Practical 7: Implementation of sorting based two pass algorithm

GAHAN SARAIYA, 18MCEC10

18mcec10@nirmauni.ac.in

I. INTRODUCTION

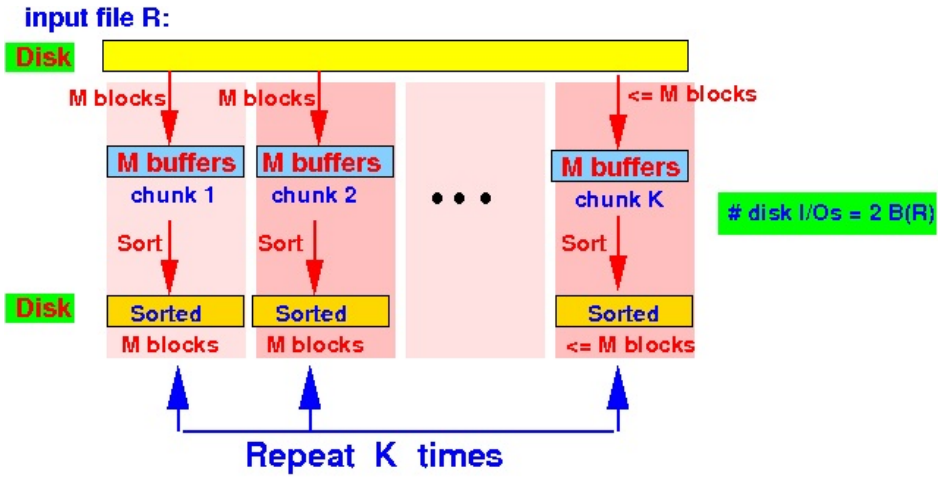
Aim of this practical is to implement sort based two pass algorithm to find distinct values.

II. LOGIC

Prerequisite setup: Created a data file and added dummy in it

I. Pass 1

1. Get $Available\ Memory\ Buffer(M)$
2. Determine file size and decide whether file can be read in one pass or two pass or more
3. For this experiment we'll focus on implementing two pass algorithm to evaluate distinct values
4. Divide file in to chunks of block
5. sort this chucks(sublist) individually with any in memory sorting algorithm
6. Write these sorted chunks(sublist) to disk



II. Pass 2

1. We (re)-use the M buffers to merge the first $(Available\ Memory\ Buffer - 1)$ chucks into a chunk of size $(Available\ Memory\ Buffer) \times (Available\ Memory\ Buffer - 1)$ blocks
2. Iterate over every first element of chunks and pick least value
3. Output if it is not same as previously picked element
4. Repeat from Step 2 until all the elements in all chunks are evaluated

III. Implementation

The code is implemented in Python as below

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
"""
Author: Gahan Saraiya
GiT: https://github.com/gahan9
StackOverflow: https://stackoverflow.com/users/story/7664524

Implementation of sorting based two pass algorithm
"""
import os
import math

from itertools import islice
from faker import Faker

fak = Faker()


class Iterator(object):
    """
    Iterator class to add tuple in form of table
    attributes -> <attrib1, attrib2, attrib3,...>
    Adding values
    values -> <val1, val2, val3, ....>
    """

    def __init__(self, attribute_tuple, file_path, *args, **kwargs):
        """
        :param attribute_tuple: attribute tuple in form of string containing
 ↪  attributes of file (if to be created)
```

```
        :param file_path: location of data file
        :param args:
        :param kwargs:
        """
        self.attributes = attribute_tuple
        self.file_path = file_path
        self.write_back_folder = kwargs.get("write_back_path",
        ↪  "phase_one_write_back")
        self.write_back_path = kwargs.get("write_back_path", "temp.write")
        self.separator = "\t"
        self.records_per_block = kwargs.get("records_per_block", 30)
        self.initialize_file()
        print("{0}\n{1}Consideration{1}\n"
              "Records per block: {2}\n"
              "Total Records: {3}\n{0}\n".format("#"*50, "-"*10,
              ↪  self.records_per_block, self.total_records)
              )

    @staticmethod
    def read_in_chunks(file_object, chunk_size=1024):
        """Lazy function (generator) to read a file piece by piece.
        Default chunk size: 1k."""
        while True:
            data = file_object.read(chunk_size)
            if not data:
                break
            yield data

    @property
    def free_memory(self):
        # calculate how many blocks can be accommodated in memory buffer
        num_lines = sum(1 for line in open(self.file_path))
        no_of_records = num_lines - 2  # remove header line and last new line
        return 101  # for now return available memory statically for basic
        ↪  implementation

    @property
    def total_blocks(self):
        # calculate total number of blocks by record size
        return math.ceil(self.total_records / self.records_per_block)

    @property
```

```
69      def total_records(self):
70          # calculate total number of blocks by record size
71          num_lines = sum(1 for line in open(self.file_path))
72          no_of_records = num_lines - 2  # remove header line and last empty line
73          return no_of_records
74
75      @property
76      def can_be_one_pass(self):
77          # return False  # for testing
78          return True if self.total_blocks < self.free_memory else False
79
80      @property
81      def can_be_two_pass(self):
82          return True if self.free_memory > math.ceil(math.sqrt(self.total_blocks))
            ↪  else False
83
84      def initialize_file(self):
85          # create write back directory for phase 1
86          os.makedirs(self.write_back_folder, exist_ok=True)
87          # check if file exits or not
88          if os.path.exists(self.file_path):
89              pass
90          else:
91              # create file with header if file not exist
92              with open(self.file_path, "w") as f:
93                  f.write(self.separator.join(self.attributes))
94                  f.write("\n")
95          return True
96
97      def add_dummy_data(self, number_of_record=100):
98          """
99
100         :param number_of_record: number of records to be inserted in given file
    ↪   path
101         :return:
102         """
103         with open(self.file_path, "a+") as file:  # open file in append mode
104             for _ in range(number_of_record):
105                 f = fak.profile()
106                 data_tuple = (
```

```
                    f['name'], f['ssn'], f['sex'], f['job'].replace("\n", ""),
                    ↪ f['company'].replace("\n", ""), f['address'].replace("\n",
                    ↪ "")
                )
                data_string = self.separator.join(data_tuple) + "\n"
                file.write(data_string)

    @staticmethod
    def summary(total_results, total_records):
        print("-"*30)
        print("Total Results: {}".format(total_results))
        print("Total Records: {}".format(total_records))
        return True

    @staticmethod
    def split_file_in_blocks(file_obj, split_size):
        blocks = []
        while True:
            block_records = list(islice(file_obj, split_size))
            if not block_records:
                break
            else:
                blocks.append(block_records)
        return blocks

    @staticmethod
    def create_file_obj(attribute):
        file_name = "output_distinct_on_{}.tsv".format(attribute)
        return open(file_name, "w")

    def get_distinct(self, attribute=None, only_summary=True,
    ↪ output_write=False):
        output_obj = self.create_file_obj(attribute) if output_write else None
        sort_key = attribute if attribute else "ssn"
        print("{0}\n DISTINCT ON {1}\n{0}".format('#'*50, sort_key))
        _result_set = []
        if self.can_be_one_pass:
            print("Processing One Pass Algorithm")
            with open(self.file_path, "r") as f:
                content = f.read().split("\n")
            for record in content:
                if record not in _result_set:
```

```
146                 _result_set.append(record)
147         elif self.can_be_two_pass:
148             # apply 2 pass algorithm to sort and use operation on database
149             print("Processing Two Pass Algorithm")
150             f = open(self.file_path, "r")
151             header = f.readline()
152             # writer = open(self.write_back_path, "w")
153             # writer.write(header)
154             _idx = header.split(self.separator).index(sort_key)
155             file_order = 0  # finally a number contains total number of
                ↪ split/sublist file
156             while True:
157                 # read blocks one by one
158                 block_records = list(islice(f, self.free_memory - 1))
159                 if not block_records:
160                     break
161                 else:
162                     file_order += 1
163                     # sort sublist by "ssn" or any other attribute
164                     writer = open(os.path.join(self.write_back_folder,
                        ↪ "temp_00{}".format(file_order)), "w")
165                     # writer.write(header)
166                     sorted_sublist = sorted(block_records, key=lambda x:
                        ↪ x.split(self.separator)[_idx])
167                     # write sorted block/sublist data back to disk(secondary
                        ↪ memory)
168                     writer.writelines(sorted_sublist)
169                     writer.close()
170             f.close()
171
172             # PHASE 2
173
174             partition_ptr_lis = [open(os.path.join(self.write_back_folder,
                ↪ "temp_00{}".format(i)), "r") for i in range(1, file_order+1)]
175             phase2_data = [i.readline().split(self.separator)[_idx] for i in
                ↪ partition_ptr_lis]  # get first element from each sublist
176             # read sublist from each block and output desire result
177             last_read = ""
178             total_results = 0
179             # for line in open(self.write_back_path, "r"):
180             while any(phase2_data):
```

```
181                 temp_lis = list(filter(None, phase2_data)) if None in phase2_data
                     ↪  else phase2_data
182                 current_record = min(temp_lis)
183                 chunk_no = phase2_data.index(current_record)
184                 next_record = partition_ptr_lis[chunk_no].readline()
185                 if next_record:
186                     phase2_data[chunk_no] =
                         ↪  next_record.split(self.separator)[_idx]
187                 else:
188                     # file/sublist has nothing to load/read
189                     del partition_ptr_lis[chunk_no]
190                     del phase2_data[chunk_no]
191                 if current_record and current_record != last_read:
192                     if not only_summary:
193                         print(current_record)
194                     if output_write:
195                         output_obj.write(current_record + "\n")
196                     last_read = current_record
197                     total_results += 1
198             self.summary(total_results, self.total_records)
199         else:
200             # can not proceed all given blocks with memory constraint
201             print("Require more than two pass to handle this large data")
202         return _result_set
203
204
205 if __name__ == "__main__":
206     table = Iterator(attribute_tuple=("name", "ssn", "gender", "job", "company",
         ↪  "address"),
207                      file_path="iterator.dbf")
208     table.get_distinct("name", only_summary=True)
209     table.get_distinct("job", only_summary=False, output_write=True)
210     table.get_distinct("ssn", only_summary=True)
211     table.get_distinct("gender", only_summary=False, output_write=True)
```

I. Output

```
Consideration:
Records per block: 30
Total Records per block: 5000
```

```
##################################################
 DISTINCT ON name
##################################################
Processing Two Pass Algorithm
-----------------------------
Total Results: 4837
Total Records: 5000
##################################################
 DISTINCT ON job
##################################################
Processing Two Pass Algorithm
Academic librarian
Accommodation manager
Accountant, chartered
Accountant, chartered certified
Accountant, chartered management
Accountant, chartered public finance
Accounting technician
Actor
Actuary
Acupuncturist
Administrator
Administrator, Civil Service
Administrator, arts
Administrator, charities/voluntary organisations
Administrator, education
Administrator, local government
Administrator, sports
Adult guidance worker
Adult nurse
Advertising account executive
Advertising account planner
Advertising art director
Advertising copywriter
Advice worker
Aeronautical engineer
Agricultural consultant
Agricultural engineer
Aid worker
Air broker
Air cabin crew
```

```
Air traffic controller
Airline pilot
Ambulance person
Amenity horticulturist
Analytical chemist
Animal nutritionist
Animal technologist
Animator
Applications developer
Arboriculturist
Archaeologist
Architect
Architectural technologist
Archivist
Armed forces logistics/support/administrative officer
Armed forces operational officer
Armed forces technical officer
Armed forces training and education officer
Art gallery manager
Art therapist
Artist
Arts administrator
Arts development officer
Associate Professor
Astronomer
Audiological scientist
Automotive engineer
Banker
Barista
Barrister
Barrister's clerk
Best boy
Biochemist, clinical
Biomedical engineer
Biomedical scientist
Bonds trader
Bookseller
Brewing technologist
Broadcast engineer
Broadcast journalist
Broadcast presenter
```

```
Building control surveyor
Building services engineer
Building surveyor
Buyer, industrial
Buyer, retail
Cabin crew
Call centre manager
Camera operator
Careers adviser
Careers information officer
Cartographer
Catering manager
Ceramics designer
Charity fundraiser
Charity officer
Chartered accountant
Chartered certified accountant
Chartered legal executive (England and Wales)
Chartered loss adjuster
Chartered management accountant
Chartered public finance accountant
Chemical engineer
Chemist, analytical
Chief Executive Officer
Chief Financial Officer
Chief Marketing Officer
Chief Operating Officer
Chief Strategy Officer
Chief Technology Officer
Chief of Staff
Child psychotherapist
Chiropodist
Chiropractor
Civil Service administrator
Civil Service fast streamer
Civil engineer, consulting
Civil engineer, contracting
Claims inspector/assessor
Clinical biochemist
Clinical cytogeneticist
Clinical embryologist
```

```
Clinical molecular geneticist
Clinical psychologist
Clinical research associate
Clinical scientist, histocompatibility and immunogenetics
Clothing/textile technologist
Colour technologist
Commercial art gallery manager
Commercial horticulturist
Commercial/residential surveyor
Commissioning editor
Communications engineer
Community arts worker
Community development worker
Community education officer
Community pharmacist
Company secretary
Comptroller
Computer games developer
Conference centre manager
Conservation officer, historic buildings
Conservation officer, nature
Conservator, furniture
Conservator, museum/gallery
Consulting civil engineer
Contracting civil engineer
Contractor
Control and instrumentation engineer
Copy
Copywriter, advertising
Corporate investment banker
Corporate treasurer
Counselling psychologist
Counsellor
Curator
Customer service manager
Cytogeneticist
Dance movement psychotherapist
Dancer
Data processing manager
Data scientist
Database administrator
```

```
Dealer
Dentist
Designer, blown glass/stained glass
Designer, ceramics/pottery
Designer, exhibition/display
Designer, fashion/clothing
Designer, furniture
Designer, graphic
Designer, industrial/product
Designer, interior/spatial
Designer, jewellery
Designer, multimedia
Designer, television/film set
Designer, textile
Development worker, community
Development worker, international aid
Diagnostic radiographer
Dietitian
Diplomatic Services operational officer
Dispensing optician
Doctor, general practice
Doctor, hospital
Dramatherapist
Drilling engineer
Early years teacher
Ecologist
Economist
Editor, commissioning
Editor, film/video
Editor, magazine features
Editorial assistant
Education administrator
Education officer, community
Education officer, environmental
Education officer, museum
Educational psychologist
Electrical engineer
Electronics engineer
Embryologist, clinical
Emergency planning/management officer
Energy engineer
```

```
Energy manager
Engineer, aeronautical
Engineer, agricultural
Engineer, automotive
Engineer, biomedical
Engineer, broadcasting (operations)
Engineer, building services
Engineer, chemical
Engineer, civil (consulting)
Engineer, civil (contracting)
Engineer, communications
Engineer, control and instrumentation
Engineer, drilling
Engineer, electrical
Engineer, electronics
Engineer, energy
Engineer, land
Engineer, maintenance
Engineer, maintenance (IT)
Engineer, manufacturing
Engineer, manufacturing systems
Engineer, materials
Engineer, mining
Engineer, petroleum
Engineer, production
Engineer, site
Engineer, structural
Engineer, technical sales
Engineer, water
Engineering geologist
English as a foreign language teacher
English as a second language teacher
Environmental consultant
Environmental education officer
Environmental health practitioner
Environmental manager
Equality and diversity officer
Equities trader
Ergonomist
Estate agent
Estate manager/land agent
```

```
Event organiser
Exercise physiologist
Exhibition designer
Exhibitions officer, museum/gallery
Facilities manager
Farm manager
Fashion designer
Fast food restaurant manager
Field seismologist
Field trials officer
Film/video editor
Financial adviser
Financial controller
Financial manager
Financial planner
Financial risk analyst
Financial trader
Fine artist
Firefighter
Fish farm manager
Fisheries officer
Fitness centre manager
Food technologist
Forensic psychologist
Forensic scientist
Forest/woodland manager
Freight forwarder
Furniture conservator/restorer
Furniture designer
Further education lecturer
Futures trader
Gaffer
Games developer
Garment/textile technologist
General practice doctor
Geneticist, molecular
Geochemist
Geographical information systems officer
Geologist, engineering
Geologist, wellsite
Geophysical data processor
```

```
Geophysicist/field seismologist
Geoscientist
Glass blower/designer
Government social research officer
Graphic designer
Haematologist
Health and safety adviser
Health and safety inspector
Health physicist
Health promotion specialist
Health service manager
Health visitor
Herbalist
Heritage manager
Herpetologist
Higher education careers adviser
Higher education lecturer
Historic buildings inspector/conservation officer
Holiday representative
Homeopath
Horticultural consultant
Horticultural therapist
Horticulturist, amenity
Horticulturist, commercial
Hospital doctor
Hospital pharmacist
Hotel manager
Housing manager/officer
Human resources officer
Hydrogeologist
Hydrographic surveyor
Hydrologist
IT consultant
IT sales professional
IT technical support officer
IT trainer
Illustrator
Immigration officer
Immunologist
Industrial buyer
Industrial/product designer
```

```
Information officer
Information systems manager
Insurance account manager
Insurance broker
Insurance claims handler
Insurance risk surveyor
Insurance underwriter
Intelligence analyst
Interior and spatial designer
International aid/development worker
Interpreter
Investment analyst
Investment banker, corporate
Investment banker, operational
Jewellery designer
Journalist, broadcasting
Journalist, magazine
Journalist, newspaper
Land
Land/geomatics surveyor
Landscape architect
Lawyer
Learning disability nurse
Learning mentor
Lecturer, further education
Lecturer, higher education
Legal executive
Legal secretary
Leisure centre manager
Lexicographer
Librarian, academic
Librarian, public
Licensed conveyancer
Lighting technician, broadcasting/film/video
Lobbyist
Local government officer
Location manager
Logistics and distribution manager
Loss adjuster, chartered
Magazine features editor
Magazine journalist
```

```
Maintenance engineer
Make
Management consultant
Manufacturing engineer
Manufacturing systems engineer
Marine scientist
Market researcher
Marketing executive
Materials engineer
Mechanical engineer
Media buyer
Media planner
Medical illustrator
Medical laboratory scientific officer
Medical physicist
Medical sales representative
Medical secretary
Medical technical officer
Mental health nurse
Merchandiser, retail
Merchant navy officer
Metallurgist
Meteorologist
Microbiologist
Midwife
Minerals surveyor
Mining engineer
Mudlogger
Multimedia programmer
Multimedia specialist
Museum education officer
Museum/gallery conservator
Museum/gallery curator
Museum/gallery exhibitions officer
Music therapist
Music tutor
Musician
Nature conservation officer
Naval architect
Network engineer
Neurosurgeon
```

```
Newspaper journalist
Nurse, adult
Nurse, children's
Nurse, learning disability
Nurse, mental health
Nutritional therapist
Occupational hygienist
Occupational psychologist
Occupational therapist
Oceanographer
Office manager
Oncologist
Operational investment banker
Operational researcher
Operations geologist
Ophthalmologist
Optician, dispensing
Optometrist
Orthoptist
Osteopath
Outdoor activities/education manager
Paediatric nurse
Paramedic
Passenger transport manager
Patent attorney
Patent examiner
Pathologist
Pension scheme manager
Pensions consultant
Personal assistant
Personnel officer
Petroleum engineer
Pharmacist, community
Pharmacist, hospital
Pharmacologist
Photographer
Physicist, medical
Physiological scientist
Physiotherapist
Phytotherapist
Pilot, airline
```

```
Planning and development surveyor
Plant breeder/geneticist
Podiatrist
Police officer
Politician's assistant
Presenter, broadcasting
Press photographer
Press sub
Primary school teacher
Print production planner
Printmaker
Prison officer
Private music teacher
Probation officer
Producer, radio
Producer, television/film/video
Product designer
Product manager
Product/process development scientist
Production assistant, radio
Production assistant, television
Production designer, theatre/television/film
Production engineer
Production manager
Professor Emeritus
Programme researcher, broadcasting/film/video
Programmer, applications
Programmer, multimedia
Programmer, systems
Proofreader
Psychiatric nurse
Psychiatrist
Psychologist, clinical
Psychologist, counselling
Psychologist, educational
Psychologist, forensic
Psychologist, occupational
Psychologist, prison and probation services
Psychologist, sport and exercise
Psychotherapist
Psychotherapist, child
```

```
Psychotherapist, dance movement
Public affairs consultant
Public house manager
Public librarian
Public relations account executive
Public relations officer
Publishing copy
Publishing rights manager
Purchasing manager
Quality manager
Quantity surveyor
Quarry manager
Race relations officer
Radiation protection practitioner
Radio broadcast assistant
Radio producer
Radiographer, diagnostic
Radiographer, therapeutic
Ranger/warden
Records manager
Recruitment consultant
Recycling officer
Regulatory affairs officer
Research officer, government
Research officer, political party
Research officer, trade union
Research scientist (life sciences)
Research scientist (maths)
Research scientist (medical)
Research scientist (physical sciences)
Restaurant manager
Restaurant manager, fast food
Retail banker
Retail buyer
Retail manager
Retail merchandiser
Risk analyst
Risk manager
Runner, broadcasting/film/video
Rural practice surveyor
Sales executive
```

```
Sales professional, IT
Sales promotion account executive
Science writer
Scientific laboratory technician
Scientist, audiological
Scientist, biomedical
Scientist, clinical (histocompatibility and immunogenetics)
Scientist, forensic
Scientist, marine
Scientist, physiological
Scientist, product/process development
Scientist, research (life sciences)
Scientist, research (maths)
Scientist, research (medical)
Scientist, research (physical sciences)
Scientist, water quality
Secondary school teacher
Secretary, company
Secretary/administrator
Seismic interpreter
Senior tax professional/tax inspector
Set designer
Ship broker
Site engineer
Social research officer, government
Social researcher
Social worker
Software engineer
Soil scientist
Solicitor
Solicitor, Scotland
Sound technician, broadcasting/film/video
Special educational needs teacher
Special effects artist
Speech and language therapist
Sport and exercise psychologist
Sports administrator
Sports coach
Sports development officer
Sports therapist
Stage manager
```

```
Statistician
Structural engineer
Sub
Surgeon
Surveyor, building
Surveyor, building control
Surveyor, commercial/residential
Surveyor, hydrographic
Surveyor, insurance
Surveyor, land/geomatics
Surveyor, minerals
Surveyor, mining
Surveyor, planning and development
Surveyor, quantity
Surveyor, rural practice
Systems analyst
Systems developer
TEFL teacher
Tax adviser
Tax inspector
Teacher, English as a foreign language
Teacher, adult education
Teacher, early years/pre
Teacher, music
Teacher, primary school
Teacher, secondary school
Teacher, special educational needs
Teaching laboratory technician
Technical author
Technical brewer
Technical sales engineer
Telecommunications researcher
Television camera operator
Television floor manager
Television production assistant
Television/film/video producer
Textile designer
Theatre director
Theatre manager
Theatre stage manager
Theme park manager
```

```
Therapeutic radiographer
Therapist, art
Therapist, drama
Therapist, horticultural
Therapist, music
Therapist, nutritional
Therapist, occupational
Therapist, speech and language
Therapist, sports
Tour manager
Tourism officer
Tourist information centre manager
Town planner
Toxicologist
Trade mark attorney
Trade union research officer
Trading standards officer
Training and development officer
Translator
Transport planner
Travel agency manager
Tree surgeon
Veterinary surgeon
Video editor
Visual merchandiser
Volunteer coordinator
Warden/ranger
Warehouse manager
Waste management officer
Water engineer
Water quality scientist
Web designer
Wellsite geologist
Writer
Youth worker
------------------------------
Total Results: 639
Total Records: 5000
##################################################
 DISTINCT ON ssn
##################################################
```

```
Processing Two Pass Algorithm
------------------------------
Total Results: 5001
Total Records: 5000
################################################
 DISTINCT ON gender
################################################
Processing Two Pass Algorithm
F
M
------------------------------
Total Results: 2
Total Records: 5000
```

IV. Summary

One pass algorithm can only work if whole block of relation can fit in to main memory otherwise we require more than one pass to determine correct result such as **Two-Pass Multiway Merge Sort (TPMMS) Algorithm** as implemented here for unary operator - distinct (δ).

For Two pass algorithm we need to fetch all blocks and then write the individually sorted blocks and again we need to read all blocks to perform query operation hence this algorithm can only work if below requirements are satisfied.

I. Requirements of Two Pass

- $number\ of\ chunks \leq Available\ Memory\ Buffer - 1$

II. File Size Constraint

- $Max\ File\ Size \leq (Available\ Memory\ Buffer) \times (Available\ Memory\ Buffer - 1)$