



# درسنامه ویژه میانترم

## مبانی برنامه نویسی

مباحث:

- عملگرها و اولویت‌ها
  - Case Sensitivity
  - Comments
  - کلمات کلیدی در برنامه نویسی

```
#include<iostream>

using namespace std;

int main()
{
    return 0;
}
```

- حل معادله درجه اول و دوم
- معرفی انواع متغیرها و دامنه آن‌ها

- **int** - stores integers (whole numbers), without decimals, such as 123 or -123

اعداد صحیحی که مثبت و منفی آنها در نظر گرفته می‌شود.

عدد صحیحی بدون علامت -> **Unsign int**

- **double** - stores floating point numbers, with decimals, such as 19.99 or -19.99
- **char** - stores single characters, such as 'a' or 'B'. Char values are surrounded by single quotes
- **string** - stores text, such as "Hello World". String values are surrounded by double quotes
- **bool** - stores values with two states: true or false
- **float**

[https://www.tutorialspoint.com/cplusplus/cpp\\_data\\_types.htm](https://www.tutorialspoint.com/cplusplus/cpp_data_types.htm)

- Initialization ○
- Casting ○
- نوشتن محاسبات کامپیوتری و اولویت عملگرها ○

$$\frac{\frac{a+b}{2} - cd}{\frac{-b+b-4ac}{2a}} \quad \blacksquare$$

$$\frac{a-b}{\frac{2x^2}{-y^2}} \quad \blacksquare$$

- اهمیت پرانتز و آکولاد گذاری ○
- تفاوت / و ./ ○
- تفاوت / در انواع عدد صحیح و اعشاری ○
- معرفی ++ و -- و نکات مربوطه (قبل یا بعد از متغیر آمدن) ○
- #define (مشابه find & replace) ○
- خلاصه نویسی عبارات محاسباتی x+=y ○
- مربع مجموع سه عدد ○
- Ternary Operator ○
- x=a>b?a:b    ■
- a?b:c?d:e?f:g?h:i    ■
- نکات مربوط به cout ○
- موارد مربوط به \n و \t    ■
- نوشتن با نماد علمی 1.23e4 ○
- توضیح کامل iomanip و نحوه استفاده از توابع آن در کنترل فرمت خروجی ○

```
int x;
double y;
x=10;
cout<<setw(10)<<setfill('*')<<x<<endl;
x=010; //octal
cout<<setw(10)<<x<<endl;
x=0x10; //hexadecimal
cout<<x<<endl;
x=2.345678e3; //scientific
cout<<scientific<<x<<endl;
y=2.345678e3; //scientific
cout<<setfill(' ');
cout<<fixed<<setprecision(2);
cout<<setw(10)<<y<<endl;
cout<<setprecision(4);
cout<<y<<endl;
cout.unsetf(ios::floatfield); //clearing the floating point flag
cout<<setprecision(5);
cout<<y<<endl;
**** input and output in other bases ****
int a,b,c;
cin>>oct>>a>>hex>>b>>dec>>c;
cout<<"a="<<a<<"\nb="<<b<<"\nc="<<c<<endl;
cout<<hex<<32;
```

○ تفاوت = و ==

• توابع برنامه نویسی پیمانه‌ای

○ توابع عادی مانند ... - int - void

○ توابع فرخوانی با مقدار و ارجاع مانند swap(&a,&b)

○ توابع بازگشتی

○ Scope متغیرها: local, global و استاتیک

• شرط

○ ساختار کلی if-else

○ معرفی بلوک دستورات {}

○ انواع عملگرهای مقایسه‌ای و توجه به ترتیب آن‌ها <, <=, >, >=, !=

○ ترکیب شروط با ! - || - && (پرانتر گذاری صحیح بسیار مهم است)

true && true = true

true && false = false

false && true = false

false && false = false

true || true = true

true || false = true

false || true = true

false || false = false

○ معرفی switch و قابلیت جایگزینی آن با if

○ معرفی while و for

▪ محاسبه میانگین n عدد

▪ گام کاهشی حلقه

▪ گرفتن n و سپس n عدد و این که در بین آن‌ها مضرب ۱۷ وجود داشته یا نه؟ و اگر بله اولین عدد

که این خاصیت را داشته کدام بوده است؟ Flag‌های اضافی در شرط if برای ادامه ندادن پس از یافتن

اولین عدد

▪ گرفتن یک عدد و یافتن اول یا مرکب بودن آن: بحث درمورد کران for

▪ گرفتن n و m چاپ n عدد تصادفی بین 1 تا m : آشنایی با srand(time(0) و time.h و

rand()%(m+1)

▪ چاپ یک سری عدد تصادفی هر ۵۰۰ میلی ثانیه تا زمانی که کلیدی زده شود: آشنایی با delay و

kbhit

• حلقه‌ها

○ محاسبات با استفاده از حلقه‌ها

▪ محاسبه سری حاصل جمع 1...n و مقایسه با فرمول گوس

- محاسبه سری  $\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$  تا  $n$  جمله: انتخاب نوع متغیر درست
- محاسبه سری  $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots$  تا  $n$  جمله و چاپ ۴ برابر آن و مقایسه نزدیکی به عدد

بی

- محاسبه سری  $\arctan x = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$  تا  $n$  جمله
- محاسبه سری  $e = \frac{1}{0!} + \frac{1}{1!} + \frac{1}{2!} + \dots$  تا  $n$  جمله (توجه به عدم لزوم تکرار محاسبات)
- محاسبه سری  $\sin x = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$  تا  $n$  جمله
- استفاده از **while** تا محقق شدن یک شرط: محاسبه سری  $\frac{1}{1} + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$  تا زمانی که حاصل بزرگتر از  $x$  بشود ( $x$  ورودی) و چاپ تعداد جملات لازم و حاصل کلی سری
- محاسبه سری‌های بازگشتی: فیبوناچی تا جمله  $n$
- گرفتن یک عدد صحیح و تجزیه ارقام آن
- گرفتن یک عدد صحیح و تجزیه آن
- گرفتن  $n$  و چاپ تمام مقسوم و علیه‌های آن
- محاسبه ب.م.م
- محاسبه ک.م.م
- چاپ اعداد اول از 1 تا  $n$  (توجه به دوبرابر شدن سرعت با شروع از ۳ و پله‌های دوتایی)
- تجزیه عدد و نمایش به صورت  $p_1^{a_1} * p_2^{a_2} * \dots$  (عدم نیاز به چک کردن اول بودن عدد جهت افزایش سرعت

- گرفتن عددی و چک کردن اینکه آیا آرمسترانگ است یا خیر؟  $\overline{abc} = a^3 + b^3 + c^3$
- چاپ اعداد آرمسترانگ ۱ تا ۱۰۰۰
- مسائل مربوط به حداقل تعداد سکه‌ها
- حلقه‌های تو در تو
- حلقه‌های مربوط به ترسیمات در محیط کنسول



- چاپ مثلث خیام – پاسکال

1 2 3 4	1 2 3 4	1 2 3 4	1 5 9 13
2 4 6 8	5 6 7 8	2 3 4 5	2 6 10 14
3 6 9 12	9 10 11 12	3 4 5 6	3 7 11 15
4 8 12 16	13 14 15 16	4 5 6 7	4 8 12 16

- آرایه‌ها و رشته‌ها

○ آرایه به عنوان متغیر اندیس دار که از صفر شروع می‌شود.

- مرتب سازی حبابی (Bubble sort)
- امکان استفاده از آرایه به عنوان پارامتر (فرخوانی با ارجاع)
- نوشتن تابع مرتب سازی
- امکان حفاظت از پارامترهای ارسالی برای عدم تغییر

```
void myTest(const int a[])
{
    int b = a[0];
    a[0]++;
}
```

- روش های مقداردهی اولیه به آرایه

Method 1: Garbage allocation

```
int a[10];
```

Method 2: Specify value

```
double a[] = {1.7, 2.0, 3.5};
```

Method 3: Specify value and size

```
int a[5] = {5, 4, 3, 2, 1}; //throws compiler error if the numbers exceed
from the size of the array - if the numbers are less than the size of
the array, the remaining cells will be set to 0
```

```
int a[] = {} → results in sizeof(a) == 0
```

```
int x[] = { 1, 2, 3 };
```

```
sizeof(x) → 12 == (3 * 4) == (3 * sizeof(int))
```

- مرتب سازی انتخابی و مقایسه با bubble sort

○ آرایه دو و چندبعدی

○ جمع دو ماتریس

○ نقطه زین اسبی

○ Big Numbers

○ وزیرهای تهدیدکننده



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:

آزمون میان ترم

درس: میانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۱ از ۶

**تذکر بسیار مهم:** محل درج پاسخ سوالات، در برگه‌ی سوالات و به اندازه‌ی مشخص شده است! پس در مدیریت صفحات و سایز فونت نوشتاری (!) کوشا بوده و حتی الامکان با مداد و به صورت دو ستونی پاسخ سوالات را بدهید تا جایی برای جبران اشتباهات وجود داشته باشد! برگه‌هایی که بیش از حد بد خط، خط خورده یا لاک اندود (!) باشند تصحیح نخواهند شد! پاسخ‌گویی با قلم قرمز ممنوع است! □

**بخش اول - درک کد (۴ نمره)** در بخش زیر، ۴ قطعه کد وجود دارد و در هر مورد باید بگویید کد نوشته شده چه کاری انجام می‌دهد؟ یک مثال نیز باید برای کارکرد آن بیاورید. به عنوان مثال به یک کد و یک پاسخ درست و یک پاسخ نادرست دقت کنید. (البته ممکن است کد نوشته شده برنامه‌ی کامل نبوده و بخشی از یک برنامه باشد که در آن صورت باید بگویید آن بخش از کد چه کاری انجام می‌دهد):

کد	پاسخ
<pre>int n,i,s=0; cin&gt;&gt;n; for (i=1;i&lt;=n/2;i++)     if (n%i==0)         s+=i; cout&lt;&lt;s;</pre>	<p>نمونه پاسخ درست:</p> <p>این برنامه یک عدد صحیح از کاربر گرفته و مجموع مقوم علیه‌های آن عدد غیر از خودش را نمایش می‌دهد. مثلاً اگر <math>n=12</math> باشد عدد 16 که مجموع <math>1+2+3+4+6</math> است در خروجی چاپ خواهد شد.</p> <p>نمونه پاسخ نادرست:</p> <p>این برنامه یک عدد صحیح از کاربر گرفته و سپس یک حلقه تشکیل داده و آنرا از یک <math>n/2</math> مقدار دهی نموده و به ازای هر <math>n</math> بررسی می‌کند که آیا <math>n</math> بر <math>i</math> بخش پذیر است یا خیر و اگر بود آن را در <math>s</math> جمع می‌کند و در آخر <math>s</math> را چاپ می‌کند. مثلاً اگر <math>n=12</math> باشد عدد 16 که مجموع <math>1+2+3+4+6</math> است در خروجی چاپ خواهد شد.</p>

سؤال ۱-

کد	پاسخ
<pre>int n, a, b=0; cin&gt;&gt;n; a=n%1000; while (a&gt;0) {     b*=10;     b+=a%10;     a/=10; } n/=1000; n*=1000; n+=b; cout&lt;&lt;n&lt;&lt;endl;</pre>	

سؤال ۲-



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۲ از ۶

کد	پاسخ
<pre>int i, a, b, temp, flag=0; cin&gt;&gt;a&gt;&gt;b; if(b&lt;a){     temp=a;     a=b;     b=temp; } i=a; while(flag==0){     if(i%a==0)         if(i%b==0){             flag=1;             cout&lt;&lt;i&lt;&lt;endl;         }     i++; }</pre>	
سؤال ۳ -	
کد	پاسخ
<pre>int a=1, b=1, c=1, count, n; cin&gt;&gt;n; if(n==1)     count=0; else     count=1; while(c&lt;n) {     c=b+a;     b=a;     a=c;     count++; } cout&lt;&lt;count&lt;&lt;endl;</pre>	
سؤال ۴ -	
کد	پاسخ
<pre>int n, temp, count=0; cin&gt;&gt;n; for(int i=1; i&lt;n; i++) {     temp=0;     if(i%2==0)         temp++;     if(i%3==0)         temp++;     if(i%5==0)         temp++;     if(i%7==0)         temp++;     if(temp==2)         count++; } cout&lt;&lt;count&lt;&lt;endl;</pre>	



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۳ از ۶

تذکر بسیار مهم تر (!): از این به بعد، سؤالاتی خواهید دهد که دوتا دوتا به صورت ا و آ پریم (مثلاً ۵ و ۵ پریم) هستند. از بین هر کدام از این دوتایی ها ( که نمرات و طبعاً درجه سختی متفاوتی دارند) باید دقیقاً یک سؤال را انتخاب کرده و پاسخ دهید. پاسخ به هر دو سؤال = نمره صفر

بخش دوم - خروجی کد:

سؤال ۵- (۱/۵ نمره) خروجی کد زیر را به ازای  $n=7$  و  $a=5$  بنویسید.

کد	پاسخ
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     int n, a, b=1;     float sum=1;     cin&gt;&gt;n&gt;&gt;a;     for(int i=1; i&lt;n; i++)     {         b*=a;         sum+=b/3;     }     cout&lt;&lt;sum&lt;&lt;endl;     return 0; }</pre>	

سؤال ۵ پریم- (۲ نمره) خروجی کد زیر را به ازای  $n=121$  بنویسید.

کد	پاسخ
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     float a, b=1;     int n, sum=0;     cin&gt;&gt;n;     for(int i=1; i&lt;=n; i++)     {         b*=i;         a=i*i*i;         sum+=a/b;     }     cout&lt;&lt;sum&lt;&lt;endl;     return 0; }</pre>	





به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۴ از ۶

بخش سوم - خطایابی کد:

\*  
\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

سؤال ۶- (۳ نمره) هدف برنامه‌ی زیر این بوده است که شکل روبرو را در محیط متنی چاپ کند. پنج نوع خطا در این برنامه موجود است. آنها را پیدا کرده و توضیح دهید چگونه این خطاها تصحیح می‌شوند.

کد

پاسخ

```
include<iostream>
using namespace std;
int main
{
    int n=5, i;
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=n-i; j++)
            cout<<" ";
        for(j=1; j<=2*i-1; j++)
            cout<<"*";
    }
    return;
}
```

\* \* \*  
\* \* \*  
\*\*\*  
\*  
\*\*\*  
\* \* \*  
\* \* \*

سؤال ۶ پریم - (۴ نمره) هدف برنامه‌ی زیر این بوده است که شکل زیر را در محیط متنی چاپ کند. شش نوع خطا در این برنامه موجود است. آنها را پیدا کرده و توضیح دهید چگونه این خطاها تصحیح می‌شوند.

کد

پاسخ

```
include<iostream>
using namespace std;
int main
{
    int n=4, i;
    for(i=1; i<=2*n-1; i++)
    {
        for(j=1; j<=2*n-1; j++)
            if((j==i) && (j==2*n-i) && (j==n))
                cout<<"*";
            else
                cout<<" ";
    }
    return;
}
```



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۵ از ۶

بخش چهارم - برنامه نویسی:

سؤال ۷- (۳ نمره) برنامه ای بنویسید که ۱۰۰۰۰ عدد تصادفی بین ۰ تا ۹۹۹۹۹ تولید کرده و نسبت تعداد اعداد اول تولید شده به کل اعداد را چاپ کند. مثلاً اگر می خواستیم ۵ عدد تصادفی ایجاد کنیم، خروجی به شکل زیر بدست می آمد.

خروجی	اگر ۵ عدد تصادفی تولید شود
0.4	5 9 102 19 48
0.2	3 0 4 18 10

سؤال ۷ پریم- (۴ نمره) برنامه ای بنویسید که ابتدا عدد  $n$  و سپس  $n$  عدد دیگر از ورودی بگیرد و عددی که بیشترین تکرار بین این  $n$  عدد ورودی را دارد، در خروجی چاپ کند. فرض کنید همواره عددی وجود دارد که بیش از  $n/2$  بار تکرار شده است. به طور مثال اگر  $n$  برابر ۱۰ بود، حتماً عددی وجود دارد که بیش از ۵ بار تکرار شده است.

خروجی نمونه	ورودی نمونه
10	4 10 10 2 10
2	9 8 2 800 2 224 2 9 2 2



به نام خدا

دانشکده  
مهندسی  
برق

آزمون میان ترم

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۶ از ۶

سوال ۸- (۴ نمره) برنامه‌ای بنویسید که عدد  $n$  (  $n$  حداکثر 10000000000 ) را از ورودی گرفته و تعداد صفرهای سمت راست عدد  $n!$  را در خروجی چاپ کند. (دقت کنید که  $10000000000!$  بسیار بزرگتر از آن است که بتوانید آن را در یک متغیر ذخیره کنید).

خروجی نمونه	ورودی نمونه
6	26
1	5

سوال ۸- (۵ نمره) برنامه‌ای بنویسید که عدد  $n$  را از ورودی گرفته و معادل مبنای ۲ آن را در خروجی چاپ کند.

خروجی نمونه	ورودی نمونه
11010	26
101	5

به نام خدا



دانشکده مهندسی برق

مبانی برنامه نویسی (۲۵۷۶۸)

نیم سال اول ۱۴۰۰-۱۳۹۹

آزمون پایانی

دکتر وثوقی وحدت - دکتر آراسته

مجموع نمرات برابر ۵۰۰ است که ۱۰۰ نمره آن (معادل ۱ نمره نهایی) امتیازی است

۶ بهمن ۹۹

۱. عدد تام (۳۰ نمره)



در این سؤال باید یک عدد صحیح از ورودی گرفته، و سپس مشخص کنید که آیا عدد تام است یا خیر؟ عدد تام عددی است که برابر مجموع مقسوم علیه های کوچک تر از خودش باشد. برای مثال 6 یک عدد تام است زیرا  $6=1+2+3$ .

ورودی:

عدد  $n$  ( $1 \leq n \leq 100000000$ )

خروجی:

در تنها خط خروجی اگر عدد تام است چاپ کنید YES و در غیر این صورت NO را چاپ کنید (با حروف بزرگ).

مثال از ورودی و خروجی صحیح:

input	output
6	YES
12	NO
28	YES

## ۲. ترانهاده (۴۰ نمره)



می‌دانید یکی از اعمالی که می‌توان بر روی یک ماتریس انجام داد، ترانهاده کردن آن است که در آن سطر و ستون‌های ماتریس با هم جابجا می‌شوند. مثلاً ترا نهادهٔ ماتریس  $A$  را که با  $A^T$  مشخص می‌شود در زیر می‌بینید:

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \Rightarrow A^T = \begin{bmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{bmatrix}$$

در این سؤال نیز باید با گرفتن سطر و ستون و سپس درایه‌های ماتریس، ترانهادهٔ آن را در خروجی چاپ کنید.

### ورودی:

در سطر اول خروجی دو عدد  $m, n$  ( $1 \leq m, n \leq 500$ ) که به ترتیب آمده‌اند و  $m$  تعداد سطرها و  $n$  تعداد ستون‌ها را مشخص می‌کند. در  $m$  سطر بعدی، در هر سطر  $n$  عدد آمده.

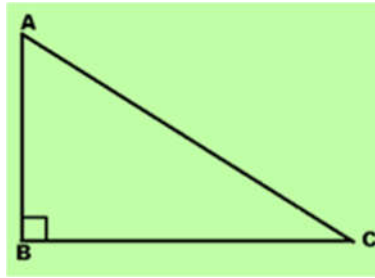
### خروجی:

در خروجی باید ترانهادهٔ ماتریس ورودی را در  $m$  سطر چاپ کنید (هر عدد با عدد بعدی یک فاصله دارد)

مثال از ورودی و خروجی صحیح:

input	output
2 3 1 2 3 4 5 6	1 4 2 5 3 6
4 1 12 10 8 20	12 10 8 20
3 3 1 2 3 4 5 6 7 8 9	1 4 7 2 5 8 3 6 9

### ۳. قائم الزاویه‌ها (۱۰۰ نمره)



در این سؤال باید برنامه‌ای بنویسید که ابتدا  $m$  و  $n$  و سپس یک آرایه‌ی دوبعدی  $m \times n$  از اعداد 0 و 1 از کاربر بگیرد و با این فرض که هر کدام از 1ها یک رأس را در صفحه‌ی  $m \times n$  تعریف می‌کنند، تعداد مثلث‌های قائم‌الزاویه که اضلاع قائمه آن (غیر از وتر) موازی محورهای  $x$  یا  $y$  باشد را چاپ کند.

#### ورودی:

در سطر اول خروجی دو عدد  $m, n$  ( $2 \leq m, n \leq 500$ ) که به ترتیب آمده‌اند و  $m$  تعداد سطرها و  $n$  تعداد ستون‌ها را مشخص می‌کند. در  $m$  سطر بعدی، در هر سطر  $n$  عدد 0,1 آمده.

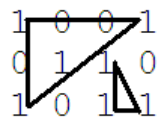
**راهنمایی:** محاسبه‌ی تعداد مثلث‌ها بسیار سخت و زمان‌بر است. فکر کنید چطور با دو آرایه یک بعدی، مسأله را حل کنید (نظیر میان‌بری که در محاسبه‌ی نقاط زین اسبی رفتیم).

#### خروجی:

تعداد مثلث‌های قائم‌الزاویه که اضلاع موازی محورهای  $x$  یا  $y$  باشد.

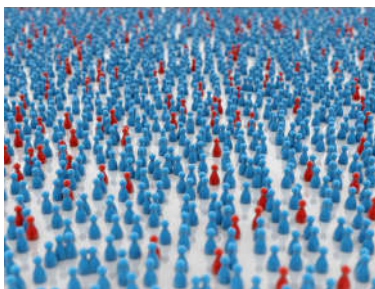
#### مثال از ورودی و خروجی صحیح:

input	output
3 4 1 0 0 1 0 1 1 0 1 0 1 1	9
2 2 1 0 1 1	1

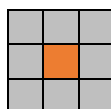


به عنوان مثال دو تا از مثلث‌های مورد نظر در مثال اول به این شکل است:

#### ۴. انتشار (۱۰۰ نمره)



می‌خواهیم ببینیم ویروس کرونا، چطور پخش می‌شود. برای این کار، یک آرایه دو بعدی در نظر می‌گیریم که هر کدام از درایه‌های آن 0 یا 1 هستند که 1 یعنی بیمار و 0 یعنی سالم. سپس در هر مرحله، برای این که بفهمیم وضعیت در مرحله بعد چه خواهد شد، برای هر خانه، به همه همسایه‌های آن نگاه می‌کنیم و می‌دانیم برای هر خانه حداکثر ۸ همسایه وجود دارد (مانند خانه‌های خاکستری شکل زیر).



سپس وضعیت هر خانه را با توجه به اکثریت همسایه‌ها تعیین می‌کنیم. مثلاً اگر یک خانه ۵ همسایه 1 و ۳ همسایه 0 داشته باشد، تبدیل به 1 می‌شود. اگر تعداد همسایه‌ها مساوی باشند، وضعیت قبلی حفظ می‌شود. دقت کنید وضعیت بعدی هر خانه در آرایه قبلی جایگزین نمی‌شود، یعنی اگر وضعیت خانه  $[i][j]$  از 0 به 1 تغییر کرد، برای محاسبه وضعیت خانه  $[i][j+1]$ ، خانه  $[i][j]$  همچنان 0 در نظر گرفته می‌شود تا مرحله بعد که همه خانه‌ها مقدار جدید خود را بگیرند. در مرحله بعد، همه خانه‌ها مقدار جدید خود را دارند و این وضعیت می‌تواند تا هر تعداد مرحله ادامه پیدا کند. هدف ما این است بفهمیم بعد از  $k$  مرحله ماتریس ورودی چه شکلی خواهد داشت.

#### ورودی:

در سطر اول خروجی دو عدد  $n, k$  ( $1 \leq n \leq 500$  و  $1 \leq k \leq 1000$ ) آمده و در  $n$  سطر بعدی، در هر سطر  $n$  عدد 0,1 آمده.

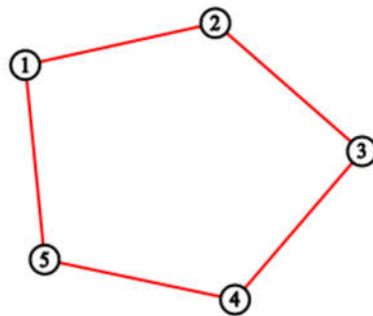
#### خروجی:

در خروجی باید  $k$ -امین مرحله از ماتریس ورودی را در  $n$  سطر چاپ کنید (هر عدد با عدد بعدی یک فاصله دارد)

input	output
3 1 1 0 1 1 1 1 0 0 0	1 1 1 0 1 0 1 1 1
2 1000 1 0 0 0	0 0 0 0
3 999 1 1 1 0 0 0 1 1 1	0 0 0 1 1 1 0 0 0

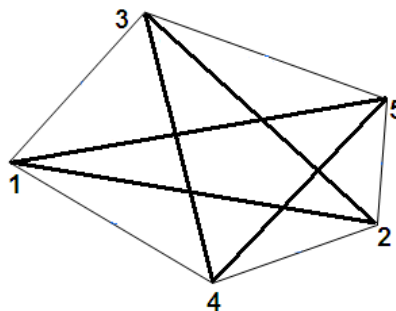


## ۵. گرافیک محدب (۱۰۰ نمره)



برنامه‌ای بنویسید که ابتدا  $n$  و سپس مختصات  $n$  نقطه را از کاربر گرفته و در نهایت وارد محیط گرافیک شده و یک  $n$ -ضلعی با این رأس ترسیم کند. اما فرض کنید کاربر ممکن است ترتیب رئوس را رعایت نکند و شما باید طوری این  $n$  رأس را به هم متصل کنید که یک چند ضلعی محدب پدید بیاید (یعنی هیچ زاویه داخلی بزرگتر یا مساوی  $180^\circ$  درجه نباشد و هیچ ضلعی، ضلع دیگر را قطع نکند). همچنین فرض می‌کنیم این رئوس طوری وارد شده‌اند که حتماً یک چند ضلعی محدب را تشکیل می‌دهند.

به عنوان مثال در شکل زیر اگر مطابق ترتیب وارد شده رئوس را به هم وصل کنیم (یعنی از ۱ تا ۵) شکل ستاره داخلی پدید می‌آید که محدب نیست، اما اگر ترتیب اتصال را ۱-۴-۲-۵-۳-۱ در نظر بگیریم، شکل محدب مورد نظر به دست می‌آید:



### ورودی:

ابتدا  $n$  که تعداد رئوس است و سپس مختصات  $n$  نقطه.

### خروجی:

ترسیم  $n$ -ضلعی محدب با رئوس داده شده در محیط گرافیک

**راهنمایی:** از نقطه اول شروع کرده و هر بار نزدیک ترین نقطه را پیدا کنید! فاصله بین دو نقطه  $i$  و  $j$  را هم که حتماً یادتان هست برابر است با  $(x[i] - x[j])^2 + (y[i] - y[j])^2$ .

**دقت کنید** در این سؤال کل فولدر پروژه گرافیک خود را zip کرده و نام آن را شماره دانشجویی خود قرار دهید. مثلاً

**99300123.zip** در غیر این صورت پاسخ شما تصحیح نخواهد شد!

## ۶. مجمع تاس‌ها! (۱۳۰ نمره)



می‌دانید اگر ۱ تاس را بریزیم، حاصل ۶ حالت دارد که به ترتیب صعودی عبارتند از:

1 2 3 4 5 6

و اگر دو تاس را بریزیم حاصل ۳۶ حالت دارد که به ترتیب صعودی عبارتند از:

11 12 13 14 15 16 21 22 23 24 25 26 31 32 33 34 35 36 41 42 43 44 45 46 51 52 53 54 55 56 61  
62 63 64 65 66

حال مثلاً اگر از بین حالات دو تاس، حالت ۸ ام را بخواهیم، ۲۲ است و اگر حالت ۳۵ ام را بخواهیم ۶۵ است. هدف این است که ابتدا  $n$ ، تعداد تاس‌ها را گرفته و سپس  $m$  که نشان‌دهنده  $m$  امین حالت از ترکیب  $n$  تاس که به ترتیب صعودی چیده شده‌اند را گرفته و حالت مورد نظر را چاپ کنیم.

### ورودی:

ابتدا  $1 \leq n \leq 8$  و سپس  $1 \leq m \leq 6^n$  می‌آید.

### خروجی:

در تنها خط خروجی باید ترکیب مورد نظر در دنبالهٔ صعودی ترکیب  $n$  تاس را بنویسید.

مثال از ورودی و خروجی صحیح:

input	output
1 4	4
2 10	24
3 7	121

به نام خدا



دانشکده مهندسی برق

مبانی برنامه نویسی (۲۵۷۶۸)

نیم سال اول ۱۴۰۰-۱۳۹۹

آزمون میان ترم

دکتر وثوقی وحدت - دکتر آراسته

۱۲ دی ۹۹

۱. آب خوردن! (۳۰ نمره)



در این سؤال باید یک عدد صحیح از ورودی گرفته، و سپس تعداد ارقام آن را در خروجی چاپ کنید.

**ورودی:**

عدد  $n$  ( $1 \leq n \leq 1000000000000000000$ )

**خروجی:**

در تنها خط خروجی باید تعداد ارقام عدد ورودی  $n$  را چاپ کنید

مثال از ورودی و خروجی صحیح:

input	output
2021	4
8	1
50000002	8

۲. باقیماندهٔ توان مند (۳۰ نمره)



تنها کاری که در این سؤال باید بکنید گرفتن سه عدد  $m, n, k$  از ورودی و محاسبه‌ی باقی‌مانده‌ی  $m^n$  بر  $k$  است. دقت دارید که اگر باقی‌مانده‌ی  $a$  بر  $b$  برابر  $c$  باشد، باقی‌مانده‌ی  $a * k$  بر  $b$  برابر  $c * k$  خواهد بود (که البته  $c * k$  خودش ممکن است بزرگتر از  $b$  باشد و باید دوباره باقی‌مانده‌ی آن را بر  $b$  حساب کرد. مثلاً باقی‌مانده‌ی ۱۸ بر ۷ برابر ۴ است، پس باقی‌مانده‌ی  $18 * 18$  بر ۷ برابر  $4 * 18$  است که چون خود  $4 * 18$  بزرگتر از ۷ است مجدداً باید باقی‌مانده‌ی آن را بر ۷ حساب کنیم که برابر ۲ می‌شود).

ورودی:

سه عدد  $m, n, k$  ( $-10000000 \leq m, n, k \leq 10000000$ ) که به ترتیب آمده‌اند

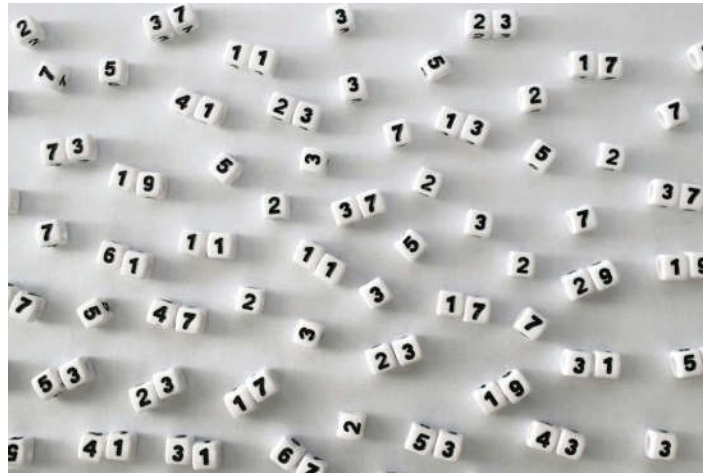
خروجی:

در تنها خط خروجی باید باقی‌مانده‌ی  $m^n$  بر  $k$  را چاپ کنید

مثال از ورودی و خروجی صحیح:

input	output
10 5 17	6
28 1 5	3
100000 500000 2	0

### ۳. عوامل اول بی توان (۸۰ نمره)



در این سؤال باید یک عدد صحیح از ورودی گرفته و کلیه عوامل اول آن را (بدون توان مربوطه) در خروجی چاپ کنید. مثلاً اگر ورودی 12 باشد خروجی 2 3 خواهد بود (هر کدام با یک فاصله)

ورودی:

عدد  $n$  ( $2 \leq n \leq 1000000000$ )

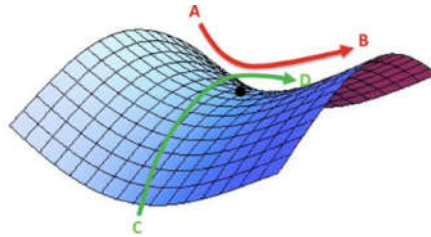
خروجی:

تک تک عوامل اول  $n$  به ترتیب صعودی که با یک فاصله از هم جدا شده‌اند (آخرین عدد فاصله‌ای بعد خود ندارد).

مثال از ورودی و خروجی صحیح:

input	output
1024	2
12	2 3
300	2 3 5

#### ۴. نقطهٔ زین اسبی (۸۰ نمره)



در یک آرایه  $n \times n$  ( $n$  سطر و  $n$  ستون) اگر یک خانه در سطر خودش بزرگترین عدد و در ستون خودش کوچکترین عدد باشد، یا برعکس در سطر خودش کوچکترین عدد و در ستون خودش بزرگترین عدد باشد به آن خانه نقطه‌ی زین/اسبی می‌گوییم.

- در آرایه‌ی زیر، عدد ۵ کوچکترین عدد ستون نخست و بزرگترین عدد سطر دوم است.

8	9	1
<u>5</u>	2	4
6	1	3

- در آرایه‌ی زیر، عدد ۱۹ بزرگترین عدد ستون چهارم و کوچکترین عدد سطر پنجم است.

41	4	64	16	9	32
6	1	3	8	5	7
8	9	92	10	47	40
11	5	13	9	12	15
77	27	33	<u>19</u>	20	36
8	9	1	17	5	11

(دقت کنید: آرایه‌ی بالا غیر از ۱۹ نقطه‌ی زین اسبی دیگری نیز دارد، آن را بیابید.)

- آرایه‌ی زیر نقطه‌ی زین اسبی ندارد.

1	2	3	4
0	8	4	7
7	10	6	11
6	13	12	-1

### ورودی

در نخستین سطر ورودی  $100 \geq n \geq 1$  و پس از آن  $n$  سطر می‌آیند که هر یک شامل  $n$  عدد صحیح است.

### خروجی

همه‌ی نقطه‌های زین اسبی آرایه باید در خروجی بیابند. اگر آرایه‌ی ورودی بیش از یک نقطه‌ی زین اسبی دارد هر نقطه را در سطری جداگانه در خروجی نمایش دهید و اگر آرایه‌ی ورودی هیچ نقطه‌ی زین اسبی ندارد در خروجی NO (با حروف بزرگ) را چاپ کنید. دقت کنید خروجی‌ها را به ترتیب سطر و ستون نمایش دهید. مثلاً اگر دو نقطهٔ زین اسبی در سطر 0 و 2 هست، ابتدا آن که در سطر 0 هست می‌آید. اگر هم دو نقطهٔ زین اسبی در یک سطر وجود دارد، ابتدا آن که شماره ستون کمتری دارد می‌آید.

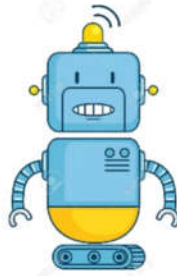
دقت کنید که اندیس آرایه‌ها در C++ از صفر آغاز می‌شود. بنابراین اندیس سطر دوم برابر 1 و اندیس ستون نخست برابر 0 خواهد بود.

مثال از ورودی و خروجی صحیح:

input	output
3 8 9 1 5 2 4 6 1 3	1 0
4 10 20 30 40 13 21 32 50 12 34 37 60 11 20 30 70	0 3 1 0
4 1 2 3 4 0 8 4 7 7 10 6 11 6 13 12 -1	NO



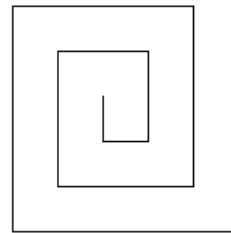
## ۵. گرافیک چپ چپ (۸۰ نمره)



هدف از این سؤال گرفتن  $n$  و سپس  $d$  و کشیدن مسیر یک ربات است که با الگوریتم چپ چپ (هر بار به سمت چپ می‌رود، یعنی حرکت پادساعت‌گرد) سعی دارد دنیای پیرامون خود را با گام‌های  $d, d, 2d, 2d, 3d, 3d, \dots$  پیمایش کند. مسیر ربات شبیه زیر است:



$n=5$



$n=10$

شروع حرکت ربات باید از وسط صفحه گرافیک باشد.

**ورودی:**

ابتدا  $n$  که تعداد گام‌هاست و سپس  $d$  که مشخص کننده طول اولین گام ربات بر حسب پیکسل است.

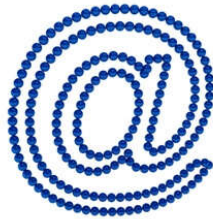
**خروجی:**

تولید شکل خواسته شده مسیر ربات بر اساس توضیحات داده شده در صفحه گرافیک.

**دقت کنید** در این سؤال کل فولدر پروژه گرافیک خود را zip کرده و نام آن را شماره دانشجویی خود قرار دهید. مثلاً

**99300123.zip** در غیر این صورت پاسخ شما تصحیح نخواهد شد!

۶. عملگر شگفت‌انگیز (@) (نمره ۱۰۰)



فرض کنید عملگر شگفت‌انگیز @ را بر روی اعداد صحیح تعریف کرده‌ایم که هر بار آن را بر روی یک عدد اعمال کنیم، آن عدد را تبدیل به حاصل جمع ارقام آن می‌کند. مثلاً @198 برابر  $1+9+8=18$  است. برنامه‌ای بنویسید که عدد X را بگیرد و تعداد دفعات لازم اعمال این عملگر بر روی X برای تبدیل آن به یک عدد یک رقمی را چاپ کند. مثلاً اگر کاربر عدد 991 را وارد کرد، خروجی 3 خواهد بود زیرا با یک بار اعمال این عملگر، 991 تبدیل به 19 می‌شود و با اعمال مجدد تبدیل به 10 و در مرتبه‌ی سوم تبدیل به 1 می‌شود.

**ورودی:**

در تنها خط ورودی عدد X وارد می‌شود.

**توجه:** دامنه‌ی قابل قبول X عبارت است از:  $0 \leq x \leq 10^{100000}$  بنابراین باید از string برای گرفتن ورودی استفاده کنید.

**خروجی:**

در تنها خط خروجی باید تعداد .

مثال از ورودی و خروجی صحیح:

input	output
991	3
6	0
1399	2

به نام خدا



دانشکده مهندسی برق

مبانی برنامه نویسی (۲۵۷۶۸)

نیمسال اول ۱۴۰۱-۱۴۰۰

آزمون میان ترم

دکتر وثوقی وحدت - دکتر آراسته

۱۸ آذر ۱۴۰۰

۱. آب خوردن! (۳۰ نمره)



در این سؤال باید یک عدد صحیح از ورودی گرفته، و سپس تعداد ارقام آن را در خروجی چاپ کنید.

ورودی:

عدد  $n$  ( $1 \leq n \leq 1000000000000000000$ )

خروجی:

در تنها خط خروجی باید تعداد ارقام عدد ورودی  $n$  را چاپ کنید

مثال از ورودی و خروجی صحیح:

input	Output
2021	4
8	1
50000002	8

۲. بخشپذیر متوالی (۱۰۰ نمره)



۲۵۲۰ کوچکترین عدد طبیعی است که بر تمام اعداد ۱ تا ۱۰ بخشپذیر است. شما باید در این سؤال یک عدد  $n$  بگیرید و محاسبه کنید کوچکترین عدد طبیعی که بر تمامی اعداد ۱ تا  $n$  بخشپذیر است کدام است.

ورودی:

عدد  $n$  ( $1 \leq n \leq 40$ ).

خروجی:

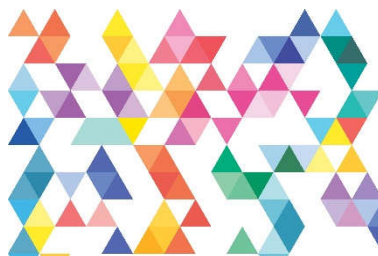
در تنها خط خروجی باید کوچکترین عدد طبیعی که بر تمامی اعداد ۱ تا  $n$  بخشپذیر است را چاپ کنید

راهنمایی: ک.م.م برابر است با ضرب دو عدد تقسیم بر ب.م.م آن دو.

مثال از ورودی و خروجی صحیح:

Input	Output
10	2520
5	60
20	232792560

### ۳. مثلثات (۱۲۰ نمره)



عدد مثلثی  $n$  ام با جمع اعداد ۱ تا  $n$  به دست می‌آید. مثلاً ۷ امین عدد مثلثی برابر است با جمع ۱ تا ۷ که ۲۸ می‌شود. بنابراین اعداد مثلثی به این صورت خواهند بود:

1, 3, 6, 10, 15, 21, 28, 36, 45, 55, ...

چنانچه مقسوم‌علیه‌های اعداد مثلثی ۱ تا ۷ را لیست کنیم:

1:1  
3:1,3  
6:1,2,3,6  
10:1,2,5,10  
15:1,3,5,15  
21:1,3,7,21  
28:1,2,4,7,14,28

متوجه می‌شویم که ۲۸ اولین عددی است که بیش از ۵ مقسوم‌علیه دارد. حال شما باید عدد  $k$  را گرفته، و مشخص کنید اولین عدد مثلثی که بیش از  $k$  مقسوم‌علیه دارد، کدام است؟

**ورودی:**

عدد  $k$  ( $1 \leq k \leq 500$ )

**خروجی:**

اولین عدد مثلثی که بیش از  $k$  مقسوم‌علیه دارد

مثال از ورودی و خروجی صحیح:

input	Output
5	28
1	3
3	6

#### ۴. جریان کاراکترها چند؟! (۱۵۰ نمره)



در این سؤال، یک سری کاراکتر وارد می‌شود تا زمانی که کاراکتر مورد نظر برابر C (C کاراکتر کوچک و نه C بزرگ) باشد. کاراکترهای دریافتی همه کاراکترهای عددی یا + و - هستند. شما باید این کاراکترها که با هم یک عبارت ریاضی می‌سازند را بگیرید و حاصل آن عبارت را محاسبه کنید. اعداد همگی در متغیر int جا می‌شوند. مثلاً ورودی میتواند  $123+23-56$  باشد که خروجی طبعاً برابر 90 خواهد بود. دقت کنید طول این دنباله را از ابتدا نمی‌دانیم و باید همینطور به گرفتن کاراکتر بعدی ادامه دهیم تا به کاراکتر C برسیم.

**استفاده از آرایه و رشته مجاز نیست!**

#### ورودی

جریانی از کاراکترها که به C ختم می‌شود.

#### خروجی

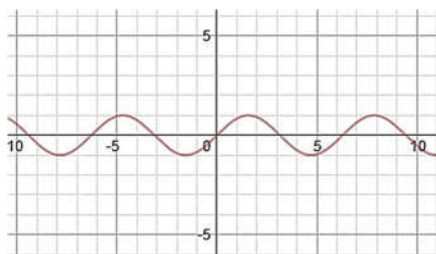
محاسبه عبارت ریاضی وارد شده حاوی جمع و تفریق.

**راهنمایی:** دقت کنید می‌توانید خیلی راحت با یک حلقه while کاراکترها را یکی یکی cin کرده و سپس عملیات مورد نظر را اجرا کنید.

مثال از ورودی و خروجی صحیح:

input	output
$123+23-56$	90
$1+1-2+2$	2
$123456-456$	123000
$-1000+100$	-900
-1	-1
110	110

## ۵. گرافیک سینوسی {امتیازی} (۷۰ نمره)



هدف از این سؤال ترسیم یک نمودار سینوسی در صفحه است. مبدأ مختصات در ابتدا در وسط صفحه است و هر بار کاربر دو بار کلیک چپ بر روی صفحه کرد، مبدأ به مختصات موس منتقل شده و نمودار مجدد کشیده می‌شود. آنچه لازم است، تنها خطوط محورهای مختصات است. ولی نوشتن اعداد و خطوط گرید (همانند شکل سؤال) نمره اضافی دارد. فرکانس موج سینوسی باید طوری باشد که حداقل ۳ و حداکثر ۶ تناوب آن دیده شود و دامنه آن حداقل ۲۰۰ و حداکثر ۴۰۰ پیکسل باشد. ترسیم باید با خط باشد و ترسیم با پیکسل مورد قبول نیست. با زدن دکمه ESC برنامه پایان می‌پذیرد.

**ورودی:**

موس و کیبورد.

**خروجی:**

تولید شکل خواسته شده سینوسی بر اساس توضیحات داده شده در صفحه گرافیک.

**دقت کنید** در این سؤال کل فولدر پروژه گرافیک خود را zip کرده و نام آن را شماره دانشجویی خود قرار دهید. مثلاً

**400300123.zip** در غیر این صورت پاسخ شما تصحیح نخواهد شد!





به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:

آزمون میان ترم

درس: میانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۱ از ۶

**تذکر بسیار مهم:** محل درج پاسخ سوالات، در برگه‌ی سوالات و به اندازه‌ی مشخص شده است! پس در مدیریت صفحات و سایز فونت نوشتاری (!) کوشا بوده و حتی الامکان با مداد و به صورت دو ستونی پاسخ سوالات را بدهید تا جایی برای جبران اشتباهات وجود داشته باشد! برگه‌هایی که بیش از حد بد خط، خط خورده یا لاک اندود (!) باشند تصحیح نخواهند شد! پاسخ‌گویی با قلم قرمز ممنوع است! □

**بخش اول - درک کد (۴ نمره)** در بخش زیر، ۴ قطعه کد وجود دارد و در هر مورد باید بگویید کد نوشته شده چه کاری انجام می‌دهد؟ یک مثال نیز باید برای کارکرد آن بیاورید. به عنوان مثال به یک کد و یک پاسخ درست و یک پاسخ نادرست دقت کنید. (البته ممکن است کد نوشته شده برنامه‌ی کامل نبوده و بخشی از یک برنامه باشد که در آن صورت باید بگویید آن بخش از کد چه کاری انجام می‌دهد):

کد	پاسخ
<pre>int n,i,s=0; cin&gt;&gt;n; for (i=1;i&lt;=n/2;i++)     if (n%i==0)         s+=i; cout&lt;&lt;s;</pre>	<p>نمونه پاسخ درست:</p> <p>این برنامه یک عدد صحیح از کاربر گرفته و مجموع مقوم علیه‌های آن عدد غیر از خودش را نمایش می‌دهد. مثلاً اگر <math>n=12</math> باشد عدد 16 که مجموع <math>1+2+3+4+6</math> است در خروجی چاپ خواهد شد.</p> <p>نمونه پاسخ نادرست:</p> <p>این برنامه یک عدد صحیح از کاربر گرفته و سپس یک حلقه تشکیل داده و آنرا از یک <math>n/2</math> مقدار دهی نموده و به ازای هر <math>n</math> بررسی می‌کند که آیا <math>n</math> بر <math>i</math> بخش پذیر است یا خیر و اگر بود آن را در <math>s</math> جمع می‌کند و در آخر <math>s</math> را چاپ می‌کند. مثلاً اگر <math>n=12</math> باشد عدد 16 که مجموع <math>1+2+3+4+6</math> است در خروجی چاپ خواهد شد.</p>

سؤال ۱-

کد	پاسخ
<pre>int n, a, b=0; cin&gt;&gt;n; a=n%1000; while (a&gt;0) {     b*=10;     b+=a%10;     a/=10; } n/=1000; n*=1000; n+=b; cout&lt;&lt;n&lt;&lt;endl;</pre>	

سؤال ۲-



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۲ از ۶

کد	پاسخ
<pre>int i, a, b, temp, flag=0; cin&gt;&gt;a&gt;&gt;b; if(b&lt;a){     temp=a;     a=b;     b=temp; } i=a; while(flag==0){     if(i%a==0)         if(i%b==0){             flag=1;             cout&lt;&lt;i&lt;&lt;endl;         }     i++; }</pre>	
سؤال ۳ -	
کد	پاسخ
<pre>int a=1, b=1, c=1, count, n; cin&gt;&gt;n; if(n==1)     count=0; else     count=1; while(c&lt;n) {     c=b+a;     b=a;     a=c;     count++; } cout&lt;&lt;count&lt;&lt;endl;</pre>	
سؤال ۴ -	
کد	پاسخ
<pre>int n, temp, count=0; cin&gt;&gt;n; for(int i=1; i&lt;n; i++) {     temp=0;     if(i%2==0)         temp++;     if(i%3==0)         temp++;     if(i%5==0)         temp++;     if(i%7==0)         temp++;     if(temp==2)         count++; } cout&lt;&lt;count&lt;&lt;endl;</pre>	



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۳ از ۶

تذکر بسیار مهم تر (!): از این به بعد، سؤالاتی خواهید دهد که دوتا دوتا به صورت ا و آ پریم (مثلاً ۵ و ۵ پریم) هستند. از بین هر کدام از این دوتایی ها ( که نمرات و طبعاً درجه سختی متفاوتی دارند) باید دقیقاً یک سؤال را انتخاب کرده و پاسخ دهید. پاسخ به هر دو سؤال = نمره صفر

بخش دوم - خروجی کد:

سؤال ۵- (۱/۵ نمره) خروجی کد زیر را به ازای  $n=7$  و  $a=5$  بنویسید.

کد	پاسخ
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     int n, a, b=1;     float sum=1;     cin&gt;&gt;n&gt;&gt;a;     for(int i=1; i&lt;n; i++)     {         b*=a;         sum+=b/3;     }     cout&lt;&lt;sum&lt;&lt;endl;     return 0; }</pre>	

سؤال ۵ پریم- (۲ نمره) خروجی کد زیر را به ازای  $n=121$  بنویسید.

کد	پاسخ
<pre>#include &lt;iostream&gt; using namespace std;  int main(){     float a, b=1;     int n, sum=0;     cin&gt;&gt;n;     for(int i=1; i&lt;=n; i++)     {         b*=i;         a=i*i*i;         sum+=a/b;     }     cout&lt;&lt;sum&lt;&lt;endl;     return 0; }</pre>	



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۴ از ۶

بخش سوم - خطایابی کد:

\*  
\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*

سؤال ۶- (۳ نمره) هدف برنامه‌ی زیر این بوده است که شکل روبرو را در محیط متنی چاپ کند. پنج نوع خطا در این برنامه موجود است. آنها را پیدا کرده و توضیح دهید چگونه این خطاها تصحیح می‌شوند.

کد

پاسخ

```
include<iostream>
using namespace std;
int main
{
    int n=5, i;
    for(i=1; i<=n; i++)
    {
        for(j=1; j<=n-i; j++)
            cout<<" ";
        for(j=1; j<=2*i-1; j++)
            cout<<"*";
    }
    return;
}
```

\* \* \*  
\* \* \*  
\*\*\*  
\*  
\*\*\*  
\* \* \*  
\* \* \*

سؤال ۶ پریم - (۴ نمره) هدف برنامه‌ی زیر این بوده است که شکل زیر را در محیط متنی چاپ کند. شش نوع خطا در این برنامه موجود است. آنها را پیدا کرده و توضیح دهید چگونه این خطاها تصحیح می‌شوند.

کد

پاسخ

```
include<iostream>
using namespace std;
int main
{
    int n=4, i;
    for(i=1; i<=2*n-1; i++)
    {
        for(j=1; j<=2*n-1; j++)
            if((j==i) && (j==2*n-i) && (j==n))
                cout<<"*";
            else
                cout<<" ";
    }
    return;
}
```



به نام خدا

دانشکده  
مهندسی  
برق

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

آزمون میان ترم

درس: مبانی برنامه سازی (۲۵۷۶۸)

مدت آزمون: ۹۰ دقیقه صفحه ۵ از ۶

بخش چهارم - برنامه نویسی:

سؤال ۷- (۳ نمره) برنامه ای بنویسید که ۱۰۰۰۰ عدد تصادفی بین ۰ تا ۹۹۹۹۹ تولید کرده و نسبت تعداد اعداد اول تولید شده به کل اعداد را چاپ کند. مثلاً اگر می خواستیم ۵ عدد تصادفی ایجاد کنیم، خروجی به شکل زیر بدست می آمد.

خروجی	اگر ۵ عدد تصادفی تولید شود
0.4	5 9 102 19 48
0.2	3 0 4 18 10

سؤال ۷ پریم- (۴ نمره) برنامه ای بنویسید که ابتدا عدد  $n$  و سپس  $n$  عدد دیگر از ورودی بگیرد و عددی که بیشترین تکرار بین این  $n$  عدد ورودی را دارد، در خروجی چاپ کند. فرض کنید همواره عددی وجود دارد که بیش از  $n/2$  بار تکرار شده است. به طور مثال اگر  $n$  برابر ۱۰ بود، حتماً عددی وجود دارد که بیش از ۵ بار تکرار شده است.

خروجی نمونه	ورودی نمونه
10	4 10 10 2 10
2	9 8 2 800 2 224 2 9 2 2



به نام خدا

دانشکده  
مهندسی  
برق

آزمون میان ترم

نام و نام خانوادگی:

شماره دانشجویی:

نام استاد:

درس: مبانی برنامه سازی (۲۵۷۶۸)


مدت آزمون: ۹۰ دقیقه صفحه ۶ از ۶

سوال ۸- (۴ نمره) برنامه‌ای بنویسید که عدد  $n$  (  $n$  حداکثر 10000000000) را از ورودی گرفته و تعداد صفرهای سمت راست عدد  $n!$  را در خروجی چاپ کند. (دقت کنید که  $10000000000!$  بسیار بزرگتر از آن است که بتوانید آن را در یک متغیر ذخیره کنید).

خروجی نمونه	ورودی نمونه
6	26
1	5

سوال ۸- (۵ نمره) برنامه‌ای بنویسید که عدد  $n$  را از ورودی گرفته و معادل مبنای ۲ آن را در خروجی چاپ کند.

خروجی نمونه	ورودی نمونه
11010	26
101	5

دانشکده مهندسی برق	باسمه تعالی		
	نام و نام خانوادگی: شماره دانشجویی: نام استاد: تاریخ برگزاری: ۱۶ آذر ۱۴۰۲	<b>آزمون میان ترم</b> <b>سال تحصیلی ۱۴۰۲-۰۳</b> درس: مبانی برنامه سازی (۲۵۷۶۸) مدت آزمون: ۱۱۰ دقیقه صفحه ۱ از ۷	

**تذکر بسیار مهم:** محل درج پاسخ سوالات، در برگه‌ی سؤالات و به اندازه‌ی مشخص شده است! پس در مدیریت صفحات و سایز فونت نوشتاری (!) کوشا بوده و حتی‌الامکان با مداد و به صورت دو ستونی پاسخ سؤالات را بدهید تا جایی برای جبران اشتباهات وجود داشته باشد! برگه‌هایی که بیش از حد بد خط، خط خورده یا لاک‌اندود (!) باشند تصحیح نخواهند شد! پاسخ‌گویی با قلم قرمز ممنوع است!

**بخش اول – درک کد (۴ نمره)** در بخش زیر، ۴ قطعه کد وجود دارد و در هر مورد باید بگویید کد نوشته شده چه کاری انجام می‌دهد؟ یک مثال نیز باید برای کارکرد آن بیاورید. (البته ممکن است کد نوشته شده برنامه‌ی کامل نبوده و بخشی از یک برنامه باشد که در آن صورت باید بگویید آن بخش از کد چه کاری انجام می‌دهد):


کد	پاسخ
<pre>int n,i,s=0; cin&gt;&gt;n; for(i=1;i&lt;=n/2;i++)     if(n%i==0)         s+=i; cout&lt;&lt;s;</pre>	<div> <input checked="" type="checkbox"/> نمونه پاسخ درست:         </div> <p>این برنامه یک عدد صحیح از کاربر گرفته و مجموع مقوم علیه‌های آن عدد غیر از خودش را نمایش می‌دهد. مثلاً اگر <math>n=12</math> باشد عدد 16 که مجموع <math>1+2+3+4+6</math> است در خروجی چاپ خواهد شد.</p>

سؤال ۱ -

کد	پاسخ
<pre>int n; cin&gt;&gt;n; for(int i=1;i&lt;=n;i++) {     int k=i-n;     for(int j=1;j&lt;=n;j++)         cout&lt;&lt;setw(4)&lt;&lt;(k+=n);     cout&lt;&lt;endl; }</pre>	

سؤال ۲ -

کد	پاسخ
<pre>int f(int n){     static int c=0;     c+=n;     return c; } int main(){     int n, s, x;     cin&gt;&gt;n;     for(int i=0;i&lt;n;i++){         cin&gt;&gt;x;         s=f(x);     }     cout&lt;&lt;s&lt;&lt;endl;     return 0; }</pre>	

<b>دانشکده مهندسی برق</b>	نام و نام خانوادگی: شماره دانشجویی: نام استاد: تاریخ برگزاری: ۱۶ آذر ۱۴۰۲	<b>آزمون میان ترم</b> <b>سال تحصیلی ۱۴۰۲-۰۳</b> درس: مبانی برنامه سازی (۲۵۷۶۸) مدت آزمون: ۱۱۰ دقیقه صفحه ۲ از ۷	
-----------------------------------	--	--	--

سؤال ۳ -

کد	پاسخ
<pre>int a, b, c, T; bool f = false; cin&gt;&gt;T; for(a=1;(a&lt;=T)&amp;&amp;(!f);a++)     for(b=1;(b&lt;=T-a)&amp;&amp;(!f);b++)     {         c=T-a-b;         if(a*a+b*b==c*c)         {             cout&lt;&lt;a&lt;&lt;"", "&lt;&lt;b&lt;&lt;"", "&lt;&lt;c&lt;&lt;endl;             f = true;         }     } if(!f)     cout&lt;&lt;"Impossible";</pre>	

سؤال ۴ -

کد	پاسخ
<pre>int g(int a[], int n) {     int m=0;     for (int i = 0; i &lt; n; i++)     {         int j;         bool f=false;         for (j = 0; (j &lt; i)&amp;&amp;(!f); j++)             if (a[i] == a[j])                 f=true;         if (!f)             a[m++] = a[i];     }     return m; }</pre>	

بخش دوم - خروجی کد:

سؤال ۵ - (۱.۵ نمره) خروجی تکه کد زیر (مقدار چاپ شده در خروجی) را به ازای  $n=1402$  و  $a$  ها به ترتیب زیر بنویسید:

**a: -1 1 -1 -1 2 -1 -1 -1 3 -1 -1 -1 -1 4 ...**

<pre>int n, p = 0, c = 0, a; cin &gt;&gt; n; for (int i = 0; i &lt; n; i++) {     cin &gt;&gt; a;     if(a &gt; 0)         p += a;     else</pre>	<pre>{     if (p == 0)         c++;     else         p--; } //else } //for cout &lt;&lt; c;</pre>
---	---

پاسخ



<b>دانشکده مهندسی برق</b>	نام و نام خانوادگی: شماره دانشجویی: نام استاد: تاریخ برگزاری: ۱۶ آذر ۱۴۰۲	<b>آزمون میان ترم</b> <b>سال تحصیلی ۱۴۰۲-۰۳</b> درس: مبانی برنامه سازی (۲۵۷۶۸) مدت آزمون: ۱۱۰ دقیقه صفحه ۳ از ۷	
	<b>سؤال ۶ - (۱ نمره) مقدار آرایه a پس از اجرای تابع h به ازای n=10 و a[]={1,2,3,4,5,6,7,8,9,10} را بنویسید:</b>		
<pre>void h(int a[], int n) {     for(int i=0;i&lt;n/2;i++)         swap(a[i],a[n-i-1]); }</pre>		<b>پاسخ:</b>	
<p align="right"><b>بخش سوم - سؤالات Syntax:</b></p> <p><b>سؤال ۷ - (۱ نمره) تفاوت و شباهت های if و switch چیست؟ توضیح دهید.</b></p> <p align="right"><b>پاسخ:</b></p>			
<p align="right"><b>بخش چهارم - برنامه نویسی:</b></p> <p><b>سؤال ۸ - (۳.۵ نمره) برنامه ای بنویسید که عدد n را به عنوان ورودی گرفته و سپس کوچکترین و بزرگترین عددی را که می توان با جابجایی ارقام آن ساخت، ساخته (یعنی متغیری با آن مقدار ساخته و در متغیری قرار دهید) و آن ها را در خروجی چاپ کند. استفاده از مرتب سازی مجاز نیست.</b></p> <p align="right"><b>ورودی: عدد حداکثر ۱۷ رقمی n</b></p> <p align="right"><b>خروجی: کوچکترین و بزرگترین عددی که می توان با جابجایی ارقام n ساخت</b></p>			
ورودی نمونه ۱		ورودی نمونه ۲	
9978234444356566		6033050350575000	
خروجی نمونه ۱		خروجی نمونه ۲	
2334444556667899 9987666554444332		333555567 7655553330000000	



آزمون میان ترم  
سال تحصیلی ۱۴۰۲-۰۳  
درس: مبانی برنامه سازی (۲۵۷۶۸)  
مدت آزمون: ۱۱۰ دقیقه صفحه ۴ از ۷

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:

تاریخ برگزاری: ۱۶ آذر ۱۴۰۲

دانشکده  
مهندسی  
برق

پاسخ:

سؤال ۹ - (۴ نمره) همواره برای انتخاب یک رمز برای استفاده‌های مختلف انسان‌ها در حال چالش با خود هستند یک نظریه وجود دارد که میگوید در میان رمزهایی که کاملاً متشکل از ارقام هستند یعنی شامل کاراکتر نیستند آن دسته از رمزهایی که جان سخت (!) باشند دارای امنیت بالایی هستند.


یک رمز  $N$  رقمی جان سخت است اگر عددی اول باشد و اگر از سمت راست آن شروع به حذف ارقام آن بکنیم همواره عدد باقیمانده نیز عددی اول باشد. مثلاً رمز ۲۳۹۹ یک رمز جان سخت است زیرا اعداد ۲۳۹۹ و ۲۳۹ و ۲۳ و ۲ همگی اول هستند. همچنین ۳۱۳۷ نیز یک رمز جان سخت است.

برنامه ای بنویسید که عدد  $N$  را بگیرد و تمام رمزهای جان سخت به طول  $N$  را چاپ کند. دقت کنید که عدد ۱ عدد اول نیست.

ورودی: یک خط شامل عدد  $N$  که  $1 \leq N \leq 8$

خروجی: رمزهای جان سخت به طول  $N$  که به صورت صعودی مرتب شده‌اند.

ورودی نمونه	خروجی نمونه
$N=2$	23 29 31 37 53 59 71 73 79

<div>دانشکده مهندسی برق</div>	<div>نام و نام خانوادگی: شماره دانشجویی: نام استاد: تاریخ برگزاری: ۱۶ آذر ۱۴۰۲</div>	<div>آزمون میان ترم سال تحصیلی ۱۴۰۲-۰۳ درس: مبانی برنامه سازی (۲۵۷۶۸) مدت آزمون: ۱۱۰ دقیقه صفحه ۵ از ۷</div>	<div></div>
<div>پاسخ:</div>			



آزمون میان ترم  
سال تحصیلی ۱۴۰۲-۰۳  
درس: مبانی برنامه سازی (۲۵۷۶۸)  
مدت آزمون: ۱۱۰ دقیقه صفحه ۶ از ۷

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:  
تاریخ برگزاری: ۱۶ آذر ۱۴۰۲

دانشکده  
مهندسی  
برق

	4				2		1	9
				3	5	1		8
3	1				9	4	7	
	9	4						7
2							8	9
			9	5	2			4
4	2			1	6	9		
1	6			8				7

سؤال ۱۰- (۵ نمره) جدول سودوکو، جدولی ۹ در ۹ است که در هیچ یک از سطرها، ستون‌ها و مربع‌های ۳ در ۳ مشخص شده این جدول نباید عدد تکراری وجود داشته باشد (شکل روبرو). یک جدول سودوکو به شما داده می‌شود که بعضی از خانه‌های این جدول توسط اعداد ۱ تا ۹ پر شده و برخی از خانه‌های آن خالی است. می‌خواهیم برای یک خانه خاص از جدول بررسی کنیم که در این مرحله چه مقداری می‌تواند در این خانه قرار بگیرد. برنامه‌ای بنویسید که تمام مقادیر ممکن که می‌توان در این مرحله در خانه مورد نظر نوشت را به صورت صعودی چاپ کند و اگر هیچ مقداری نمی‌تواند جایگزین خانه مشخص شده شود صفر چاپ کند.

**ورودی:** ورودی شامل ۱۰ خط است و در ۹ خط اول هر خط شامل ۹ عدد است که مقادیر خانه‌های جدول را مشخص می‌کنند. اگر مقدار یک خانه صفر باشد به این معناست که هنوز این خانه از جدول خالی است. در آخرین خط ورودی به ترتیب شماره سطر و شماره ستون خانه مورد نظر از جدول که قرار است مقدار آن بررسی شود می‌آید.

**خروجی:** در تنها خط خروجی، هر یک از اعداد بین ۱ تا ۹ که می‌توان در خانه مذکور جایگذاری کرد را به صورت صعودی چاپ کنید و در غیر این صورت صفر را چاپ کنید.

ورودی نمونه ۱	ورودی نمونه ۲
000000000 000002000 000000001 005000008 000000000 000000000 000000000 000000305 000072640 000000900 89	609000804 000900630 000080079 540096083 000030000 930410065 360080000 082004000 704000306 22
خروجی نمونه ۱	خروجی نمونه ۲
0	1257

پاسخ:



آزمون میان ترم  
سال تحصیلی ۱۴۰۲-۰۳  
درس: مبانی برنامه سازی (۲۵۷۶۸)  
مدت آزمون: ۱۱۰ دقیقه صفحه ۷ از ۷

نام و نام خانوادگی:  
شماره دانشجویی:  
نام استاد:  
تاریخ برگزاری: ۱۶ آذر ۱۴۰۲

دانشکده  
مهندسی  
برق