

İSİM

bzip2 – blok sıralamalı dosya sıkıştırıcı v1.0.2
bunzip2 – sıkıştırılmış dosyaları açar
bzcat – dosyaları standart çıktıda açar
bzip2recover – zarar görmüş bzip2 dosyalarındaki verileri kurtarır

KULLANIM

```
bzip2    [ -cdfkqstvwVL123456789 ] [ dosyalar ... ]
bunzip2 [ -fkvsVL ] [ dosyalar ... ]
bzcat   [ -s ] [ dosyalar ... ]
bzip2recover dosyaismi
```

AÇIKLAMA

bzip2, Burrows–Wheeler blok sıralamalı metin sıkıştırma algoritmasını ve Huffman kodlamasını kullanarak dosyaları sıkıştırır. Sıkıştırma oranı geleneksel LZ77/LZ78 tabanlı sıkıştırma araçlarından çok daha iyidir. PPM ailesi istatistiksel sıkıştırıcıların performansına oldukça yakındır.

Komut satırı seçenekleri kasıtlı olarak GNU **gzip**'e oldukça benzer şekilde yapılmıştır, fakat birebir aynı değildir.

bzip2, komut satırı seçeneklerine eşlik edecek dosya isimlerinin listesini ister. Her bir dosya sıkıştırılmış sürümü ile değiştirilir ve ismi "*orjinal_dosya.bz2*" şekline dönüştürülür. Her bir sıkıştırılmış dosya, orjinal dosya ile aynı izinlere, aynı düzenlenme tarihlerine ve mümkün ise aynı iyeliklere sahiptir. Bu sayede sıkıştırılmış dosya açıldığı zaman orjinal dosya elde edilmiş olur. Dosya isimlerini yönetmek işi, orjinal dosya ismini koruyamayacak, izinler ve iyelikleri veya tarihleri kendi dosya sistemi içinde bulunduramayacak dosya sistemlerinin (mesela MS–DOS) olduğu düşüncesi ile yürütülmektedir. Muhtemel dosya ismi uzunluğu kısıtlamalarını da unutmamak gerekir.

bzip2 ve **bunzip2** öntanımlı olarak, varolan dosyanın üzerine yazmazlar. Şayet bunu yapmak isterseniz **-f** seçeneğini kullanmanız gerekir.

Şayet dosya ismi belirtilmezse, **bzip2** standart girdiden standart çıktıya sıkıştırma işlemi yapar. Bu durumda, **bzip2** sıkıştırılmış çıktıyı bir terminale yazmayı reddedecektir, bu nedenle böyle bir şey yapmak tamamen anlamsız ve amaçsız olacaktır.

bunzip2 (veya **bzip2 -d**) belirtilen bütün dosyaları açar. **bzip2** tarafından sıkıştırılmamış dosyalar tespit edilir ve yok sayılır, ardından bir uyarı mesajı verilir. **bzip2**, sıkıştırılmış dosyadan açılacak dosyaya aktarılacak dosya ismini şu şekilde tespit etmeye çalışır:

```
dosyaismi.bz2    =>    dosyaismi
dosyaismi.bz     =>    dosyaismi
dosyaismi.tbz2   =>    dosyaismi.tar
dosyaismi.tbz    =>    dosyaismi.tar
başkabirisim     =>    başkabirisim.out
```

Şayet dosya yukarıda belirtilen soneklerden birisine sahip değilse, **bzip2** mızdırılır ve orijinal dosya ismini sonuna *.out* ekleyerek kullanır.

Sıkıştırma işleminde olduğu gibi, dosya ismi belirtmeden yapılan açma işleminde standart girdiden standart çıktıya açma işlemi yapılır.

bunzip2, iki ya da daha fazla sıkıştırılmış dosyadan oluşan birleşik bir dosyayı doğru bir şekilde açacaktır. Sonuçta açılmış dosyalar birleşik olarak elde edilir. Birleşik sıkıştırılmış dosyaların bütünlük sınaması yapılması (**-t**) özelliği de desteklenmektedir.

Ayrıca **-c** seçeneğini belirterek standart çıktıda dosyaları sıkıştırabilir veya açabilirsiniz. Çoklu dosyalar bu şekilde sıkıştırılabilir veya açılabilir. Sonuç çıktıları standart çıktıya peşpeşe gönderilir. Bu şekilde çoklu dosya sıkıştırma çoklu sıkıştırılmış dosya temsilcilerinin bulunduğu bir akım üretir. Bu tür bir akım sadece bzip 0.9.0 veya daha sonraki bir sürümle düzgün bir şekilde açılabilir. Daha eski sürümler ilk dosyayı açtıktan sonra dururlar.

bzcat (veya **bzip2 -dc**) belirtilen bütün dosyaları standart çıktıya açar.

bzip2, argümanları sırayla **BZIP2** ve **BZIP** ortam değişkenlerinden okurlar ve bu argümanları komut satırı argümanlarını okumadan önce uygularlar. Bu, öntanımlı argümanların uygulanması için elverişli bir yol sağlar.

Sıkıştırma işlemi, sıkıştırılmış dosya orijinalinden biraz büyük olsa bile, mutlaka yapılır. Yüz bayttan daha küçük dosyalar şişmanlamaya meyillidir. Çünkü sıkıştırma algoritması ilk 50 bayta yazılacak bir sabite sahiptir. Rasgele veri (pek çok dosya sıkıştırıcısının çıktısı dahil) bayt başına 8.05 bitlik bir oran ile kodlanır. Bu da %0.5lik bir genişlemeye sebep verir.

Bir özdenetim mekanizması olarak, **bzip2** 32 bitlik CRC kullanır ve açılan dosya ile orijinal sıkıştırılmış dosyanın aynı olup olmadığını kontrol eder. Bu tedbir sıkıştırılmış dosyada meydana gelebilecek bozulmalar ve **bzip2**'de olması muhtemel yazılım hatalarına karşı güvenlik sağlar. Elden geçirilen her bir dosya için tespit edilemeyecek hata oranı 4 milyarda bir değişiklik oranının altındadır. Bu orandan daha küçük değişiklikler mikroskobiktir ve tespit edilemeyebilir. Bununla birlikte uyanık olmakta fayda vardır, kontrol açma işlemi esnasında yapılır ve sadece bir takım şeylerin yanlış olduğunu söyler. Orijinal açılmış verileri kurtarmanıza yardımcı olmaz. Bu durumda, verileri kurtarmak için **bzip2recover** kullanmalısınız.

Geri dönen değerler

Normal çıktı için 0, çevresel sorunlar için 1 (dosya bulunamaması, geçersiz seçenekler, G/Ç hataları, vs..), sıkıştırılmış dosyada bozukluk için 2, bzip2'nin paniklemesine sebep olan dahili bir tutarlılık hatası için 3 (yazılım hatası).

SEÇENEKLER

-a

--ascii

Ascii metin kipi: satır sonlarını yerel ayarlara göre dönüştürür. Bu seçenek, sadece Unix dışı birkaç sistemde desteklenir. Örneğin MSDOS'da sıkıştırma esnasında CRLF'ler LF'ye çevrilir ve açma işleminde LF'ler CRLF'ye dönüştürülür.

-c

--stdout

Standart çıktıda sıkıştırılır veya açar.

-d

--decompress

Mutlaka açma işlemi yapılır. **bzip2**, **bunzip2** ve **bzcat** aslında aynı yazılımdır ve yapılması gereken işlem, kullanılan yazılım ismine göre kararlaştırılır. Bu seçenek, bu yöntemi yoksayar ve **bzip2**'yi açma işlemine zorlar.

-z

--compress

-d nin zıddı: hangi isim ile çağrılırsa çağrılın mutlaka sıkıştırma yapılır (örneğin, **bunzip2** komutuna bu seçeneği vererek sıkıştırma yapmasını sağlayabilirsiniz).

-t

--test

Belirtilen dosya veya dosyaların bütünlüğünü kontrol eder, gerçekte açma işlemi yapmaz. Bu seçenek ile bir deneme açılımı yapılmış olur ve sonuçlar belirtilir.

-f

--force

Çıktı dosyasının üzerine yazmaya zorlar. Normal olarak, **bzip2** varolan bir dosyanın üzerine yazmayacaktır. Ayrıca bu seçenek **bzip2**'yi dosyalara olan sabit bağları kırmaya zorlar, böylece bu bağlarda üzerine yazma yapılmaz.

bzip2 normalde doğru sihirli başlık baytlarına sahip olmayan dosyaları açmayı rededecektir. Şayet **-f** kullanılır ise, bu dosyaları değiştirilmeden geçecektir. GNU **gzip** de bu şekilde davranır.

-k

--keep

Sıkıştırma veya açma işlemi esnasında orjinal dosyaları silmez, saklar.

-s

--small

Sıkıştırma, açma ve sınaıma işlemleri için bellek ihtiyacını azaltır. Dosyalar bayt bloğu başına 2.5 bayt gerektiren deęişiklik yapılmış bir algoritma ile açılır ve sınanır. Bunun anlamı, herhangi bir dosyanın 2300k lık bir bellek alanı içinde açılabilceğidir ancak başarıım, normal hızın yarısına kadar düşer.

Sıkıştırma esnasında, **-s** seçeneęi ile 200k lık blok boyutu seçilir, bu belleęi de aynı şekilde sınırlar ve sıkıştırma oranını artırır. Uzun lafın kıtası, makinenizin belleęi 8MB'dan daha az ise **-s** seçeneęini her şeyde kullanın. Aşağıdaki [BELLEK YÖNETİMİ](#) (sayfa: 4) bölümüne bakınız.

-q

--quiet

Gerekli olmayan uyarılar verilmez. G/Ç ve dięer önemli hata iletileri gizlenmez.

-v

--verbose

Ayrıntı kipi: işlenen her bir dosya için sıkıştırma oranlarını gösterir. Daha fazla **-v** ayrıntı oranını artıracaktır. Teşhis amaçlı kullanıma uygun olarak pek çok bilgi yazılım tarafından bildirilecektir.

-L

--license

-V

--version

Yazılımın sürüm, lisans, koşul ve kurallarını görüntüler.

-1 (veya **--fast**) den **-9** (veya **--best**) a kadar

Sıkıştırma yaparken blok boyutlarını 100k, 200k, ..., 900k şeklinde ayarlar. Çözme işleminde bir etkisi yoktur. Aşağıdaki [BELLEK YÖNETİMİ](#) (sayfa: 4) bölümüne bakınız. **--fast** ve **--best** seçenek isimleri GNU **gzip** uyumluluęunu saęlamak içindir. Açıkçası, **--fast** gözle görölür bir hız artışı saęlamaz ve **--best** sadece öntanımlı deęerlerin seçilmesini saęlar.

--

Ardından belirtilen bütün argümanları dosya isimleri olarak kabul eder. Bu argümanların tek çizgi ile başlamış olmaları bile önemli deęildir. Böylece tek çizgi ile başlayan dosya isimlerini kullanabilirsiniz. Örneęin: **bzip2 -- -benim_dosyam**.

--repetitive-fast

--repetitive-best

Bu seçenekler sürüm 0.9.5 ve üzerinde gereksizdir. Daha eski sürümlerde, bazı kaba kontrollerin yapılmasını sağlamaya yönelik seçeneklerdir. Bazen faydalı olabilmekte idi. 0.9.5 ve üstü sürümler bu ayarların yapılmasına gerek bırakmayan daha iyi algoritmalar kullanmaktadırlar.

BELLEK YÖNETİMİ

bzip2 büyük dosyaları bloklar halinde sıkıştırır. Blok boyutları, hem sıkıştırma oranını, hem de sıkıştırma/açma için gereken bellek miktarını etkiler. **-1** den **-9** a kadarki seçenekler ile istenen blok boyutlarını 100.000 bayttan 900.000 bayta (varsayılan değer budur) kadar tanımlamasını sağlar. Açma işlemi sırasında, sıkıştırılmış dosyanın baş kısmından sıkıştırma esnasında kullanılan blok boyutu okunur ve daha sonra **bunzip2** kendisi için gerekli olan bellek alanını ayırır ve oraya yerleşir. Sıkıştırma blok boyutları sıkıştırılmış dosya içinde saklandığı için, **-1** den **-9** a kadar olan seçenekler açma işlemi esnasında geçersizdir. Şayet bu seçenekleri belirtirseniz, program bunları gözardı edecektir.

Sıkıştırma ve açma gereksinimleri, bayt cinsinden, yaklaşık olarak:

Sıkıştırma: $400k + (8 \times \text{blok boyu})$

Açma: $100k + (4 \times \text{blok boyu})$ veya
 $100k + (2.5 \times \text{blok boyu})$

Daha büyük blok boyutları hızlı bir şekilde azalan sınıra yakın dönüşler verir. Sıkıştırmanın pek çoğu, blok boyutunun ilk iki veya üç yüz k lık kısmından elde edilir. Bu kuralı, küçük makinelerde **bzip2** kullanırken aklınızdan çıkarmayın. Ayrıca aklınızdan çıkarmamanız gereken diğer bir noktada, açma sırasındaki bellek gereksinimlerinin, sıkıştırma sırasında kullandığınız blok boyutları ile birebir alakalı olduğudur.

900k lık öntanımlı blok boyutu ile sıkıştırılan dosyalar için, **bunzip2** açma işlemi sırasında 3700 kB lık belleğe ihtiyaç duyacaktır. 4 MB lık belleğe sahip bir makinede herhangi bir dosyayı açmak için, **bunzip2**, ihtiyaç duyulan bellek miktarının yarısı ile (takriben 2300KB) açma işlemini gerçekleştirecek bir argümana sahiptir. Bu argümanın kullanılması açma hızını da yarı yarıya düşürecektir. Çok gerekli olmadıkça bu seçeneği (**-s**) kullanmayın. (*Ç.N: günümüzde kullanılan makineler için bu seçenek oldukça nostaljik görünebilir ama test ve eğitim amaçlı oldukça düşük özelliklere sahip ikinci veya üçüncü makineleri olanlar bu seçeneğin değerini anlayacaktır. P233 makinede 16 MB edoram ile harikalar :-)) yaratıyor.)

Genel kural olarak, bellek sınırlarınız elverdiği sürece, mümkün olan en büyük blokları kullanın. Böylece azami sıkıştırma elde edilir. Sıkıştırma ya da açma hızının aslında blok boyutu ile bir ilgisi yoktur.

Diğer bir önemli nokta da dosyalara tek bir bloğa sığacak dosyalara yapılan işlemdir. Aslında karşılaşılan pek çok dosya bir bloktan küçüktür. Dosya bir bloktan küçük olduğundan, gerçekte kullanılan bellek miktarı normalde gerekenden küçük olur. Örneğin: 20.000 baytlık bir dosyayı **-9** seçeneği ile sıkıştırmak, sıkıştırıcının 7600k belleğe ihtiyaç duymasına sebep olacaktır ama gerçekte sadece $400k + 20000 \times 8 = 560k$ bellek kullanılacaktır. Aynı şekilde açma işlemi de 3700k lık bir bellek ayıracak ama sadece $100k + 20000 \times 4 = 180k$ lık bir alanı kullanacaktır.

Aşağıda değişik blok boyutlarının bellek kullanımını gösteren bir tablo bulacaksınız. Ayrıca Calgary Text Compression Corpus'da (Calgary Metin Sıkıştırma Külliyyatı) 14 dosyalık sıkıştırılmış toplam boyut olarak 3.141.622 bayt kayıtlıdır. Aşağıdaki sütunlar, blok boyutu ile sıkıştırma arasındaki ilgi hakkında bir fikir verecektir. Büyük dosyalar için büyük blok boyutlarını kullanmak daha avantajlıdır. Çünkü Külliyyat daha küçük dosyalar üzerine kuruludur.

| | Sıkıştırma | Açma | -s ile açma | Külliyyat |
|---------|------------|-----------|-------------|-----------|
| Seçenek | kullanımı | kullanımı | kullanımı | boyutu |

| | | | | |
|----|-------|-------|-------|--------|
| -1 | 1200k | 500k | 350k | 914704 |
| -2 | 2000k | 900k | 600k | 877703 |
| -3 | 2800k | 1300k | 850k | 860338 |
| -4 | 3600k | 1700k | 1100k | 846899 |
| -5 | 4400k | 2100k | 1350k | 845160 |
| -6 | 5200k | 2500k | 1600k | 838626 |
| -7 | 6100k | 2900k | 1850k | 834096 |
| -8 | 6800k | 3300k | 2100k | 828642 |
| -9 | 7600k | 3700k | 2350k | 828642 |

HASARLI DOSYADAN VERİ KURTARMA

bzip2 dosyaları genellikle 900k lık bloklar halinde sıkıştırır. Her bir blok birbirinden bağımsız olarak ele alınır. Bir aygıt ya da aktarım hatası pek çok **.bz2** bloğunun hasar görmesine sebep olursa, zarar görmemiş bloklardaki verileri kurtarmak mümkün olabilir.

Sıkıştırılmış her bir blok 48 bitlik bir şablon ile temsil edilir. Böylece blok sınırlarını kesin olarak bulmak mümkün olabilir. Ayrıca her bir blok kendi 32 bitlik CRC'sini bulundurur, böylece bozuk bloklar sağlam olanlardan ayırt edilebilir.

bzip2recover, basitçe, **.bz2** dosyaların içindeki blokları tarayan bir yazılımdır. Buradaki her bir taranmış bloğu kendi **.bz2** dosyasına aktarır. Sonuçların bütünlüğünü kontrol etmek için **bzip2 -t** komutunu kullanabilirsiniz. Ardından bozulmamış verileri açabilirsiniz.

bzip2recover sadece hasarlı dosya ismini argüman olarak alır ve kurtardığı blokları "rec00001file.bz2", "rec00002file.bz2" şeklinde yazar. Bu isim numaralandırması sonradan yapılacak işlemlerdeki simgelemlerin kullanımı içindir. Örneğin, **bzip2 -dc rec*file.bz2 > kurtarılan_veri** komutu dosyaları doğru sıralama ile üretir.

bzip2recover daha çok büyük **.bz2** dosyaları için faydalıdır çünkü bunlar daha çok blok içerirler. Tek bloktan oluşmuş hasarlı bir dosya ile bu yazılımı kullanmanın hiçbir anlamı yoktur, çünkü bozuk blok kurtarılamaz. Şayet herhangi bir hasar durumunda zararınızı en azda tutmak istiyorsanız, daha küçük blok boyutları kullanmanız akıllıca olacaktır.

BAŞARIM HAKKINDA

Sıkıştırmanın sınıflandırma aşaması, birbirine benzer dizgeleri bir araya getirmektir. Bu nedenle tekrarlanan sembollerden oluşmuş dosyaların sıkıştırması (aabaabaabaab ... gibi) daha uzun tutabilir. Sürüm 0.9.5 ve üzeridekilerde başarımlar daha iyidir. En kötü durum ile ortalama durum sıkıştırma zamanı arasındaki oran 10:1 civarındadır. Daha önceki sürümlerde bu durum 100:1 gibiydi. İsterseniz sürecin bütün ayrıntılarını görebilmek için **-vvv** seçeneğini kullanabilirsiniz.

Açma hızı bu durumdan bağımsızdır.

bzip2 genellikle işlem yapmak için, çok MB lı bellek alanları ayırır ve daha sonra tarafsız ve rasgele bir şekilde saldırır. Bunun anlamı, açma ve sıkıştırma performanslarının makinenizin önbelleğine erişim süresi ile alakalı olduğudur. Önbelleğe erişim kayıplarını düşürecek küçük kod değişiklikleri büyük performans artışları getirebilir. Çok büyük önbelleklere sahip makinelerde **bzip2** çok yüksek başarımlar göstermektedir.

YETERSİZLİKLER

G/Ç hata iletileri olmaları gerektiği kadar faydalı değildiler. **bzip2** G/Ç hatalarını tespit etmek ve temiz bir şekilde çıkmak için oldukça ter döker ama bazen sorunun ayrıntıları gözden kaçabilmektedir.

Bu kılavuz sayfası **bzip2**'nin 1.0.2 sürümüne aittir. Bu sürüm ile sıkıştırılmış veriler geçmiş sürümler ile tamamen uyumludur (sürüm 0.1pl2, 0.9.0, 0.9.5, 1.0.0 ve 1.0.1 ile), tek istisna 0.9.0 ve üzeri sürümler

çoklu dosya sıkıştırılmaları düzgünce açabilirler, 0.1p12 bunu yapamaz, ilk dosyayı açtıktan sonra duracak ve akımın gerisini getirmeyecektir.

bzip2recover sürümleri 1.0.2 den önceki sürümlere göredir. Sıkıştırılmış dosya içindeki bit konumlarını belirtmek için 32 bitlik tamsayılar kullanılır ve bu yüzden 512 MB'den daha büyük sıkıştırılmış dosyalar elde edilemezler. Sürüm 1.0.2 ve üzeri 64 bitlik tam sayılar kullanılır ve GNU ile Windows platformlarını desteklerler. Bu tip bir sınır ile kurulup kurulmadığını anlamak için **bzip2recover**'i argümentsiz çalıştırın. Her halükarda, bu sınırı ortadan kaldırmak mümkündür. Şayet yazılımı `MaybeUInt64` ile yeniden derleyebilirsiniz, işaretli 64 bitlik tamsayı ayarlanmış olacağından bu sınır ortadan kalkmış olur.

YAZAN

Julian Seward, <jseward@acm.org> tarafından yazılmıştır.

<http://sources.redhat.com/bzip2>

bzip2'deki fikirlerin sahipleri şunlardır (en azından): Michael Burrows ve David Wheeler (blok sıralamalı dönüşüm için), David Wheeler (tekrar, Huffman kodlayıcısı), Peter Fenwick (orijinal bzip'teki yapısal kodlama modeli ve pek çok tasfiye için) ve Alistair Moffat, Radford Neal ve Ian Witten (orijinal bzip'teki aritmetik kodlayıcı için). Onlara yardımları, destek ve tavsiyeleri için şükran borçluyum. Belgeler için kaynak kod paketine bakınız. Christian von Roques, daha hızlı sıralama algoritması için beni cesaretlendirdi ve böylece daha hızlı bir sıkıştırma elde edebildik. Bela Lubkin, en kötü durum sıkıştırması için cesaret verdi. **bz*** betikleri GNU **gzip**'inkilerden türetildi. Bir çok kimse de, yamalar gönderdi, taşınabilirlik sorunlarında yardımcı oldu, tavsiyelerde bulundu; bunlar çoğunlukla faydalı oldu.

ÇEVİREN

Yalçın Kolukısa <yalcink01@yahoo.com>, Ocak 2004

YASAL UYARI

Bu çevirinin telif hakkı yukarıda belirtilen çevirmen(ler)e aittir. Özgün belgenin telif hakkı ve lisans bilgileri varsa ve belge içinde belirtilmemişse belge sonunda belirtilmiş olacaktır. Bu çevirinin lisansı, özgün belge için belirtilmiş bir lisans varsa ve bu lisans çevirinin de aynı lisansa sahip olmasını gerektiriyorsa onunla aynıdır, yoksa GNU GPL lisansı ve her iki durumda da ek olarak aşağıdaki koşullar geçerlidir. GNU GPL lisansı <<http://www.gnu.org/licenses/gpl.html>> adresinden edinilebilir.

BU BELGE ÜCRETSİZ OLARAK RUHSATLANDIĞI İÇİN, BELGENİN İÇERDİĞİ BİLGİLERİN VEYA KODLARIN NİTELİKLERİ İÇİN İLGİLİ KANUNLARIN İZİN VERDİĞİ ÖLÇÜDE HERHANGİ BİR GARANTİ VERİLMEMEKTEDİR. AKSİ YAZILI OLARAK BELİRTİLMEDİĞİ MÜDDETÇE TELİF HAKKI SAHİPLERİ VE/VEYA BAŞKA ŞAHISLAR BELGELERİ "OLDUĞU GİBİ", AŞIKAR VEYA ZIMNEN, SATILABİLİRLİĞİ VEYA HERHANGİ BİR AMACA UYGUNLUĞU DA DAHİL OLMAK ÜZERE HİÇBİR GARANTİ VERMEKSİZİN DAĞITMAKTADIRLAR. BELGELERİN KALİTESİ VEYA PERFORMANSI İLE İLGİLİ TÜM SORUNLAR SİZE AİTTİR. HERHANGİ BİR HATA VEYA EKSİKLİKTEN DOLAYI DOĞABİLECEK OLAN BÜTÜN SERVİS, TAMİR VEYA DÜZELTME MASRAFLARI SİZE AİTTİR.

İLGİLİ KANUNUN İCBAR ETTİĞİ DURUMLAR VEYA YAZILI ANLAŞMA HARİCİNDE HERHANGİ BİR ŞEKİLDE TELİF HAKKI SAHİBİ VEYA YUKARIDA İZİN VERİLDİĞİ ŞEKİLDE BELGEYİ DEĞİŞTİREN VEYA YENİDEN DAĞITAN HERHANGİ BİR KİŞİ, BELGENİN İÇERDİĞİ BİLGİNİN KULLANIMI VEYA KULLANILAMAMASI (VEYA VERİ KAYBI OLUŞMASI, VERİNİN YANLIŞ HALE GELMESİ, SİZİN VEYA ÜÇÜNCÜ ŞAHISLARIN ZARARA UĞRAMASI VEYA BİLGİNİN BAŞKA BİLGİLERLE UYUMSUZ OLMASI) YÜZÜNDEN OLUŞAN GENEL, ÖZEL, DOĞRUDAN YA DA DOLAYLI HERHANGİ BİR ZARARDAN, BÖYLE BİR TAZMİNAT TALEBİ TELİF HAKKI SAHİBİ VEYA İLGİLİ KİŞİYE BİLDİRİLMİŞ OLSA DAHİ, SORUMLU DEĞİLDİR.

bzip2(1)

Bu dosya (man1-bzip2.pdf), belgenin XML biçiminin \TeX Live ve belgeler-xsl paketlerindeki araçlar kullanılarak PDF biçimine dönüştürülmesiyle elde edilmiştir.

19 Ocak 2007