

1) زیرروال زیر اعداد در رجیستر های r0 و r1 به ترتیب صعودی مرتب می کند. (با استفاده از return pass by value) زیرروال را بازنویسی کنید به صورتی که کارا تر باشد با توجه به استفاده از register ها و تعداد دستورات بدون تغییر دادن ساختار زیرروال. نحوه ی انتقال پارامتر ها را تغییر ندهید. هیچ یک از کاربردها را تغییر ندهید.

(2

```

3) Sort      STR LR,[SP,#-4]!
4)  STR R2,[SP,#-4]!
5)  STR R3,[SP,#-4]!
6)  STR R4,[SP,#-4]!
7)    CMP R0,R1
8)    BLE Done
9)    MOV R1,R0
10)   MOV R0,R2
11) Done  LDR R4,[SP],#4
12) LDR R3,[SP],#4
13) LDR R2,[SP],#4
14) LDR LR,[SP],#4
15) MOV PC,LR

```

2) برنامه ای بنویسید که در آن تابعی با عملکرد زیر صدا زده می شود:

عدد n را از stack بخواند

N عدد دیگر از حافظه بخواند و max و min آن n عدد را برگرداند

در تابع main مقدار برگردانده شده را استفاده کند و min را در R10 و همچنین max را در R11 بریزد.

توجه: صدا زدن تابع باید با استفاده از BL باشد و پاس دادن مقادیر به تابع و همچنین برگرداندن جواب باید تماماً با استفاده از stack باشد و به پیاده سازی های دیگر نمره تعلق نمی گیرد.

{ تمرین خود را فقط با فرمت HW3_G(group number)_StudentNumber.zip آپلود کنید. در صورت رعایت نکردن فرمت ذکر شده عواقب آن به عهده ی دانشجو می باشد }

مثال: HW3_G2_9531066.zip

{ برای این تمرین یک فایل زیپ آپلود کنید که هر دو سوال به صورت دو کد اسمبلی در فایل هستند. }

{ تمرین به صورت انفرادی می باشد }

{ ددلاین تمرین جمعه 8/24 می باشد }

{ در صورت داشتن سوال با ایمیل درس در ارتباط باشید }