

## PROGRAMLAMA DİLLERİ VİZE SINAVI SORULARI

1. Aşağıdaki BNF'yi, "+"yı "\*" üzerine öncelikli ve "+"yı sağ birleşmeli olacak şekilde yeniden düzenleyiniz.

```
<assign> → <id> = <expr>
<id> → A | B | C
<expr> → <expr> + <term> | <term>
<term> → <term> * <factor> | <factor>
<factor> → (<expr>) | <id>
```

2.

a) Yandaki C programında main'deki atama ifadesinden sonra x'in değeri, operantlar soldan sağa değerlendirildiğinde ne olur?

b) Yandaki C programında main'deki atama ifadesinden sonra x'in değeri, operantlar sağdan sola değerlendirildiğinde ne olur?

```
int fun(int *i) {
    *i += 5;
    return 4;
}

void main ( ) {
    int x = 3;
    x = x + fun(&x);
}
```

3. YAZMUH adlı yeni bir programlama dili yazacağımızı varsayınız. Bir görev de, biçimsel olarak dilin gramerini belirlemektir. Aşağıda listelenen çeşitli öncelik ve birleşmeliliklere sahip farklı operatörler istediğimizi varsayınız.

Öncelik	Operatörler	Operatör Tipi	Birleşmelilik
En yüksek	~, !	prefix unary	Sağdan sola (sağ birleşmeli)
	@	infix binary	Yok
	#	infix binary	Soldan sağa (sol birleşmeli)
En düşük	%, &	infix binary	Soldan sağa (sol birleşmeli)

Operatör önceliği ve birleşmeliliğe göre, aşağıdaki BNF'yi tamamlayınız. Terminal olmayan <identifier> ve <number>'ın sizler için tanımlandığını ve değişken ve sayısal değerleri temsil ettiğini varsayınız. Parantez kullanmak zorunda değilsiniz.

<elem> ::= <identifier> | <number>

<expr> ::=

4. Aşağıdaki top metodunda (C) etiketli noktadaki referans çevre nedir?

```
public static int top(int[] a) {
    int top = 0;
    for (int i = 0; i < a.length; i++) {
        int v = a[i];
    }
}
```

```
        // (C)
        top += v;
    }
    return top;
}
```

5. Aşağıda bir program verilmiştir.

- a) Eğer dil dinamik kapsamlıysa ne basılır?
- b) Eğer dil statik kapsamlı ise ne basılır?

```
const int b = 18;
int foo()
{
    int a = b + 5;
    printf("%d\n", a);
}

int bar()
{
    int b = 39;
    return foo();
}

int main()
{
    foo();
    bar();
    return 0;
}
```