

چند مثال با سه ++

برنامه نویسی کامپیوتر

شهرز صیادک



۱) الگوریتمی که دو عدد **a** و **b** را از ورودی گرفته و عدد بزرگتر را نمایش دهد.
توضیح: ابتدا دو عدد از ورودی گرفته شده و با یک مقایسه، عدد بزرگتر پیدا شده و چاپ می شود.

۱- شروع

۲- متغیرهای **a** و **b** را از نوع عددی تعریف کن

۳- **a** و **b** را از ورودی بگیر

۴- اگر $a > b$ آنگاه **a** را چاپ کن در غیراینصورت **b** را چاپ کن

۵- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int a , b ;
    cout<<"2 adad vared konid : " ;
    cin>>a>>b ;
    if (a > b)
        cout<<"adade bozorgtar : "<<a ;
    else
        cout<<"adade bozorgtar : "<<b ;
    getch() ;
}
```

۲) الگوریتمی که دو عدد **a** و **b** را گرفته، حاصل جمع و حاصل ضرب آن دو را چاپ کند.
توضیح: ابتدا دو عدد از ورودی گرفته شده و با دو محاسبه، حاصل جمع و ضرب آن ها را چاپ می نماید.

۱- شروع

۲- متغیرهای **a** و **b** را از نوع عددی تعریف کن

۳- **a** و **b** را از ورودی بگیر

۴- مقدار $a+b$ را چاپ کن

۵- مقدار $a*b$ را چاپ کن

۶- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int a , b ;
    cout<<"2 adad vared konid : " ;
    cin>>a>>b ;
    cout<<"\nhasele jam : "<<a+b ;
    cout<<"\nhasele zarb : "<<a*b ;
    getch() ;
}
```

۳) الگوریتمی که عدد **a** را گرفته اگر بر ۵ بخش پذیر باشد عبارت **"bakhsh pazir"** و اگر نباشد عبارت **"bakhsh napazir"** را چاپ نماید.

توضیح: اگر عددی مثل **a** بخواهد بر عدد دیگری مثل **b** بخش پذیر باشد باید باقیمانده تقسیم **a** بر **b** برابر صفر گردد. عملگر باقیمانده در زبان ++C عملگر % است و در این مثال بخش پذیری **a** را بر ۵ چک می کنیم.

- ۱- شروع
- ۲- a را از نوع عددی تعریف کن
- ۳- a را از ورودی بگیر
- ۴- اگر باقیمانده تقسیم a بر ۵ برابر صفر بود آنگاه عبارت "bakhsh pazir" را چاپ کن در غیر این صورت عبارت "bakhsh napazir" را چاپ کن
- ۵- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int a ;
    cout<<"adadi ra vared konid : " ;
    cin>>a ;
    if (a % 5 == 0)
        cout<<"bakhsh pazir" ;
    else
        cout<<"bakhsh napazir" ;
    getch() ;
}
```

۴) الگوریتمی که مقدار a و b را از ورودی گرفته و به کمک متغیر کمکی Temp جابه جا نموده و چاپ نماید. توضیح: ابتدا a را داخل temp ریخته، سپس b را داخل a قرار داده و در نهایت temp را داخل b می ریزد.

- ۱- شروع
- ۲- a و b و temp را از نوع عددی تعریف کن
- ۳- a و b را از ورودی بگیر
- ۴- a را در temp قرار بده
- ۵- b را در a قرار بده
- ۶- temp را در b قرار بده
- ۷- a و b را چاپ کن
- ۸- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int a , b , temp ;
    cout<<"2 adad vared konid : " ;
    cin>>a>>b ;
    temp = a ;
    a = b ;
    b = temp ;
    cout<<"2 adade taavuz shode : " ;
    cout<<a<<"\n"<<b;
    getch() ;
}
```

۵) الگوریتمی که ۳ متغیر **a** و **b** و **c** را از ورودی گرفته و بزرگترین آن ها را در **max** ریخته و چاپ کند.

توضیح: ابتدا میان **a** و **b** عدد بزرگتر را انتخاب کرده و به طور موقت در **max** قرار می دهیم. سپس **c** را با مقدار فعلی **max** مقایسه کرده و اگر **c** از **max** بزرگتر بود **c** را در **max** قرار می دهیم.

۱- شروع

۲- **a** و **b** و **c** را از نوع عددی تعریف کن

۳- **a** و **b** و **c** را از ورودی بگیر

۴- اگر $a > b$ بود آنگاه **a** را در **max** قرار بده در غیراینصورت **b** را در **max** قرار بده

۵- اگر $c > \text{max}$ بود آنگاه **c** را در **max** قرار بده

۶- **max** را چاپ کن

۷- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int a , b , c , max ;
    cout<<"3 adad vared konid : " ;
    cin>>a>>b>>c ;
    if (a > b)
        max = a ;
    else
        max = b ;
    if (c > max)
        max = c ;
    cout<<"bozorgtarin adad : "<<max ;
    getch() ;
}
```

۶) الگوریتمی که **a** و **b** را از ورودی گرفته و اگر بر هم بخش پذیر بودند عبارت "**bakhsh pazir**" و در غیراینصورت عبارت "**bakhsh napazir**" را چاپ نماید. فرض هم بر این است که در هنگام ورودی گرفتن ، **a** از **b** بزرگتر باشد توضیح: اگر عددی مثل **a** بخواهد بر عدد دیگری مثل **b** بخش پذیر باشد باید باقیمانده تقسیم **a** بر **b** برابر صفر گردد. عملگر باقیمانده در زبان C++ عملگر % است.

۱- شروع

۲- **a** و **b** را از نوع عددی تعریف کن

۳- **a** و **b** را از ورودی بگیر

۴- اگر باقیمانده تقسیم **a** بر **b** برابر صفر بود عبارت "**bakhsh pazir**" و در غیراینصورت عبارت "**bakhsh napazir**"

را چاپ کن

۵- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int a , b , ;
    cout<<"2 adad vared konid : " ;
    cin>>a>>b ;
```

```

if(a % b == 0)
    cout<<"bakhsh pazir" ;
else
    cout<<"bakhsh napazir" ;
getch( ) ;
}

```

۷) الگوریتمی که طول و عرض مستطیلی را گرفته و محیط و مساحت آن را چاپ نماید.

توضیح : دو عدد X و Y را به عنوان طول و عرض مستطیل گرفته و محیط را از رابطه $2(x+y)$ در متغیر mohit ریخته و مساحت را از رابطه $x*y$ در متغیر masahat قرار می دهد.

۱- شروع

۲- X و Y و mohit و masahat را از نوع عددی تعریف کن

۳- X و Y را از ورودی بگیر

۴- مقدار $2*(x+y)$ را در mohit قرار بده

۵- مقدار $x*y$ را در masahat قرار بده

۶- mohit و masahat را چاپ کن

۷- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int x , y , mohit , masahat ;
    cout<<"toole mostatil ra vared konid : " ;
    cin>>x ;
    cout<<"arze mostatil ra vared konid : " ;
    cin>>y ;
    mohit = 2*(x+y) ;
    masahat = x * y ;
    cout<<"\nmohite mostatil : "<<mohit ;
    cout<<"\nmasahate mostatil : "<<masahat ;
    getch( ) ;
}

```

۸) الگوریتمی که X را به عنوان ضلع مربعی گرفته و محیط و مساحت آن را چاپ نماید.

توضیح : در مربع ، طول و عرض با هم برابر بوده لذا محیط از رابطه $4*x$ و مساحت از رابطه $x*x$ بدست می آید.

۱- شروع

۲- X و mohit و masahat را از نوع عددی تعریف کن

۳- X را از ورودی بگیر

۴- مقدار $4*x$ را در mohit قرار بده

۵- مقدار $x*x$ را در masahat قرار بده

۶- mohit و masahat را چاپ کن

۷- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int x , mohit , masahat ;
    cout<<"andazeye zele morabba ra vared konid : " ;
    cin>>x ;
    mohit = 4 * x ;
    masahat = x * x ;
    cout<<"\nmohite morabba : "<<mohit ;
    cout<<"\nmasahate morabba : "<<masahat ;
    getch( ) ;
}

```

۹) الگوریتمی که n عدد مختلف را از ورودی گرفته و حاصل جمع این n عدد را در sum و حاصل ضرب آن ها را $mult$ قرار دهد.

توضیح: ابتدا باید خود n را از ورودی گرفته تا مشخص شود که قرار است چند عدد خوانده شود. این برنامه نیاز به حلقه تکرار دارد که n بار تکرار شده و شمارنده آن i است و در هر بار یک عدد جدید از ورودی گرفته و در a قرار می دهد و مقدار sum را با مقدار a جدید جمع کرده و در sum ریخته و مقدار $mult$ را نیز در a جدید ضرب کرده و در $mult$ می ریزد. قبل از ورود به حلقه تکرار باید sum را با صفر مقداردهی اولیه کرده و $mult$ را با یک پر کنیم. (زیرا در جمع صفر بی تاثیر است و در ضرب یک) پس از خروج از حلقه نیز مقدار نهایی sum و $mult$ را چاپ می نماییم.

- ۱- شروع
- ۲- متغیرهای a و n و sum و $mult$ و i را از نوع عددی تعریف کن
- ۳- n را از ورودی بگیر
- ۴- i را برابر یک قرار بده
- ۵- sum را برابر صفر قرار بده
- ۶- $mult$ را برابر یک قرار بده
- ۷- تا زمانی که $i \leq n$ است مراحل ۸ تا ۱۱ را تکرار کن
- ۸- a را از ورودی بگیر
- ۹- sum را برابر $sum + a$ قرار بده
- ۱۰- $mult$ را برابر $mult * a$ قرار بده
- ۱۱- i را برابر $i+1$ قرار بده
- ۱۲- مقدار sum و $mult$ را چاپ کن
- ۱۳- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int n , a , i , sum , mult ;
    cout<<"tedade adade voroodi ra vared konid : " ;
    cin>>n ;
    sum = 0 ;
    mult = 1 ;
}

```

```

for ( i=1 ; i<=n ; i++ )
{
    cout<<"\nadad ra vared konid : " ;
    cin>>a ;
    sum = sum + a ;
    mult = mult * a ;
}
cout<<"\nhasele jam : "<<sum ;
cout<<"\nhasele zarb : "<<mult ;
getch( ) ;
}

```

۱۰) الگوریتمی که n عدد را از ورودی گرفته و تعداد اعداد مثبت و منفی و صفر را چاپ نماید.

توضیح: ابتدا باید خود n را از ورودی گرفته تا مشخص شود که قرار است چند عدد خوانده شود. این برنامه نیاز به حلقه تکرار دارد که n بار تکرار شده و شمارنده آن i است و در هر بار یک عدد جدید از ورودی گرفته و در a قرار می دهد. همچنین نیاز به سه شمارنده است X برای مثبت ها ، Y برای منفی ها و Z برای صفرها. در یک شرط بررسی می کند که اگر عدد صفر باشد به Z یکی اضافه می شود. اگر مثبت باشد به X و اگر منفی باشد شمارنده Y اضافه می شود. این سه شمارنده نیز قبل از ورود به حلقه باید مقدار اولیه صفر بگیرند.

۱- شروع

۲- متغیرهای n و i و X و Y و Z و a را از نوع عددی تعریف کن

۳- مقدار X و Y و Z را برابر صفر قرار بده

۴- n را از ورودی بگیر

۵- i را برابر یک قرار بده

۶- تا زمانی که $i \leq n$ است مراحل ۷ تا ۱۱ را تکرار کن

۷- مقدار a را از ورودی بگیر

۸- اگر a برابر صفر بود آنگاه به Z یکواحد اضافه کن

۹- اگر a بزرگتر از صفر بود آنگاه به X یکواحد اضافه کن

۱۰- اگر a کوچکتر از صفر بود آنگاه به Y یکواحد اضافه کن

۱۱- مقدار i را برابر $i+1$ قرار بده

۱۲- مقدار X و Y و Z را چاپ کن

۱۳- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int n , a , i , x , y , z ;
    cout<<"tedade adade voroodi ra vared konid : " ;
    cin>>n ;
    x = 0 ;
    y = 0 ;
    z = 0 ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\nadad ra vared konid : " ;

```

```

cin>>a ;
if (a == 0)
    z = z + 1 ;
if (a > 0)
    x = x + 1 ;
if (a < 0)
    y = y + 1 ;
}
cout<<"\ntedade sefrha : "<<z ;
cout<<"\ntedade mosbatha : "<<x ;
cout<<"\ntedade manfiha : "<<y ;
getch( ) ;
}

```

(۱) الگوریتمی که n عدد را از ورودی گرفته و بزرگترین آن ها را یافته و در max قرار داده و چاپ نماید.

توضیح : برای این کار ابتدا باید یک عدد گرفته و آن را در max قرار دهیم. سپس در یک حلقه تکرار اعداد را از دومی تا n امی یکی یکی گرفته و با max مقایسه نماییم و هر کدام از max فعلی بزرگتر بود در max قرار داده تا max جدید بدست آید.

۱- شروع

۲- متغیرهای a و n و i و max را از نوع عددی تعریف کن

۳- n را به عنوان تعداد اعداد از ورودی بگیر

۴- متغیر a را به عنوان عدد اول از ورودی بگیر

۵- مقدار a را در max قرار بده

۶- i را برابر دو قرار بده

۷- تا زمانی که $i \leq n$ است مراحل ۸ تا ۱۰ را تکرار کن

۸- مقدار a را از ورودی بگیر

۹- اگر $a > \max$ بود آنگاه a را در max قرار بده

۱۰- مقدار i را برابر $i+1$ قرار بده

۱۱- مقدار max نهایی را چاپ کن

۱۲- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int n , a , i , max ;
    cout<<"tedade adade voroodi ra vared konid : " ;
    cin>>n ;
    cout<<"\nadad ra vared konid : " ;
    cin>>a ;
    max = a ;
    for ( i=2 ; i<=n ; i++ )
    {
        cout<<"\nadad ra vared konid : " ;
        cin>>a ;
        if (a > max)
            max = a ;
    }
}

```



```

cout<<"\nbozorgtarin adad : "<<max ;
getch( ) ;
}

```

۱۲) الگوریتمی بنویسید که عدد n را گرفته و فاکتوریل آن را محاسبه و چاپ نماید.

توضیح : روند کار این برنامه مشابه الگوریتم محاسبه حاصل ضرب 1 تا n می باشد. بدین صورت که ابتدا خود n را از ورودی گرفته و با استفاده از شمارنده i که از یک تا n می شمارد هر بار مقدار $fact$ را در i ضرب کرده تا i به n برسد. لازم به ذکر است که در شروع کار و قبل از ورود به حلقه ابتدا باید $fact$ با یک ، مقداردهی اولیه شود که در ضرب بی تاثیر است.

- ۱- شروع
- ۲- متغیرهای n و i و $fact$ را از نوع عددی تعریف کن
- ۳- مقدار n را از ورودی بگیر
- ۴- مقدار i را برابر یک قرار بده
- ۵- مقدار $fact$ را برابر یک قرار بده
- ۶- تا زمانی که $i \leq n$ است مراحل ۷ تا ۸ را تکرار کن
- ۷- مقدار $fact$ را برابر $fact * i$ قرار بده
- ۸- به i یک واحد اضافه کن
- ۹- مقدار $fact$ را چاپ کن
- ۱۰- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int n , i , fact ;
    cout<<"tedade adade voroodi ra vared konid : " ;
    cin>>n ;
    fact = 1 ;
    for ( i=1 ; i<=n ; i++ )
        fact = fact * i ;
    cout<<"\nfactoriel : "<<fact ;
    getch( ) ;
}

```

۱۳) الگوریتمی بنویسید که n نمره یک دانشجو را گرفته و معدل آن را محاسبه نماید.

توضیح : ابتدا تعداد نمرات را از ورودی گرفته و در n قرار می دهیم. سپس به کمک یک حلقه تکرار ، که شمارنده آن i است و n بار تکرار می شود در هر بار یک نمره دانشجو را گرفته و به متغیر حاصل جمع نمرات که sum است اضافه می کنیم. در بیرون از حلقه حاصل جمع نمرات را تقسیم بر تعداد آن ها یعنی n کرده و معدل بدست می آید.

- ۱- شروع
- ۲- متغیرهای n و i را از نوع عدد صحیح تعریف کن
- ۳- متغیرهای a و sum و ave را از نوع عدد اعشاری تعریف کن
- ۴- مقدار n را از ورودی بگیر
- ۵- مقدار sum را برابر صفر قرار بده
- ۶- مقدار i را برابر یک قرار بده
- ۷- تا زمانی که $i \leq n$ است مراحل تا را تکرار کن

- ۸- مقدار a را به عنوان نمره دانشجو از ورودی بگیر
- ۹- مقدار a را به sum اضافه کن (sum=sum+a)
- ۱۰- به i یک واحد اضافه کن
- ۱۱- مقدار sum را بر n تقسیم کرده و در ave قرار بده
- ۱۲- مقدار ave را چاپ کن
- ۱۳- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int n , i ;
    float a , sum , ave ;
    cout<<"tedade nomarate daneshjoo ra vared konid : " ;
    cin>>n ;
    sum = 0 ;
    for ( i=1 ; i<=n ; i++)
    {
        cout<<"\nnomre : " ;
        cin>>a ;
        sum = sum + a ;
    }
    ave = sum / n ;
    cout<<"\nmoaddele daneshjoo : "<<ave ;
    getch( ) ;
}
```

۱۴) الگوریتمی بنویسید که نام و نمره تعدادی دانشجو را گرفته و نام و نمره دانشجویان تجدید را چاپ نماید.

توضیح : ابتدا تعداد دانشجویان و نمراتشان را از ورودی گرفته و در n قرار می دهیم. سپس به کمک یک حلقه تکرار ، که شمارنده آن i است و n بار تکرار می شود در هر بار یک نام دانشجو و نمره آن را گرفته و به کمک یک شرط ، بررسی می کنیم که اگر نمره وی زیر ۱۰ باشد نام و نمره آن را به عنوان دانشجوی تجدیدی چاپ نماید. در این برنامه برای نام هر دانشجو از متغیر رشته ای باید استفاده شود که نحوه تعریف آن در زبان C++ می تواند به صورت ; char name [15] باشد که یک رشته ۱۵ حرفی را در خود جای می دهد.

- ۱- شروع
- ۲- متغیرهای n و i را از نوع عدد صحیح و متغیر a را (به عنوان نمره) از نوع عدد اعشاری تعریف کن
- ۳- متغیر name را از نوع رشته تعریف کن
- ۴- متغیر n را از ورودی بگیر
- ۵- مقدار i را برابر یک قرار بده
- ۶- تا زمانی که i<=n است مراحل ۷ تا ۱۰ را تکرار کن
- ۷- نام دانشجو را از ورودی بگیر و در name قرار بده
- ۸- نمره دانشجو را از ورودی بگیر و در a قرار بده
- ۹- اگر a<10 آنگاه نام دانشجو و نمره آن را چاپ کن
- ۱۰- به i یک واحد اضافه کن
- ۱۱- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int n , i ;
    float a ;
    char name [15] ;
    cout<<"tedade daneshjoo ra vared konid : " ;
    cin>>n ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\nesme daneshjoo : " ;
        cin>>name ;
        cout<<"\nnomreya daneshjoo : " ;
        cin>>a ;
        if (a<10)
            cout<<"\ndaneshjoo "<<name<<" ba nomreya "<<a<<" tajdid shod east" ;
    }
    getch() ;
}

```

۱۵) الگوریتمی بنویسید که اعداد ۱ تا ۱۰ را در خطوط مجزا چاپ نماید.

توضیح : این برنامه نیازمند یک حلقه تکرار است که شمارنده آن (i) از ۱ تا ۱۰ می شمارد و در هر بار خود شمارنده چاپ شده و با کاراکتر '\n' به خط بعد می رود.

- ۱- شروع
- ۲- متغیر i را از نوع عدد صحیح تعریف کن
- ۳- مقدار i را برابر یک قرار بده
- ۴- تا زمانی که $i \leq 10$ است مراحل ۵ تا ۶ را تکرار کن
- ۵- مقدار i را چاپ کن و به خط بعد برو
- ۶- به i یک واحد اضافه کن
- ۷- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i ;
    for ( i=1 ; i<=10 ; i++ )
        cout<<i<<"\n" ;
    getch() ;
}

```

۱۶) الگوریتمی بنویسید که اعداد ۱ تا ۱۰ را به صورت معکوس در خطوط مجزا چاپ نماید.

توضیح : این برنامه نیازمند یک حلقه تکرار است که شمارنده آن (i) از ۱۰ تا ۱ می شمارد و در هر بار خود شمارنده چاپ شده و با کاراکتر '\n' به خط بعد می رود.

- ۱- شروع
- ۲- متغیر i را از نوع عدد صحیح تعریف کن

- ۳- مقدار i را برابر ده قرار بده
- ۴- تا زمانی که $i \geq 1$ است مراحل ۵ تا ۶ را تکرار کن
- ۵- مقدار i را چاپ کرده و به خط بعد برو
- ۶- از i یک واحد کم کن
- ۷- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i;
    for ( i=10; i>=1; i-- )
        cout<<i<<'\n';
    getch();
}
```

۱۷) الگوریتمی بنویسید که n را از ورودی گرفته و اعداد ۱ تا n را به صورت عادی و سپس به صورت معکوس چاپ نماید. توضیح: در این برنامه ابتدا باید خود n از ورودی گرفته شود و سپس نیازمند دو حلقه تکرار مجزا است که شمارنده حلقه اول (i) از ۱ تا n می شمارد و در هر بار خود شمارنده چاپ شده و با کاراکتر '\n' به خط بعد می رود و شمارنده حلقه دوم (i) از n تا ۱ می شمارد و در هر بار خود شمارنده چاپ شده و با کاراکتر '\n' به خط بعد می رود.

- ۱- شروع
- ۲- متغیرهای i و n را از نوع عددی صحیح تعریف کن
- ۳- مقدار n را از ورودی بگیر
- ۴- مقدار i را برابر یک قرار بده
- ۵- تا زمانی که $i \leq n$ است مراحل ۶ تا ۷ را تکرار کن
- ۶- مقدار i را چاپ کرده و به خط بعد برو
- ۷- به i یک واحد اضافه کن
- ۸- مقدار i را برابر n قرار بده
- ۹- تا زمانی که $i \geq 1$ است مراحل ۱۰ تا ۱۱ را تکرار کن
- ۱۰- مقدار i را چاپ کرده و به خط بعد برو
- ۱۱- از i یک واحد کم کن
- ۱۲- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i, n;
    for ( i=1; i<=n; i++ )
        cout<<i<<'\n';
    for ( i=n; i>=1; i-- )
        cout<<i<<'\n';
    getch();
}
```

۱۸) الگوریتمی بنویسید که اعداد زوج میان ۱ تا ۲۰ را چاپ نماید.

توضیح : این برنامه نیازمند یک حلقه تکرار است که شمارنده آن (i) از ۱ تا ۲۰ می شمارد و در هر بار از طریق یک دستور شرطی چک می شود که اگر i بر ۲ بخش پذیر بود (زوج بود) خود i چاپ شود. البته می توان این برنامه را به شکل دیگری نیز نوشت. اولین عدد زوج بین ۱ تا ۲۰ عدد ۲ است. پس حلقه تکرار باید از ۲ شروع شده و ۲ واحد ۲ واحد اضافه شود تا به بیست برسد و در هر بار i چاپ می گردد.

روش اول :

- ۱- شروع
- ۲- متغیر i را از نوع عدد صحیح تعریف کن
- ۳- مقدار را برابر یک قرار بده
- ۴- تا زمانی که $i \leq 20$ است مراحل ۵ تا ۶ را تکرار کن
- ۵- اگر باقیمانده تقسیم i بر ۲ برابر صفر بود آنگاه i را چاپ کن
- ۶- به i یک واحد اضافه کن
- ۷- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i;
    for ( i=1 ; i<=20 ; i++)
        if (i % 2 == 0)
            cout<<i<<'\n' ;
    getch() ;
}
```

روش دوم :

- ۱- شروع
- ۲- متغیر i را از نوع عدد صحیح تعریف کن
- ۳- مقدار را برابر دو قرار بده
- ۴- تا زمانی که $i \leq 20$ است مراحل ۵ تا ۶ را تکرار کن
- ۵- مقدار i را چاپ کن
- ۶- به i دو واحد اضافه کن
- ۷- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i;
    for ( i=2 ; i<=20 ; i=i+2 )
        cout<<i<<'\n' ;
    getch() ;
}
```

۱۹) الگوریتمی بنویسید که دو عدد **a** و **b** را از ورودی گرفته و اعداد فرد میان **a** تا **b** را چاپ نماید.

توضیح : در این برنامه ابتدا دو متغیر **a** و **b** از ورودی گرفته شده و نیازمند یک حلقه تکرار است که شمارنده آن (**i**) از **a** تا **b** می شمارد و در هر بار از طریق یک دستور شرطی چک می شود که اگر باقیمانده تقسیم **i** بر ۲ برابر ۱ بود (فرد بود) خود **i** چاپ شود.

۱- شروع

۲- متغیرهای **i** و **a** و **b** را از نوع عدد صحیح تعریف کن

۳- مقدار **a** و **b** را از ورودی بگیر

۴- مقدار **i** را برابر **a** قرار بده

۵- تا زمانی که $i \leq b$ است مراحل ۶ تا ۷ را تکرار کن

۶- اگر باقیمانده تقسیم **i** بر ۲ برابر یک شد آنگاه **i** را چاپ کن

۷- به **i** یک واحد اضافه کن

۸- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i , a , b ;
    cout<<"\nadade ebteda ra vared konid : " ;
    cin>>a ;
    cout<<"\nadade enteha ra vared konid : " ;
    cin>>b ;
    for ( i=a ; i<=b ; i++ )
        if (i % 2 == 1)
            cout<<i<<"\n" ;
    getch() ;
}
```

۲۰) الگوریتمی بنویسید که دو عدد **a** و **b** را از ورودی گرفته و مضارب ۵ میان **a** تا **b** را چاپ نماید.

توضیح : در این برنامه ابتدا دو متغیر **a** و **b** از ورودی گرفته شده و نیازمند یک حلقه تکرار است که شمارنده آن (**i**) از **a** تا **b** می شمارد و در هر بار از طریق یک دستور شرطی چک می شود که اگر باقیمانده تقسیم **i** بر ۵ برابر صفر بود خود **i** چاپ شود.

۱- شروع

۲- متغیرهای **i** و **a** و **b** را از نوع عدد صحیح تعریف کن

۳- مقدار **a** و **b** را از ورودی بگیر

۴- مقدار **i** را برابر **a** قرار بده

۵- تا زمانی که $i \leq b$ است مراحل ۶ تا ۷ را تکرار کن

۶- اگر باقیمانده تقسیم **i** بر ۵ برابر یک شد آنگاه **i** را چاپ کن

۷- به **i** یک واحد اضافه کن

۸- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i , a , b ;
    cout<<"\nadade ebteda ra vared konid : " ;
    cin>>a ;
```

```

cout<<"\nadade enteha ra vared konid : " ;
cin>>b ;
for ( i=a ; i<=b ; i++ )
    if (i % 5 == 0)
        cout<<i<<"\n" ;
getch( ) ;
}

```

۲۱) الگوریتمی بنویسید که مقدار a را از ورودی گرفته و مقسوم علیه های آن را چاپ نماید.

توضیح : مقسوم علیه های a اعدادی هستند میان ۱ تا a که a بر آن ها بخش پذیر است. لذا نیازمند یک حلقه تکرار هستیم که شمارنده آن (i) از ۱ تا a تکرار شود و در هر بار به کمک یک شرط چک شود که اگر a بر i بخش پذیر باشد i را چاپ کند.

- ۱- شروع
- ۲- متغیرهای a و i را از نوع عدد صحیح تعریف کن
- ۳- مقدار a را از ورودی بگیر
- ۴- مقدار i را برابر یک قرار بده
- ۵- تا زمانی که $i \leq a$ است مراحل ۶ تا ۷ را تکرار کن
- ۶- اگر باقیمانده تقسیم a بر i برابر صفر بود آنگاه i را چاپ کن
- ۷- به i یک واحد اضافه کن
- ۸- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int i , a ;
    cout<<"\nadadi ra vared konid : " ;
    cin>>a ;
    for ( i=1 ; i<=a ; i++ )
        if (a % i == 0)
            cout<<i<<"\n" ;
    getch( ) ;
}

```

۲۲) الگوریتمی بنویسید که مقدار a را از ورودی گرفته و تعداد مقسوم علیه های آن را چاپ نماید.

توضیح : مقسوم علیه های a اعدادی هستند میان ۱ تا a که a بر آن ها بخش پذیر است. لذا نیازمند یک حلقه تکرار هستیم که شمارنده آن (i) از ۱ تا a تکرار شود و در هر بار به کمک یک شرط چک شود که اگر a بر i بخش پذیر باشد آنگاه باید یک واحد به مقدار یک شمارنده مانند $count$ اضافه گردد. لازم به ذکر است که مقدار $count$ باید قبل از ورود به حلقه ، صفر باشد زیرا در شروع کار هیچ مقسوم علیه‌ی یافت نشده است.

- ۱- شروع
- ۲- متغیرهای a و i و $count$ را از نوع عدد صحیح تعریف کن
- ۳- مقدار a را از ورودی بگیر
- ۴- مقدار $count$ را برابر صفر قرار بده
- ۵- مقدار i را برابر یک قرار بده
- ۶- تا زمانی که $i \leq a$ است مراحل ۷ تا ۸ را تکرار کن
- ۷- اگر باقیمانده تقسیم a بر i برابر صفر بود آنگاه به $count$ یک واحد اضافه کن

- ۸- به i یک واحد اضافه کن
- ۹- مقدار `count` را چاپ کن
- ۱۰- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int i , a , count ;
    cout<<"\nadadi ra vared konid : " ;
    cin>>a ;
    count = 0 ;
    for ( i=1 ; i<=a ; i++ )
        if (a % i == 0)
            count = count + 1 ;
    cout<<"\ntedade maghsoom alayh : "<<count ;
    getch( ) ;
}
```

۲۳) الگوریتمی بنویسید که یک عدد را از ورودی گرفته و مشخص کند که عدد اول است یا خیر.

توضیح : عدد اول عددی است که به غیر از یک و خودش مقسوم علیه دیگری نداشته باشد یعنی فقط ۲ مقسوم علیه داشته باشد. این برنامه را می توان به ۲ روش زیر نوشت :

روش اول : با توجه به الگوریتم مثال قبل ، بعد از خروج از حلقه مقدار `count` که بیانگر تعداد مقسوم علیه ها است را چک می کنیم و اگر ۲ بود یعنی عدد اول است و در صورتیکه بیشتر از ۲ بود یعنی عدد a به اعداد دیگری نیز بخش پذیر بوده در نتیجه اول نیست. روش دوم : می توان از یک `flag` استفاده نمود که یک بودن آن به معنی اول بودن عدد a است و صفر شدن آن به معنی اول نبودن عدد a است. قبل از شروع به حلقه تکرار ، ابتدا `flag` را برابر یک قرار می دهیم یعنی فرض می کنیم که عدد اول است مگر آنکه خلاف آن ثابت شود. و در داخل حلقه تکرار که شمارنده i آن از ۲ تا قبل از a می چرخد چک می کنیم که اگر i به هر کدام از اعداد میان ۲ تا قبل از a بخش پذیر باشد به معنی اول نبودن عدد a است و در نتیجه `flag` را برابر صفر قرار می دهیم. در انتها و در خارج از حلقه با یک شرط مقدار `flag` را بررسی کرده و اگر همچنان یک باشد یعنی عدد a اول است ولی اگر صفر شده باشد یعنی عدد a اول نیست.

روش اول :

- ۱- شروع
- ۲- متغیرهای a و i و `count` را از نوع عدد صحیح تعریف کن
- ۳- مقدار a را از ورودی بگیر
- ۴- مقدار `count` را برابر صفر قرار بده
- ۵- مقدار i را برابر یک قرار بده
- ۶- تا زمانی که $i \leq a$ است مراحل ۷ تا ۸ را تکرار کن
- ۷- اگر باقیمانده تقسیم a بر i برابر صفر بود آنگاه به `count` یک واحد اضافه کن
- ۸- به i یک واحد اضافه کن
- ۹- اگر مقدار `count` برابر ۲ بود چاپ کن "adad aval ast"
- ۱۰- اگر مقدار `count` بزرگتر از ۲ بود چاپ کن "adad aval nist"

۱۱- پایان


```

main() {
    int i , a , count ;
    cout<<"\nadadi ra vared konid : " ;
    cin>>a ;
    count = 0 ;
    for ( i=1 ; i<=a ; i++ )
        if (a % i == 0)
            count = count + 1 ;
    if (count == 2)
        cout<<"\nadad aval ast" ;
    if (count >= 2)
        cout<<"\nadad aval nist" ;
    getch( ) ;
}

```

روش دوم :

- ۱- شروع
- ۲- متغیرهای a و i و $flag$ را از نوع عدد صحیح تعریف کن
- ۳- مقدار a را از ورودی بگیر
- ۴- مقدار $flag$ را برابر یک قرار بده
- ۵- مقدار i را برابر دو قرار بده
- ۶- تا زمانی که $i < a$ است مراحل ۷ تا ۸ را تکرار کن
- ۷- اگر باقیمانده تقسیم a بر i برابر صفر بود آنگاه مقدار $flag$ را برابر صفر قرار بده
- ۸- به i یک واحد اضافه کن
- ۹- اگر مقدار $flag$ برابر یک بود چاپ کن "adad aval ast"
- ۱۰- اگر مقدار $flag$ برابر صفر بود چاپ کن "adad aval nist"
- ۱۱- پایان

```

main() {
    int i , a , flag ;
    cout<<"\nadadi ra vared konid : " ;
    cin>>a ;
    flag = 1 ;
    for ( i=2 ; i < a ; i++ )
        if (a % i == 0)
            flag = 0 ;
    if (flag == 1)
        cout<<"\nadad aval ast" ;
    if (flag == 0)
        cout<<"\nadad aval nist" ;
    getch( ) ;
}

```

۲۴) الگوریتمی که عدد n را از ورودی گرفته و حاصل جمع اعداد ۱ تا n و حاصل ضرب اعداد ۱ تا n را محاسبه و چاپ نماید.

توضیح : ابتدا باید خود n را از ورودی گرفته تا مشخص شود که قرار است حاصل جمع و ضرب ۱ تا چند را محاسبه نماییم این برنامه نیاز به حلقه تکرار دارد که از ۱ تا n تکرار شده و شمارنده آن i است و در هر بار مقدار خود شمارنده یعنی i با sum جمع شده و در

mult ضرب می شود. قبل از ورود به حلقه تکرار باید sum را با صفر مقداردهی اولیه کرده و mult را با یک پر کنیم. (زیرا در جمع صفر بی تاثیر است و در ضرب یک) پس از خروج از حلقه نیز مقدار نهایی sum و mult را چاپ می نماییم.

- ۱- شروع
- ۲- متغیرهای n و sum و mult و i را از نوع عددی تعریف کن
- ۳- n را از ورودی بگیر
- ۴- i را برابر یک قرار بده
- ۵- sum را برابر صفر قرار بده
- ۶- mult را برابر یک قرار بده
- ۷- تا زمانی که $i \leq n$ است مراحل ۸ تا ۱۰ را تکرار کن
- ۸- sum را برابر $sum + i$ قرار بده
- ۹- mult را برابر $mult * i$ قرار بده
- ۱۰- i را برابر $i+1$ قرار بده
- ۱۱- مقدار sum و mult را چاپ کن
- ۱۲- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int n , i , sum , mult ;
    cout<<"adadi ra vared konid : " ;
    cin>>n ;
    sum = 0 ;
    mult = 1 ;
    for ( i=1 ; i<=n ; i++ )
    {
        sum = sum + i ;
        mult = mult * i ;
    }
    cout<<"\nhasele jame 1 ta n : "<<sum ;
    cout<<"\nhasele zarbe 1 ta n : "<<mult ;
    getch( ) ;
}
```

۲۵) الگوریتمی بنویسید که درآمد ماهانه تعدادی کارمند را به همراه نام آن ها گرفته و براساس تعرفه زیر، مالیات آن ها را محاسبه کرده و دریافتی خالص و میزان مالیات را نمایش دهد.

مالیات	درآمد ماهانه (تومان)
0	کمتر از ۳۰۰۰۰۰ تومان
۷٪ نسبت به مازاد	بین ۳۰۰۰۰۰ تا ۸۰۰۰۰۰ تومان
۱۲٪ نسبت به مازاد	بیشتر از ۸۰۰۰۰۰ تومان

توضیح: ابتدا باید n را از ورودی به عنوان تعداد کارکنان دریافت کرد و حلقه تکراری با شمارنده i داریم که از ۱ تا n یعنی n بار تکرار شده و در هر بار، نام کارمند را گرفته و در متغیر رشته ای name قرار داده و حقوق آن را نیز گرفته و در S قرار می دهد. سپس به کمک سه دستور شرطی، با توجه به جدول فوق، میزان مالیات کارمند مشخص شده و در T قرار می گیرد. با توجه به مالیات هر

شخص و میزان حقوق آن ، میزان خالص دریافتی از رابطه S-T بدست آمده و در متغیر P قرار می گیرد. در انتها نام شخص و خالص دریافتی یعنی name و P را چاپ می کنیم.

با توجه به جدول اگر سه شخص مختلف ، درآمدهای زیر را داشته باشند میزان مالیات هریک به این گونه محاسبه می شود :

$$\text{اگر } S = 200000 \rightarrow T = 0$$

$$\text{اگر } S = 500000 \rightarrow T = (S-300000) * 0.07$$

$$\text{اگر } S = 2000000 \rightarrow T = (S-800000) * 0.12 + (800000-300000) * 0.07$$

۱- شروع

۲- متغیرهای n (تعداد کارمند) و i (شمارنده) و P (خالص دریافتی) و S (درآمد) و T (مالیات) را از نوع عدد صحیح

تعریف کن

۳- متغیر name را از نوع رشته ای تعریف کن

۴- مقدار n را از ورودی بگیر

۵- مقدار i را برابر یک قرار بده

۶- تا زمانی که $i \leq n$ است مراحل ۷ تا ۱۴ را تکرار کن

۷- نام کارمند را از ورودی گرفته و در name قرار بده

۸- میزان درآمد کارمند را از ورودی گرفته و در S قرار بده

۹- اگر $S \leq 300000$ بود آنگاه $T = 0$ قرار بده

۱۰- اگر $S > 300000$ و $S \leq 800000$ بود آنگاه $T = (S-300000) * 0.07$ قرار بده

۱۱- اگر $S > 800000$ بود آنگاه $T = (S-800000) * 0.12 + (800000-300000) * 0.07$ قرار بده

۱۲- مقدار S-T را به عنوان خالص دریافتی در P قرار بده

۱۳- مقدار T و P را به عنوان میزان مالیات و دریافتی خالص چاپ کن

۱۴- مقدار i را یک واحد افزایش بده

۱۵- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int n , i ;
    long int S , T , P ;
    char name [20] ;
    cout<<"tedade karmandan ra vared konid : " ;
    cin>>n ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\nesme karmand : " ;
        cin>>name ;
        cout<<"\nmizane daramad : " ;
        cin>>S ;
        if (S<=300000)
            T = 0 ;
        if (S>300000 && S<=800000)
            T = (S-300000) * 0.07 ;
        if (S>800000)
            T = (S-800000) * 0.12 + (800000-300000) * 0.07 ;
```

```

P = S - T ;
cout<<"\nmizane maliat : "<<T ;
cout<<"\ndaryaftie khales : "<<P ;
}
getch( ) ;
}

```

۲۶) الگوریتمی بنویسید که تعداد پالس های تلفن مشتریان را همراه آن ها خوانده و هزینه تلفن آن ها را به شرح زیر محاسبه نماید :

از ۱ تا ۱۰۰۰ پالس ، هزینه هر پالس ۱۰ تومان ، از ۱۰۰۱ پالس به بالا ، هزینه پالس های مازاد ۱۲ تومان توضیح : ابتدا تعداد مشتریان را از ورودی گرفته و در n قرار می دهیم. حلقه تکراری نیاز که n بار تکرار شده و در هر بار نام مشتری را گرفته و در name قرار داده و تعداد پالس مصرفی اش را نیز گرفته و در P قرار دهد. اگر تعداد پالس ها کمتر مساوی ۱۰۰۰ بود هزینه تلفن را از رابطه $P*10$ محاسبه کرده و در M قرار دهد. و اگر تعدد پالس ها بیش از ۱۰۰۰ بود ۱۰۰۰ پالس ۱۰ تومانی را به علاوه P-1000 پالس ۱۲ تومانی محاسبه کرده و در M قرار دهد. در انتها نیز نام و هزینه را یعنی name و M را چاپ نماید.

- ۱- شروع
- ۲- متغیرهای n و i و P و M را از نوع عدد صحیح تعریف کن
- ۳- متغیر name را از نوع رشته ای تعریف کن
- ۴- مقدار n را از ورودی دریافت کن
- ۵- مقدار i را برابر یک قرار بده
- ۶- تا زمانی که $i \leq n$ است مراحل ۷ تا ۱۲ را تکرار کن
- ۷- نام مشتری را خوانده و در name قرار بده
- ۸- تعداد پالس مصرفی را خوانده و در P قرار بده
- ۹- اگر $P \leq 1000$ بود آنگاه $P*1000$ را در M قرار بده
- ۱۰- اگر $P > 1000$ بود آنگاه $12*(P-1000)+10*1000$ را در M قرار بده
- ۱۱- مقدار M را به عنوان هزینه تلفن چاپ کن
- ۱۲- مقدار i را برابر $i+1$ قرار بده
- ۱۳- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int n , i , P ;
    char name [20] ;
    cout<<"Tedade moshtari ra vared konid : " ;
    cin>>n ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\nesme moshtari : " ;
        cin>>name ;
        cout<<"\ntedade palse masrafi : " ;
        cin>>P ;
        if (P<=1000)
            M = 1000 * 10 ;
        else

```

```

        M = 1000 * 10 + (P-1000) * 12 ;
        cout<<"\nhazineye telephone moshtari : "<<M ;
    }
    getch( ) ;
}

```

۲۷) الگوریتمی بنویسید که اطلاعات مربوط به دانشجویان که شامل نام ، نمره کتبی ، نمره عملی و نمره تکالیف کلاسی است را از ۱۰۰ نمره خوانده و نمره نهایی را از ۲۰ نمره به کمک نسبت های زیر محاسبه نماید :

تکالیف کلاسی ۲۰٪ ، نمره عملی ۳۰٪ و نمره کتبی ۵۰٪ نمره را تشکیل می دهند.

توضیح : ابتدا n را به عنوان تعداد دانشجو گرفته و حلقه تکراری داریم که n بار تکرار می شود و در هر بار اسم دانشجو را گرفته و در name قرار می دهد و نمرات تکالیف کلاسی ، عملی و کتبی را نیز از ۱۰۰ نمره گرفته و به ترتیب در متغیرهای T و M و F قرار می دهد. نمره نهایی را ابتدا از ۱۰۰ نمره و با رابطه $T*0.20 + M*0.30 + F*0.50$ محاسبه کرده و در G قرار می دهد و برای تبدیل آن به ۲۰ نمره G را در 0.20 ضرب می کند.

- ۱- شروع
- ۲- متغیرهای n و i و F و M و T را از نوع عدد صحیح تعریف کن
- ۳- متغیر G را از نوع عدد اعشاری تعریف کن
- ۴- متغیر name را از نوع رشته ای تعریف کن
- ۵- مقدار n را از ورودی بخوان
- ۶- مقدار i را برابر یک قرار بده
- ۷- تا زمانی که $i \leq n$ است مراحل ۸ تا ۱۵ را تکرار کن
- ۸- نام دانشجو را خوانده و در name قرار بده
- ۹- نمره تکالیف کلاسی را خوانده و در T قرار بده
- ۱۰- نمره عملی را خوانده و در M قرار بده
- ۱۱- نمره کتبی را خوانده و در F قرار بده
- ۱۲- مقدار $T*0.20 + M*0.30 + F*0.50$ را در G قرار بده
- ۱۳- برای تبدیل G به ۲۰ نمره G را در 0.20 ضرب کن و در G قرار بده
- ۱۴- مقدار name و G را به عنوان نام دانشجو و نمره نهایی چاپ کن
- ۱۵- مقدار i را یک واحد افزایش بده
- ۱۶- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int n , i , T , M , F ;
    float G ;
    char name [20] ;
    cout<<"Tedade daneshjoo ra vared konid : " ;
    cin>>n ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\nesme daneshjoo : " ;
        cin>>name ;
        cout<<"\nomreya classi : " ;
    }
}

```

```

cin>>T ;
cout<<"\nnomreya amali : " ;
cin>>M ;
cout<<"\nnomreya katbi : " ;
cin>>F ;
G = T*0.20 + M*0.30 + F*0.50 ;
G = G * 0.20 ;
cout<<"\nesme daneshjoo : "<<name ;
cout<<"\nnomreya nahaae : "<<G ;
}
getch( ) ;
}

```

۲۸) الگوریتمی بنویسید که میزان یارانه پرداختی به تعدادی خانوار را از طریق دریافت تعداد نفرات خانواده محاسبه کرده و چاپ نماید. یارانه پرداختی به هر نفر معادل ۴۵۵۰۰ تومان می باشد.

توضیح : ابتدا مقدار n را به عنوان تعداد خانوارهایی که باید یارانه شان محاسبه شود از ورودی دریافت می کنیم. سپس در یک حلقه تکرار از ۱ تا n به ازای هر خانواده، تعداد نفرات را گرفته و در m قرار می دهیم و هزینه یارانه خانواده را از رابطه $m*45500$ محاسبه کرده و در S قرار می دهیم. در انتها مقدار S یعنی سهم یارانه هر خانواده را چاپ می نماییم.

۱- شروع

۲- متغیرهای n و m و i و S را از نوع عدد صحیح و متغیر $name$ را از نوع رشته ای تعریف کن

۳- مقدار n را از ورودی بگیر

۴- مقدار i را برابر یک قرار بده

۵- تا زمانی که $i \leq n$ است مراحل ۶ تا ۹ را تکرار کن

۶- تعداد نفرات خانواده را گرفته و در m قرار بده

۷- مقدار $m*45500$ را در S قرار بده

۸- مقدار S را چاپ کن

۹- به مقدار i یک واحد اضافه کن

۱۰- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main( ) {
    int n , m , i ;
    long int S ;
    char name [20] ;
    cout<<"Tedade khanevar ra vared konid : " ;
    cin>>n ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\ntedade nafarate in khanevade : " ;
        cin>>m ;
        S = m*45500 ;
        cout<<"\nmizane yaraneye in khanevade : "<<S ;
    }
    getch( ) ;
}

```

۲۹) الگوریتمی بنویسید که n عدد را از ورودی خوانده و قدر مطلق آن ها را چاپ نماید.

توضیح : ابتدا n را به عنوان تعداد اعداد ورودی می گیریم. سپس در یک حلقه تکرار که n بار تکرار می شود در هر بار یک عدد را گرفته و با یک شرط قدر مطلق آن را محاسبه و چاپ می نماییم. به طوری که اگر عدد بزرگتر از صفر باشد قدر مطلق آن ، خود عدد می باشد و اگر عدد کوچکتر از صفر باشد ، قدر مطلق آن ، معکوس عدد می باشد. مثالی از قدر مطلق : $|5| = 5$ و $|-5| = 5$

- ۱- شروع
- ۲- متغیرهای n و a و i را از نوع عدد صحیح تعریف کن
- ۳- مقدار n را از ورودی بگیر
- ۴- مقدار i را برابر یک قرار بده
- ۵- تا زمانی که $i \leq n$ است مراحل ۶ تا ۹ را تکرار کن
- ۶- مقدار a را از ورودی بگیر
- ۷- اگر $a > 0$ بود مقدار a را چاپ کن
- ۸- اگر $a < 0$ بود مقدار $-a$ را چاپ کن
- ۹- به مقدار i یک واحد اضافه کن
- ۱۰- پایان

```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int n , i , a ;
    cout<<"Tedade adad ra vared konid : " ;
    cin>>n ;
    for ( i=1 ; i<=n ; i++ )
    {
        cout<<"\nadade voroodi : " ;
        cin>>a ;
        if (a > 0)
            cout<<"\nghadre motlaghe adad : "<<a ;
        if (a < 0)
            cout<<"\nghadre motlaghe adad : "<<-a ;
    }
    getch( ) ;
}
```

۳۰) الگوریتمی بنویسید که درجه حرارت را برحسب سانتیگراد خوانده و به فارنهایت تبدیل نماید.

توضیح : ابتدا درجه را برحسب سانتیگراد خوانده و در C قرار می دهیم. سپس از رابطه $F = 1.8 * C + 32$ درجه را برحسب فارنهایت بدست می آوریم.

- ۱- شروع
- ۲- متغیرهای C و F را از نوع عدد اعشاری تعریف کن
- ۳- مقدار C را از ورودی بگیر
- ۴- مقدار $1.8 * C + 32$ را در F قرار بده
- ۵- مقدار F را چاپ کن
- ۶- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    int C , F ;
    cout<<"daraje ra bar hasbe cantigrad vared konid : " ;
    cin>>C ;
    F = 1.8 * C + 32 ;
    cout<<"\ndaraje bar hasbe farenhait : "<<F ;
    getch( ) ;
}

```

۳۱) الگوریتمی بنویسید که شعاع دایره ای را از ورودی گرفته و محیط و مساحت آن را محاسبه و چاپ نماید.
توضیح: مقدار r را به عنوان شعاع دایره از ورودی خوانده و مقدار $2*3.14*r$ را به عنوان محیط در P قرار داده و مقدار $3.14*r*r$ را به عنوان مساحت در S قرار می دهیم.

- ۱- شروع
- ۲- متغیرهای r و P و S را از نوع عدد اعشاری تعریف کن
- ۳- مقدار r را به عنوان شعاع از ورودی بگیر
- ۴- مقدار $2*3.14*r$ را در P قرار بده
- ۵- مقدار $3.14*r*r$ را در S قرار بده
- ۶- مقدار P را به عنوان محیط دایره و مقدار S را به عنوان مساحت دایره چاپ کن
- ۷- پایان

```

#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    float r , P , S ;
    cout<<"Shoae dayere ra vared konid : " ;
    cin>>r ;
    P = 2*3.14*r ;
    S = 3.14*r*r ;
    cout<<"\nmohite dayere : "<<P ;
    cout<<"\nmasahate dayere : "<<S ;
    getch( ) ;
}

```

۳۲) الگوریتمی بنویسید که ارتفاع و قاعده مثلثی را گرفته و مساحت آن را محاسبه و چاپ نماید.
توضیح: این برنامه باید H را به عنوان ارتفاع و B را به عنوان قاعده از ورودی گرفته و مساحت مثلث را از رابطه $H*B/2$ محاسبه و در S قرار داده و چاپ نماید.

- ۱- شروع
- ۲- متغیرهای H و B و S را از نوع عدد اعشاری تعریف کن
- ۳- متغیرهای H و B را از ورودی بگیر
- ۴- مقدار $H*B/2$ را به عنوان مساحت در S قرار بده
- ۵- مقدار S را به عنوان مساحت مثلث چاپ کن
- ۶- پایان


```
#include <iostream.h>
#include <stdio.h>
#include <conio.h>
main() {
    float S , B , H ;
    cout<<"ertefae mosallas ra vared konid : " ;
    cin>>H ;
    cout<<"ghaedeye mosallas ra vared konid : " ;
    cin>>B ;
    S = H*B/2 ;
    cout<<"\nmasahate dayere : "<<S ;
    getch() ;
}
```