



مرور و حل تمرین میانترم (:



انواع سوالات ممکن!

سوالات حفظی!

سوالات پیدا کردن خروجی (خروجی معمولی و مخصوص!)

سوالات رفع ایراد

سوالات پیاده سازی

سوالات حفظی:

اجزای سخت افزاری سیستم های کامپیوتری را نام برده و هریک را به اختصار شرح دهید.

کجا رو بخونیم؟

اسلاید اول (: به طور کامل این مطالب رو پوشش داده.

سوالات خروجی (ساده):

```
int i;  
for(i = 0; i < 10; i++){  
    if(i >= 5)  
        continue;  
    switch(i % 3){  
        case 0:  
            printf("case 0\n");  
            break;  
        case 1:  
        case 2:  
            printf("case 1\n");  
            break;  
        default:  
            printf("case default \n");  
    }  
}
```

سوالات خروجی (ساده):

پاسخ:

case 0

case 1

case 1

case 0

case 1

سوالات خروجی (ساده):

```
int i, j;
void f(void){
    int k, m = 10;
    static int n = 0;
    for(k = 0; k < 2; k++){
        int m = 20;
        n++;
        printf("i=%d, j=%d, k=%d, n=%d, m=%d\n", i, j, k, n, m);
    }
}

void main(void){
    f();
    i = 10;
    f();
}
```

سوالات خروجی (ساده):

پاسخ:

$i=0, j=0, k=0, n=1, m=20$

$i=0, j=0, k=1, n=2, m=20$

$i=10, j=0, k=0, n=3, m=20$

$i=10, j=0, k=1, n=4, m=20$

سوالات خروجی (مخصوص!):

خروجی قطعه کد زیر را بدست بیاورید؛ ضمناً مواردی که cast رخ میدهد را مشخص کنید.

سوالات خروجی (مخصوص!):

```
int i, j;  
double d;  
float f;  
d = f = j = i = 1;  
d = j + i + 1.0;  
i = i % (int)(f + j);  
printf("%lf, %d\n", d, i);
```

پاسخ

3.000000, 1

سوالات خروجی (مخصوص):

```
int i, j; double d;  
d = 1.0;  
j = i = 1;  
d = (i++) + (--j);  
i += j++ + 1;  
printf("%lf, %d, %d\n", d, i, j);
```

پاسخ

1.000000, 3, 1

سوالات خروجی (مخصوص):

```
int i = 0, j = 1;  
bool p, q;  
p = i;  
q = p ? p : !p;  
i = p && q;  
j = q || q;  
printf("%d, %d\n", i, j);
```

پاسخ

0, 1

سوالات پیاده سازی:

یک تابع بنویسید که یک عدد $n > 2$ بگیرد و در صورتی که جمله n ام فیبوناچی باشد؛ جمله بعدی را برگرداند و در غیر این صورت مقدار -1 را برگرداند. با استفاده از این برنامه یک برنامه کامل بنویسید که یک ورودی از کاربر بگیرد. و در صورتی که عضو دنباله فیبوناچی باشد جمله قبلی و بعدی را چاپ کند. در غیر این صورت یک پیغام خطای مناسب چاپ کند. (از فرمول مستقیم رابطه فیبوناچی استفاده نکنید.)

سوالات پیاده سازی:

فرض کنید تابعی برای تشخیص همپوشانی دو بازه داریم. به این صورت که `bool overlap (int a1, int b1, int a2, int b2)` این تابع بازه‌های `[a1,b1],[a2,b2]` بررسی میکند و چنانچه حداقل یک نقطه مشترک هم داشته باشد مقدار `true` را برمیگرداند و در غیر این صورت مقدار `false` را برمیگرداند. حال با استفاده از این تابع میخواهیم تابعی به نام `cover` بنویسیم که در آن 6 عدد ورودی میگیریم `a1,b1,a2,b2,a3,b3` و بازه‌های ایجاد شده را بررسی میکنیم و چنانچه بازه‌ای بزرگتر ایجاد میکردند، `true` و در غیر این صورت `false` برمیگردانند.