

## ELMU 4088 - Bilgisayar Mimarisi

### Çalışma 2

**Alan Hesaplama:** Küp, küre ve silindirin alanını hesaplamak için bir MIPS birleştirici dil (assembly language) kodunu yazınız. Her şekil için bir numara belirlenecektir ve kullanıcıdan hangi şekle ait alan hesaplaması yapılacağı seçilecektir. Ayrıca kullanıcıdan alanı hesaplanacak şekle ait olan parametreler istenecektir. Hesaplanan alan değerleri, tanımlanmış on (10) elemanlı bir diziye (AlanDizisi) kayıt edilecektir. On (10) adet değer hesaplanınca program bitirilecektir. Hesaplanan alan değerleri konsolda gösterilecektir.

Örneğin;	No.	Alan
Küp	1	$6*a^2$
Küre	2	$4*pi*r^2$
Silindir	3	$2*pi*r*h + 2*pi*r^2$

$r \rightarrow$  yarıçap

$a \rightarrow$  kenar

$h \rightarrow$  uzunluk

$pi \rightarrow 314 / 100$  pi sayısını şu şekilde kullanabilirsiniz (önce 314 ile çarpma ve sonra 100' e bölme)

Bir dizi için yer tahsis etme;

AlanDizisi: .space 400 (byte cinsindendir, bu alana 100 eleman yerleştirilebilir).

Bir dizinin taban adresini alma;

la \$t9, AlanDizisi

•div \$t1, \$t2, \$t3

Bölme: \$t2'yi \$t3'e bölerek \$t1'e yazar, tamsayı bölme.

•mul \$t1, \$t2, \$t3

Multiplication: Sözdde komuttur. 32-bit bir çarpma yapar ve sonucu 32-bit bir kaydediciye yazar. Taşma kontrolü yoktur. Hi kaydedicisinin bitleri kontrol edilmez veya saklanmaz.

Bu durum programcının kontrolü bir sonraki komut yerine programdaki bir başka konuma dandandırmasına [branch] (atlamasına [jump]) izin verir. Bu tip komutlar birleştirici (assembler) dilde koşullu işlemleri yapabilen program yürütülebilmesine olanak tanır.

Eğer \$rs kaydedicisi \$rt kaydedicisinden küçük ise, \$rd kaydedicisi set (buna 1 yazılır) edilir. Aksi halde \$rd kaydedicisine sıfır yazılır;

SLT RD, RS, RT

# RD = (RS < RT) ? 1 : 0

ELMU 4088 - Bilgisayar Mimarisi  
Çalışma 2

İki kaydedicinin içeriği eşit ise dallanma olur;

**BEQ RS, RT, Label**                      #eğer  $RS = RT$ , *Label'a dallan*

İki kaydedicinin içeriği eşit değil ise dallanma olur;

**BNE RS, RT, Label**                      #eğer  $RS \neq RT$ , *Label'a dallan*