



سوالات

سوال اول

دو تابع `xSemaphoreCreateBinaryStatic` و `xSemaphoreCreateBinary` را در `free RTOS` توضیح دهید و تفاوتشان را بیان کنید.

سوال دوم

دو مورد از آنومالی هایی که در زمانبندی روی چند پردازنده ممکن است بوجود بیاید را با ذکر مثال توضیح دهید.

سوال سوم

توضیح دهید چگونه میتوان یک سیستم عامل معمولی را به سیستم عامل `real time` تبدیل کرد؟ این کار چه مزایا و چه معایبی دارد؟

سوال چهارم

وظایف زیر را در نظر بگیرید

	C_i	T_i
t_1	۱	۴
t_2	۲	۶
t_3	۳	۸

زمان بندی براساس الگوریتم های `RM` و `EDF` را در یک `Hyperperiod` رسم کنید و مشخص کنید کدام یک از الگوریتم های ذکر شده باعث نقض موعد می شود.

سوال پنجم

در ابتدا کتابخانه `FreeRTOS Arduino` را بر روی `IDE Arduino` خود نصب کنید . بعد از نصب کتابخانه ، `IDE` را باز کرده و از مسیر `File->Examples->FreeRTOS` ، مثال `BlinkAnalogread` را انتخاب کنید . حال تابع `TaskBlink` را به صورت زیر تغییر بدهید.

```

void TaskBlink(void *pvParameters) {

    (void) pvParameters;

    pinMode(LED_BUILTIN, OUTPUT);

    volatile int i = 0;
    for (;;) // A Task shall never return or exit.
    {
        digitalWrite(LED_BUILTIN, HIGH);
        for(i=0;i<30000;i++);
        digitalWrite(LED_BUILTIN, LOW);
        for(i=0;i<30000;i++);
    }
}

```

برنامه را اجرا کرده و به سوالات زیر پاسخ دهید:

۱. تابع TaskBlink اولیه با تابع جدید چه تفاوتی دارد؟
۲. متغیر i به چه دلیل volatile تعریف شده است؟
۳. در صورتی که متغیر i را به صورت volatile تعریف نکنیم و برنامه را اجرا کنیم ، چه تفاوتی با volatile دارد؟

موفق باشید