

ذخیره و بازیابی داده ها

مدیریت فایل از رکوردها

احمد خادم زاده
khademzaeh@mail.iust.ac.ir
 دانشگاه آزاد اسلامی واحد مشهد - بهار ۱۳۸۳

فایل بصورت جریانی از بایتها

- بطور کلی و با یک نگاه ساده می توان فایل را بصورت جریانی از بایتها در نظر گرفت که پشت سر هم قرار گرفته اند.
- می توانیم این جریان بایتها را با قطعه برنامه ای (مانند زیر) بخوانیم:

```
while (! infile.fail()) {
    cout <<ch;
    infile >>ch;
}
```

- اگر بخوایم در یک فایل اطلاعات را بصورت رکورد و فیلدهای آن ذخیره کنیم ، در زمان خواندن اطلاعات (که این خواندن بصورت جریانی از بایتها صورت گیرد) ، چگونه مرز بین فیلدها را تشخیص دهیم ؟ چگونه مرز بین رکوردها را تشخیص دهیم و متوجه شویم اطلاعات یک رکورد خاص به اتمام رسیده و اطلاعات رکورد جدیدی را داریم می خوانیم ؟

khademzadeh@iust.ac.ir

یک فایل که بصورت جریانی از بایت نوشته شده است.

- فرض کنیم محتویات فایل A.TXT بصورت زیر باشد :

```
87358CARROLLALICE IN WONDERLAND <n1>
03818FOLK FILE STRUCTURES <n1>
79733KNUTH THE ART OF COMPUTER PROGR<n1>
86683KNUTH SURREAL NUMBERS <n1>
18395TOLKIEN THE HOBITT <n1>
```

- در این فایل از نماد <n1> به عنوان کاراکتر خط جدید استفاده شده است.

khademzadeh@iust.ac.ir

خواندن اطلاعات این فایل بصورت جریانی از بایتها

- در این جریان از بایتها موقعیت جاری در هر لحظه ، مشخص می باشد.
- در زمان باز کردن فایل ، اشاره گر به ابتدای فایل اشاره می کند.
- در زمان صدور اولین دستور `ch >> infile` کاراکتر 8 از این فایل را خوانده و به متغیر `ch` منتقل نسبت خواهد داد و اشاره گر موقعیت جاری در فایل را یک واحد به جلو خواهد برد.
- دستور بعدی `ch >> infile` کاراکتر 7 از فایل را خوانده و به متغیر `ch` نسبت خواهد داد و اشاره گر داخل فایل را یک واحد به جلو خواهد برد.

khademzadeh@iust.ac.ir

خواندن اطلاعات این فایل بصورت جریانی از بایتها

- ۳۸ امین دستور `ch >> infile` کاراکتر <n1> را خواهد خواند و به `ch` نسبت خواهد داد. و همچنین اشاره گر داخل فایل
- ۳۹ امین دستور `ch >> infile` کاراکتر 0 را خوانده و به `ch` نسبت خواهد داد. و همچنین موقعیت جاری در فایل را به اندازه یک واحد افزایش خواهد داد. و ...

khademzadeh@iust.ac.ir

خواندن اطلاعات این فایل بصورت جریانی از بایتها

- همانطور که مشاهده شد ، با توجه به اینکه توانستیم تمام اطلاعات فایل را بخوانیم ، اما سیاستی در مورد اینکه چه زمان یک فیلد تمام شده و فیلد جدیدی شروع شده ، نداشتیم. بنابراین نمی توانیم مرز بین فیلدها را تشخیص دهیم.
- در مورد رکوردها هم این نکته صادق است.
- پس نیاز است که فیلدها و رکوردها را طوری سازماندهی کنیم تا در زمان خواندن اطلاعات فایل ، بتوانیم مرز بین آنها را تشخیص دهیم.

khademzadeh@iust.ac.ir

ساختار فیلد و رکورد

- **رکورد** : مجموعه ای از فیلدهای مرتبط به هم را رکورد می نامیم.
- **فیلد** : کوچکترین واحد منطقی و معنی دار از اطلاعات در یک فایل را فیلد می نامیم.
- **کلید** : مجموعه ای از یک یا چند فیلد که برای هر رکورد ، با سایر رکوردها متفاوت می باشد و یک رکورد خاص را مشخص می کند.

khademzadeh@iust.ac.ir

سازماندهی فایل بصورت فیلدها و رکوردها

- پس یک فایل را می توان به دو صورت زیر داشت :
- همانطور که در بالا مشاهده شده ، به صورت جریانی از بایتها ، یا
- به صورت مجموعه ای از رکوردها که شامل فیلد می باشند ،

khademzadeh@iust.ac.ir

ساختار فیلد و رکورد – مثال

- کلید ثانویه (Secondary key) : کلید های دیگری که ممکن است از آنها جهت انجام عمل جستجو در فایل استفاده شود. مثالی از کلید ثانویه برای فایل A.TXT چه می باشد ؟
- در حالت کلی ، کلید شامل یک فیلد از فیلدهای فایل نمی باشد و ممکن است ، ترکیبی از فیلدها ، کلید را تشکیل دهند.

khademzadeh@iust.ac.ir

ساختار فیلد و رکورد – مثال

- در فایل نمونه A.TXT که اطلاعاتی در مورد چند کتاب اطلاعاتی را نگهداری می کند :
- هر خط از فایل به یک کتاب خاص نسبت داده شده است و یک رکورد نامیده می شود.
- هر یک از مقادیر موجود در هر یک از رکوردها (خط های فایل) ، مثل شماره کتاب ، نام مولف کتاب ، و یا نام کتاب ، یک فیلد از این رکورد می باشد.
- کلید اصلی (primary key) : کلید اصلی ، یک رکورد خاص را مشخص می کند. برای فایل فوق کلید اصلی ، چه می تواند باشد ؟

khademzadeh@iust.ac.ir

ساختار فیلد

- روش دوم – می توانیم ، طول هر یک از فیلدها را در ابتدای آن قید نماییم :
- ```
058735907CARROLL19ALICE IN WONDERLAND
050381804FOLK15FILE STRUCTURES
```
- روش سوم – می توانیم جهت جدا کردن مرز بین فیلدها از یک کاراکتر جداکننده استفاده نماییم :

```
87359|CARROLL|ALICE IN WONDERLAND|
03818|FOLK|FILE STRUCTURES|
```

khademzadeh@iust.ac.ir

## ساختار فیلد

- چگونه فیلدهای یک فایل می توانند سازماندهی و از یکدیگر متمایز شوند، تا در زمان خواندن اطلاعات از فایل بتوان مرز بین فیلدها را تشخیص داد ؟
- روش اول – فیلدهای با طول ثابت :
- در فایل A.TXT می توانیم برای هر یک از فیلدها طولی ثابت در نظر بگیریم . مثلا طول فیلدها به ترتیب برابر با ۵، ۷ و ۲۵ باشد :

```
87358CARROLLALICE IN WONDERLAND
03818FOLK FILE STRUCTURES
79733KNUTH THE ART OF COMPUTER PROGR
```

khademzadeh@iust.ac.ir

## سازماندهی فیلدها – مزیت ها و معایب هر یک از روش های آرایه شده

| روشی                   | مزیت                                                                                                       | عیب                                                                                             |
|------------------------|------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| طول ثابت (Fixed)       | راحتی خواندن و ذخیره سازی فیلدها                                                                           | هدر دادن فضا حافظه                                                                              |
| با نشانگر طول          | پرش به انتهای فیلد مقدور می باشد.                                                                          | رکوردهای طولانی بیش از یک بایت جهت نگهداری طول رکورد نیاز دارند. (رکوردهای با طول بیشتر از ۲۵۵) |
| جدا شده با یک فیلد خاص | ممکن است فضای هدر رفته کمتر از روش با نشانگر طول باشد.                                                     | باید تمامی بایت با کاراکتری که جداکننده می باشد، مقایسه شوند.                                   |
| نام فیلد = مقدار       | فیلدها خود توصیفگر می باشند و اجازه می دهد تا بعضی از فیلدها را بطور کل برای بعضی از رکوردها ذخیره نماییم. | فضای هدر رفته جهت ذخیره سازی نام فیلدها زیاد می باشد.                                           |

khademzadeh@iust.ac.ir

## ساختار فیلد

- روش چهارم – هر یک از فیلدها را بصورت FieldName=Value ذخیره کنیم:
- در این روش می توانیم از کاراکتر جداکننده استفاده نکنیم.

ISBN=87359|AU=CARROLL|TI=ALICE IN WONDERLAND|  
ISBN=03818|AU=FOLK|TI=FILE STRUCTURES|

khademzadeh@iust.ac.ir

## ساختار رکورد

- روش اول – حالت دوم: استفاده از فیلدهای با طول متغیر – که در این صورت باید در انتهای هر رکورد به میزان مورد نیاز کاراکتر خالی قرار دهیم، تا به طول مورد نظر برسیم.
- استفاده از فیلدهای به طول متغیر و کاراکتر جدا کننده:

```
87359|CARROLL|ALICE IN WONDERLAND
03818|FOLK|FILE STRUCTURES
79733|KNUTH|THE ART OF COMPUTER PROGR
```

- استفاده از فیلدهای به طول متغیر با نشانگر طول در ابتدای فیلد:

```
058735907CARROLL19ALICE IN WONDERLAND
050381804FOLK15FILE STRUCTURES
```

khademzadeh@iust.ac.ir

## ساختار رکورد

- برای مدیریت و سازماندهی رکوردها می توانیم بصورت ها مختلفی عمل کنیم، در زیر این روشها را معرفی می کنیم:
- روش اول – رکوردهای با طول ثابت:
- حالت اول: استفاده از فیلدهای با طول ثابت.

```
87358CARROLLALICE IN WONDERLAND
03818FOLK FILE STRUCTURES
79733KNUTH THE ART OF COMPUTER PROGR
```

khademzadeh@iust.ac.ir

## ساختار رکورد

- روش سوم – استفاده از نشانگر طول در ابتدای هر رکورد
- مثال (جدا سازی فیلدها با استفاده از کاراکتر جدا کننده)

```
3387359|CARROLL|ALICE IN WONDERLAND
2603818|FOLK|FILE STRUCTURES
```

- آیا می توانیم این روش را در ترکیب با روشهای، فیلد دارای شاخص طول یا فیلدهای دارای زوج نام فیلد=مقدار، استفاده کنیم؟

khademzadeh@iust.ac.ir

## ساختار رکورد

- روش دوم – رکوردهای با تعداد فیلد ثابت و طول متغیر: این روش می تواند با هر یک از ساختارهای ذخیره سازی فیلد با طول متغیر ترکیب شود. به عنوان مثال یک فایل با رکوردهای ۳ فیلدی را بررسی می کنیم.
- فیلدهای جدا شده با کاراکتر جدا کنند:

```
87359|CARROLL|ALICE IN WONDERLAND|03818|FOLK|...
```

- استفاده از نشانگر طول برای فیلدها:

```
058735907CARROLL19ALICE IN WONDERLAND0503818...
```

- در دو حالت فوق چگونه از پایان یک رکورد مطلع می شویم؟

khademzadeh@iust.ac.ir

## ساختار رکورد

- روش پنجم – استفاده از یک کاراکتر جداکننده در انتهای رکورد
- کاراکتر پایان خط (eol) یک جداکننده رایج می باشد.

```
87358|CARROLL|ALICE IN WONDERLAND|<n1>
03818|FOLK|FILE STRUCTURES|<n1>
79733|KNUTH|THE ART OF COMPUTER PROGR|<n1>
```

khademzadeh@iust.ac.ir

## ساختار رکورد

- روش چهارم – استفاده از شاخص (Index) برای تعیین آدرس
- شاخص ، شماره بایت (byte offset) شروع هر یک از رکوردها را نگهداری می نماید. انجام این عمل به ما کمک می کند که با جستجو در شاخص بتوانیم آدرس شروع رکورد مورد نظر را بیابیم.

فایل داده :

```
87359|CARROLL|ALICE IN WONDERLAND|03818|FOLK|...
```

فایل شاخص :

1 | 35

khademzadeh@iust.ac.ir

سوال ؟

## خلاصه و بررسی روشها

| عیب                                                                           | مزیت                                          | روش                |
|-------------------------------------------------------------------------------|-----------------------------------------------|--------------------|
| فضای حافظه را هدر می دهد.                                                     | پرسیدن به رکورد دلخواه به راحتی انجام می گیرد | رکوردهای طول ثابت  |
| به جز در حالت استفاده از شاخص ، نمی توانیم به رکورد مورد نظر پرش داشته باشیم. | صرفه جویی در میزان حافظه                      | رکوردهای طول متغیر |

khademzadeh@iust.ac.ir