

Fatih Sultan Mehmet Vakıf Üniversitesi

Bilgisayar Mühendisliği

Hazırlayan Mevlüt Emre TANRIVERDİ

Okul NO 1821221023

Ders Bilgisayar Programlama III

Konu LAB – Ödev 2
(Seyrek matris sıkıştırma
algoritmaları)



Amaç

C'de dizi ve işaretçi (int: pointer) işlemlerini kullanmak ve kütüphane oluşturmak

Yöntem

İlk olarak:

Main() dışında

"**compress.h**" isimli kütüphane ve "**compress.c**" adlı source file oluşturduk.

Main dosyasında #define X ve #define Y olarak verdiğimiz ölçütlerde random sayılardan oluşan A[X][Y]; iki boyutlu matrisi oluşturulmuştur.

Daha sonra bu matrisi dönerek matriste yer alan 0 dışındaki numaraların adetini nsize olarak kaydedilmiştir.

Daha sonra tek boyutlu nz[nsize] oluşturulmuştur. Bu dosyaya da A[X][Y] de yer alan 0 dışı sayıları kaydedilmiştir.

Devamında:

1. Sıkıştırılmış Satır Saklama Algoritması:

nz ile birlikte iki tane daha tek boyutlu dizi oluşturulur.
Bunlardan biri *nz*'deki elemanların **sütun** indislerinden oluşan *col_ind*
Diğeri ise e *A* matrisinin her bir **satırının** sıfırdan farklı ilk
elemanının *nz*'deki indisini tutan *row_ptr* dizisini oluşturulmuştur

2. Sıkıştırılmış Sütun Saklama Algoritması:

nz ile birlikte iki tane daha tek boyutlu dizi oluşturulur.
Bunlardan biri *nz*'deki elemanların **satır** indislerinden oluşan *row_ind*
Diğeri ise e *A* matrisinin her bir **sütunun** sıfırdan farklı ilk
elemanının *nz*'deki indisini tutan *col_ptr* dizisini oluşturulmuştur

3. Koordinat Sıkıştırma Algoritması:

compress.c dosyasına:

```
void compress_IJ(int **A, int *nz, int *rows, int *cols);  
int **decompress_IJ(int *nz, int *rows, int *cols);  
void compress_CRS(int **A, int *nz, int *col_ind, int *row_ptr);  
int **decompress_CRS(int *nz, int *col_ind, int *row_ptr);  
void compress_CCS(int **A, int *nz, int *row_ind, int *col_ptr);  
int **decompress_CCS(int *nz, int *row_ind, int *col_ptr);
```

methodlarını oluşturulmuştur.

compress.h kütüphanesine:

```
#define COMPRESS_H_  
void compress_IJ(int **A, int *nz, int *rows, int *cols);  
int **decompress_IJ(int *nz, int *rows, int *cols);  
void compress_CRS(int **A, int *nz, int *col_ind, int *row_ptr);  
int **decompress_CRS(int *nz, int *col_ind, int *row_ptr);  
void compress_CCS(int **A, int *nz, int *row_ind, int *col_ptr);  
int **decompress_CCS(int *nz, int *row_ind, int *col_ptr);  
methodlarını eklenmiştir.
```

Yorum

Bu yöntemlerle oluşturulmuş koordinatlarda ihtiyacımız olmayan yerleri sıkıştırarak daha az yer kaplamasını sağlayabiliyoruz. Sadece yer açmakla kalmayıp işlem sürelerini de hızlandırmış oluyoruz.