضمن از متغیرهای پویا نیز استفاده شود)

```
#include<iostream.h>  
#include<conio.h>  
#include<stdlib.h>  
main()  
{  
float *p,s;  
int i,n;  
cout<<"tedad vared shavad :"<<endl;  
cin>>n;  
p=new float[n];  
if(!p)  
{  
cout<<"bevojud nayamad";  
getch();  
exit(1);  
}  
cout<<"adad vared shavad :"<<endl;  
for(i=0;i<n;i++)  
cin>>*(p+i);  
s=0;  
for(i=0;i<n;i++)  
s+=*(p+i);  
s/=n;  
for(i=0;i<n;i++)  
if(*(p+i)>=s)  
cout<<endl<<*(p+i);  
delete p;  
}
```

فرم کلی قالب تابع :

```
Template<class t>  
T a,b,c;
```

۱۱۹ _ برنامه ای که برای بار اول سه عدد صحیح و برای بار دوم سه عدد حقیقی را خوانده و برای بار سوم سه کاراکتر بخواند و به کمک تابع بزرگترین

هر کدام را خوانده و بنویسد .


```

#include<iostream.h>
template<class t>
t ff(t a,t b,t c)
{
t d;
d=a>b ? a:b;
d=d>c ? d:c;
return d;
}
main()
{
// adad sahih
int a,b,c;
cout<<"3 adad sahih vared shavad :"<<endl;
cin>>a>>b>>c;
cout<<"bozorgtarin sahih :"<<ff(a,b,c);
// adad haghghi
float m,n,p;
cout<<endl<<"3 adad haghghi vared shavad :"<<endl;
cin>>m>>n>>p;
cout<<"bozorgtarin haghghi :"<<ff(m,n,p);
// horof
char w,s,v;
cout<<endl<<"3 harf vared shavad :"<<endl;
cin>>w>>s>>v;
cout<<"bozorgtarin harf :"<<ff(w,s,v);
}

```

تذکر :

```

Char a[10],b[10];
Cin>>a;

```

b=a; این روش غلط است

برای قرار دادن رشته a در رشته b دو روش زیر را در پیش داریم :

روش اول :

```

For(i=0;i<10;i++)
b[i]=a[i];
یا
For(i=0;a[i];i++)
b[i]=a[i];

```

روش دوم :

```
strcpy(a,b)
```

_ نکته : نوع (,) strcpy فقط مختص رشته ها میباشد .

مقایسه دو رشته :

a==b; این روش غلط است

برای مقایسه دو رشته دو روش زیر را پیش رو داریم :

روش اول :

مقایسه توسط حلقه

روش دوم :

```
Strcmp(a,b)
```

اگر $a < b$ جواب میشود "منفی"

اگر $a = b$ جواب میشود "صفر"

اگر $a > b$ جواب میشود "مثبت"

Showpoint

Dec

Hex

قطعه اعشار را نمایان میکند

عدد را در مبنای ۱۰ نشان میدهد

عدد را در مبنای ۱۶ نشان میدهد

Oct عدد را در مبنای ۸ نشان میدهد
Fixed عدد را بصورت عادی نشان میدهد
Right از سمت راست مرتب میکند
Left از سمت چپ مرتب میکند
Showbase مبنای عدد را نشان میدهد
Showpose علامت مثبت کنار عدد را نشان میدهد

۱۲۰ _ نمونه های بالا را در قسمت (cout.setf(ios::*)) در برنامه زیر بجای علامت * قرار داده و عملکرد هریک را مشاهده نمایید.

```
#include<iostream.h>
#include<conio.h>
#include<stdio.h>
main()
{
clrscr();
cout.setf(ios::*);
cout<<20.0;
}
```

1 _ cout.setf(ios::showpoint | ios::showpose); نقطه اعشار یا علامت مثبت کنار عدد را نشان میدهد
2 _ cout.width(عدد) میدان عدد نشان دهنده این است که داده ها در چند خانه جای گذاری شوند
3 _ cout.precision(تعداد اعشار) تعداد اعشار را نمایان میکند
4 _ cout.fill(' * ') * چاپ میکند