

۱- خروجی قطعه کد زیر چه خواهد بود؟

```
1  #include <iostream>
2  using namespace std;
3  int func1(int a) {
4      a++;
5      return a;
6  }
7  int main()
8  {
9      int a = func1(1024);
10     cout<<a;
11 }
```

۲- خروجی قطعه کد زیر چه خواهد بود؟

```
1  #include <iostream>
2  using namespace std;
3  int func2(int a) {
4      return 2 + a * a;
5  }
6  int func1(int a) {
7      if(a % 2 == 0) {
8          return func2(a - 2);
9      }
10     else {
11         return func2(a + 2);
12     }
13 }
14 int main()
15 {
16     int a = func1(1);
17     int b = func1(2);
18     cout<<a<<" "<<b;
19 }
```

۳- متاسفانه!! اجرای برنامه زیر با خطا همراه بوده است! این کد را با کمترین تغییرات به گونه ای کامل کنید؛ که توان دوم ۷ را در خروجی چاپ کند.
تبصره ۱: حق تغییر دادن تابع func1 را ندارید!
تبصره ۲: باید از تابع func1 استفاده شود!

```
1  #include <iostream>
2  using namespace std;
3  void func1(int a) {
4      cout<<a * a;
5  }
6  int main()
7  {
8      int a = func1(7);
9      cout<<a;
10 }
```

```
1  #include <iostream>
2  using namespace std;
3  struct Date{
4      int day;
5      int month;
6      int year;
7  };
8  struct Person{
9      char name[20];
10     int age;
11     Date birthday;
12     Person close_friend;
13 };
14 int main()
15 {
16     Person p1;
17     p1.age = 13;
18     cout<<p1.age;
19 }
20
```

۴- آیا برنامه روبرو به درستی اجرا می شود؟ اگر بله؛ خروجی برابر چه خواهد بود؟ و اگر نه چرا؟ توضیح دهید!

۵- ساختار (structure) یک چراغ راهنمایی و رانندگی را با ۴ ویژگی، پیاده سازی کنید:

بخش اول سوال:

موارد زیر را در نظر بگیرید: (دقت کنید! در هر ویژگی نوع مناسب داده را در نظر بگیرید؛

هدف پیاده سازی این ساختار است، نه مقداردهی یا محاسبه مقادیر)

الف) رنگ فعلی چراغ (قرمز، زرد یا سبز)

ب) رنگ قبلی چراغ (قرمز، زرد یا سبز)

پ) مدت کل زمانی که باید با رنگ فعلی روشن بماند (با واحد ثانیه)

ت) مدت زمانی که با رنگ فعلی روشن بوده (با واحد ثانیه)

ث) مدت زمانی که تا تغییر رنگ باقی مانده (با واحد ثانیه)

ج) آیا دکمه مخصوص عابر پیاده فشار داده شده است؟ (تنها دو حالت دارد)

بخش دوم سوال (امتیازی):

در این ساختار؛ یک تابع با نام `update_time` و یک تابع با نام `update_color` وجود دارد. تابع `update_time` هر ثانیه یک بار؛ توسط پردازنده داخلی، صدا زده می‌شود (اجرا می‌شود)؛ وظیفه این تابع به روز رسانی پارامترهای مربوط به زمان (موارد پ؛ ت و ث) و در صورت لزوم فراخوانی تابع `update_color` است. هیچ‌کدام از دو تابع هیچ ورودی‌ای ندارند! بدنه این دو تابع را پیاده سازی کنید!

مهم ترین فعالیت تابع `update_time` این است که با هر بار صدا زده شدن؛ یک ثانیه تغییر را در هر ویژگی مربوط به زمان اعمال میکند؛ (مثلا مدت زمان قسمت ت را یک ثانیه زیاد می‌کند)

می‌دانیم هر گاه موارد پ و ت با هم برابر شوند تابع `update_color` صدا زده می‌شود؛ همچنین؛ به روز رسانی مورد ث (که محتوای درون آن به عنوان زمان سنج در چهار راه ها نمایش داده میشود) را فراموش نکنید! (رابطه ریاضی بین پ و ت و ث برقرار کنید) در پیاده سازی `update_color` نیز فرض کنید همیشه رنگ ها به ترتیب عوض می‌شوند (سبز - زرد - قرمز - زرد - سبز) (با دقت به مورد الف و ب می‌توانید پیاده سازی خود را انجام دهید)

۶- (امتیازی) برنامه‌ای بنویسید که همه اعداد طبیعی؛ خودمقلوب و اول کوچکتر از ۱۰۰۰ را چاپ کند!

به عنوان مثال؛ ۱۳۱ چنین ویژگی‌ای دارد؛ چون هم اول است هم خودمقلوب است!
(عدد خودمقلوب عددی است که از هر دو سمت به یک صورت خوانده می‌شود مثل ۹۴۹)
(حل سوال با تابع نمره مثبت خواهد داشت: یعنی تابعی بنویسید که خودمقلوب بودن ورودی را تشخیص دهد، یک تابع هم برای تشخیص اول بودن ورودی بنویسید؛ با استفاده از آن‌ها اقدام به حل سوال کنید!)