

Dedocumenter

Hazırlayan: Mehmet Said Yolcu

Tarih: 11.07.2023

Giriş:

Dedocumenter programı excel'de tanımlı mesaj paketlerini alıp bunları tanım kıstasları uyarınca bir C structure'ına dönüştürüyor.

Program Arayüzü ve Kullanımı:

Programı komutunun genel formatı (normalde tek satırda yazılıyor):

```
python3 dedocumenter.py ( -d | -c) -i <girdi dosyası> -o  
<çıktı dosyası> -l <ilk satır> <son satır> -r <rezerve  
değişken sayısı> -b <byte count>
```

Bu komutun terminalde girdi dosyasının bulunduğu adreste çalıştırılması lazım.

İlk komut:

Program şu komut ile çalıştırabilirsiniz:

```
python3 dedocumenter.py -d -i <girdi dosyası> -o <çıktı  
dosyası> -l <ilk satır> <son satır> -p
```

-d seçeneği çıktı dosyasındaki halihazırdaki verinin silinerek, programın çıktısının çıktı dosyasına yazılmasını sağlıyor. Yani overwrite seçeneği.

-i seçeneğinin sonrasında girdi dosyası verilmeli.

-o seçeneği sonrasında çıktı dosyası veriliyor. Eğer -o seçeneği hiç kullanılmayıp bir çıktı dosyası verilmezse, program girdi dosyasının adının sonuna bir .h uzantısı ekleyerek çıktı dosyasını oluşturuyor.

-l seçeneği sonrasında sırasıyla tanımlı mesaj paketinin girdi dosyasında başladığı ve bittiği satırlar verilmeli.

-p flagını girmek zorunsuz. Ancak girildiğinde, -l flagında girilen bitiş satırının işlenmesi ile tüm mesaj paketinin işlenmiş olacağı bilgisini veriyor. Programın mesaj paketi structure'ını otomatikman kapatmasını sağlıyor. Bu flag girilmediği durumda structure'ın bitişi kullanıcı tarafından yazılmalı.

Devamındaki komutlar:

Program dökümanda bir hata ile karşılaştığında, eğer hata kendi düzeltebileceği bir hata ise terminale bir prompt vererek kullanıcıya hatayı manuel mi düzeltmek istediğini yoksa programın otomatik düzeltmesini mi istediğini soruyor. Kullanıcının verdiği cevaba göre ya programı sonlandırıp kontrolü kullanıcıya devrediyor, veya hatayı bizzat düzeltiyor. Örneğin dökümanda byte sayımında bir hata yapılmışsa, program bunu terminal yoluyla kullanıcıya bildirip, kullanıcıya hatayı manuel düzeltmek isteyip istemediğini sorup, eğer "otomatik düzeltme" yanıtı alırsa C structure'ına kendi saydığı byte sayısına göre byte değerlerini yazıyor. Ancak sorusuna "manuel düzeltme" yanıtı alırsa, program sonlanıyor.

Program kendi düzeltemeyeceği bir hata ile karşılaştığında ise kullanıcıya bir prompt vererek hatanın hangi satırda olduğunu bildirerek, programı sonlandırıyor. Örneğin eğer bir satırda bit aralığı ile bit uzunluğu değerleri uyumşuyor ise, program bu satırı yazdıktan sonra bu uyumsuzluğu kullanıcıya bildirip programı sonlandırıyor. Bu durumda kullanıcı çıktı dosyasına açıp mevzubahis hatayı düzeltmeli. Ardında program aşağıdaki komut ile kaldığı yerden çalışmaya devam ettirilebilir:

```
python3 dedocumenter.py -c -i <girdi dosyası> -o <çıktı dosyası> -l <devam satırı> <son satır> -r <rezerve değişken sayısı> -b <byte sayısı> -p
```

-c continue yani devam seçeneğini ifade ediyor. Bu seçenekte çıktı dosyası silinmiyor, çıktılar dosyanın sonuna ekleniyor.

Bu komutta girilen girdi ve çıktı dosyaları ilk komutta verilen ile aynı olmalı. Bittabii eğer girdi ve çıktı dosyaları aynıysa, çıktı dosyainı vermeye gerek yok.

-l flagına verilen devam satırı, hatalı satırdan sonraki ilk satır veya ilk anlamlı satır olmalı.

-r dökümanda "Reserved" olarak adlandırılan rezerve değişken sayısı. Rezerve değişkenlerin adlarında bir çakışma olmaması için, program adlarının sonuna sayı ekliyor. Mesela 5. rezerve değişkeni "Reserved5" olarak adlandırıyor. Dolayısıyla program devam modunda çalıştırıldığında çıktıya geçirilmiş olan rezerve değişken sayısı da verilmeli. Yazılan son rezerve değişkenin numarası ne ise devam modunda -r flagıyla girilmeli.

-b flagına, işlenmiş olan byte sayısı yazılmalı. Direkt

tamamıyla işlenmiş olan son byte'ın numarasına bir ekleyerek hesaplayabilirsiniz. Not henüz tüm bitleri işlenmemiş byte'ları işlenmiş kabul etmiyoruz.

Eğer ilk komutta -p flagı girildiyse bu komutta da girilmeli. Eğer girilmezse, mesaj paketi tamamen aktarıldıktan sonra paket structure'ını sonlandıran süslü parantez (}) ve structure adı kullanıcı tarafından manuel olarak yazılmalı.

Not: Eğer bit bit tanımlanan bir byte'ta hata çıkarsa, o byte'ın yazılan bitlerinin çıktı dosyasından silinip, hata düzeltildikten sonrasında, byte'ın başlangıç satırından itibaren tekrardan yazdırılması tavsiye edilir.

Özellikler:

1. Değişken adındaki yasak karakterler siliniyor.
2. Değişken adındaki Türkçe karakterler dönüştürülüyor.
3. Rezerve değişkenler numaralandırılıyor.
4. Byte numaralandırmasında dökümanda bir tutarsızlık varsa, program kullanıcıyı bilgilendirip, C kodunda doğru numaralandırıyor.
5. Bit numaralandırılmasında dökümanda bir hata varsa program kullanıcıyı bilgilendirip, aktarım işlemini durduruyor.
- 5.a. Byte sütunu hatasında kullanıcıya manuel ve otomatik düzeltme seçeneği sunuyor.
- 5.b. Bit sütunu, length sütunu hatalarında hatayı kullanıcıya bildirip, programı sonlandırıyor.
6. Uygun flag girilirse, mesaj paketi structure'ı program tarafından başlatıp bitirilebiliyor.

Nasıl Geliştirilebilir:

1. Programın ilk ve son satırının kullanıcı tarafından komut satırına girilmesi lazım. Son satırı program kendi tanıyacak şekilde geliştirilebilir.
2. İşlenecek excel belgesinin program ile aynı klasörde bulunması gerekiyor. Program herhangi bir adresteki belgeyi işleyecek şekilde geliştirilebilir.
3. Bir hata ile karşılaştığında, hatayı düzeltmek için daha çok komut satırı opsiyonu sunulabilir.
4. Hata durumunda programı sonlandırmak yerine, program komut satırına "devam" emri girilene dek beklemeye alabilir.