

درسنامه ویژه میانترم مبانی برنامه نویسی – کل حل تمرین چهارشنبه‌ها (۱۹ الی ۲۱) – معین مکیان

مباحث:

- عملگرها و اولویت‌ها
 - Case Sensitivity
 - Coments
 - 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

- 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 $\backslash n$ و $\backslash t$
- 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