

BigAdd Dili

BigAdd dili , büyük tamsayıları programlı olarak eklemek ve çıkarmak ve sonucu ekranda göstermek için tasarlanmış küçük bir programlama dilidir. BigAdd , çevrilmiş bir dildir. BigAdd yorumlayıcı komut BA komut satırından çalışır ve sadece nitelik olarak komut dosyanın adını alır. Betik dosyasının uzantıya sahip olduğu kabul edilir . *ba* .

örnek : c: \> ba komutu myscript

yük ve myscript.ba denilen komut dosyasını çalıştırmak gerekir

Kod Satırları: Nokta ('.') , Satır karakterinin sonu . Satırlar anahtar kelimeler, parantezler, değişkenler ve sabitler içerebilir. Bu elemanlar arasında herhangi bir sayıda boşluk veya yorum kullanılabilir. Bir program satırı, birden çok metin satırı arasında bölünebilir.

Bir kod satırı aşağıdakilerden biridir:

- Bir değişken bildirimi .
- Bir atama ifadesi.
- Bir ek deyimi.
- Bir çıkarma ifadesi.
- Bir çıktı ifadesi.
- Bir döngü ifadesi.

Yorumlar:

Yorumlar küme parantezleri arasında yazılmıştır {}

örnek : {bu bir yorumdur} {ve

buda bir yorumdur

Veri tipleri: Değişkenler için veri türü *tamsayı*. Tamsayı, **ondalık veri türü** olarak temsil edilen imzalı bir tam sayıdır . Bir tamsayı **100 ondalık basamak** kadar büyük olabilir .

Dizeler yalnızca sabitler olarak bulunur ve yalnızca *out* deyimlerinde kullanılır .

örnek : 123113, -5, 0, -314159 geçerli tamsayı gösterimleridir .

3.14159 geçerli bir tam sayı değil (gerçek bir sayıdır)

3.0 geçerli bir tamsayı değil (ondalık nokta gösterilmemeli)

3. geçerli bir tamsayı değil (ondalık nokta gösterilmemeli)

- 5 geçerli bir tamsayı değil (eksi işareti ile ilk rakam arasında boş bırakılmamalıdır.

--5 geçerli bir tam sayı değil (yalnızca bir eksi işaretine izin verilir)

+5 geçerli bir tamsayı değil (artı işaretine izin verilmez)

Değişkenler: Tüm değişkenler bir tamsayı olarak bildirilmelidir . Değişkenler kullanılmadan önce bildirilmelidir. Tüm değişkenler geneldir (ve statiktir).

int <değişken>.

Değişken isimleri büyük / küçük harf duyarlıdır ve en fazla 20 karakter uzunluğundadır. Değişken isimleri bir harfle (alfabetik karakter) başlar ve alfanümerik karakterlerle veya alt çizgi karakteriyle devam eder.

örnek : int myVar.

Tüm değişkenler oluşturma sırasında 0 değeriyle başlatılır.

Atama deyimi:

<int_value > ögesini <değişken> 'e taşıyın .

örnek : 25'i Varma'ya götür . { 25'i myVar'a atar }

hareket yourVar için MyVar. { atar yourVar için myVar}

Ek beyan:

<int_value > ögesini <değişken> ' e ekleyin .

Değişkeni int_value ile arttırır .

örnek : toplamı 2 ekleyin. {Toplaman değeri 2 artar}

Çıkarma ifadesi:

alt < int_value > , <değişken> 'den.

örnek : alt ttoplam. {Toplam değer t ile azalır}

Çıktı ifadesi:

dışarı < dışarı_listesi >.

örnek : “Sonuç şudur:” , toplam .

Bir tamsayı değeri değişken veya sabittir.

<int_value > → <değişken> | <int_const >

Bir çıkış listesi virgülle ayrılmış dizge ve tamsayı değerlerinin listesi.

<out_list > → <out_list > , <list_element> | <list_element >

<list_element > → <int_value> | <string> | Yeni hat

Bir dize, iki tırnak işareti arasındaki herhangi bir karakter dizisidir.

örnek : “Merhaba, bu bir dize!”

Döngü:

döngü < int_value > times <line>

{VEYA}

döngü < int_value > times <code_block >

Bir döngü int_value ile başlar ve her yinelemede int_value değerini bir azaltır . İnt_value değişken ise, değişkenin değeri erişilebilir ve döngü sırasında modifiye edilebilir. Döngüden sonra değişken 0 değerini alır. Döngünün son yinelemesi 1 değeriyle çalışır.

örnek :

10 kez dışarı döngü ``*``. { Ekrana 10 yıldız yazar }

Kod Bloğu:

Bir kod bloğu, köşeli ayraçlar arasındaki kod satırlarının bir listesidir. Bir kod bloğu, bir döngüde, tek bir kod satırı ile değiştirilebilir bir şekilde kullanılabilir. Kod bloklarını kullanarak döngüler birbirlerinin içine yerleştirilebilir.

örnek :

int boyut

int sum.

hareket 5 boyuta kadar.

döngü boyutu süreleri

[boyut, yeni satır.

 toplama boyut ekleyin .

]

yeni satır dışarı, “Sum:” toplamı.

Çıktı:

5

4

3

2

1

Toplam : 15

Hatalar: Hata algılama , başarılı bir BigAdd uygulamasının önemli bir yönüdür . Bir tercüman projesinden yüksek puan alabilmek için hataları tam olarak tespit edebilmelidir. Hata mesajları kesin hata tipini ve hata yerini göstermelidir. Tercüman, bildirilmemiş değişken kullanımına, anahtar kelimelerin yazım hatalarına veya dilbilgisi kötü koduna tolerans göstermemelidir .