

เอกสารอ้างอิง



- (1) Barlara J. Burian. A simplified approach to S/370 assembly language programming. Englewood Chifts, N.J. prentice-Hall, 1977.
- (2) IBM DOS/VS Access method service User's guide
- (3) IBM DOS/VS System control Statement
- (4) IBM DOS/VS System management guide
- (5) IBM DOS/VS Supervisor and I/O Macro
- (6) IBM DOS full American National Standard COBOL Programmer guide
- (7) IBM DOS/VS COBOL Compiler and Library Programmer's guide
- (8) IBM System /360 and System /370 FORTRAN IV Language
- (9) IBM DOS/VS Data management guide
- (10) IBM Student Text Introduction to IBM System /360 Direct access Storage device and Organization method
- (11) Introduction to data management
- (12) Introduction to IBM DASD and Organization method
- (13) Jame Martin. Computer Data-base Organization. Englewood Chifts, N.J. prentice-Hall, 1977
- (14) Kendred, Alton R. Data system and management. Englewood Chifts, N.J. prentice-Hall, 1973
- (15) Nancy Stun, Alden Sager, Robert A. Stun. 370/360 Assembler language programming. John Wiley & Sen, 1979
- (16) Reference Manual for IBM 3340/3344 Disk Storage
- (17) Robert T. grauer, Marshal A. Crawford. The COBOL ENVIRONMENT. Englewood Chifts, N.J. prentice-Hall, 1979.

ภาคผนวก ก .

ระบบแผ่นจานแม่เหล็ก
(Magnetic Disk System)

๑. ลักษณะส่วนประกอบของจานแม่เหล็ก

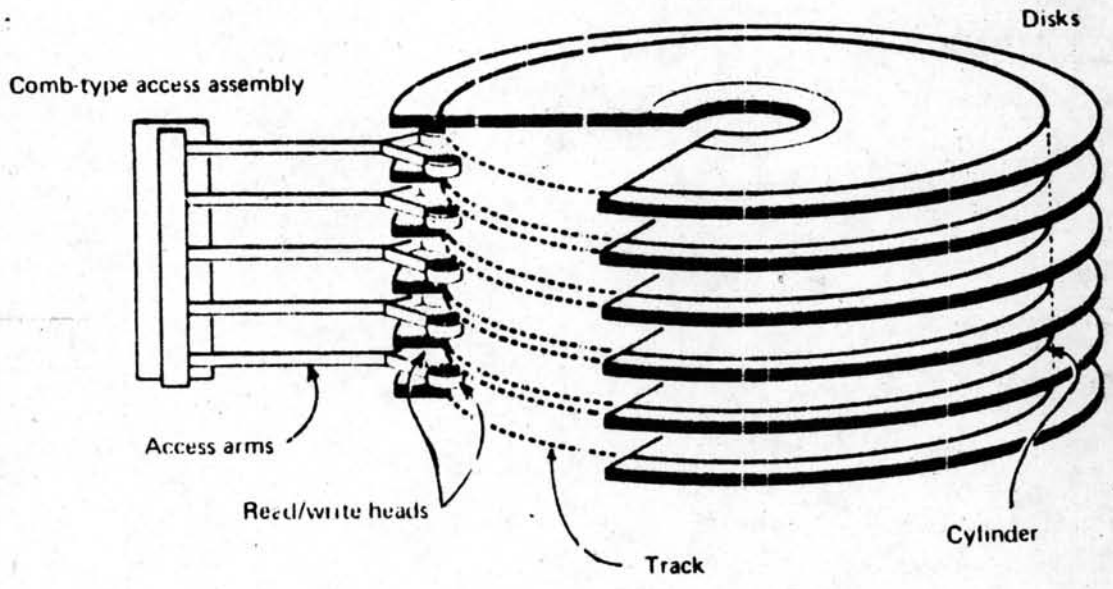
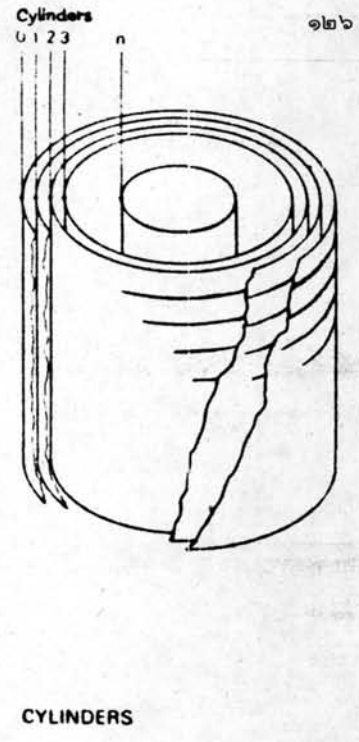
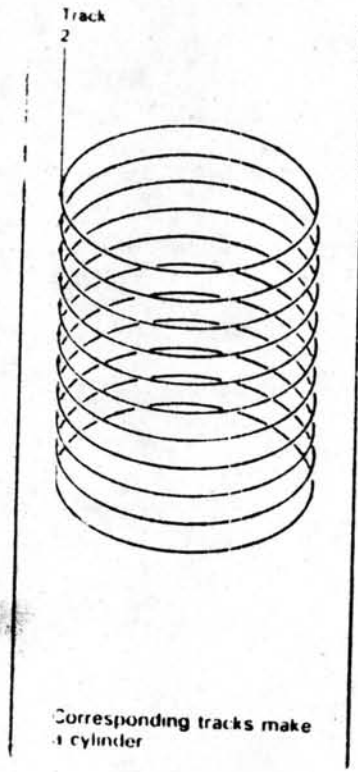
ระบบแผ่นจานแม่เหล็ก แบ่งได้เป็น ๒ ส่วนใหญ่ ๆ คือ

ก) แผ่นจานแม่เหล็ก (Disk Pack)

ข) ตู้จานแม่เหล็ก (Disk Drive)

ก) แผ่นจานแม่เหล็ก

แผ่นจานแม่เหล็ก ใช้เป็นสื่อบันทึกข้อมูล มีลักษณะคล้ายแผ่นเสียงหลายแผ่นมาเรียงซ้อนกัน ระหว่างแผ่นมีช่องว่างพอที่จะให้หัวอ่านและบันทึก (Read / Write head) เคลื่อนเข้าออกได้ แต่ละแผ่นทำด้วย สตีล เบอริลเลียม (Steel beryelium) ผิวหน้า (Surface) ทั้งสองฉาบด้วย เฟอร์ริกออกไซด์ (Ferric Oxide) แผ่นเหล่านี้ถูกนำมาประกอบเพื่อหมุน รอบแกนเดียวกันด้วยความเร็วคงที่ เพื่ออ่านและบันทึกข้อมูลบนพื้นผิวทั้งด้านบนและด้านล่าง ในแต่ละด้านของพื้นผิว จะประกอบไปด้วยวง เป็นจำนวนมาก เพื่อใช้บันทึกข้อมูลซึ่งเรียกว่า แทรค (Track) แต่ละแทรคสามารถจุข้อมูลได้เท่า ๆ กัน แต่ละแทรคที่อยู่ตรงกันของแต่ละพื้นผิวรวมกันเป็นไซลินเดอร์ (Cylinder)



รูป ก. แสดงลักษณะของแผ่นจานแม่เหล็ก

แผ่นจานแม่เหล็กนี้ แบ่งได้เป็น ๒ ชนิดคือ

๑) ชนิดที่ถอดเปลี่ยนได้ (Removable Disk pack) คือแผ่นจานแม่เหล็กที่สามารถใส่เข้าหรือถอดออกจากตู้จานแม่เหล็กได้ (Disk Drive) กล่าวคือ เมื่อต้องการใช้แผ่นจานแม่เหล็กใดทำงานก็นำไปสับเปลี่ยนกับแผ่นจานแม่เหล็กในตู้จานแม่เหล็กที่ไม่ได้ใช้นั้น

๒) ชนิดที่ถอดเปลี่ยนไม่ได้ (Fixed Disk pack) คือแผ่นจานแม่เหล็กชนิดที่แผ่นจานแม่เหล็กจะติดกับตู้จานแม่เหล็กอยู่ตลอดเวลา ไม่สามารถถอดเข้าหรือออกได้

ข) ตู้จานแม่เหล็ก ประกอบด้วยส่วนที่สำคัญ ๆ ดังนี้

๑) หัวอ่าน/บันทึก (Read/Write head) ทำหน้าที่อ่านหรือบันทึกข้อมูลบนแผ่นจานแม่เหล็ก โดยปกติแล้วหัวอ่าน/บันทึกข้อมูลนี้จะมีจำนวน เท่ากับจำนวนผิวของแผ่นจานแม่เหล็กและแต่ละหัวจะอยู่บนแขน (Access arms) แยกจากกัน แต่ทุกแขนจะถูกยึดรวมกัน (Access Assembly) ลักษณะหัวอ่านโดยทั่วไป สามารถแบ่งออกได้เป็น ๒ ชนิดคือ

๑.๑ ชนิดเคลื่อนที่ได้ (Moving head) หัวอ่าน/บันทึก ชนิดนี้ จะมีหัวอ่าน/บันทึกในแต่ละพื้นผิวเพียงชุดเดียวทั้งในรูปประกอบ และในการวิ่ง (Access) ข้อมูลจะต้องเสียเวลาจำนวนหนึ่ง ซึ่งเรียกว่าเวลาค้นหาตำแหน่งข้อมูล (Access time หรือ Seek time) เพื่อให้หัวอ่าน/บันทึกนี้เคลื่อนไปยังไซลินเดอร์ ที่ต้องการ และก่อนที่ หัวอ่าน/บันทึก จะสามารถทำงานตรงตำแหน่งที่ต้องการได้ ต้องรอให้ข้อมูลที่ต้องการอ่านหรือบันทึกนั้นหมุนผ่านมาตรงตำแหน่งที่หัวอ่าน/บันทึก นี้รออยู่ ซึ่งเวลาที่เสียไป ส่วนนี้เรียกว่า Rotational Delay

๑.๒ ชนิดคงที่ (Fixed head) หัวอ่าน/บันทึกชนิดนี้ จะมีหัวอ่าน/บันทึกหลายหัว ซึ่งอาจจะ เท่ากับจำนวนแทรคของแต่ละหน้า บางครั้งจะเรียกจานแม่เหล็กชนิดนี้ว่า "Head per track" ตู้จานแม่เหล็กชนิดหัวอ่าน/บันทึกนี้จะประหยัด เวลาในการเคลื่อนหัวอ่าน/บันทึกไปยังไซลินเดอร์ที่ต้องการ โดยไม่ต้อง เสียเวลาสำหรับการค้นหาตำแหน่งข้อมูล ส่วนจะใช้หัวอ่าน/บันทึกใดทำงานนั้น หัวควบคุม (Controller) ของแผ่นจานแม่เหล็ก จะเป็นตัวกำหนดให้โดยอัตโนมัติ

๒) แอกทูเอเตอร์ (Actuator) ทำหน้าที่เลื่อนหัวอ่าน/บันทึกให้เลื่อนจากแทรคหนึ่งไปยังอีกแทรคหนึ่งบนผิวของจานแม่เหล็ก

๒. การบันทึกข้อมูล

ข้อมูลจะบันทึกลงในแทรค โดยอาศัยคุณสมบัติของสารแม่เหล็กที่ก่อตัวเป็นจุด ๆ บนแทรคจุด ๆ หนึ่งคือ Magnetic Spot ใช้แทนค่าเลขฐาน ๒ (Binary digit) ๑ หลัก เรียกว่าบิต ซึ่งเป็นหน่วยที่เล็กที่สุดในการบันทึกข้อมูล ในแต่ละแทรค นอกจากมีข้อมูลที่บันทึกแล้วยังมีบางส่วนที่ฮาร์ดแวร์สร้างขึ้นมาเอง เพื่อใช้บอกตำแหน่งของข้อมูลบนแทรค บอกความหมายของข้อมูลแต่ละหน่วย หรือใช้บอกสภาวะหรือสภาพของแทรคที่เก็บข้อมูล

๓. การดึงข้อมูล

ในการดึงข้อมูลในแผ่นจานแม่เหล็กออกมาใช้ ต้องเสียเวลา ๔ ช่วง คือ

ก) เวลาที่หัวอ่าน/บันทึกข้อมูลเคลื่อนที่ไปยังไซลินเดอร์ที่เก็บข้อมูลที่ต้องการ (Access motion time หรือ Seek time) ซึ่งขึ้นอยู่กับชนิดของแผ่นจานแม่เหล็ก สำหรับแผ่นจานแม่เหล็กของ ไอ.บี.เอ็ม รุ่น ๓๓๔๐ นี้ จะใช้เวลาไม่มากกว่า ๑๓๐ ms (เศษหนึ่งส่วนพันวินาที) และไม่น้อยกว่า ๒๕ ms ซึ่งโดยปกติแล้วจะใช้เวลาเฉลี่ยไม่น้อยกว่า ๖๐ ms

ข) เวลาที่ใช้ในการเลือกหัวที่จะอ่านข้อมูลแต่เวลานี้น้อยมากจนไม่จำเป็นต้องนำมาคำนวณ (Head Selection time)

ค) เวลาที่รอให้ระเบียนข้อมูลที่ต้องการอ่านหรือบันทึกผ่านหัวอ่าน/บันทึก (Rotational Delay) ก่อนที่จะเริ่มอ่านหรือบันทึกข้อมูลนั้น สำหรับแผ่นจานแม่เหล็กของ ไอ.บี.เอ็ม ๓๓๔๐ นี้ จะใช้เวลาไม่มากกว่า ๕๐ ms และไม่น้อยกว่า ๐ คือไม่เสียเวลาเลย หรือโดยเฉลี่ยแล้วประมาณ ๒๕ ms

ง) อัตราเร็วที่หัวอ่าน/บันทึก อ่านข้อมูลเพื่อส่งไปยัง Core storage หรือส่งข้อมูลจาก Core Storage ไปบันทึกลงจานแม่เหล็ก (Data transfer rate) ซึ่งขึ้นอยู่กับความเร็วในการหมุนของแผ่นจานแม่เหล็กและความหนาแน่นของข้อมูลบนจานแม่เหล็ก สำหรับจานแม่เหล็กของ ไอ.บี.เอ็ม ๓๓๔๐ นี้จะมีค่าเป็น ๔๔๔ KB/sec หรือ ๐.๐๐๑๑๓ millisecc/bytes.

การคำนวณ ข

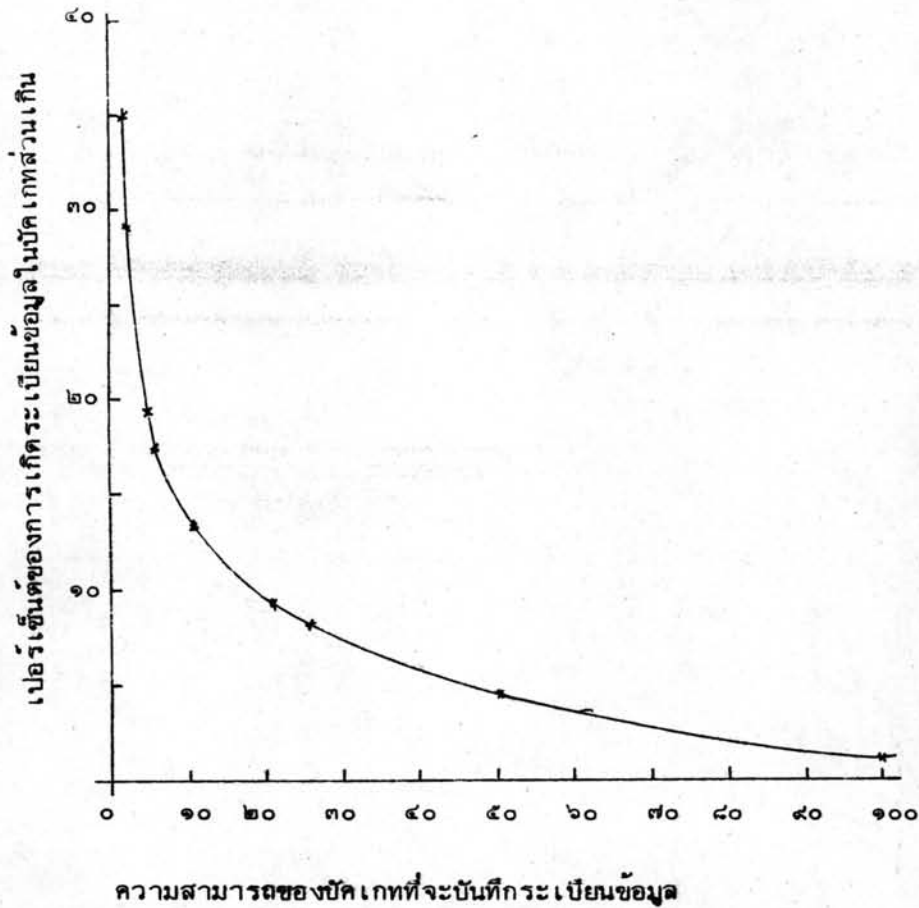
ประสิทธิภาพของการจัดเก็บข้อมูลแบบโคเร็ค (Direct Access Organization)

แฟ้มข้อมูลที่จัดเก็บแบบโคเร็คจะทำงานอย่างมีประสิทธิภาพขึ้นอยู่กับองค์ประกอบต่าง ๆ ดังนี้^(๑๓)

- ๑) ขนาดของบัคเกต (Bucket size)
- ๒) แพ็คกิงเดนซิตี (Packing Density)
- ๓) แอลซิงฟังก์ชันที่ใช้ (Hashing key to-address transform)
- ๔) วิธีการที่จะจัดการกับระเบียบข้อมูลในพื้นที่ส่วนเกิน (Method of handling Overflow)

๑. ขนาดของบัคเกต

บัคเกต เป็นหน่วยหนึ่งเพื่อเก็บระเบียบข้อมูล ซึ่งจะ เป็นเนื้อที่เท่าใดก็ได้ในจานแม่เหล็ก แล้วแต่กำหนด โดยปกติแล้วหนึ่งบัคเกตจะ เท่ากับหนึ่งแทรคหรือหนึ่งไซลินเดอร์ การบันทึกระเบียบข้อมูลลงในบัค เกตนั้นจะบันทึกระเบียบข้อมูลก็ระเบียบข้อมูลลงในบัค เกตขึ้นอยู่กับผู้ใช้จะกำหนดขนาดของ บัคเกตและความยาวของระเบียบข้อมูล ขนาดของบัคเกตที่ผู้ใช้กำหนดจะมีผลต่อการเกิดระเบียบข้อมูล ในบัคเกตส่วนเกินคือ ถ้าบัคเกตหนึ่ง ๆ สามารถบันทึกระเบียบข้อมูลได้มากโอกาสที่ระเบียบข้อมูลจะ ไปอยู่ในบัคเกตส่วนเกินก็จะน้อย แต่ถ้ากำหนดให้บัคเกตหนึ่ง ๆ สามารถบันทึกระเบียบข้อมูลได้น้อย โอกาสที่ระเบียบข้อมูลจะอยู่ในบัคเกตส่วนเกินก็จะมามาก เช่น ถ้ามีบัคเกตอยู่ ๑๐๐ บัคเกต แต่ละ บัคเกตสามารถบันทึกระเบียบข้อมูลได้ ๑ ระเบียบข้อมูล โอกาสที่จะเกิดระเบียบข้อมูลในพื้นที่ส่วนเกิน ก็จะมีมากกว่าถ้ามีบัคเกตอยู่ ๑๐ บัคเกตและแต่ละบัคเกตสามารถบันทึกระเบียบได้ ๑๐ ระเบียบข้อมูล ดังแสดงในรูป ก.



รูป ก. แสดงความสัมพันธ์ของการเกิดระเบียนข้อมูลในบิต เกทส่วนเกินกับความสามารถของบิต เกทที่จะบันทึกระเบียนข้อมูล

๒. แพ็คกิง เดนซิตี

คือ สัดส่วนของจำนวนของระเบียนข้อมูลที่บรรจุในโหมบิต เกทกับจำนวนสูงสุดของระเบียนข้อมูลที่สามารถบรรจุได้ในโหมบิต เกท แพ็คกิง เดนซิตีจะมีผลต่อการเกิดระเบียนข้อมูลในบิต เกทส่วนเกิน ถ้าแพ็คกิง เดนซิตีมีค่าสูงโอกาสที่จะเกิดระเบียนข้อมูลในบิต เกทส่วนเกินก็มีมาก ซึ่งผู้ใช้สามารถคำนวณได้จากสูตร

$$\text{แพ็คกิง เดนซิตี} = \frac{\text{จำนวนของระเบียนข้อมูลที่บรรจุในโหมบิต เกท}}{\text{จำนวนสูงสุดของระเบียนข้อมูลที่สามารถบรรจุได้}}$$

เมื่อใช้หลักการนี้ในการอ้างอิงถึงโหมบิต เกทโดยไม่คำนึงถึง บิต เกทส่วนเกินจะเรียกว่า ไพรมแพ็คกิง เดนซิตี (Prime packing Density) ผู้ใช้สามารถคำนวณหาไพรมแพ็คกิง เดนซิตี

ขนาดของบ้คเกท และโอกาสที่จะเกิดระเบียนข้อมูลในบ้คเกทส่วนเกินโดยใช้ตาราง ก. และในรูป ข. เป็นการแสดงผลที่ได้จากการคำนวณ

ให้ $N =$ จำนวนของระเบียนข้อมูลทั้งหมด

$M =$ จำนวนของบ้คเกท

$C =$ ความสามารถที่จะบันทึกระเบียนข้อมูลได้ในแต่ละบ้คเกท

$$\therefore \text{Packing Density} = \frac{N}{CM} \text{-----2.1}$$

โดยใช้ Binomial Distribution (๑๘) จะสามารถคำนวณโอกาสที่ที่ว่างแต่ละที่ จะบรรจุระเบียนข้อมูลได้ x ระเบียนข้อมูล จากสมการ ๒.๒

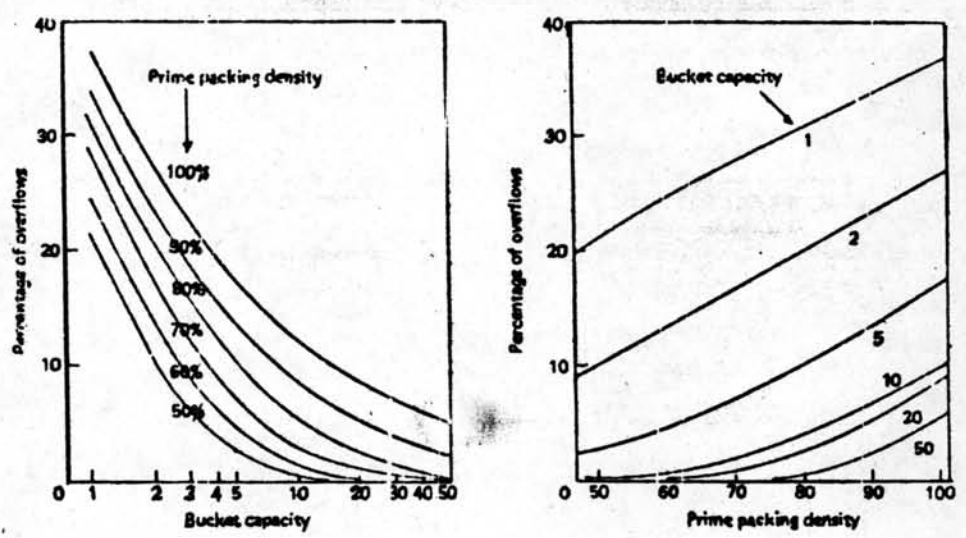
$$\text{Prob}(x) = \frac{N!}{x!(N-x)!} \left(\frac{1}{M}\right)^x \left(\frac{1-1}{M}\right)^{n-x} \text{-----2.2}$$

$$\text{และโอกาสที่จะเกิด } Y \text{ Overflow} = \text{Prob}(C+Y) \text{-----2.3}$$

$$\therefore \text{ค่าเฉลี่ยของการเกิด Overflow สำหรับที่ว่างแต่ละที่} = \sum_{Y=1}^{\infty} \text{Prob}(C+Y) \cdot Y \text{-----2.4}$$

$$\therefore \% \text{ ของการเกิด Overflow สำหรับที่ว่างทั้งหมด} = \frac{M}{N} \times 100 \cdot \sum_{Y=1}^{\infty} \text{Prob}(C+Y) \cdot Y \text{-----2.5}$$

ตาราง ก. แสดงการคำนวณหาไพรม์แพ็คกิงเดนซิตี้ ขนาดของบ้คเกทและโอกาสที่จะเกิดระเบียนข้อมูล ในบ้คเกทส่วนเกิน



รูป ข. แสดงขนาดของบัคเกต ไพรมแพคกิ้ง เดนซิตีที่มีผลต่อการเกิดระเหินข้อมูลในบัคเกต ส่วนเกิน

จากรูป ถ้าไพรมแพคกิ้ง เดนซิตีคงที่ = ๕๐ เปอร์เซ็นต์ เมื่อความสามารถของบัคเกต ที่จะบรรจุระเหินข้อมูล (bucket capacity) เพิ่มขึ้น เปอร์เซ็นต์ของการเกิดระเหินข้อมูลใน บัคเกตส่วนเกินก็มีน้อย

ถ้าความสามารถของบัคเกตที่จะบรรจุระเหินข้อมูลมีค่าคงที่ เปอร์เซ็นต์ของระเหินข้อมูล ที่จะไปอยู่ในบัคเกตส่วนเกินจะเพิ่มขึ้น เมื่อไพรมแพคกิ้ง เดนซิตีมีค่าเพิ่มขึ้น

ถ้าต้องการให้เปอร์เซ็นต์ของการเกิดระเหินข้อมูลในบัคเกตส่วนเกินไม่เกินค่าคงที่ค่าใด ค่าหนึ่ง เช่นไม่เกิน ๑๐% ผู้ใช้จะต้องให้บัคเกตสามารถบรรจุระเหินข้อมูลได้มากขึ้น โดยให้ไพรม

แพคกิง เคนซีที่มีค่าสูงชัน

จะเห็นได้ว่า การที่จะกำหนดขนาดของโปรแกรมแพคกิง เคนซีที่ ความสามารถของบัค เกทที่จะ บรรจุนั้นจะขึ้นอยู่กับการตัดสินใจของผู้ใช้ว่าต้องการประหยัดเวลาในการค้นหาหรือประหยัด Storage

๓. แฮชซิงฟังก์ชันที่ใช้

แฮชซิงฟังก์ชันเป็นฟังก์ชันใด ๆ ที่ใช้ในการคำนวณเปลี่ยนค่าสียให้ เป็นตำแหน่งที่เก็บระเบียบ ข้อมูลบนแผ่นจานแม่เหล็กนั้นจะมีผลต่อการเกิดระเบียบข้อมูลในบัค เกทส่วนเกิน แฮชซิงฟังก์ชันบางวิธี เมื่อใช้ในการคำนวณแล้วโอกาสที่จะเกิดระเบียบข้อมูลในบัค เกทส่วนเกินมีน้อย ดังนั้นผู้ใช้จะต้องเลือก แฮชซิงฟังก์ชันที่ดี เพื่อให้ผลการคำนวณได้ระเบียบข้อมูลในบัค เกทส่วนเกินน้อยหรือไม่เกิดเลย

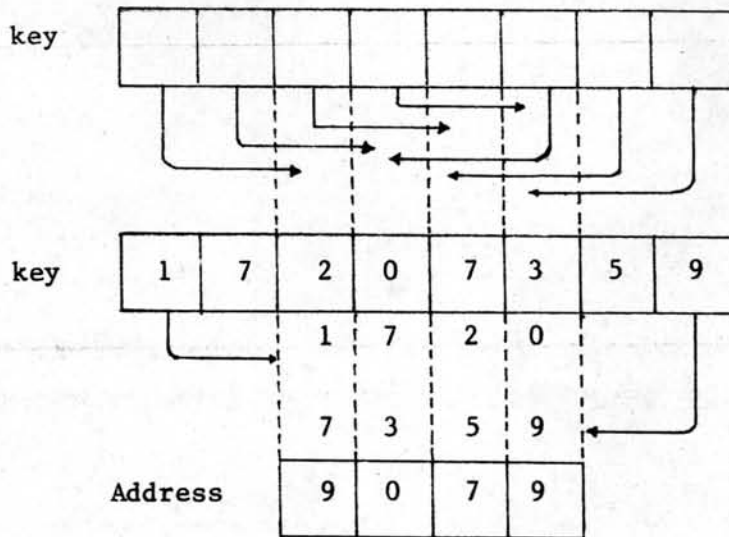
ตัวอย่างแฮชซิงฟังก์ชัน

แฮชซิงฟังก์ชันนั้นแบ่งได้หลายวิธี คือ

๑) มิดสแควร์ (Mid Square Method) วิธีนี้จะนำตัวเลขภายในคีย์มาคูณกัน เลือก เฉพาะตัวเลขส่วนกลางมาปรับ (Adjust) ให้อยู่ในช่วงของตำแหน่งที่เก็บ เช่น คีย์มีค่าเป็น ๑๗๒๑๔๔ เมื่อคูณกันแล้วจะได้ค่าเป็น ๑๒๔๖๓๔๕๓๓๐๔ นำเอาค่า ๓๔๕๓ มาปรับให้อยู่ในช่วงของ ตำแหน่งที่เก็บต่อไป

๒) การหาร (Dividing) นำคีย์มาหารด้วยเลขจำนวนหนึ่งที่ใกล้เคียงกับจำนวน ตำแหน่งที่เก็บ (โดยปกติอาจใช้ Prime number) เศษที่ได้จะใช้เป็น Relative bucket address เช่นคีย์มีค่าเป็น ๑๗๒๑๔๔ ใช้บัค เกท ๓๐๐๐ บัค เกท นำเอา ๑๗๒๑๔๔ หารด้วย ๖๔๕๗ จะได้เศษเป็น ๔๒๒๐ ซึ่งจะใช้เป็น Relative bucket address

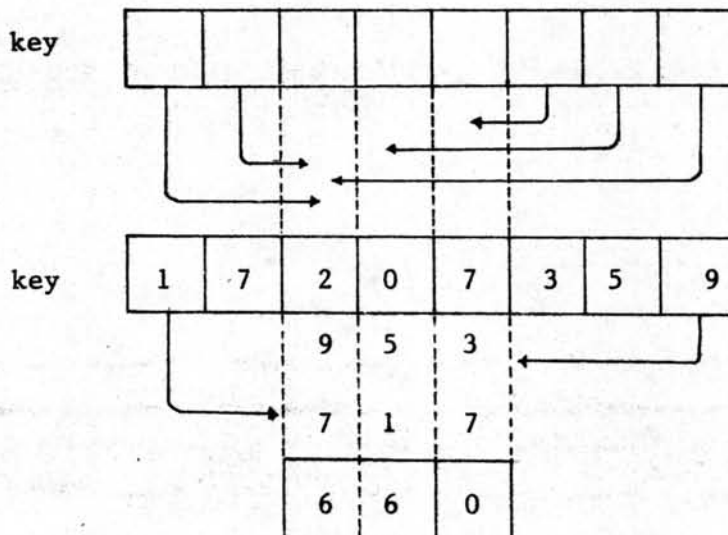
๓) ชิฟติง (Shifting) นำเอาตัวเลขนอกสุดของทั้งสองข้างของคีย์เลื่อนเข้าข้างใน โดยให้ทับกันเท่ากับความยาวของตำแหน่งที่เก็บ



รูป ค. แสดงการหาค่าแ่งที่เก็บโดยใช้วิธีฮพิง

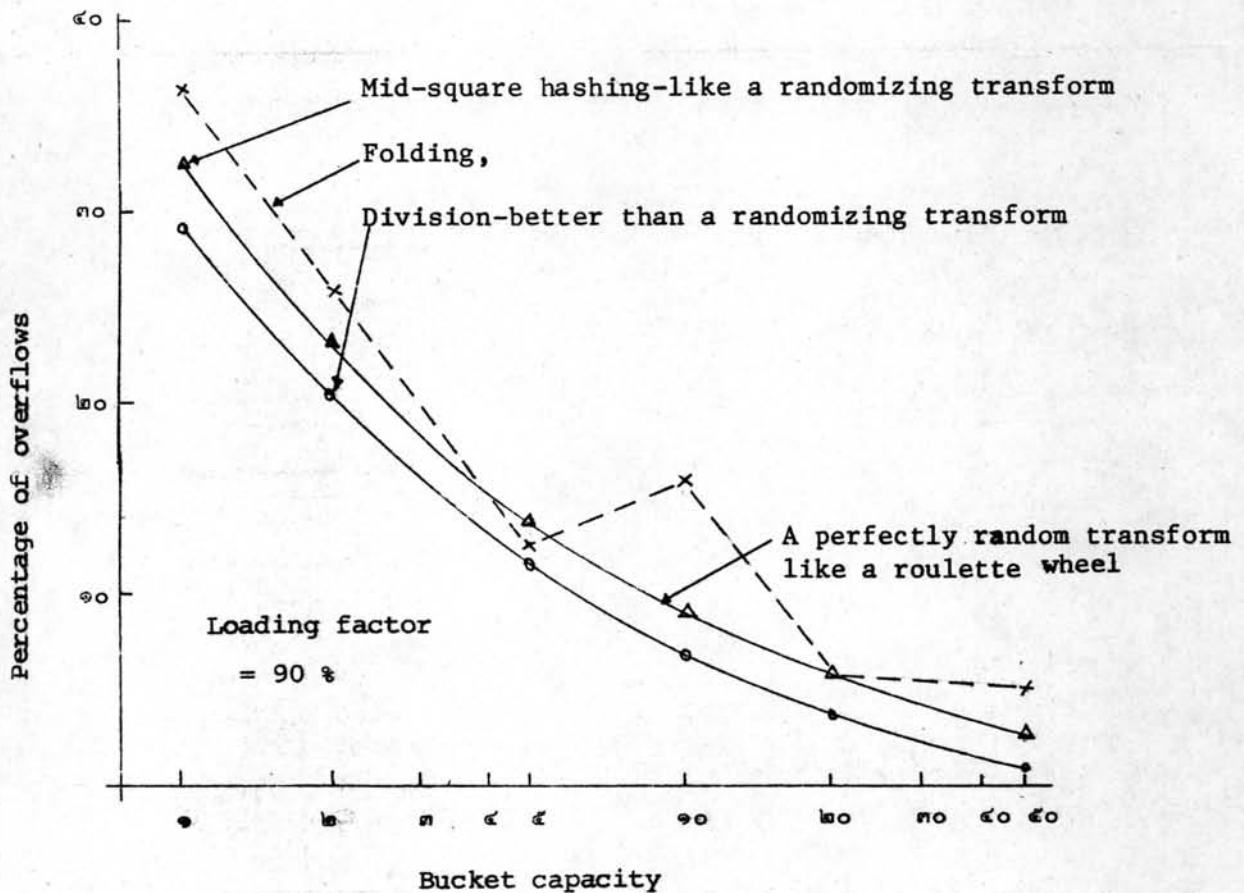
- ๔) โฟลคคิง (Folding) วิธีนี้ตัวเลขในคีย์ส่วนที่เกินความยาวของตำแหน่งที่เก็บจะถูกพับ (fold) เข้าข้างใน แล้วนำมารวมกัน วิธีนี้เหมาะสำหรับคีย์ ที่มีขนาดใหญ่

ความยาวของ Address



รูป ง. แสดงการหาค่าแ่งที่เก็บโดยใช้วิธีโฟลคคิง

จะเห็นได้ว่า แอสซิงฟังก์ชันมีหลายวิธีซึ่งผู้ใช้จะเลือกวิธีใดก็ได้ตามความเหมาะสมในแต่ละชนิดของงาน แต่อย่างไรก็ดีแอสซิงฟังก์ชันที่ดีนั้นจะต้องทำให้ระเบียนข้อมูลในบัคเกตส่วนเกินน้อยที่สุด ซึ่งได้มีการค้นคว้าเกี่ยวกับเรื่องนี้มาแล้ว และสามารถสรุปผลการค้นคว้าออกมาได้ดังแสดงในรูป จ.



จากรูป จ. จะเห็นได้ว่า ถ้า loading factor และความสามารถของบัคเกตที่จะบรรจุระเบียนข้อมูลมีค่าคงที่ คือมีค่า ๙๐% เมื่อได้แอสซิงฟังก์ชันวิธีที่ ๒ คือการหารจะทำให้มีระเบียนข้อมูลในบัคเกตส่วนเกินน้อยที่สุด

จะเห็นได้ว่าถ้าผู้ใช้สามารถเลือกแอสซิงฟังก์ชันที่เหมาะสมแล้ว โอกาสที่จะเกิดระเบียนข้อมูลในบัคเกตส่วนเกินก็มีน้อย

วิธีการจัดการระเบียบข้อมูลในบัคเกทส่วนเกิน

ดังได้กล่าวมาแล้วว่าการจัดแฟ้มข้อมูลแบบโคเรคที่ตินั้นจะต้องมีระเบียบข้อมูลในบัคเกทส่วนเกินน้อยหรือไม่มีเลย แต่ถ้าผู้ใช้ไม่สามารถหลีกเลี่ยงได้ ก็มีวิธีการที่จะจัดบัคเกทส่วนเกินเพื่อให้สามารถค้นหาระเบียบข้อมูลในบัคเกทส่วนเกินได้รวดเร็ว ซึ่งในที่นี้จะกล่าวถึงโดยแยกตามลักษณะของบัคเกทส่วนเกินดังนี้

- ก) Overflow record ใน Prime area
- ข) Seperate overflow area
- ค) Distributed Overflow

ก) Overflow ใน Prime area

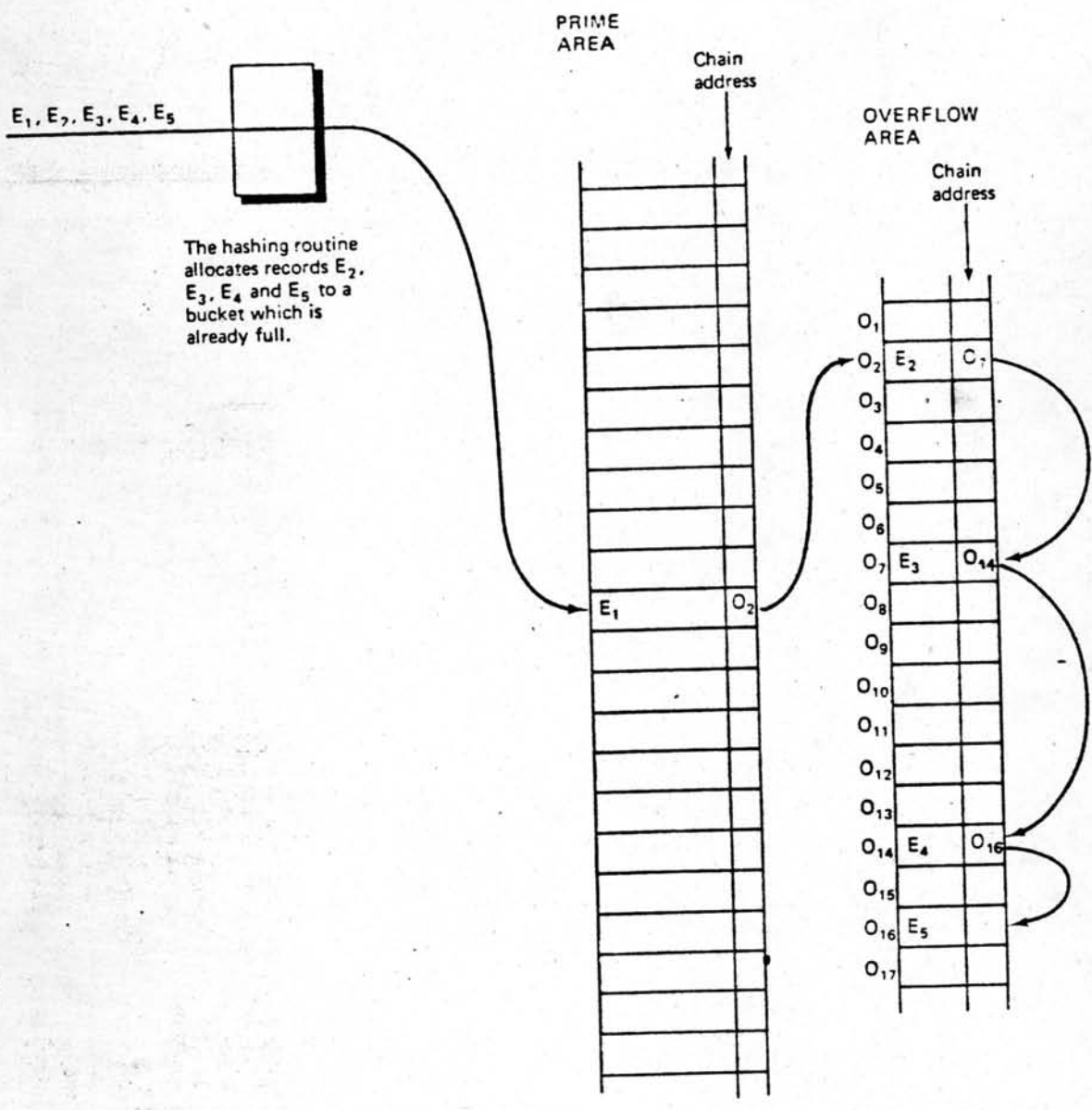
ผู้ใช้สามารถที่จะจัด Overflow ชนิดนี้ให้สามารถทำงานได้อย่างมีประสิทธิภาพ โดยอาศัยวิธีการต่าง ๆ เช่น Consecutive spill method วิธีนี้เมื่อได้ค่าตำแหน่งที่เก็บจากการคำนวณผ่านแอสซิงฟังก์ชันแล้ว ชั้นแรกจะบันทึกระเบียบข้อมูลลงในโฮมบัคเกทที่คำนวณได้ ถ้าโฮมบัคเกทเต็มก็จะบันทึกลงในโฮมบัคเกทถัดไป ซึ่งวิธีนี้จะทำให้ไม่เสียเวลาในการค้นหา

ข) Seperate Overflow area

Seperate Overflow area นี้ ผู้ใช้สามารถจัด Overflow ได้ ๒ วิธีคือ

- ๑) โดยใช้ chain
 - ๒) โดยไม่ใช้ chain
- ๑) โดยใช้ chain

เมื่อจะบันทึกระเบียบข้อมูลลงในโฮมบัคเกทที่เต็มแล้วบัคเกทส่วนเกินจะถูกนำมาใช้ โดยจะบันทึกระเบียบข้อมูลนั้นลงในบัคเกทส่วนเกิน และตำแหน่งที่เก็บในบัคเกทส่วนเกินของระเบียบข้อมูลนั้นจะถูกบันทึกลงในโฮมบัคเกท ดังแสดงในรูป ฉ.



รูป ฉ. แสดง Separate overflow ที่ใช้ chain

๒) โดยไม่ใช้ Chain

วิธีนี้แบ่งได้เป็น ๒ กรณีคือ

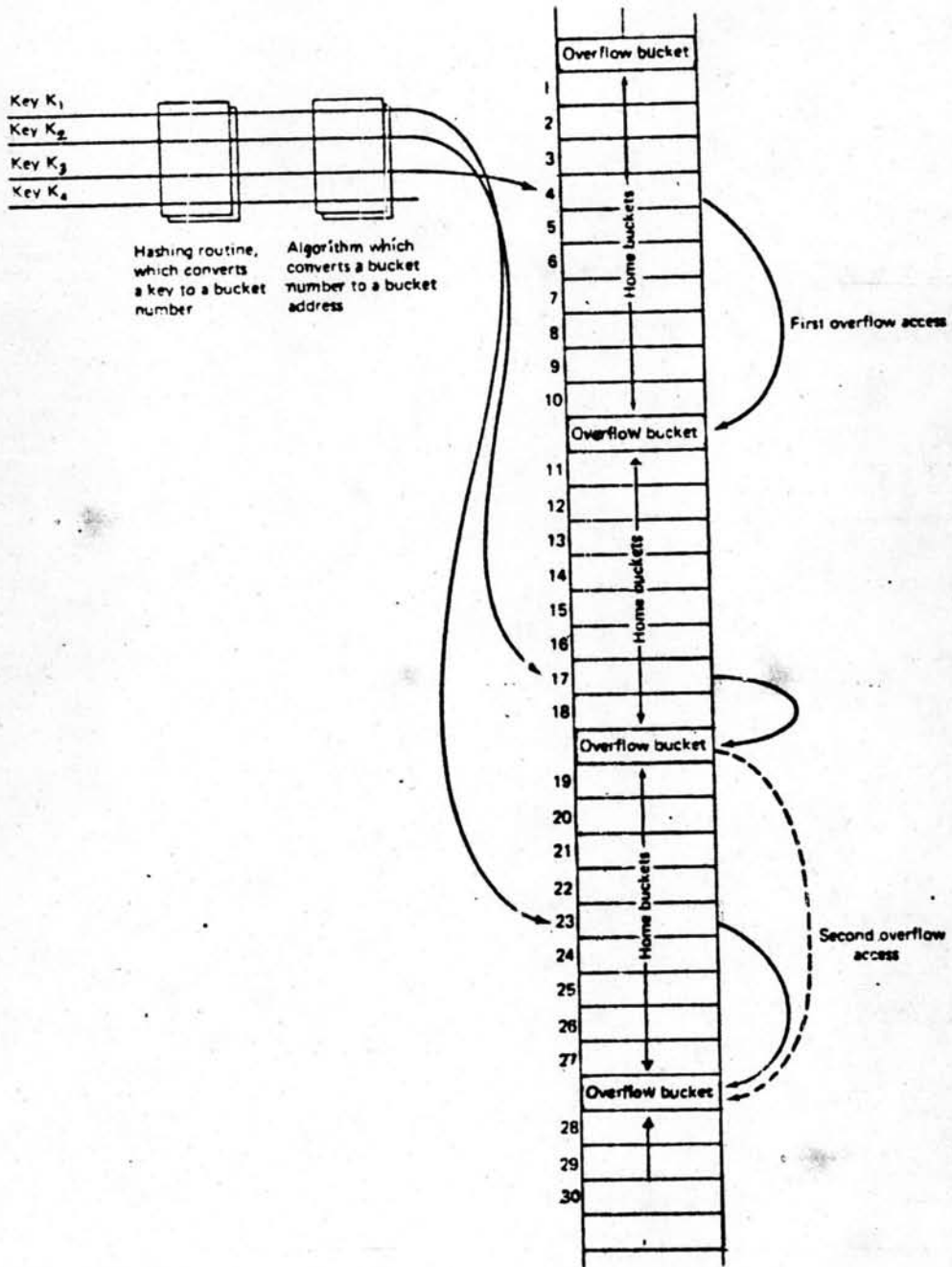
๒.๑) Overflow buckets with consecutive spill

๒.๒) Directory of empty overflow buckets

ทั้ง ๒ กรณีนี้จะมีวิธีการเช่นเดียวกับที่กล่าวมาแล้วในข้อ ก)

ค) Distributed Overflow space

การจัดพื้นที่ส่วนเกิน ชนิดนี้ จะเตรียมที่ว่างเหลือเผื่อไว้ในโคมบักเกต เลย์ถ้า โคมบักเกต ไม่สามารถบันทึกระเบียบข้อมูลได้อีกก็จะบันทึกระเบียบข้อมูลลงในบักเกตส่วนเกิน และถ้า บักเกตส่วนเกินเองเกิดเต็ม ก็จะทำบันทึกระเบียบข้อมูลลงในพื้นที่ส่วนเกินถัดไป ดังแสดงในรูป ข.



รูป ข แสดงการจัดพื้นที่เกินแบบมีที่ว่างเหลือเฟือไว้ (Distribute Overflow)

ภาคผนวก ค.

การใช้คำสั่ง Define เมื่อต้องการใช้แฟ้มข้อมูลแบบวีแรม

คำสั่ง Define ที่ใช้ สำหรับแฟ้มข้อมูลแบบวีแรมแบ่งได้เป็น ๔ อย่างคือ

๑. Define Mastercatalog
๒. Define Space
๓. Define Cluster
๔. Define Usercatalog

๑) Define Mastercatalog

ใช้คำสั่งนี้ในการกำหนดมาสเตอร์แคตตาล็อก ซึ่งจะต้องกำหนดขึ้นเป็นอันดับแรก ถ้าปราศจากการกำหนดนี้ ก็ไม่สามารถกำหนดที่ว่าง, คลัสเตอร์ หรือยูสเซอร์แคตตาล็อกได้ โดยปกติ System Programmer จะเป็นผู้กำหนด มีรูปแบบดังนี้

DEFINE MASTERCATALOG

NAME (VSAM.MASTER.CAT)-	VSAM.MASTER.CAT	= จำนวน Mastercatalog
VOL (CUDAT6)-	CUDAT6	= ชื่อของวอลุ่มที่ใช้
TRACK(238)-	238	= จำนวนแทรคที่ใช้
FILE (IJSYSCT)-	IJSYSCT	= กำหนดชื่อแฟ้มข้อมูลปกติต้องกำหนดเป็น IJSYSCT
UPDPW(C SCSUPPW)	C SCSUPPW	= ชื่อของพาสเวิร์ด

๒) Define Space

ใช้คำสั่งนี้ในการกำหนดที่ว่างสำหรับวีแรม โดยมีรูปแบบในการกำหนดดังนี้

DEFINE SPACE-

(FILE (KPSPACE)-	KPSPACE	= ชื่อแฟ้มข้อมูล
VOL (CUDAT6)-	CUDAT6	= ชื่อ วอลุ่ม
CYL(18 1)-	18	= จำนวนไซลินเดอร์ที่ใช้
)-		
CATALOG(KP.UCAT 1/KPUPPW1)	KP.UCAT1	= ชื่อของยูสเซอร์แคตตาล็อก
	KPUPPW1	= พาสเวิร์ด

๓) Define Cluster

ใช้คำสั่งนี้กำหนดลักษณะของแฟ้มข้อมูลที่จะเก็บไว้ในแฟ้มข้อมูลแบบวิ้นแชม โดยรูปแบบในการกำหนดเป็นดังนี้

DEFINE CLUSTER

(NAME (D.MSCRS.Q.FILE) -	D.MSCRS.Q.FILE	= ชื่อแฟ้มข้อมูล
MRPW (MSCRSMPW) -	MSCRSMPW	
UPDPW (MSCRSUPW) -	MSCRSUPW	= พาสเวิร์ด
RDPW (MSCRSRPW) -	MSCRSRPW	
FILE (KPSPACE) -	KPSPACE	= ชื่อของที่ว่างที่กำหนดไว้
CYL(6 1) -	6	= จำนวนไซลินเดอร์ที่ใช้
FOR(15) -	15	= จำนวนวันที่เก็บแฟ้มข้อมูลไว้
VOL(CUDAT6) -	CUDAT6	= ชื่อของวอลุ่มที่ใช้
RECORDSIZE (25 25)	25	= ขนาดของระเบียนข้อมูล ในแฟ้มข้อมูล
KEY (6 0) -	6	= ความยาวของคีย์ที่ใช้
FREESPACE (50 50) -	50	= จำนวน% ของที่ว่างที่เก็บไว้
) -		
DATA -		
(NAME (D.MSCRS.Q.DATA) -	D.MSCRS.Q.DATA	= ชื่อของระเบียนข้อมูล
) -		
INDEX -		
(NAME (D.MSCRS.Q.KEY) -	D.MSCRS.Q.KEY	= ชื่อของคีย์
) -		
CATALOG (KP.UCAT1/KPUPPW1)	KP.UCAT1	= ชื่อของยูสเซอร์แคตตาล็อก
	KPUPPW1	= ชื่อของพาสเวิร์ด

๔) Define Usercatalog

สำหรับยูสเซอร์แคตตาลอกนี้จะกำหนดหรือไม่ก็ได้ แล้วแต่ผู้ใช้ ถ้ากำหนดจะมีรูปแบบดังนี้

DEFINE UCAT-

(NAME (KP.UCAT1) -	KP.UCAT1	=	ชื่อของยูสเซอร์แคตตาลอก
MRPW (KPMRDW1) -	KPMRPW1	=	ชื่อพาสเวิร์ด
UPDPW (KPUPPW1) -	KPUPPW1		
FILE (KPUCAT1) -	KPUCAT1	=	ชื่อของแฟ้มข้อมูล
VOL (CUDAT6) -	CUDAT6	=	ชื่อของวอลุ่มที่ใช้
TRACKS (48 2) -	48	=	จำนวนแทรค
RVBL TO (80330) -	80330	=	จำนวนวันที่เก็บไว้
OWNER (RATANA) -	RATANA	=	ชื่อผู้ใช้
CATALOG (VSAM.MASTER.CAT/C -	VSAM.MASTER.CAT	=	ชื่อมาสเตอร์แคตตาลอก
(VSAM.MASTER.CAT/C SC*UPPW)	CSC*UPPW	=	ชื่อพาสเวิร์ด

ภาคผนวก ง

```

0001 C * * * * *
0002 C * THIS PROGRAM USE TO CREATE SEQUENTIAL FILE ON DISK 3360 *
0003 C * * * * *
0004 DIMENSION CCSI(4),NAME(9)
0005 INTEGER CN,S,COST,A,CD,AD,GD,B,SM,YR,SN,PN
0006 DATA IA,IR /'A','R'/
0007 INPUT=0
0008 IOPTC=0
0009 C READ DATA FROM TAPE-FILE
0010 5 READ(13,20,END=35)IED,CN,S,COST,A,CD,AD,GD,MJ,D,IO,NAME,SM,YR,SN,
0011 PN
0012 20 FORMAT(A4,A2,I2,4A4,I2,4A4,A2,I2,A1,A1,I4,A1,I6,9A4,I1,I2,I3,I7)
0013 INPUT=INPUTC+1
0014 IF(B.EQ.1A1) GO TO 30
0015 IC2=2000000
0016 IDAR =ID+IC2
0017 C WRITE DATA ON DISK 3360
0018 25 WRITE(14,26)IED,CN,S,COST,A,CD,AD,GD,MJ,IDAB,NAME,SM,YR,SN,PN
0019 26 FORMAT(A4,A2,I2,4A4,A2,I2,4A4,A2,I2,A1,A1,I4,I4,I7,9A4,I1,I2,I3,I7)
0020 30 IC1 =1000000
0021 IDAR =ID+IC1
0022 60 TO 25
0023 35 ENDFILE 14
0024 REMIND 14
0025 C PRINT CONCLUSION
0026 WRITE(3,40)
0027 40 FORMAT(14,'CREATE SEQUENTIAL FILE')
0028 WRITE(3,50)
0029 50 FORMAT(74,'-----')
0030 WRITE(3,60) INPUTC
0031 60 FORMAT(135,'TOTAL INPUT RECORD = ',I6)
0032 WRITE(3,70) IOPTC
0033 70 FORMAT(135,'TOTAL OUTPUT RECORD = ',I6)
0034 WRITE(3,80)
0035 80 FORMAT(135,'PROGRAM BY MISS FATANA BENJAPOLKUL')
0036 STOP
0037 END

```



```

C
0001 MAIN PROGRAM
0002 DIMENSION COST(4),NAME(9),IPNT1(16),IPNT2(91),OPCODE(10)
0003 DIMENSION DEPT(10,8),MJCDE(10),MAJR(10,8),DEP(8),MAJ(8)
0004 INTEGER CN,S,COST,A,CD,AD,GD,B,SN,YR,SN,PN
0005 INTEGER OPCODE,DEPT,MJCDE,MAJR,DEP,MAJ
0006 DATA IPNT1,IPNT2 /16*1H*,91*1H-/
0007 IMPT=0
0008 IOPT=0
0009 IERR=0
0010 IFCO=0
0011 CALL DEPART(OPCODE,DEPT)
0012 CALL MAJOR(MJCDE,MAJ)
0013 5 READ(1,10,END=65)TRC,IOC,IOPT,IMJ
0014 10 FORMAT (A1,I4,I3,A4)
0015 IMPT=IMPT+1
0016 CALL CCNV(TRC,IOC,IOPT)
0017 6 READ(1,20,END=65)IED,CN,S,COST,A,CD,AD,GD,MJ,IOA,NAME,SM,YR,SN,PN
0018 20 FORMAT (A4,A2,I2,4A6,A2,I2,A1,A1,I4,I7,9A4,I1,I2,I3,I7)
0019 45 IF (IOA-IDA) 30,40,50
0020 30 IERR=IERR+1
0021 WRITE(12,15) IOB
0022 15 FORMAT (I40,'ERROR-TRANSACTION# 17')
0023 GO TO 5
0024 40 DO25 I=1,IO
0025 IF (IOA.NE.MJCDE(I)) GO TO 25
0026 DC 26J=1,8
0027 26 DEP(J)=DEPT(I,J)
0028 GO TO 27
0029 25 CONTINUE
0030 GO TO 35
0031 27 DO51 I=1,IC
0032 IF (IMJ.NE.MJCDE(I)) GO TO 55
0033 DE 54J=1,8
0034 MAJ(J)=MAJR(I,J)
0035 GO TO 57
0036 55 CONTINUE
0037 GO TO 35
0038 57 WRITE(3,60)
0039 60 FORMAT (H1,I59,'STUDENT REPORT')
0040 IOPT=IOPT+1
0041 WRITE(3,70) IPNT1
0042 WRITE(3,80) DEP,MAJ
0043 80 FORMAT (H2,T34,'DEPARTMENT OF',3X,8A6,'MAJOR',5X,8A4)
0044 WRITE(3,90) IOC,IOA,IOB
0045 90 FORMAT (H2,T24,'NUMBER',3X,A1,I5,5X,'NAME',5X,9A4)
0046 WRITE(3,95) IPNT2
0047 95 FORMAT (H2,T23,91A1)
0048 WRITE(3,100)
0049 100 FORMAT (H2,T73,'COURSE-NO',8X,'COURSE-NAME',5X,'SECTION',5X,
- 'CREDIT',5X,'AUDIT',3X,'GRADE',5X,'SEMIESTEP',5X,'AVEAGE')
0050 WRITE(3,95) IPNT2
0051 105 WRITE(3,110) IED,CN,S,COST,A,CD,AD,GD,SM,YR
0052 110 FORMAT (23X,A4,A2,7X,4A4,A2,4X,I2,2X,I2,9X,A1,7X,A1,10X,I1,10X,I2)

```

```

0053 READ(14,20,END=65)ID,CH,S,COST,A,CO,AD,GO,MJ,IDA,NAME,SM,YR,SN,PI
0054 IF(IOR.EQ.IDA) GO TO 105
0055 GO TO 5
0056
0057 35 IERR=IERRD+1
0058 WRITE(3,120)IJ,IOP
0059 120 FORMAT(7C,1ERRDP-DEPT-MAJOR,A4,3X,13)
0060 GO TO 5
0061 50 GO TO 6
0062 65 WRITE(3,130) INPT
0063 130 FORMAT(1H1,140,1TOTAL TRANSACTION = ,17)
0064 140 WRITE(3,140) IERR
0065 140 FORMAT(140,1TOTAL ERRDP CARD = ,17)
0066 WRITE(3,150) IERR
0067 150 FORMAT(140,1TOTAL ERR DEPT = ,17)
0068 WRITE(3,160)IOP
0069 160 FORMAT(140,1TOTAL WRITE REPORT = ,17)
0070 WRITE(3,170)
0071 170 FORMAT(140,1PROGRAM BY MESS PATANA BENJAPKUL*)
0072 STOP
0073 END
    
```



```

0001 C SUBROUTINE SUBPROGRAM NAMED CONV
0002 SUBROUTINE CONV(BC, ID, IOR )
0003 DIMENSION COSTC(4), NAMEC(9), COST(4), NAME(9)
0004 DATA TA, TB, /A, /B, /
0005 IF (BC.FQ.TA) GO TO 35
0006 IC2 = 2000000
0007 IDB = ID+IC2
0008 RETURN
0009 35 IC1 = 1000000
0010 IOR = ID+IC1
0011 RETURN
0012 END

```

```

0001 C SUBROUTINE SUBPROGRAM NAMED DEPART
0002 SUBROUTINE DEPART(DPCODE,DEPT)
0003 INTEGER DPCODE(10), DEPT(10,8)
0004 DO 201=1,10
0005 20 READ(1,10)DPCODE(1),(DEPT(1,J),J=1,8)
0006 10 FORMAT(13,8A4)
0007 RETURN
0008 END

```



```

0001 C SUBROUTINE SUBPRGPM NAMED MAJOR
0002 SUBROUTINE MAJOR(MJCODE,MAJR)
0003 INTEGER MJCODE(10),MAJR(10,8)
0004 DO 30 I=1,10
0005 30 READ(1,40) MJCODE(I),(MAJR(I,J),J=1,8)
0006 40 FORMAT(A6,8A1)
0007 RETURN
0008 END

```

```

00001 ID DIVISION.
00002 PROGRAM-ID. CREATE.
00003 AUTHOR. MISS PATANA BENJAPOLKUL.
00004 DATE-COMPILED. 24/12/80
00005 REMARKS.
00006 * * * * *
00007 * THIS PROGRAM USE TO CREATE *
00008 * SEQUENTIAL FILE ON DISK 3340 *
00009 * * * * *
00010 ENVIRONMENT DIVISION.
00011 CONFIGURATION SECTION.
00012 OBJECT-COMPUTER. IBM-370.
00013 INPUT-OUTPUT SECTION.
00014 FILE-CONTROL.
00015 SELECT TAPE-MS-I-FILE ASSIGN TO SYS011-UT-2400-S.
00016 SELECT D-MS-D-FILE ASSIGN TO SYS013-DA-3340-S-3AM.
00017 DATA
00018 FILE TAPE-MS-I-FILE
00019 BLOCK CONTAINS 40 RECORDS
00020 RECORD CONTAINS 90 CHARACTERS
00021 RECORDING MODE IS F
00022 LABEL RECFD IS OMITTED
00023 DATA RECFD IS TAPE-MS-I-REC.
00024 PIC X(90).
00025 FD D-MS-D-FILE
00026 BLOCK CONTAINS 40 RECORDS
00027 RECORD CONTAINS 90 CHARACTERS
00028 RECORDING MODE IS F
00029 LABEL RECFD IS STANDARD
00030 DATA RECFD IS D-MS-D-REC.
00031 PIC X(90).
00032 WORKING-STORAGE SECTION.
00033 77 INPUT-REC-CNT PIC 9(5) COMP VALUE ZER0.
00034 77 OUTPUT-REC-CNT PIC 9(5) COMP VALUE ZER0.
00035 77 INPUT-ERROR-CNT PIC 9(5) COMP VALUE ZER0.
00036 77 INVALID-REC-CNT PIC 9(5) COMP VALUE ZER0.
00037 01 SWITCH-AREA.
00038 05 VALID-SWITCH-A PIC X.
00039 05 BR VALID-SWITCH VALUE 'Y'.
00040 05 WS-TAPE-I-SWITCH PIC X.
00041 05 WS-TAPE-SWITCH VALUE 'Y'.
00042 01 WS-TAPE-I-REC.
00043 03 TAPE-COURSE-NO PIC X(6).
00044 03 TAPE-COURSE PIC X(28).
00045 03 TAPE-STUDENT-ID.
00046 05 ID-ALPHA PIC X.
00047 05 ID-NUMERIC PIC X(6).
00048 03 TAPE-STUDENT PIC X(49).
00049 PROCEDURE DIVISION.
00050 * MAIN LEVEL STARTING HERE
00051 * ---- INITIALIZATION ----
00052 * OPEN FILES
00053 OPEN INPUT TAPE-MS-I-FILE

```

2 CREATE 14.15.51 24/12/80

```

00054 OUTPUT D-MS-O-FILE.
00055 MAIN PROCESS
00056 010-READ-TAPE-CHECK THRU READ-TAPE-EXT
00057 UNTIL WS-TAPE-SWITCH.
00058 CONCLUSION
00059 * DISPLAY * TOTAL INPUT RECORD = * INPUT-REC-CNT.
00060 * DISPLAY *
00061 * DISPLAY *
00062 * DISPLAY * TOTAL OUTPUT RECORD = * OUTPUT-REC-CNT.
00063 * DISPLAY *
00064 * DISPLAY *
00065 * DISPLAY * TOTAL INPUT ERROR = * INPUT-ERROR-CNT.
00066 * DISPLAY *
00067 * DISPLAY * TOTAL INVALID RECORD = * INVALID-REC-CNT.
00068 * DISPLAY *
00069 * DISPLAY *
00070 * DISPLAY *
00071 * DISPLAY * PROGRAMMED BY RATANA BENJAPOLKUL.
00072 * DISPLAY *
00073 * DISPLAY * EOJ.
00074 * CLOSE TAPE-MS-I-FILE
00075 * D-MS-O-FILE.
00076 * STOP RUN.
00077 010-READ-TAPE-CHECK.
00078 PEAD TAPE-MS-I-FILE INTO WS-TAPE-I-REC
00079 AT END
00080 MOVE *Y* TO WS-TAPE-I-SWITCH,
00081 GC TO PEAD-TAPE-EXT.
00082 ADD 1 TO INPUT-REC-CNT.
00083 IF ( I0-ALPHA IS ALPHABETIC AND
00084 I0-NUMERIC IS NUMERIC )
00085 WRITE D-MS-O-REC FROM WS-TAPE-I-REC
00086 ADD 1 TO OUTPUT-REC-CNT
00087 ELSE
00088 DISPLAY 'INPUT RECORD ERROR' WS-TAPE-I-REC
00089 ADD 1 TO INPUT-ERROR-CNT.
00090 READ-TAPE-EXT. EXIT.
00091 * *** END FILE ***

```

* STATISTICS *
* STATISTICS *
* OPTIONS IN EFFECT *
* OPTIONS IN EFFECT *
* OPTIONS IN EFFECT *
* OPTIONS IN EFFECT *
* LISTER OPTIONS *

SOURCE RECORDS = 91 DATA ITEMS = 20
PARTITION SIZE = 524168 LINE COUNT = 56
PMAP RELOC ADR = NONE SPACING =
NOLISTX APOST NO SYM NO CATALD
NOCLIST FLAGH ZMB NO SUPMAP
NOSTATE TRUNC SEQ NO SYMAMP
NONE NOCOUNT

PACC DIV SZ =
512 BUFFER SIZE =
NONE FLDK
NO SIXTT LIMP
NO SXEFP ERRS
NO SYNTX NOVERB
NOCLV NOVERBSUM NOVERDEF

CBL STATE	TO	DIVISION.	
CBL LIB	PROGRAM-ID.	RETRIEVE-SAM.	
00001	AUTHOR.	MISS KATANA BENJAPOLKUL.	
00002	DATE-COMPILED.	24/04/80.	
00003	REMARKS.	*****	
00004		* THIS PROGRAM USE TO RETRIEVE *	
00005		* SEQUENTIAL FILE ON DISK 3340 *	
00006		*****	
00007			
00008			
00009	ENVIRONMENT	DIVISION.	
00010	CONFIGURATION	SECTION.	
00011	OBJECT-COMPUTER.	IBM-370.	
00012	INPUT-OUTPUT	SECTION.	
00013	FILE-CONTROL.		
00014	SELECT	CRD-FILE	ASSIGN TO SYS004-UR-2540R-S.
00015	SELECT	PRT-FILE	ASSIGN TO SYS006-UR-1403-S.
00016	SELECT	D-MS-I-FILE	ASSIGN TO SYS013-DA-3340-S-SAM.
00017	DATA	DIVISION.	
00018	FILE	SECTION.	
00019	FD	CRD-FILE	
00020	RECORD	CONTAINS	80 CHARACTERS
00021	LABEL	MODE	IS F
00022	DATA	RECORD	IS OMITTED
00023	01	RECORD	IS CRO-REC.
00024	FD	CRD-REC	PIC X(80).
00025	RECORD	CONTAINS	132 CHARACTERS
00026	LABEL	MODE	IS F
00027	DATA	RECORD	IS OMITTED
00028	01	RECORD	IS PRT-REC.
00029	FD	D-MS-I-FILE	PIC X(132).
00030	RECORD	CONTAINS	40 RECORDS
00031	LABEL	CONTAINS	90 CHARACTERS
00032	DATA	MODE	IS F
00033	01	RECORD	IS STANDARD
00034	FD	D-MS-I-REC.	PIC X(61).
00035	RECORD	03	PIC XX.
00036	LABEL	03	PIC X(18).
00037	DATA	03	PIC XX.
00038	01	D-COURSE-NO	PIC X.
00039	03	D-SECTION	PIC X.
00040	03	D-COURSE-NAME	PIC X(18).
00041	03	D-CREDIT	PIC XX.
00042	03	D-AUDIT	PIC X.
00043	03	D-GRADE	PIC X.
00044	03	STUDENT.	
00045	05	MAJOR.	
00046	07	DEPT	PIC XXX.
00047	07	FILLER	PIC X.
00048	03	D-STUDENT-ID	PIC X(7).
00049	03	D-STUDENT-NAME	PIC X(36).
00050	03	D-SEMESTER	PIC X.
00051	03	D-YEAR	PIC XX.

2 RETRIEVE 12-08-57 24/04/80

00052	03 FILLER	PIC	X(10)
00053	WORKING-STORAGE SECTION.	PIC	X(10)
00054	77 TRANS-CNT	PIC	9(5) COMP VALUE ZER0.
00055	77 INPUT-REC-CNT	PIC	9(5) COMP VALUE ZER0.
00056	77 PRT-REC-CNT	PIC	9(5) COMP VALUE ZER0.
00057	77 ERROR-CARD-CNT	PIC	9(5) COMP VALUE ZER0.
00058	77 ERROR-SEQ-CNT	PIC	9(5) COMP VALUE ZER0.
00059	77 MAJRN	PIC	X(13) VALUE SPACE.
00060	77 DEPT-N	PIC	X(32) VALUE SPACE.
00061	77 COMPARE-ID	PIC	X(17) VALUE *B25187*.
00062	01 DNAME COPY CYREDEPT SUPPRESS.		
00217	01 DNAME REDEFINES DNAME	CPY	CYRECADP SUPPRESS.
00224	01 MNAME COPY CYREMAJR SUPPRESS.		
00335	01 MAJNAME REDEFINES MNAME	CPY	CYRECAHJ SUPPRESS.
00342	01 HEAD1.		
00343	03 FILLER	PIC	X(58) VALUE SPACE.
00344	03 FILLER	PIC	X(15) VALUE *STUDENT REPORT*.
00345	03 FILLER	PIC	X(59) VALUE SPACE.
00346	01 HEAD2.		
00349	03 FILLER	PIC	X(33) VALUE SPACE.
00350	03 FILLER	PIC	X(13) VALUE *DEPARTMENT OF*.
00351	03 FILLER	PIC	XXX VALUE SPACE.
00352	03 HEAD2-1	PIC	X(32) VALUE SPACE.
00353	03 FILLER	PIC	X(5) VALUE *MAJOR*.
00354	03 HEAD2-2	PIC	X(15) VALUE SPACE.
00355	03 FILLER		
00356	01 HEAD3.		
00357	03 FILLER	PIC	X(33) VALUE SPACE.
00358	03 FILLER	PIC	X(9) VALUE *TO NUMBER*.
00359	03 FILLER	PIC	XXX VALUE SPACE.
00360	03 HEAD3-1	PIC	X(12) VALUE SPACE.
00361	03 FILLER	PIC	X(4) VALUE *NAME*.
00362	03 HEAD3-2	PIC	X(5) VALUE SPACE.
00363	03 FILLER	PIC	X(36) VALUE SPACE.
00364	01 HEAD4.		
00365	03 FILLER	PIC	X(30) VALUE SPACE.
00366	03 FILLER	PIC	X(22) VALUE SPACE.
00367	03 FILLER	PIC	X(9) VALUE *COURSE-NO*.
00368	03 FILLER	PIC	X(8) VALUE SPACE.
00369	03 FILLER	PIC	X(11) VALUE *COURSE-NAME*.
00370	03 FILLER	PIC	X(5) VALUE SPACE.
00371	03 FILLER	PIC	X(7) VALUE *SECTION*.
00372	03 FILLER	PIC	X(5) VALUE SPACE.
00373	03 FILLER	PIC	X(6) VALUE *CREDIT*.
00374	03 FILLER	PIC	X(5) VALUE SPACE.
00375	03 FILLER	PIC	X(5) VALUE *AUDIT*.
00376	03 FILLER	PIC	XXX VALUE SPACE.
00377	03 FILLER	PIC	X(5) VALUE *GRADE*.
00378	03 FILLER	PIC	X(5) VALUE SPACE.
00379	03 FILLER	PIC	X(8) VALUE *SEMESTER*.
00380	03 FILLER	PIC	X(5) VALUE SPACE.
00381	03 FILLER	PIC	X(4) VALUE *YEAR*.
00382	03 FILLER	PIC	X(19) VALUE SPACE.


```

00381 01 HEAD5.
00382 03 FILLER PIC X(123) VALUE SPACE.
00383 03 PFT-COURSE-NO PIC X(16).
00384 03 FILLER PIC X(7).
00385 03 PFT-COURSE-NAME PIC X(118).
00386 03 FILLER PIC X(14).
00387 03 PFT-SECTION PIC XX.
00388 03 FILLER PIC X(9).
00389 03 PRT-CREDIT PIC XX.
00390 03 FILLER PIC X(9).
00391 03 PRT-AUDIT PIC X.
00392 03 FILLER PIC X(17).
00393 03 PRT-GRADE PIC X.
00394 03 FILLER PIC X(10).
00395 03 PRT-SEMESTER PIC X.
00396 03 FILLER PIC X(10).
00397 03 PRT-YEAR PIC XX.
00398 03 FILLER PIC X(30).
00399 01 HEAD6.
00400 03 FILLER PIC X(158) VALUE SPACE.
00401 03 FILLER PIC X(16).
00402 01 HEAD7.
00403 03 FILLER PIC X(22).
00404 03 S PIC X(91).
00405 03 FILLER PIC X(58).
00406 01 #5-CRD-REC.
00407 03 CPD-STUDENT-ID.
00408 05 ID-ALPHA PIC X.
00409 05 ID-NUMERIC PIC 9(6).
00410 01 SWITCH-AREA.
00411 05 TRANS-VALID-SWITCH-A PIC X.
00412 08 TRANS-VALID-SWITCH VALUE YY.
00413 05 WS-CRD-SWITCH-A PIC X.
00414 08 WS-CRD-SWITCH VALUE YY.
PROCEDURE DIVISION.
* MAIN LEVEL STARTING HERE
* ---- INITIALIZATION ----
* OPEN FILES
OPEN INPUT CRD-FILE
D-MS-I-FILE
OUTPUT PRT-FILE.
* ---- MAIN PROCESS ----
PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
UNTIL TRANS-VALID-SWITCH.
PERFORM 020-READ-MASTER THRU 020-READ-MASTER-EXT.
PERFORM 030-COMPARE-MS-CRD THRU 030-COMPARE-MS-CRD-EXT
UNTIL MS-CRD-SWITCH.
* ---- CONCLUSION ----
DISPLAY 'TOTAL READ TRANS' = ' TRANS-CNT.
DISPLAY 'TOTAL D-MS-I-RECORD' = ' INPUT-REC-CNT.
DISPLAY 'TOTAL WRITE REPORT' = ' PRT-REC-CNT.
DISPLAY 'TOTAL ERROR CARD' = ' ERROR-CARD-CNT.
DISPLAY 'TOTAL ERROR SEQ CARD' = ' ERROR-SEQ-CNT.
00428
00429
00430
00431
00432
00433

```

RETRIEVE 12.08.57 24/04/80

BENJAPOLKUL.

```

00434 DISPLAY *PROGRAMMED BY MISS RATANA BENJAPOLKUL.
00435 CRD-FILE
00436 CLOSE
00437 D-MS-I-FILE
00438 PRT-FILE.
00439 RUN.
00440 STJP SECTION. COPY CYREFDEP.
00441 DEP-FF SECTION.
00442 DEP-FIND SECTION.
00443 SEARCH ALL DTNAME
00444 AT END GO TO NOT-F-DEP
00445 WHEN DEPT = DND (D)
00446 GO TO FOUND-D.
00447 FOUND-D.
00448 MOVE D-NAME (D) TC DEPT-N.
00449 GO TO FIND-D.
00450 NOT-F-DEP.
00451 MOVE SPACE TO DEPT-N.
00452 FIND-D.
00453 EXIT.
00454 MAJ-FF SECTION. COPY CYREFMAJ.
00455 MAJ-FIND SECTION.
00456 SEARCH ALL MNAME
00457 AT END GO TO NOT-FOUND-M
00458 WHEN MAJOF = MND (A)
00459 GO TO FOUND-M.
00460 FOUND-M.
00461 MOVE M-NAME (A) TC MAJORN.
00462 GO TO FIND-E.
00463 NOT-FOUND-M.
00464 MOVE SPACE TO MAJORN.
00465 FIND-E.
00466 EXIT.
00467 010-READ-TRANS.
00468 READ CRD-FILE INTO MS-CRD-REC
00469 AT END
00470 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00471 MOVE 'Y' TO MS-CRD-SWITCH-A
00472 GO TO 010-READ-TRANS-EXT.
00473 ADD 1 TO TRANS-CNT.
00474 IF (ID-ALPHA AND
00475 ID-NUMERIC IS ALPHABETIC AND
00476 CRD-STUDENT-ID < COMPARE-ID) IS NUMERIC AND
00477 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00478 ELSE
00479 DISPLAY 'ERROR-CARD' MS-CRD-REC
00480 MOVE 'N' TO TRANS-VALID-SWITCH-A
00481 ADD 1 TO ERRGR-CARD-CNT.
00482 010-READ-TRANS-EXT. EXIT.
00483 020-READ-MASTER.
00484 READ D-MS-I-FILE
00485 AT END GO TO 020-READ-MASTER-EXT.
00486

```

```

00487 ADD 1 TO INPUT-REC-CNT.
00488 020-REAJ-MASTER-EXT. EXIT.
00489 030-COMPARE-MS-CRD.
00490 IF CRD-STUDENT-ID = D-STUDENT-ID
00491 PERFORM 040-WRITE-HEAD
00492 PERFORM 060-COMPARE-MS-CRD THRU 060-COMPARE-EXT
00493 UNTIL CRD-STUDENT-ID NOT = D-STUDENT-ID
00494 MOVE *N* TO TRANS-VALID-SWITCH-A
00495 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00496 UNTIL TRANS-VALID-SWITCH

```

```

00497 ELSE IF CRD-STUDENT-ID < D-STUDENT-ID
00498 DISPLAY *ERROR-SEQ-CARD* MS-CRD-FEC
00499 ADD 1 TO ERROR-SEQ-CNT
00500 MOVE *N* TO TRANS-VALID-SWITCH-A
00501 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00502 UNTIL TRANS-VALID-SWITCH
00503

```

```

00504 ELSE PERFORM 020-READ-MASTER THRU 020-READ-MASTER-EXT.
00505
00506 030-COMPARE-MS-CRD-EXT. EXIT.
00507 040-WRITE-HEAD.

```

```

00508 ADD 1 TO PRT-REC-CNT.
00509 WRITE PRT-REC FROM HEAD1 AFTER POSITIONING 0 LINES.
00510 WRITE PRT-REC FROM HEAD6 AFTER POSITIONING 1 LINES.
00511 PERFORM FIND-DEP THRU FIND-D.
00512 PERFORM FIND-MAJ THRU FIND-E.
00513 MOVE DEPT-N TO HEAD2-1.
00514 MOVE MAJOPN TO HEAD2-2.
00515 WRITE PRT-REC FROM HEAD2 AFTER POSITIONING 2 LINES.
00516 MOVE D-STUDENT-ID TO HEAD3-1.
00517 MOVE D-STUDENT-NAME TO HEAD3-2.
00518 WRITE PRT-REC FROM HEAD3 AFTER POSITIONING 2 LINES.
00519 MOVE ALL *S* TO S.
00520 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
00521 WRITE PRT-REC FROM HEAD4 AFTER POSITIONING 2 LINES.
00522 MOVE ALL *S* TO S.
00523 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
00524 050-WRITE-MESSAGE.

```

```

00525 MOVE D-COURSE-NO TO PRT-COURSE-NO.
00526 MOVE D-COURSE-NAME TO PRT-COURSE-NAME.
00527 MOVE D-SECTION TO PRT-SECTION.
00528 MOVE D-CREDIT TO PRT-CREDIT.
00529 MOVE D-GRADE TO PRT-GRADE.
00530 MOVE D-SEMESTER TO PRT-SEMESTER.
00531 MOVE D-YEAR TO PRT-YEAR.
00532 WRITE PRT-REC FROM HEAD5 AFTER POSITIONING 2 LINES.
00533 060-COMPARE-MS-CRD.

```

```

00534 PERFORM 050-WRITE-MESSAGE.
00535 PERFORM 020-READ-MASTER THRU 020-READ-MASTER-EXT.
00536 060-COMPARE-EXT. EXIT.

```


LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000000				1	PRINT CN,ACGEN
000000	05C6			2	START 0
000000				3	BALR 12,0
000000				4	USING #,12
000002	5880 C266			5	USING BEGIN+4C96,11
000006	58A0 C26A			6	USING BEGIN+8192,10
				7	L 11,X1
				8	L 10,X2
				9	OPEN INFILE,INFILE1,PRINT
				19	LGOP
				25	GET INFILE1,RECCRC
				25	PUT INFILE,RECCRD
				31	LOOP
				32	EOFTAPE
				31	PRINT,SK,1
				38	MVC PRINTAREA,SPACES
000054	0283 8EF3 8EF2 01EF3 C1EF2			39	MVC LINE,ENCLINE
00005A	0217 8F29 8F77 01F29 01F77			40	PUT PRINT,PRINTAREA
				46	EOFDISK
				56	EOJ
				59	FILE
				210	INFILE1
					PRINT
					BLKSIZE=3600,
					DEVADDR=SYSCLG,
					FILEBL=NO,
					ECFADDR=ECFTAPE,
					ICAREAL=BUFFERIN,
					RECFORM=FIXBLK,
					RECSIZE=90,
					TYPEFL=INPLT,
					WORKA=YES
				244	INFILE DTFS
					BLKSIZE=2608,
					DEVADDR=SYSOL4,
					DEVICE=334C,
					IOAREAL=BUFFEROUT,
					RECFRM=FIXBLK,
					RECSIZE=90,
					TYPEFL=OUTPUT,
					WORKA=YES
				295	X1 DC A(BEGIN+4096)
				296	X2 DC A(BEGIN+8192)
				297	BUFFERIN DS CL3600
				298	BUFFEROUT DS CL3608
				299	RECORD DS CL90
				300	SPACES DC CL1
				301	PRINTAREA DS OCLL32
				302	DS CL34
				303	LINE DS CL24
				304	DS CL54
				305	ENDLINE DC C**END OF SEQUENTIAL FILE**
				306	BEGIN
				307	=C**BCPEN *
				308	=C**BGLSEE *
				309	=A(INFILE1)
				000000	
001F90	5858C3067C 5D54C				
001F98	5858C30366E2C5				
001FA0	00000168				
001FA4					
001F77	58C305C4 40D6C640				
001FA4	00001598			310	=A(RECORD)
001FA8	00000168			311	=A(INFILE1)
001FAC	00000088			312	=A(PRINT)
001FB0	00001EF3			313	=A(PRINTAREA)

LDC	OBJECT CODE	ADDR1	ADDR2	SMT	SOURCE STATEMENT
	000000			1	PRINT GA,AGEN
	000000			2	START 0
	000000			3	BALR L2,0
	000002			4	USING *,12
	010000			5	USING BEGIN+4C9C,11
	C2C0C			6	USING BEGIN+8192,10
	00518			7	L 11,X1
	000002	5880	C516	8	L 10,X2
	000006	58A0	C51A	9	CPEN INFIL,CAFD,PFINT
				19	CNTRL PRINT,SR,1
	000030	0289	8420	25	MVC OUTAREA,OUTAREA-1
	000036	0208	8448	26	MVC CUTAREA,CUTAREA-1
	00003C	4570	C122	27	BAL 7,CREATE
	000040	4580	C0FC	28	BAL 8,WRITE
	000044	4580	C802	29	BAL 8,READ
	000048	4580	C0EA	30	BAL 8,READOK
	00004C	8506	8418	31	CLC IDC,100
	000052	478C	C070	32	BE CUTA
	000056	4720	C046	33	BH OUTC
	00005A	0206	8448	34	MVC OUTAREA+4C(7),IDC
	000060	020E	8452	35	MVC OUTAREA+50(15),=C*VC-RECURJ-FOUA,0
	000066	4580	C0FC	36	BAL 9,WRITE
	00006A	4580	C0D2	37	BAL 8,READ
	00006E	47F0	C04A	38	B COMP
	000072	0206	868E	39	ICDA,000
	000078	0223	86C0	40	MVC DATAT(36),STN
	00007E	0206	84F7	41	MVC CODE,TEM*7
	000084	4570	C180	42	BAL 7,SEARCH
	000088	0205	83A1	43	MVC CRNC,CRS
	00008E	0211	83AE	44	MVC CRN4,NAME
	000094	0201	83CF	45	MVC PRCR,CRC
	00009A	0200	83DA	46	MVC PRUHAUD
	0000A0	0200	83E2	47	MVC PRGR,GRC
	0000A6	0200	83ED	48	MVC PRSEM,SEM
	0000AC	0201	83F8	49	MVC PRYR,YR
				50	PUT PRINT,OUTARE1
	0000C2	4580	C0EA	56	BAL 8,READOK
	0000C6	0506	8418	57	CLC IDC,100
	0000CC	4780	C086	58	BE OUTB
	0000D0	4720	C0D2	59	BH REAC
				60	GET CARD,TEM
	0000E4	0206	8418	66	MVC IDC,TEM
	0000EA	07F8		67	BR 8
				68	INFIL,RECCHU
	0000FC	07F8		74	BR 3
				75	PRINT,CUTAREA
	00010E	0283	8420	81	MVC OUTAREA,OUTAREA-1
				82	CNTRL PRINT,SP,1
	000122	07F8		88	BR 3
	000124	4130	8614	89	3,TABDEF
	000128	4840	A338	90	LH 4,=H*0

LJC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	CARD,TEM
000130	0503 64A4 A310 014A4	02310		91	READDEP	GET
000142	4780 C15A 0015C			97	CLC	TEM(4),=C
000146	D21E 3002 84A4 00002	C14A4		98	BE	MAJ
00014C	4043 0000			99	MVC	2131,31,TEM
000150	4440 A33A 0233A			100	STH	4,0(3)
000154	5430 A314			101	AH	4,=H,150
000158	47F0 C124			102	A	3,=F,33
00015C	4130 8A07	0012C		103	B	READDEP
000170	D202 84F4 84A4 014F4	C14A4		104	LA	3,TABMAJ
000176	D210 3000 84A4 00000	014A4		105	GET	CARC,TEM
00017C	1943			111	MVC	TEST,TEM
00017E	5430 A318			112	MVC	0130,31,TEM
000192	D503 84A4 A310 014A4	02310		113	LR	4,3
000198	0787			114	A	3,=F,150
00019A	0502 84A4 84F4 014A4	014F4		115	GET	CARC,TEM
0001A0	4770 C16E 00170			121	CLC	TEM(4),=C
0001A4	5440 A31C			122	BER	7
0001A8	0210 4000 84A4 00000	014A4		123	CLC	TEM(3),TEST
0001AE	47F0 C180			124	BNE	MOVE
0001B2	4130 8818			125	A	4,=F,30
0001B6	0502 84F7 3002 014F7	00002		126	MVC	0130,4,TEM
0001BC	4780 C1C6			127	B	READMAJ
0001C0	5430 A314			128	SEARCH	3,TABDEF
0001C4	47F0 C184	00186		129	COMPARE1	CLC
0001C8	0218 843E 36C5 01638	00005		130	BE	SPAJCR
0001CC	4140 8A07			131	A	3,=F,33
0001D2	4443 0080			132	B	COMPARE1
0001D6	0503 84FA 4000 014FA	00000		133	MVC	DEPART,513
0001DC	4780 C1E6			134	LA	4,TABMAJ
0001E0	5440 A31C			135	AH	4,0(3)
0001E4	47F0 C1D4			136	COMPARE2	CLC
0001E8	D219 8662 4004 01662	001D6		137	BE	MAJCODE,0(4)
				138	A	4,=F,30
				139	B	COMPARE2
				140	CUIPLT	MVC
				141	ENTRL	MAJCR,4(4)
				147	FUT	PRINT,SK,1
				153	ENTRL	PRINT,SP,1
				159	PLI	PRINT,SP,1
				165	ENTRL	PRINT,SP,2
				171	PUT	PRINT,CUT3
				177	ENTRL	PRINT,SP,2
				183	PUT	PRINT,OUT4
				189	ENTRL	PRINT,SP,2
				195	PUT	PRINT,OUT6
				201	ENTRL	PRINT,SP,2
				207	PUT	PRINT,OUT5
				213	ENTRL	PRINT,SP,2
				219	PUT	PRINT,OUT6
				225	B	OUTE
				226	EDFRIN	ENTRL PRINT,SP,1

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0002D2	D20E 8448 A348	01448	0234B	232	MVC OUTAREA*4C(15),=C*END-OF-RETRIEVE
0002D8	458C C0FC		000FE	233	BAL 8,WRITE
				234	CLUSE INFIL, CARC, PRINT
				244	EOJ
000302	D20A 8448 A35A	01448	C235A	247	ECFDISK
000303	458D C0FC		000FE	248	CATRL PRINT, SP, I
				253	MVC OUTAREA*4C(11),=C*END-OF-DISK
				254	BAL 8,WRITE
				255	CLUSE CARC, PRINT, INFIL
				265	ECJ
				468	FILE PRINT
				422	FILE PRINT
				573	INFIL DTFS
					BLKSIZE=3600, ECF ADDR=ECFDISK, DEV ADDR=SYS014, DEVICE=3340, ICAREAL=BLFFIN, RECFORM=FIXBLK, RECSIZE=9C, TYPEFLE=INFUT, WORKA=YES
000518	00001000			619	X1 DC
00051C	00002C00			620	X2 DC
001330				621	BLFFIN DS
001330				622	RECORD DS
001336				623	CRS DS
001338				624	SEC DS
001344				625	NAME DS
00134C				626	CRS DS
00134D				627	AUD DS
00134E				628	GRD DS
001352				629	MAR DS
001359				630	100 DS
00137D				631	STN DS
00137E				632	SEM DS
001380				633	YR DS
00138A				634	DAFAL DS
00138A				635	CUTAREI DS
0013A1				636	CRNC DS
0013A7				637	CRNC DS
0013AE				638	CRNA DS
0013C0				639	CRNA DS
0013C0				640	PRSEC DS
0013C0				641	PRSEC DS
0013C0				642	FPCR DS
0013CF				643	FPCR DS
0013D1				644	PRAU DS
0013DA				645	PRAU DS
0013DB				646	PRGR DS
0013E2				647	PRGR DS
0013E3				648	FRSEM DS
0013ED				649	FRSEM DS

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0013EE				650	DS CL10
0013F8				651	DS CL2
0013FA				652	DS CL30
001418				653	DS CCL8C
00141B				654	DS CL7
00141F	40			655	DC CL1*
001420				656	DS CCL132
001420				657	DS CL4C
001443				658	DS CL92
001444				659	DS CL8C
0014F4				660	DS CL3
0014F7				661	DS OCL7
0014F7				662	DS CL3
0014FA				663	DS CL4
0014FE	4C			664	DC C*
0014FF				665	DS OCL132
0014FF	40404040404040			666	DC 58C*
001539	E2E3E4C4C3D5E340			667	DC C*STUDENT REPORT*
001543	40404040404040			668	DC 59C*
001583				669	DS OCL132
001583	40404040404040			670	DC 58C*
001580	5C5C5E5E5E5E5E			671	DC 15C*#1
0015CC	40404040404040			672	DC 59C*
001607				673	DC OCL132
001607	40404040404040			674	DC 33C*
001628	C4C5C7C1E9E304C5			675	DC C*DEPARTMENT OF
001638				676	CL28
001654	40404040404040			677	DC C* MAJCP
001662				678	DC CL26
00167C	40404040404040			679	DC 15C*
001688	40404040404040			680	DS OCL132
001688	C9C440D5E4D4C2C5			681	DC 33C*
001688				682	DC C*ID NUMBER
001688				683	DS CL7
00168F	4040404040D5C1D4			684	DC C* NAME
0016CD	40404040404040			685	DC CL49*
0016FE	40404040404040			686	DC 17C*
00170F				687	DS OCL132
00170F	40404040404040			688	DC 22C*
001725	C3D4E4D9E2C560D5			689	DC C* COURSE-NG
001736	C3D4E4D9E2C560D5			690	DC C* COURSE-NAME
001749	E2C5C3E3C9D6C540			691	DC C* SECTION
001752	C3D5C4C5E34C40			692	DC C* CREDIT
00175J	C1E4C4C9E34C4040			693	DC C* AUDIT GRADE*
00176A	4040404040E2C5C4			694	DC C* SEMESTER*
001777	4C4C4040CE8C5C1			695	DC C* YEAR*
00178J	4040404040404040			696	DC 19C*
001793				697	DC OCL132
001793	404C4C40404040			698	DC 22C*
0017A3	608C08C08C08C080			699	DC 51C*#1
001804	4040404040404040			700	DC 19C*

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
001818				701	DS 1 OF
001819				702	TABDEP DS 15CL33
001820				703	TABMAJ DS 75CL30
001821				704	BEGIN
001822				705	=C' \$8GFEN '
001823				706	=C' \$8BGL0SE '
001824				707	=AIPRINTJ
001825				708	=C'RETRIEVE-SAM'
001826				709	=AICUTAREL1
001827				710	=AICARDJ
001828				711	=AITEM
001829				712	=AIAFILEJ
001830				713	=AIRECORDJ
001831				714	=AICUTATEA1
001832				715	=C' .
001833				716	=F'33'
001834				717	=F'150'
001835				718	=F'30'
001836				719	=AICUT11
001837				720	=AICUT2J
001838				721	=AICUT3J
001839				722	=AICUT4J
001840				723	=AICUT6J
001841				724	=AICUT5J
001842				725	=H' C'
001843				726	=M'450'
001844				727	=CFAC-RECORD-FCUND'
001845				728	=C'END-OF-RETRIEVE'
001846				729	=C'END-OF-CISA'

001847
001848
001849
001850
001851
001852
001853
001854
001855
001856
001857
001858
001859
001860
001861
001862
001863
001864
001865
001866
001867
001868
001869
001870
001871
001872
001873
001874
001875
001876
001877
001878
001879
001880
001881
001882
001883
001884
001885
001886
001887
001888
001889
001890
001891
001892
001893
001894
001895
001896
001897
001898
001899
001900

ภาคผนวก จ

CLEAR DISK UTILITY
UTILITY CONTROL CARDS
// UCL B=(K=0,D=47),X100,PN,E=(3340)

// END
SPECIFIED PARAMETERS

KEY LENGTH - C
DATA LENGTH - 47

FIELD CHARACTER - X100Y
OUTPUT PARAMETER - N

RECORDS/TRACE - 39

EXTENT	BP	LOWER	LIMIT	UPPER	LIMIT
SEQ. NO.		CC	HH	CC	HH
000	000	407	000	456	011

END OF JOB

```

0001 DIMENSION NAME(5)
0002 DATA IA,IB,IC,JD,KE,LF
0003 DEFINE FIL 14(28),L,M)
0004 INPUT=C
0005 IDPIC=C
0006 N=78
0007 5 READ(13,20,FMT=60)IC,JD,MJ,NAME
0008 20 FORMATT(13,34,0)
0009 IDPIC=INPUTC+1
0010 CALL CPANUTC(10,1)
0011 MN=10-(130/MJ)
0012 N=2000+1
0013 READ(14,M)
0014 IF(1)25,30,25
0015 30 WRITE(14,50)NAME
0016 IDPIC=INPUTC+1
0017 40 FORMAT(17,56,0)
0018 GO TO 5
0019 25 MN=50+1
0020 M=3000+1
0021 WRITE(14,50)NAME
0022 GO TO 5
0023 50 WRITE(3,70) INPUT
0024 70 FORMATT(40,TOTAL INPUT RECORD = 14)
0025 WRITE(3,80)IDPIC
0026 80 FORMATT(40,TOTAL OUTPUT RECORD = 14)
0027 STOP
0028 END

```

* * * * *
 * THIS PROGRAM USE TO CREATE COURSE-FILE ON DISK 3340 *
 * * * * *
 C
 * * * * *
 * DIMENSION ICNAME(4) *
 * INPUTC=0 *
 * IOPIC=0 *
 C
 * * * * *

```

0001 DIMENSION ICNAME(4)
0002 INPUTC=0
0003 IOPIC=0
0004 5 READ(10,20,END=35)IDC,INDC1,ICNAME,ICNI
0005 20 FORMAT(A4,A2,A4,A2)
0006 INPUTC=INPUTC+1
0007 WRITE(13,20)IDC,INDC1,ICNAME,ICNI
0008 IOPIC = IOPIC+1
0009 GO TO 5
0010 35 ENDFILE 13
0011 REWIND 10
0012 WRITE(3,40)
0013 40 FORMAT(1H1,T40,'CREATE COURSE-FILE')
0014 50 FORMAT(T40, '-----')
0015 WRITE(3,50)
0016 WRITE(3,60)INPUTC
0017 60 FORMAT(T35,'TOTAL INPUT RECORD = ',I6)
0018 WRITE(3,70)IOPIC
0019 70 FORMAT(T35,'TOTAL OUTPUT RECORD = ',I6)
0020 WRITE(3,80)
0021 80 FORMAT(T35,'PROGRAM BY MISS RATANA RENJAPOLKUL')
0022 STOP
0023 END
    
```

DDJ FORTAN IV 360M-EC-479 3-8 MAINPGM DATE 04/01/81 TIME 10.09.00

C * * * * *
C * THIS PROGRAM USE TO CREATE GRADE-FILE ON DISK 3340 * * * * *
C * * * * *

```

0001 DIMENSION IC(17)
0002 DATA IA,IB,IC /1A,1B,1C,17*14+1/
0003 INPUTC=0
0004 INPUTC=0
0005 5 READ(9,20,FND=35)IDG1, IDG2, IVSM, ICS, ICST, ICN, IAD, ISEC, IGD
0006 20 FORMAT(A1,16,15,A4,A2,12,A1,12,A1)
0007 INPUTC=INPUTC+1
0008 IF(IDG1.EQ.1A)GO TO 30
0009 IDAB =IDG2*200000
0010 WRITE(12,26)IDAB, IVSM, ICS, ICST, ICN, IAD, ISEC, IGD
0011 26 FORMAT(I7,15,A4,A2,12,A1,12,A1)
0012 INPUTC =INPUTC+1
0013 GO TO 5
0014 30 IDAB =IDG2*1000000
0015 GO TO 25
0016 35 ENDFILE 12
0017 REWIND 12
0018 WRITE(3,40)
0019 40 FORMAT(I11,140,'CREATE GRADE-FILE')
0020 WRITE(3,50)IC
0021 50 FORMAT(I40,17A1)
0022 WRITE(3,60)INPUTC
0023 60 FORMAT(I35,'TOTAL INPUT RECORD = * 16)
0024 WRITE(3,70)INPUTC
0025 70 FORMAT(I35,'TOTAL OUTPUT RECORD = * 16)
0026 80 FORMAT(I35,'PROGRAM BY MISS RATANA BENJAPOLKUL')
0027 STOP
0028 END
0029

```




```

0051 WRITE(3,9C)IBB,NAME
0052 FORMAT(1H2,T34,'ID-NUMBER',3X,17,5X,'NAME',5X,544)
0053 WRITE(3,55)IPNT2
0054 WRITE(3,100)
0055 WRITE(3,100)
0056 --CR ED 11,54,ADU11,3X,40,ADU1,5X,SEMIESTER,5X,YEAR)
0057 WRITE(3,75)IPNT2
0058 READ(12,110)IDAB,IS,IY,IQ,ICS,ICSL,ICG,IAG,ISEC,ICD
0059 FORMAT(17,1,12,12,44,A2,12,A1,12,A1)
0060 IF(ICS-ICD)GO TO 105
0061 GO TO 105
0062 WRITE(3,150)IG
0063 FORMAT(120,'ERRR=TRANS',17)
0064 IERR=IERR+1
0065 GO TO 5
0066
0067 FORMAT(44,A2,44,A2)
0068 READ(13,160)ICG5,ICCS1,ICNAME,ICM1
0069 IF(ICS.LI-IDCS)GO TO 140
0070 IF(ICS.EQ-IDCS)GO TO 170
0071 GO TO 130
0072 WRITE(3,150)ICG5,ICCS1,ICNAME,ICM1,ISEC,IUL,IAC,I5,I2,IY
0073 READ(12,110)ICAE,I5,IY,IQ,ICS,ICG,IAD,ISEG,ICD
0074 IF(103-EQ-ICAR)GO TO 170
0075 GO TO 5
0076
0077 WRITE(3,200)INPI
0078 FORMAT(1H1,740,'TOTAL TRANSACTION =',17)
0079 WRITE(3,210)ICR
0080
0081 FORMAT(140,'TOTAL ERROR CARD =',17)
0082 WRITE(3,20)ICPT
0083 WRITE(3,230)
0084
0085 FORMAT(140,'PROGRAM BY MISS RATANA BENJAPUKUL')
0086 STOP
0087 END

```

```

CBL STATE
00001 ID DIVISION.
00002 PROGRAM-ID. CREATE-DAN-NAME.
00003 AUTHOR. MISS PATANA BENJAPOLKUL.
00004 DATE-COMPILED. 01/07/80
00009 ENVIRONMENT DIVISION.
00010 CONFIGURATION SECTION.
00011 OBJECT-COMPUTER. IBM-370.
00012 INPUT-OUTPUT SECTION.
00013 FILE-CONTROL.
00014 SELECT TAPE-MS-I-FILE ASSIGN TO SYS011-UT-2400-S.
00015 SELECT D-MSNAME-O-FILE ASSIGN TO SYS013-DN-33&0-D-MSNW
00016 ACCESS IS RANDOM
00017 ACTUAL KEY IS ACT-KEY.
00018 DIVISION.
00019 DATA
00020 FILE
00021 F9 TAPE-MS-I-FILE
00022 RECORD CONTAINS 47 CHARACTERS
00023 RECORDING MODE IS F
00024 LABEL RECORD IS OMITTED
00025 DATA
00026 TAPE-MS-I-REC.
00027 05 TAPE-STUDENT-ID. PIC X.
00028 07 ID-ALPHA PIC 9(6).
00029 07 ID-NUMERIC PIC X(40).
00030 F0 D-MSNAME-O-FILE
00031 RECORD CONTAINS 47 CHARACTERS
00032 RECORDING MODE IS F
00033 LABEL RECORD IS STANDARD
00034 DATA
00035 01 D-MSNAME-O-REC.
00036 05 DISK-STUDENT-ID PIC X(7).
00037 05 DISK-STUDENT PIC X(40).
00038 WORKING-STORAGE SECTION.
00039 77 INPUT-REC-CNT PIC 9(5) COMP VALUE ZERO.
00040 77 OUTPUT-REC-CNT PIC 9(5) COMP VALUE ZERO.
00041 77 INPUT-ERROR-CNT PIC 9(7) COMP SYNC.
00042 77 SAVE PIC 9(7) COMP SYNC.
00043 77 QUOTIENT PIC 9(7) VALUE 1000000.
00044 77 CONVA PIC 9(7) VALUE 2000000.
00045 77 CONVR PIC 9(7) VALUE 2000000.
00046 01 ERROR-COND.
00047 02 FILLER
00048 02 ERR
00049 02 FILLER
00050 01 ACT-KEY.
00051 03 TRACK-ID PIC 99 VALUE ZERO.
00052 03 REC-ID PIC 9 VALUE ZERO.
00053 01 SWITCH-AREA.
00054 05 WS-TAPE-SWITCH-A PIC 9(7) COMP SYNC.
00055 88 WS-TAPE-SWITCH VALUE X.
00056 05 WS-READ-A PIC 9(7) COMP SYNC.

```

2 CREATE0 09.56.56 01/07/80

```

00057 88 MS-READ DIVISION. VALUE 'Y'.
00058 → PROCEDURES.
00059 → DECLARATIVES.
00060 ERROR-PROCEDURE SECTION. USE AFTER STANDARD ERROR PROCEDURE ON
00061 D-MSNAME-O-FILE GIVING ERROR-COND.
00062 ERROR-ROUTINE.
00063 EXHIBIT NAMED ERROR-COND.
00064 IF ERR = 1
00065 IF TRACK-ID < 71
00066 ADD 1 TO TRACK-ID
00067 MOVE 'Y' TO MS-READ-A
00068 GO TO 030-WRITE-DISK
00069 ELSE
00070 DISPLAY 'NO ROOM FOUND' REC-ID
00071 GO TO END-ROUTINE.
00072 →END DECLARATIVES.
00073 OPEN INPUT TAPE-MS-I-FILE
00074 OUTPUT D-MSNAME-O-FILE.
00075 PERFORM 010-READ-TAPE THRU 010-READ-TAPE-EXT
00076 UNTIL MS-TAPE-SWITCH.
00077 END-ROUTINE.
00078 DISPLAY ' TOTAL INPUT RECORD = ' INPUT-REC-CNT.
00079 DISPLAY ' TOTAL INPUT ERROR = ' INPUT-ERROR-CNT.
00080 DISPLAY ' TOTAL OUTPUT RECORD = ' OUTPUT-REC-CNT.
00081 DISPLAY ' PROGRAM BY MISS RATANA RENJAPOLKUL'.
00082 CLOSE TAPE-MS-I-FILE
00083 D-MSNAME-O-FILE.
00084 RUN.
00085 → 010-READ-TAPE.
00086 READ TAPE-MS-I-FILE
00087 AT END.
00088 MOVE 'Y' TO MS-TAPE-SWITCH-A
00089 GO TO 010-READ-TAPE-EXT.
00090 ADD 1 TO INPUT-REC-CNT.
00091 IF (ID-ALPHA IS ALPHABETIC AND
00092 ID-NUMERIC IS NUMERIC)
00093 PERFORM 020-CONVERT THRU 020-CONVERT-EXT
00094 PERFORM 030-WRITE-DISK THRU 030-WRITE-DISK-EXT
00095 ELSE
00096 DISPLAY 'ERROR-INPUT' TAPE-MS-I-REC
00097 ADD 1 TO INPUT-ERROR-CNT.
00098 EXIT.
00099 → 010-READ-TAPE-EXT.
00100 → 020-CONVERT.
00101 MOVE TAPE-STUDENT-ID TO DISK-STUDENT-ID RFC-ID.
00102 MOVE TAPE-STUDENT TO DISK-STUDENT.
00103 IF ID-ALPHA = 'A'
00104 ADD CONVA ID-NUMERIC GIVING SAVE
00105 ELSE
00106 ADD CONVA ID-NUMERIC GIVING SAVE.
00107 DIVIDE SAVE BY 71 GIVING QUOTIENT
00108 REMAINDER TRACK-ID.
00109 DISPLAY QUOTIENT.
00110 DISPLAY TRACK-ID.

```


3 CREATEUD 09.56.56 01/07/80

00110 020-CONVERT-EXT. EXIT.
00111 030-WRITE-DISK.
00112 WRITE D-MSNAME-O-REC
00113 INVALID KEY
00114 DISPLAY 'NO ROOM FOUND' REC-ID
00115 GG TO END-ROUTINF.
00116 ADD 1 TO OUTPUT-REC-CNT.
00117 EXHIBIT NAMED TRACK-ID TAPE-MS-1-REC.
00118 IF MS-READ
00119 GO TO 010-READ-TAPE.
00120 030-WRITE-DISK-EXT. EXIT.

CBL STATE
CBL LIR

00001 ID DIVISION.
 00002 PROGRAM-ID. RETRIEVE-CAM.
 00003 AUTHOR. MISS RATANA BENJAPOLKUL.
 00004 DATE-COMPILED. 01/07/80.
 00005 REMARK. * * * * *
 00006 * THIS PROGRAM USED TO RETRIEVE *
 00007 * DIRECT FILE ON DISK 3340 *
 00008 * * * * *

00009 ENVIRONMENT DIVISION.
 00010 CONFIGURATION SECTION.
 00011 OBJECT-COMPUTER. IBM-370.
 00012 INPUT-OUTPUT SECTION.
 00013 FILE-CONTROL.

00014 SELECT CRD-FILE ASSIGN TO SYS004-UR-2540R-5.
 00015 SELECT PRT-FILE ASSIGN TO SYS006-UR-1403-5.
 00016 SELECT D-COURSE-FILE ASSIGN TO SYS014-DA-3340-1-MAST
 00017 ACCESS IS RANDOM
 00018 NOMINAL KEY IS COURSE-KEY
 00019 RECORD KEY IS DISK-CS-ID.
 00020 SELECT D-GRADE-FILE ASSIGN TO SYS007-DA-3340-1-MA SC
 00021 ACCESS IS SEQUENTIAL
 00022 NOMINAL KEY IS GRADE-KEY
 00023 RECORD KEY IS DISK-GRADE.
 00024 SELECT D-MSNAME-C-FILE ASSIGN TO SYS013-DA-3340-D-MA SN
 00025 ACCESS IS RANDOM
 00026 ACTUAL KEY IS ACT-KEY.
 00027 DIVISION.
 00028 SECTION.

DATA
FILE

FD CRD-FILE CONTAINS 80 CHARACTERS
 RECORD MODE IS F
 RECORDING LABEL IS OMITTED
 DATA RECORD IS CPD-REC.
 01 CRD-REC PIC X(80).
 FD PRT-FILE CONTAINS 132 CHARACTERS
 RECORD MODE IS F
 RECORDING LABEL IS OMITTED
 DATA RECORD IS PRT-REC.
 01 PRT-REC PIC X(132).
 FD D-MSNAME-O-FILE CONTAINS 40 RECORDS
 BLOCK CONTAINS 48 CHARACTERS
 RECORD MODE IS F
 RECORDING LABEL IS STANDARD
 DATA RECORD IS D-MSNAME-O-REC.
 01 D-MSNAME-O-REC.

00041 FD D-MSNAME-O-FILE CONTAINS 40 RECORDS
 00042 BLOCK CONTAINS 48 CHARACTERS
 00043 RECORD MODE IS F
 00044 RECORDING LABEL IS STANDARD
 00045 DATA RECORD IS D-MSNAME-O-REC.
 00046 01 D-MSNAME-O-REC.
 00047 03 DISK-STUDENT-ID PIC X(7).
 00048 03 STUDENT.
 00049 05 MAJCR.
 00050 07 DEPT PIC XXX.
 00051

2 RETRIEVE 10.05.03 01/07/80

00052	07 FILLER	PIC X.	
00053	03 DISK-NAME	PIC X(36).	
00054	FD D-GRADE-FILE		
00055	BLOCK CONTAINS	40 RECORDS	
00056	RECORD CONTAINS	25 CHARACTERS	
00057	RECORDING MODE	F	
00058	LABEL RECORD	IS STANDARD	
00059	DATA RECORD	IS D-GRADE-REC.	
00060	01 D-GRADE-REC.		
00061	03 DELETE-CODE-GRADE	PIC X.	
00062	03 DISK-GRADE.		
00063	05 DISK-ID	PIC X(7).	
00064	05 DISK-YEAR	PIC XX.	
00065	05 DISK-SEMESTER	PIC X.	
00066	05 DISK-SEQUENCE	PIC XX.	
00067	03 DISK-COURSE-ID	PIC X(16).	
00068	03 DISK-CREDIT	PIC XX.	
00069	03 DISK-AUDIT	PIC X.	
00070	03 DISK-SECTION	PIC XX.	
00071	03 DISK-GRADE-A	PIC X.	
00072	FD D-COURSE-FILE		
00073	BLOCK CONTAINS	40 RECORDS	
00074	RECORD CONTAINS	25 CHARACTERS	
00075	RECORDING MODE	F	
00076	LABEL RECORD	IS STANDARD	
00077	DATA RECORD	IS D-COURSE-REC.	
00078	01 D-COURSE-REC.		
00079	03 DELETE-CODE-CS	PIC X.	
00080	03 DISK-CS-ID	PIC X(16).	
00081	03 DISK-CS-NAME	PIC X(18).	
00082	WARNING-STORAGE SECTION.		
00083	77 TRANS-CNT	PIC 9(5) COMP	VALUE ZERO.
00084	77 ERROR-CARD-CNT	PIC 9(5) COMP	VALUE ZERO.
00085	77 PRT-REC-CNT	PIC 9(5) COMP	VALUE ZERO.
00086	77 ERR-SEQ-CNT	PIC 9(5) COMP	VALUE ZERO.
00087	77 INVALID-REC-CNT	PIC 9(5) COMP	VALUE ZERO.
00088	77 NAME-KEY	PIC X(17).	
00089	77 COURSE-KEY	PIC X(16).	
00090	77 GRADE-KEY	PIC X(12).	
00091	77 MAJORN	PIC X(13)	VALUE SPACE.
00092	77 DEPT-N	PIC X(132)	VALUE SPACE.
00093	77 SAVE	PIC 9(17) COMP	VALUE ZERO.
00094	77 QUOTIENT	PIC 9(17) COMP	VALUE ZERO.
00095	77 CONVA	PIC 9(17) VALUE	1000000.
00096	77 CONVB	PIC 9(17) VALUE	2000000.
00097	01 ACT-KEY.		
00098	03 TRACK-ID	PIC 9(17) COMP	SYNC.
00099	03 REC-ID	PIC 9(17).	
00100	01 DNAME COPY CYREDEPT	SUPPRESS.	
00256	01 DNAME COPY CYREDEFINES DNAME	COPY CYRECADP	SUPPRESS.
00263	01 MNAME COPY CYREMAJP	SUPPRESS.	
00375	01 MNAME REDEFINES MNAME	COPY CYRECAMJ	SUPPRESS.
00382	01 HEAD1.		

00383	03	FILLER	PIC	X(58)	VALUE	SPACE	'STUDENT
00384	03	FILLER	PIC	X(15)	VALUE	SPACE	'REPORT'
00385	03	FILLER	PIC	X(59)	VALUE	SPACE	
00386	01	HEAD2	PIC	X(33)	VALUE	SPACE	
00387	03	FILLER	PIC	X(13)	VALUE	SPACE	'DEPARTMENT (F'
00388	03	FILLER	PIC	XXX	VALUE	SPACE	
00389	03	FILLER	PIC	X(32)	VALUE	SPACE	
00390	03	HEAD2-1	PIC	X(5)	VALUE	SPACE	'MAJOR'
00391	03	FILLER	PIC	X(5)	VALUE	SPACE	
00392	03	FILLER	PIC	X(15)	VALUE	SPACE	
00393	03	HEAD2-2	PIC	X(15)	VALUE	SPACE	
00394	03	FILLER	PIC	X(26)	VALUE	SPACE	
00395	01	HEAD3	PIC	X(33)	VALUE	SPACE	
00396	03	FILLER	PIC	X(9)	VALUE	SPACE	'ID NUMBER'
00397	03	FILLER	PIC	XXX	VALUE	SPACE	
00398	03	FILLER	PIC	X(12)	VALUE	SPACE	
00399	03	HEAD3-1	PIC	X(4)	VALUE	SPACE	'NAME'
00400	03	FILLER	PIC	X(5)	VALUE	SPACE	
00401	03	FILLER	PIC	X(36)	VALUE	SPACE	
00402	03	HEAD3-2	PIC	X(30)	VALUE	SPACE	
00403	03	FILLER	PIC	X(22)	VALUE	SPACE	'COURSE-NO'
00404	03	FILLER	PIC	X(9)	VALUE	SPACE	
00405	03	FILLER	PIC	X(8)	VALUE	SPACE	
00406	03	FILLER	PIC	X(11)	VALUE	SPACE	'COURSE-NAME'
00407	03	FILLER	PIC	X(17)	VALUE	SPACE	'SECTION'
00408	03	FILLER	PIC	X(5)	VALUE	SPACE	
00409	03	FILLER	PIC	X(6)	VALUE	SPACE	'CREDIT'
00410	03	FILLER	PIC	X(5)	VALUE	SPACE	
00411	03	FILLER	PIC	X(5)	VALUE	SPACE	'AUDIT'
00412	03	FILLER	PIC	XX	VALUE	SPACE	
00413	03	FILLER	PIC	X(5)	VALUE	SPACE	'GRADE'
00414	03	FILLER	PIC	X(5)	VALUE	SPACE	
00415	03	FILLER	PIC	X(8)	VALUE	SPACE	'SEMESTER'
00416	03	FILLER	PIC	X(5)	VALUE	SPACE	
00417	03	FILLER	PIC	X(5)	VALUE	SPACE	
00418	03	FILLER	PIC	X(4)	VALUE	SPACE	'YEAR'
00419	03	FILLER	PIC	X(19)	VALUE	SPACE	
00420	03	FILLER	PIC	X(23)	VALUE	SPACE	
00421	03	FILLER	PIC	X(6)	VALUE	SPACE	
00422	01	HEAD5	PIC	X(17)	VALUE	SPACE	
00423	03	FILLER	PIC	X(18)	VALUE	SPACE	
00424	03	PRT-COURSE-NO	PIC	X(4)	VALUE	SPACE	
00425	03	FILLER	PIC	XX	VALUE	SPACE	
00426	03	PRT-COURSE-NAME	PIC	X(9)	VALUE	SPACE	
00427	03	FILLER	PIC	XX	VALUE	SPACE	
00428	03	PRT-SECTION	PIC	X(9)	VALUE	SPACE	
00429	03	FILLER	PIC	XX	VALUE	SPACE	
00430	03	PRT-CREDIT	PIC	X(9)	VALUE	SPACE	
00431	03	FILLER	PIC	X	VALUE	SPACE	
00432	03	PRT-AUDIT	PIC	X(7)	VALUE	SPACE	
00433	03	FILLER	PIC	X	VALUE	SPACE	
00434	03	PRT-GRADE	PIC	X(10)	VALUE	SPACE	
00435	03	FILLER	PIC	X	VALUE	SPACE	

4 RETRIEVE 10.05.03 01707/80

```

00436 03 PRT-SEMESTER PIC X.
00437 03 FILLER PIC X(10) VALUE SPACE.
00438 03 PRT-YEAR PIC XX.
00439 03 FILLER PIC X(30) VALUE SPACE.
00440 01 HEAD6.
00441 03 FILLER PIC X(58) VALUE SPACE.
00442 03 FILLER PIC X(16) VALUE *****.
00443 03 FILLER PIC X(58) VALUE SPACF.
00444 01 HEAD7.
00445 03 FILLER PIC X(22) VALUE SPACE.
00446 03 S PIC X(91).
00447 03 FILLER PIC X(58) VALUE SPACE.
00448 01 MS-CRD-REC.
00449 03 CRD-ID.
00450 05 ID-ALPHA PIC X.
00451 05 ID-NUMERIC PIC 9(6).
00452 03 CRD-YEAR PIC XX.
00453 03 CRD-SEMESTER PIC X.
00454 03 CRD-SEQUENCE PIC XX.
00455 01 SWITCH-AREA.
00456 05 MS-CRD-SWITCH-A PIC X.
00457 08 MS-CRD-SWITCH VALUE 'Y'.
00458 05 TRANS-VALID-SWITCH-A PIC X.
00459 08 TRANS-VALID-SWITCH VALUE 'Y'.
00460 05 EQ-SWITCH-NAME-A PIC X.
00461 08 EQ-SWITCH-NAME VALUE 'Y'.
00462 PROCEDURE DIVISION.
00463 * MAIN LEVEL STARTING HERE
00464 * ---- INITIALIZATION ----
00465 * OPEN FILES
00466 OPEN INPUT CRD-FILE
00467 D-MSNAME-C-FILE
00468 D-GRADE-FILE
00469 D-COURSE-FILE
00470 OUTPUT PRT-FILE.
00471 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00472 UNTIL TRANS-VALID-SWITCH.
00473 PERFORM 020-PEAD-NAME.
00474 PERFORM 030-COMPARE-NAME-CRD THRU 030-COMPARE-NAME-CRD-EXT
00475 UNTIL MS-CRD-SWITCH.
00476 * ----- CONCLUSION -----
00477 DISPLAY 'TOTAL READ TRANS = ' TRANS-CNT.
00478 DISPLAY 'TOTAL ERROR CARD = ' ERROR-CARD-CNT.
00479 DISPLAY 'TOTAL WRITE REPORT = ' PRT-REC-CNT.
00480 DISPLAY 'TOTAL ERROR SEQ CARD = ' ERR-SEC-CNT.
00481 DISPLAY 'TOTAL INVALID CARD = ' INVALID-REC-CNT.
00482 CLOSE CRD-FILE
00483 D-MSNAME-O-FILE
00484 D-GRADE-FILE
00485 D-COURSE-FILE
00486 PRT-FILE.
00487 STOP
00488 010-READ-TRANS.

```



```

00489 READ CRD-FILE INTO WS-CRD-REC
00490 AT END
00491 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00492 MOVE 'Y' TO WS-CRD-SWITCH-A
00493 GO TO 010-READ-TRANS-EXT.
00494 ADD 1 TO TRANS-CNT.
00495 IF (ID-ALPHA IS ALPHABETIC AND
00496 ID-NUMERIC IS NUMERIC)
00497 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00498 ELSE
00499 MOVE 'N' TO TRANS-VALID-SWITCH-A
00500 DISPLAY 'ERROR-CARD' WS-CRD-REC
00501 ADD 1 TO ERROR-CARD-CNT.
00502 010-READ-TRANS-EXT.
00503 020-CONVERT.
00504 MOVE CRD-ID TO REC-ID.
00505 IF ID-ALPHA = 'A'
00506 ADD CONVA ID-NUMERIC GIVING SAVE
00507 ELSE
00508 ADD CONVB ID-NUMERIC GIVING SAVE.
00509 MOVE CRD-ID TO SAVE.
00510 DIVIDE SAVE BY 71 GIVING QUOTIENT
00511 REMAINDER TRACK-ID.
00512 020-CONVERT-EXT.
00513 020-READ-NAME.
00514 READ D-MSNAME-O-FILE
00515 INVALID KEY
00516 MOVE 'Y' TO EQ-SWITCH-NAME-A.
00517 IF EQ-SWITCH-NAME
00518 MOVE 'N' TO EQ-SWITCH-NAME-A
00519 IF TRACK-ID < 71
00520 ADD 1 TO TRACK-ID
00521 PERFORM 020-READ-NAME
00522 GO TO 030-COMPARE-NAME-CRD
00523 ELSE
00524 DISPLAY 'NOT FOUND' CRD-ID
00525 MOVE 'N' TO TRANS-VALID-SWITCH-A
00526 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00527 TRANS-VALID-SWITCH
00528 PERFORM 020-CONVERT THRU 020-CONVERT-EXT
00529 PERFORM 020-READ-NAME
00530 UNTIL
00531 ELSE
00532 PERFORM 060-WRITE-HEAD
00533 PERFORM 040-START-GRADE THRU 040-START-GRADE-EXT
00534 PERFORM 050-READ-GRADE THRU 050-READ-GRADE-EXT
00535 UNTIL CRD-ID NOT = DISK-ID
00536 MOVE 'N' TO TRANS-VALID-SWITCH-A
00537 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00538 UNTIL TRANS-VALID-SWITCH
00539 PERFORM 020-CONVERT THRU 020-CONVERT-EXT
00540 PERFORM 020-READ-NAME.
00541 030-COMPARE-NAME-CRD-EXT. EXIT.

```

```

0542 040-START-GRADE.
0543 MOVE WS-CRD-REC TO GRADE-KEY.
0544 START D-GRADE-FILE KEY IS EQUAL TO CRD-ID
0545 INVALID KEY
0546 DISPLAY 'ERROR-ID' WS-CRD-REC
0547 ADD 1 TC INVALID-REC-CNT
0548 GC TO 040-START-GRADE-EXT.
0549 READ D-GRADE-FILE AT END DISPLAY 'END OF GRADE-FILE'.
0550 EXIT.
0551 040-START-GRADE-EXT.
0552 MOVE DSK-COURSE-ID TO COURSE-KEY.
0553 READ D-COURSE-FILE
0554 INVALID KEY
0555 DISPLAY 'ERROR-COURSE-FILE' WS-CRD-REC
0556 ADD 1 TC INVALID-REC-CNT
0557 GO TO 050-READ-GRADE-EXT.
0558
0559 MOVE DISK-CS-ID TO PRT-COURSE-NO.
0560 MOVE DISK-CS-NAME TO PRT-COURSE-NAME.
0561 MOVE DISK-SECTION TO PRT-SECTION.
0562 MOVE DISK-CREDIT TO PRT-CREDIT.
0563 MOVE DISK-AUDIT TO PRT-AUDIT.
0564 MOVE DISK-GRADE TO PRT-GRADE.
0565 MOVE DISK-SEMESTER TO PRT-SEMESTER.
0566 MOVE DISK-YEAR TO PRT-YEAR.
0567 WRITE PRT-REC FROM HEAD5 AFTER POSITIONING 2 LINES.
0568 READ D-GRADE-FILE AT END DISPLAY 'END OF GRADE-FILE'.
0569 EXIT.
0570 050-READ-GRADE-EXT.
0571 ADD 1 TO PRT-REC-CNT.
0572 WRITE PRT-REC FROM HEAD1 AFTER POSITIONING 0 LINES.
0573 WRITE PRT-REC FROM HEAD6 AFTER POSITIONING 1 LINES.
0574 PERFORM FIND-DEP THRU FIND-D.
0575 PERFORM FIND-MAJ THRU FIND-E.
0576 MOVE DEPT-N TO HEAD2-1.
0577 MOVE MAJOPN TO HEAD2-2.
0578 WRITE PRT-REC FROM HEAD2 AFTER POSITIONING 2 LINES.
0579 MOVE DISK-STUDENT-ID TO HEAD3-1.
0580 MOVE DISK-NAME TO HEAD3-2.
0581 WRITE PRT-REC FROM HEAD3 AFTER POSITIONING 2 LINES.
0582 MOVE ALL TO S.
0583 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
0584 WRITE PRT-REC FROM HEAD4 AFTER POSITIONING 2 LINES.
0585 MOVE ALL TO S.
0586 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
0587 DEP-FF SECTION. COPY CYREDEP.
0588 DEP-FIND SECTION.
0589 FIND-DEP.
0590 SEARCH ALL DTNAME
0591 AT END GO TO NOT-F-DEP
0592 WHEN DEPT = DNO (D)
0593 GO TO FCUND-D.
0594 FOUND-D.
0595 MOVE D-NAME (D) TO DEPT-N.

```

7 RETRIEVE 10.05.03 01/07/80

00595 C GO TO FIND-D.
00596 C NOT-FOUND.
00597 C MOVE SPACE TO DEPT-N.
00598 C FIND-D.
00599 C EXIT.
00600 MAJ-FF SECTION. COPY CYREFMAJ.
00601 C MAJ-FIND SECTION.
00602 C FIND-MAJ.
00603 C SEARCH ALL MJNAME
00604 C AT END GO TO NOT-FOUND-M
00605 C WHEN MAJOR = MND (A)
00606 C GO TO FOUND-M.
00607 C FOUND-M.
00608 C MOVE H-NAME (A) TO MAJRN.
00609 C GO TO FIND-E.
00610 C NOT-FOUND-M.
00611 C MOVE SPACE TO MAJRN.
00612 C FIND-E.
00613 C EXIT.

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000000				1	PRINT ON,NOGEN
000000	05C0			2	START 0
				3	BALR 12,0
		00002		4	USING *,12
				5	OPEN DISK,PRINT,CARD
00001A	D283 C445 C444	00447	00446	15	MVC OUTAREA,OUTAREA-1
000020	D20F C46D C406	0046F	004D8	16	MVC OUTAREA+40(16),=C*SUCCESS-OPEN-JAM*
				17	PUT PRINT,OUTAREA
				23	GET CARD,INPUT
				29	MVC B,KEY
				30	MVC KEYARG,KEY
				31	MVC SEEKADR,KEY
				32	MVC C(46),INPUT
				33	MVC OUTAREA,OUTAREA-1
				34	MVC OUTAREA+40(16),=C*READ
				35	PUT PRINT,OUTAREA
				41	WRITE DISK,AFTER
				46	WAITF DISK
				51	MVC OUTAREA,OUTAREA-1
000092	D283 C445 C444	00447	00446	51	MVC OUTAREA+40(16),=C*WRITE
000098	D20F C46D C4F6	0046F	004F8	52	MVC OUTAREA
				53	PUT PRINT,OUTAREA
				59	R READ
				60	MVC OUTAREA,OUTAREA-1
				61	MVC OUTAREA+40(16),=C*PROGRAM BY MISS RATANA BENJAPOLKUL*
0000B8	D221 C46D C522	0046F	00524	62	PUT PRINT,OUTAREA
				68	CLOSE DISK,CARD,PRINT
				78	E0J
				81	FILE CARD
235					PRINT
386					DTFDA
					DISK
					BLKSIZE=64,
					DEVICE=3340,
					ERRBYTE=ERRBYTE,
					IOAREAL=IOAREAL,
					TYPEFL=OUTPUT,
					AFTER=YES,
					DEVADDR=SYS014,
					DSKXINT=1,
					KEYARG=KEYARG,
					KEYLEN=10,
					RECFORM=FIXUMB,
					RELTYPE=DEC,
					SEEKADR=SEEKADR,
					WRITEKY=YES
					CLZ
483	ERRBYTE	DS			CL2
484	IOAREAL	DS			OCL64
485	A	DS			CL8
486	B	DS			CL10
487	C	DS			CL36
488	KEYARG	DS			CL10
489	SEEKADR	DS			CL10
490	INPUT	DS			OCL80

LOC	OBJECT CODE	ADDR 1	ADDR 2	STMT	SOURCE	STATEMENT
0003F6				491	KEY	DS CL10
000400				492	DATA	DS CL30
00041E				493		DS CL40
000446	40			494		DC CL10
000447				495	CUTAREA	DS OCL132
000447				496		DS CL40
00046F				497	HEADER	DS CL92
		00000		498		BEGIN
000400	5858C2D607C5D540			499		=C'\$\$ROPEN'
0004D8	E2E4C3C5E2E260			500		=C'SUCCESS-OPEN-DAM'
0004E8	D9C5C1C440404040			501		=C'READ
0004F8	E6D9C9E3C5404040			502		=C'WRITE
000508	5858C2C303D6E2C5			503		=C'\$\$BCLOSE'
000510	00000170			504		=A(PRINT)
000514	00000447			505		=A(CUTAREA)
000518	000000E8			506		=A(CARD)
00051C	000003F6			507		=A(INPUT)
000520	00000250			508		=A(DISK)
000524	7D9D6C7D9C1D440			509		=C'PROGRAM BY MISS PATANA BENJAPOLKUL'

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000000				1	PRINT ON,NOGEN
000000	0590			2	START 0
000002	98AB 900A	00002		3	BALR 9,0
000006	47F0 9012	0000C		4	USING HERE,9,10,11
00000A	0000	00014		5	LM 10,11,BASE
00000C	0000100200002002			6	OPEN
000040	0283 A0FE A0FE	01101	0110C	7	DC A(HERE+4096,HREF+8192)
000046	D20B A127 B7FA	01129	027FC	8	OPEN CARD,DISK,PRINT,ISGRD,ISCRS
00005C	4570 9166	00168		20	CNTRL PRINT,SK,1
000070	D209 9890 A981	00892	01983	26	MVC OUTAREA,OUTAREA-1
000076	D209 9886 A981	00888	01983	27	MVC OUTAREA+40(12),=C'RETRIEVL-DATA'
00007C	D233 A0FF A0FE	01101	01100	28	PUT PRINT,OUTAREA
000082	D208 A127 B84E	01129	02850	34	BAL 7,CREATE
000080	D209 9890 A981	00892	01983	35	GET CARD,TEM
000076	D209 9886 A981	00888	01983	41	MVC CRSDA,TEM
00007C	D233 A0FF A0FE	01101	01100	42	MVC KEYARG,TEM
000082	D208 A127 B84E	01129	02850	43	MVC OUTAREA,OUTAREA-1
000080	D209 9890 A981	00892	01983	44	MVC OUTAREA+40(9),=C'READ CARD'
000076	D209 9886 A981	00888	01983	45	PUT PRINT,OUTAREA
00007C	D233 A0FF A0FE	01101	01100	51	READ DISK,KEY
000082	D208 A127 B84E	01129	02850	56	WAITF DISK
000080	D283 A0FF A0FE	01101	01100	61	MVC OUTAREA,OUTAREA-1
000086	D223 A8DA 9862	0180C	00364	62	MVC DATA(36),NAME
00008C	D206 ABC5 A981	018C7	01983	63	MVC ID(1),TEM
0000C2	D206 AA04 A9C7	01A06	019C9	64	MVC CODE,TEM+22
0000C8	4570 91F4	001F6		65	BAL 7,SEARCH
0000CC	D20B 9CAL A98B	00CA3	0198D	66	MVC GROKEY,TEM+10
0000EA	D205 989A A0A2	0089C	010A4	67	READ ISGRD,KEY
000108	D283 A90A A0FE	0190C	01100	72	WAITF ISGRD
00010E	D205 A921 989A	01923	0089C	77	MVC CRKEY,CID
000114	D211 A92E 9C9F	01930	00C91	78	READ ISCRS,KEY
00011A	D201 A94F A0A8	01951	010AA	83	WAITF ISCRS
000120	D200 A95A A0AA	0195C	010AC	88	MVC OUTARE1,SPACES
000126	D200 A962 A0AD	01964	010AF	89	MVC CRD,CRSKEY
00012C	D200 A960 A09F	0196F	010A1	90	MVC CRNA,DATAL
000132	D201 A978 A09D	0197A	0109F	91	MVC PEC,CCR
000148	D206 A0AE A981	01080	01983	92	MVC PRAU,CAD
00015A	D506 A0AE A096	01080	01098	93	MVC PRG,CGD
000160	4780 90E8	000EA		94	MVC PRSEM,CSM
000164	4720 905E	00060		95	MVC PRYR,CYR
000168	4130 AD26		01D28	96	PUT PRINT,OUTARCI
00016C	4940 884A		0284C	102	MVC IDC,TEM
000180	D503 A981 8822	01983	02824	103	GET ISGRD
000180	D503 A981 8822	01983	02824	108	CLC IDC,GRID
000180	D503 A981 8822	01983	02824	109	BE DUTC
000180	D503 A981 8822	01983	02824	110	8H LOOP
000180	D503 A981 8822	01983	02824	111	LA 3,TABDEP
000180	D503 A981 8822	01983	02824	112	LH 4,=H'0'
000180	D503 A981 8822	01983	02824	113	READDEP GET CARD,TEM
000180	D503 A981 8822	01983	02824	119	TEM(4),=C'

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000186	4780 919E	001A0		120	MVC MAJ
000187	0018A D21E 3002 A9B1 00002	01983		121	MVC 2(31,31),TEM
000188	00190 4043 0000	00000		122	STH 4,0(3)
000189	00194 5440 884C	C284E		123	AH 4,=H,150*
000190	00198 5A30 8826	02828		124	A 3,=F,33*
000191	0019C 47F0 916E			125	R READDEP
000192	001A0 4130 AF15	00170	01F17	126	LA 3,TARMAJ
000193				127	GET CARD,TEM
000194	D202 AA01 A981 01A03 01983			133	MVC TEST,TEM
000195	JJJ19A D21D 3000 A981 00000 01983			134	MVC 0(30,3),TEM
000196	001C0 1943			135	LR 4,3
000197	0001C2 5A30 882A	0282C		136	A 3,=F,150*
000198				137	GET CARD,TEM
000199				143	CLC TEM(4),=C*
000200	D503 A9B1 9822 01983 02824			144	BER 7
000201	0010C 0787			145	CLC TEM(3),TEST
000202	0001DE D502 A981 AA01 01983 01A03			146	BNE MOVE
000203	0001E4 4770 9182	00184		147	A 4,=F,30*
000204	0001E8 5A40 882E	02830		148	MVC 0(30,4),TEM
000205	0001EC D21D 4000 A981 00000 01983			149	B READMAJ
000206	0001F2 47F0 91C4	001C6		150	LA 3,TABDEP
000207	0001F6 4130 A026	01D28		151	CLC DEPCODE,2(3)
000208	0001FA D502 AA04 3002 01A06 00002			152	RE SMAJOR
000209	000200 4780 929A	0020C		153	A 3,=F,33*
000210	000204 5A30 8826			154	R COMPARE1
000211	000208 47F0 91F8	001FA		155	MVC DEPART,5(3)
000212	00020C D218 AB45 3005 01847 01F17			156	LA 4,TARMAJ
000213	000212 4140 AF15			157	AH 4,0(3)
000214	000216 4A43 0000			158	CLC MAJCODE,0(4)
000215	00021A D503 AA07 4000 01A09 00000			159	RE 4,=F,33*
000216	000220 4780 922A	0022C		160	A COMPARE2
000217	000224 5A4C 882E			161	B 4,=F,33*
000218	000228 47F0 9218	0021A		162	MVC MAJOR,4(4)
000219	00022C D219 AB6F 4000 01871 00004			163	CLC MAJCODE,2(3)
000304	47F0 90CA	000CC		164	PUT PRINT,SP,1
000316	D283 A0FF A0FE 01101 01100			169	PUT PRINT,OUT1
00031C	D20E A127 8857 01129 02859			175	PUT PRINT,SP,1
				181	PUT PRINT,OUT2
				187	PUT PRINT,SP,2
				193	PUT PRINT,OUT3
				199	PUT PRINT,SP,2
				205	PUT PRINT,OUT4
				211	PUT PRINT,SP,2
				217	PUT PRINT,OUT6
				223	CLC PRINT,SP,2
				229	PUT PRINT,OUT5
				235	CLC PRINT,SP,2
				241	PUT PRINT,OUT6
				247	B OUTR
				248	CLC PRINT,SP,1
				254	EDFRTN OUTAREA,SPACES
				255	MVC OUTAREA,4(15),=C*END-OF-RETRIEVE*

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

256 PUT PRINT,OUTAREA
 262 CLOSE DISK,ISCRS,ISGRD,PRINT,CARD
 274 EOJ
 277 FILE
 431 FILE PRINT
 582 DISK DTFDA
 BLKSIZE=64,
 DEVICE=3340,
 ERRBYTE=DSKERR,
 IOAREAR=DSKIN,
 SEEKADR=CRSDA,
 TYPEFL=INPUT,
 DEVADR=SYS014,
 DSKXTNT=1,
 KEYARG=KEYARG,
 KEYLEN=10,
 MODNAME=IJJFZZZ,
 READKEY=YES,
 RECFORM=FIXUNB,
 RELTYPE=DEC
 665 ISCRS DTFFS
 DSKXTNT=3,
 IOROUT=RETRVE,
 KEYLEN=006,
 NRECS=040,
 HINDEX=3340,
 RECFORM=FIXBLK,
 MODNAME=JHZZRY,
 REGSIZE=0025,
 DEVICE=3340,
 WORKR=CRSRT,
 IOAREAR=CRRTVE,
 KEYARG=CRSKEY,
 RDONLY=YES,
 TYPEFL=RANDOM,
 KEYLOC=2,
 VERIFY=YES
 747 ISGRD DTFFS
 DSKXTNT=3,
 IOROUT=RETRVE,
 KEYLEN=012,
 NRECS=040,
 HINDEX=3340,
 RECFORM=FIXBLK,
 REGSIZE=0025,
 DEVICE=3340,
 ICAREAR=GDPTVE,
 MODNAME=JHZZRY,
 IOAREAS=ONAM,
 KEYARG=GRDKEY,
 RDONLY=YES,
 TYPEFL=RAMSEC,
 WORKR=GRDRT,
 KEYLOC=2,

LDC	OBJECT CODE	ADDR 1	ADDR 2	STMT	SOURCE STATEMENT	VERIFY=YES, WORKS=YES
000850				829	DSKERR DS	CL2
000852				830	DSKIN DS	OCL64
000852				831	A DS	CL8
000854				832	B DS	CL10
000854				833	NAME DS	CL36
000888				834	KEYARG DS	CL10
000892				835	CRSDA DS	CL10
00089C				836	CRSKEY DS	CL6
0008A2				837	CRRTVE DS	CL1000
000C8A				838	CRSRT DS	OCL25
000C8A				839	DS DS	CL1
000C8B				840	CKEY DS	CL6
000C91				841	DATA1 DS	CL18
000CA3				842	GRDKEY DS	CL12
000CAF				843	GRDTRV DS	CL1000
001097				844	GRDRT DS	OCL25
001097				845	DS DS	CL1
001098				846	KEY DS	OCL12
001098				847	GFID DS	CL7
00109F				848	CYR DS	CL2
0010A1				849	CSM DS	CL1
0010A2				850	CSQ DS	CL2
0010A4				851	CID DS	CL6
0010AA				852	CCR DS	CL2
0010AC				853	CAD DS	CL1
0010AD				854	CSC DS	CL2
0010AF				855	CGD DS	CL1
0010B0				856	INAREA DS	OCL80
0010B0				857	IDC DS	CL7
0010B7				858	DS DS	CL73
001100 40				859	SPACFS DC	CL1*
001101				860	OUTAREA DS	OCL132
001101				861	HEADER DS	CL40
001129				862	KEYTRV DS	CL92
001185				863	KEYTRV DS	CL7
00118C				864	MRTVE DS	CL1920
00190C				865	OUTAREI DS	OCL132
00190C				866	DS DS	CL23
001923				867	CRND DS	CL6
001929				868	CRND DS	CL7
001930				869	CRNA DS	CL18
001942				870	DS DS	CL4
001946				871	PRSEC DS	CL2
001948				872	DS DS	CL9
001951				873	PRCR DS	CL2
001953				874	DS DS	CL9
00195C				875	PRAU DS	CL1
00195D				876	DS DS	CL7
001964				877	PRGR DS	CL1

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
001965				878	DS CL10
001966				879	DS CL11
001970				880	DS CL10
00197A				881	DS CL2
00197C				882	DS CL30
00199A				883	DS CL25
001983				884	DS CL80
001A03				885	DS CL3
001A06				886	DS OCL7
001A06				887	DS CL3
001A09				888	DS CL4
001A0D 40				889	DC C*
001A0E				890	DS OCL132
001A0E 40404040404040				891	DC 58C*
001A08 E2E3E4C505E340				892	DC C*STUDENT REPORT*
001A57 40404040404040				893	DC 59C*
001A92				894	DS OCL132
001A92 40404040404040				895	DC 58C*
001ACC 5C5C5C5C5C5C5C				896	DC 15C**
001ADR 40404040404040				897	DC 59C*
001B16				898	DS OCL132
001B16 40404040404040				899	DC 33C*
001B37 C4C5D7C1D9E3D4C5				900	DC C*DEPARTMENT OF
001B47				901	DS CL28
001B63 40404040404040				902	DC C*
001B71				903	DS MAJOR
001B88 40404040404040				904	DC 15C*
001B9A				905	DS OCL132
001B9A 40404040404040				906	DC 33C*
001B38 C9C440D5E4D4C2C5				907	DC C*ID NUMBER
001B07				908	DS CL7
001BCE 40404040404040				909	DC C*TIME
001B0C 40404040404040				910	DC CL4**
001C0D 40404040404040				911	DC 17C*
001C1E				912	DS OCL132
001C1E 40404040404040				913	DC 22C*
001C34 C3D6E4D9E2C560D5				914	DC C*COURSE-NO
001C45 C3D6E4D9E2C560D5				915	DC C*COURSE-NAME
001C55 E2C5C3E3C9D6D9540				916	DC C*SECTION
001C61 C3D9C5C9C9E34040				917	DC C*CREDIT
001C6C C1E4C9E3404040				918	DC C*AUDIT
001C79 40404040404040				919	DC C*GRADE*
001C86 40404040404040				920	DC C*SEMESTER*
001C8F 40404040404040				921	DC C*YEAR*
001CA2				922	DS 19C*
001CA2 40404040404040				923	DS OCL132
001C88 60606060606060				924	DC 22C*
001D13 40404040404040				925	DC 91C**
001D28				926	DC 19C*
001D28				927	DS OF
001F17				928	DS TABDEP
					DS TABMAJ
					DS 15CL33
					DS 75CL30

LOC	OBJECT CODE	ADDR 1	ADDR 2	STMT	SOURCE STATEMENT
			00000	929	BEGIN
0027E8	5B58C206D7C30540			930	=C' \$OPEN
0027F0	5B58C2C30304E2C5			931	=C' \$CLOSE
0027F8	000003E0			932	=A(PRINT)
0027FC	D9C5E309C9C5E5E5			933	=C'RETRIEVE-DAN*
002808	00001101			934	=A(OUTAREA)
00280C	00000358			935	=A(CARD)
002810	00001983			936	=A(EN)
002814	000094C0			937	=A(DISK)
002818	00000718			938	=A(LSQRD)
00281C	000005E0			939	=A(LSCR S)
002820	0000190C			940	=A(OUTAREA)
002824	00404340			941	=C' *
002828	00000021			942	=F'EE
00282C	00000096			943	=F'50
002830	0000001E			944	=F'30
002834	0000140E			945	=A(OUT1)
002838	00001A92			946	=A(OUT2)
00283C	00001B16			947	=A(OUT3)
002840	0000199A			948	=A(OUT4)
002844	00001C42			949	=A(OUT5)
002848	00001C1E			950	=A(OUT5)
00284C	0000			951	=H'0
00284E	0096			952	=H'150
002850	D9C5C1C440C3C1D9			953	=C'READ CARD*
002854	C5D5C4600A6660D9			954	=C'END-OF-RETRIEVE*

ภาคผนวก ฉ


```

CPL STATE ID DIVISION
00001 PROGRAM-NO. DIVISION
00002 PROGRAM-NO. CREAT.
00003 AUTHOR MISS PATANIA BENJAPOLKUL
00004 DATE-COMPILED. C3/07/80
00005 REMARKS * * * * *
00006 * * * * * THIS PROGRAM USE TO CREATE INDEX *
00007 * * * * *
00008 ENVIRONMENT DIVISION
00009 CONFIGURATION SECTION
00010 OBJECT-COMPUTER IBM-370
00011 INPUT-OUTPUT SECTION
00012 FILE-CONTROL
00013 SELECT TAPE-MS-I-FILE ASSIGN TO SYS010-UI-2400-5
00014 SELECT D-MSGRD-O-FILE ASSIGN TO SYS007-DA-3340-I-MASG
00015 ACCESS IS SEQUENTIAL
00016 RECORD KEY IS DISK-KEY-GRADE
00017 DIVISION
00018 SECTION
00019 TO D-RECORD-G-FILE
00020 BLOCK CONTAINS 40 RECORDS
00021 RECORD CONTAINS 48 CHARACTERS
00022 RECORDING MODE IS F
00023 LABEL RECORD IS STANDARD
00024 DATA RECORD IS D-MSGRD-O-REC.
00025 D) D-MSGRD-O-REC. PIC X
00026 05 DISK-DELETE-CODE PIC X
00027 05 DISK-KEY-GRADE
00028 07 DISK-ID PIC X(17)
00029 07 DISK-YEAR PIC XX
00030 07 DISK-SEMESTER PIC X
00031 07 DISK-SEQUENCE PIC XX
00032 05 DISK-COURSE PIC X(12)
00033 FD TAPE-MS-I-FILE
00034 BLOCK CONTAINS 40 RECORDS
00035 RECORD CONTAINS 47 CHARACTERS
00036 RECORDING MODE IS F
00037 LABEL RECORD IS OMITTED
00038 DATA RECORD IS TAPE-MS-I-REC.
00039 01 TAPE-MS-I-REC.
00040 05 TAPE-KEY-GRADE. PIC X
00041 07 ID-ALPHA PIC X(6)
00042 07 ID-RUNPIC PIC XX
00043 07 TAPE-YEAR PIC XX
00044 07 TAPE-SEMESTER PIC X
00045 07 TAPE-SEQUENCE PIC XX
00046 05 TAPE-COURSE PIC X(12)
00047 MESSAGE-STORAGE SECTION
00048 77 INPUT-REC-CNT PIC 9(5) COMP VALUE ZERO
00049 77 OUTPUT-REC-CNT PIC 9(5) COMP VALUE ZERO
00050 77 INPUT-ERROR-CNT PIC 9(5) COMP VALUE ZERO
00051 77 INVALID-REC-CNT PIC 9(5) COMP VALUE ZERO
00052 01 SWITCH-ARFA.

```

16-19136 03/07/80

```

00053 05 MS-TAPE-I-SWITCH-A PIC 'X'.
00054 06 MS-TAPE-I-SWITCH VALUE 'VI'.
PROCEDURE DIVISION.
00055 * MAIN LEVEL STARTING HERE
00056 * INITIALIZATION
00057 * OPEN FILES
00058 * OPEN INPUT TAPE-MS-I-FILE
00059 * OPEN OUTPUT D-MSGRD-D-FILE.
00060 * MAIN PROCESS
00061 PERFORM 010-READ-TAPE-MS-I THRU 010-READ-TAPE-EXT.
00062 UNTIL MS-TAPE-I-SWITCH.
00063 * CONCLUSION
00064 DISPLAY ' TOTAL INPUT RECORD = ' INPUT-REC-CNT.
00065 DISPLAY ' TOTAL OUTPUT RECORD = ' OUTPUT-REC-CNT.
00066 DISPLAY ' TOTAL INPUT ERROR = ' INPUT-ERROR-CNT.
00067 DISPLAY ' TOTAL INVALID RECORD = ' INVALID-REC-CNT.
00068 CLOSE TAPE-MS-I-FILE
00069 D-MSGRD-D-FILE.
00070 STOP RUN.
00071 010-READ-TAPE-MS-I.
00072 READ TAPE-MS-I-FILE
00073 AT END
00074 MOVE 'VI' TO MS-TAPE-I-SWITCH-A
00075 GO TO 010-READ-TAPE-EXT.
00076
00077 ADD 1 TO INPUT-REC-CNT.
00078 IF IJID-NUMERIC IS NUMERIC AND
00079 IJID-ALPHA IS ALPHABETIC
00080 PERFORM 020-WRITE-D-MS-I THRU 020-WRITE-D-MS-EXT
00081 ELSE
00082 DISPLAY 'ERROR-INPUT' TAPE-MS-I-REC
00083 ADD 1 TO INPUT-ERRPR-CNT.
00084 EXIT.
00085 010-WRITE-D-MS-I.
00086 MOVE IJID-VALUE TO DISK-DELETE-CODE.
00087 MOVE TAPE-KEY-GRADE TO DISK-KEY-GRADE.
00088 MOVE TAPE-COURSE TO DISK-COURSE.
00089 WRITE D-MSGRD-D-REC
00090 INVALID KEY
00091 DISPLAY 'INVALID-RECORD' TAPE-KEY-GRADE
00092 ADD 1 TO INVALID-REC-CNT
00093 GO TO 020-WRITE-D-MS-EXT.
00094
00095 ADD 1 TO OUTPUT-REC-CNT.
00096 EXIT.

```

00001 ID DIVISION:
 00002 CREATA-ISAM-COURSE.
 00003 AUTHPR. MISS PATANA BENJAPLNJL.
 00004 DATE-COMPILED. 03/07/80
 00005 REMARKS. * * * * *
 00006 * * * * *
 00007 * * * * *
 00008 * * * * *
 00009 * * * * *
 00010 * * * * *

00011 ENVIRONMENT DIVISION.
 00012 CONFIGURATION SECTION.
 00013 OBJECT-COMPUTER. IBM-370.
 00014 INPUT-OUTPUT SECTION.
 00015 FILE-CONTROL.

00016 SELECT TAPE-MS-I-FILE ASSIGN TO SYS011-UT-2400-5.
 00017 SELECT D-MSCRS-Q-FILE ASSIGN TO SYS016-DA-3340-1-MAST
 00018 SELECT ACCESS IS SEQUENTIAL
 00019 SELECT CLEARD. KEY IS DISK-COURSE-ID.

00020 DATA DIVISION.
 00021 FILE SECTION.
 00022 FD D-MSCRS-Q-FILE
 00023 BLOCK CONTAINS 40 RECORDS
 00024 RECORD CONTAINS 25 CHARACTERS
 00025 RECORDING MODE IS F
 00026 LABEL RECORDS IS STANDARD
 00027 DATA RECORD IS D-MSCRS-Q-REC.

00028 01 D-MSCRS-Q-REC.
 00029 05 DISK-DELETE-CODE PIC X.
 00030 05 DISK-COURSE-ID PIC X(6).
 00031 05 DISK-COURSE-NAME PIC X(18).
 00032 FD TAPE-MS-I-FILE
 00033 BLOCK CONTAINS 40 RECORDS
 00034 RECORD CONTAINS 25 CHARACTERS
 00035 RECORDING MODE IS F
 00036 LABEL RECORD IS OMITTED
 00037 DATA RECORD IS TAPE-MS-I-REC.

00038 01 TAPE-MS-I-REC.
 00039 05 TAPE-COURSE-ID PIC X(6).
 00040 05 TAPE-COURSE-NAME PIC X(18).
 00041 WORKING-STORAGE SECTION.
 00042 77 INPUT-REC-CNT PIC 9(15) COMP VALUE ZERO.
 00043 77 OUTPUT-REC-CNT PIC 9(15) COMP VALUE ZERO.
 00044 77 INPUT-ERR-CNT PIC 9(15) COMP VALUE ZERO.
 00045 77 INVALID-REC-CNT PIC 9(15) COMP VALUE ZERO.
 00046 01 SWITCH-AFFA.

00047 05 MS-TAPE-I-SWITCH-A PIC X.
 00048 99 MS-TAPE-I-SWITCH VALUE *Y*.
 00049 PROCEDURE DIVISION.
 00050 *OPEN FILES
 00051 OPEN INPUT TAPE-MS-I-FILE
 00052 OUTPUT D-MSCRS-Q-FILE.
 00053 MAIN PROCESS
 00054 PERFORM 010-READ-TAPE-MS-I THRU 010-READ-TAPE-EXT
 00055 UNTIL MS-TAPE-I-SWITCH.
 00056 CONCLUSION

CREATAD1 16-13-19 03/07/80

```
00054 DISPLAY , TOTAL INPUT RECORD = , INPUT-REC-CNT.
00055 DISPLAY , TOTAL OUTPUT RECORD = , OUTPUT-REC-CNT.
00056 DISPLAY , TOTAL INPUT FRPOT = , INPUT-FRPOS-CNT.
00057 DISPLAY , TOTAL INVALID RECORD = , INVALID-REC-CNT.
00058 CLOSE TAPE-MS-1-FILE
00059 D-MSCPS-Q-FILE.
00060 STOP
00061 010-READ-TAPE-MS-1.
00062 READ TAPE-MS-1-FILE
00063 AT END
00064 MOVE ,V, TC MS-TAPE-1-SWITCH-A
00065 GO TO 010-READ-TAPE-EXT.
00066 ADD 1 TO INPUT-REC-CNT.
00067 IF TAPE-COURSE-ID IS NUMERIC
00068 PERFORM 020-WRITE-D-MS-Q THRU 020-WRITE-D-MS-QXT
00069 ELSE
00070 DISPLAY 'LPROP-INPUT' TAPE-MS-1-REC
00071 ADD 1 TO INPUT-ERROR-CNT.
00072 EXIT.
010-READ-TAPE-EXT.
020-WRITE-D-MS-Q.
00073 MOVE LOW-VALUE TO DISK-DELETE-CODE.
00074 MOVE TAPE-COURSE-ID TO DISK-COURSE-ID.
00075 MOVE TAPE-COURSE-NAME TO DISK-COURSE-NAME.
00076 WRITE D-MSCPS-Q-REC
00077 INVAL ID KEY
00078 DISPLAY 'INVALID-RECORD' TAPE-COURSE-ID
00079 ADD 1 TO INVALID-REC-CNT
00080 GO TO 020-WRITE-D-MS-EXT.
00081 ADD 1 TO OUTPUT-REC-CNT.
00082 020-WRITE-D-MS-EXT.
00083 EXIT.
```



```

00001 ID DIVISION.
00002 PROGRAM-ID. CREATE.
00003 AUTHOR. MISS KATANA BENJAPOLKUL.
00004 DATE-COMPILED. 03/07/80
00005 * * * * *
00006 * * * * *
00007 * * * * *
00008 * * * * *
00009 * * * * *
00010 * * * * *
00011 * * * * *
00012 * * * * *
00013 * * * * *
00014 * * * * *
00015 * * * * *
00016 * * * * *
00017 * * * * *
00018 * * * * *
00019 * * * * *
00020 * * * * *
00021 * * * * *
00022 * * * * *
00023 * * * * *
00024 * * * * *
00025 * * * * *
00026 * * * * *
00027 * * * * *
00028 * * * * *
00029 * * * * *
00030 * * * * *
00031 * * * * *
00032 * * * * *
00033 * * * * *
00034 * * * * *
00035 * * * * *
00036 * * * * *
00037 * * * * *
00038 * * * * *
00039 * * * * *
00040 * * * * *
00041 * * * * *
00042 * * * * *
00043 * * * * *
00044 * * * * *
00045 * * * * *
00046 * * * * *
00047 * * * * *
00048 * * * * *
00049 * * * * *
00050 * * * * *
00051 * * * * *
00052 * * * * *
00053 * * * * *

```

ENVIRONMENT DIVISION.
 CONFIGURATION SECTION.
 OBJECT-COMPUTER. IBM-370.
 INPUT-OUTPUT SECTION.
 FILE-CONTROL.
 00014. SELECT TAPE-MS-I-FILE ASSIGN TO SYS010-UT-2400-S.
 00015. D-MSNAME-D-FILE ASSIGN TO SYS013-DA-3340-I-MSN
 00016. ACCESS IS SEQUENTIAL.
 RECORD KEY IS DISK-STUDENT-ID.
 DIVISION.
 SECTION.
 FD D-MSNAME-D-FILE
 BLOCK CONTAINS 40 RECORDS-
 RECORD CONTAINS 48 CHARACTERS
 LABEL IS F
 DATA IS STANDARD
 01 D-MSNAME-D-REC. IS D-MSNAME-D-REC.
 05 DISK-DELETE-CODE PIC X.
 05 DISK-STUDENT-ID PIC X(7).
 05 DISK-STUDENT PIC X(40).
 FD TAPE-MS-I-FILE
 BLOCK CONTAINS 40 RECORDS
 RECORD CONTAINS 47 CHARACTERS
 RECORDING MODE IS F
 LABEL IS EMITTED
 DATA IS TAPE-MS-I-REC.
 05 TAPE-STUDENT-ID. PIC A.
 07 ID-ALPHA PIC X(6).
 05 TAPE-STUDENT PIC X(40).
 WORKING-STORAGE SECTION.
 77 INPUT-REC-CNT PIC 9(15) COMP VALUE ZERO.
 77 OUTPUT-REC-CNT PIC 9(15) COMP VALUE ZERO.
 77 INPUT-ERROR-CNT PIC 9(15) COMP VALUE ZERO.
 77 INVALID-REC-CNT PIC 9(15) COMP VALUE ZERO.
 01 SWITCH-AREA.
 05 MS-TAPE-I-SWITCH-A PIC X.
 88 MS-TAPE-I-SWITCH VALUE 'Y'.
 PROCEDURE DIVISION.
 * MAIN LEVEL STARTING HERE
 * INITIALIZATION
 * OPEN FILES INPUT TAPE-MS-I-FILE
 OPEN

CREATE 15.17.01 03/07/80

```

00054 OUTPUT D-MSNAME-D-FILE.
00055 MAIN DELGESS
00056 PERFORM 010-READ-TAPE-MS-I THRU 010-READ-TAPE-EXT
00057 UNTIL MS-TAPE-I-SWITCH.
00058 *****
00059 DISPLAY * TOTAL INPUT RECORDS = * INPUT-REC-CNT.
00060 DISPLAY * TOTAL OUTPUT RECORDS = * OUTPUT-RFC-CNT.
00061 DISPLAY * TOTAL INPUT ERRORS = * INPUT-ERRS-CNT.
00062 DISPLAY * TOTAL INVALID RECORDS = * INVALID-REC-CNT.
00063 CLOSE TAPE-MS-I-FILE
00064 D-MSNAME-D-FILE.
00065 STOP.
00066 010-READ-TAPE-MS-I.
00067 READ TAPE-MS-I-FILE
00068 AT END
00069 MOVE MS-I-ID-MS-TAPE-I-SWITCH-A
00070 GO TO 010-READ-TAPE-EXT.
00071 ADD 1 TO INPUT-REC-CNT.
00072 IF ( ID-ALPHA IS ALPHABETIC AND
00073 ID-NUMERIC IS NUMERIC )
00074 PERFORM 020-WRITE-D-MS-D THRU 020-WRITE-D-MS-EXT
00075 ELSE
00076 DISPLAY *ERROR-INPUT* TAPE-MS-I-REC
00077 ADD 1 TO INPUT-ERRS-CNT.
00078 010-READ-TAPE-EXT. EXIT.
00079 020-WRITE-D-MS-D.
00080 MOVE LOW-VALUE TO DISK-DELETE-CODE.
00081 MOVE TAPE-STUDENT-ID TO DISK-STUDENT-ID.
00082 MOVE TAPE-STUDENT
00083 WRITE D-MSNAME-D-REC
00084 INVALID KEY
00085 DISPLAY *INVALID-RECORD* TAPE-STUDENT-ID
00086 ADD 1 TO INVALID-REC-CNT
00087 GO TO 020-WRITE-D-MS-EXT.
00088 ADD 1 TO
00089 020-WRITE-D-MS-EXT. EXIT.

```

STATISTICS

SUBREC RECORDS =	89	DATA ITEMS =	18	PROG DU/22 =	1
PARTITION SIZE =	40940	LIVE COUNT =	56	NUMBER OF	1
MAP FELC AD =	NONE	SPACING =	1	LINK	REC TRX
MULTIX	APOST	MSY4	APCATAL	NOXRC	NOXRC
NOCLIST	FLAG	ZMB	MSUPMAP	MCDECK	MCDECK
NOSTATE	TRUNC	SEP	MSYMDMP	MEMVERB	MEMVERB
ACCOUNT	ACCOUNT	MSYMB	MSYMB	MEMVERB	MEMVERB
NONE	NONE	MSYMB	MSYMB	MEMVERB	MEMVERB

LISTER OPTIONS

CBL STATE
CBL LIB

00001 ID DIVISION.
 00002 PROGRAM-ID. RETRIEVE-ISAM.
 00003 AUTHOR. MISS CATANA BENJAPLKUL.
 00004 DATE-COMPILED. 03/07/80.
 00005 * * * * *
 00006 * THIS PROGRAM USE TO RETRIEVE INDEXED *
 00007 * SEQUENTIAL FILE ON DISK 334C *
 00008 * * * * *

ENVIRONMENT
 00009 CONFIGURATION SECTION.
 00010 OBJECT-COMPUTER. IBM-370.
 00011 INPUT-OUTPUT SECTION.
 00012 FILE-CONTROL.

00013 SELECT CRD-FILE ASSIGN TO SYS004-DA-2540R-S.
 00014 SELECT RET-FILE ASSIGN TO SYS006-UR-1403-S.
 00015 SELECT D-NAME-FILE ASSIGN TO SYS013-DA-3340-I-MASS.
 00016 ACCESS IS RANDOM
 00017 NOMINAL KEY IS NAME-KEY
 00018 RECORD KEY IS DISK-STUDENT-ID.
 00019 SELECT D-GRADE-FILE ASSIGN TO SYS007-DA-3340-I-MASS
 00020 ACCESS IS SEQUENTIAL
 00021 NOMINAL KEY IS GRADE-KEY
 00022 RECORD KEY IS DISK-GRADE.
 00023 SELECT D-COURSE-FILE ASSIGN TO SYS014-DA-3340-I-MASS
 00024 ACCESS IS RANDOM
 00025 NOMINAL KEY IS COURSE-KEY
 00026 RECORD KEY IS DISK-CS-ID.
 00027 DIVISION.
 00028 SECTION.

DATA
 00029 FILE
 00030 FD CRD-FILE
 00031 RECORD MODE IS F
 00032 LABEL IS OMITTED
 00033 DATA IS CRD-REC.
 00034 PIC X(180).
 00035
 00036 FD RET-FILE
 00037 RECORD MODE IS F
 00038 LABEL IS OMITTED
 00039 DATA IS PR1-REC.
 00040 PIC X(132).
 00041
 00042 FD D-NAME-FILE
 00043 RECORD MODE IS F
 00044 LABEL IS STANDARD
 00045 DATA IS D-NAME-REC.
 00046 PIC X.
 00047
 00048 FD D-NAME-REC.
 00049 RECORD MODE IS F
 00050 LABEL IS STANDARD
 00051 DATA IS D-NAME-REC.
 00052 PIC X(17).
 00053
 00054 FD DISK-STUDENT-ID
 00055 RECORD MODE IS F
 00056 LABEL IS STANDARD
 00057 DATA IS D-NAME-REC.
 00058 PIC X(17).
 00059
 00060 FD STUDENT.
 00061 RECORD MODE IS F
 00062 LABEL IS STANDARD
 00063 DATA IS D-NAME-REC.
 00064 PIC X(17).
 00065

03/07/00

16.21.1R

BEIRIFVE

03	FILLER	PIC	X(13)	VALUE	DEPARTMENT
00383	03 FILLER	PIC	X(13)	VALUE	SPACE
00384	03 FILLER	PIC	X(13)	VALUE	SPACE
00385	03 HEAD2-1	PIC	X(13)	VALUE	SPACE
00386	03 FILLER	PIC	X(15)	VALUE	MAJOR
00387	03 FILLER	PIC	X(15)	VALUE	SPACE
00388	03 HEAD2-2	PIC	X(15)	VALUE	SPACE
00389	03 FILLER	PIC	X(126)	VALUE	SPACE
00390	01 HEAD3	PIC	X(33)	VALUE	SPACE
00391	03 FILLER	PIC	X(19)	VALUE	IC NUMBER
00392	03 FILLER	PIC	X(19)	VALUE	SPACE
00393	03 FILLER	PIC	X(12)	VALUE	SPACE
00394	03 HEAD3-1	PIC	X(14)	VALUE	NAME
00395	03 FILLER	PIC	X(15)	VALUE	SPACE
00396	03 FILLER	PIC	X(16)	VALUE	SPACE
00397	03 HEAD3-2	PIC	X(130)	VALUE	SPACE
00398	03 FILLER	PIC	X(130)	VALUE	SPACE
00399	03 FILLER	PIC	X(122)	VALUE	SPACE
00400	01 HEAD4	PIC	X(19)	VALUE	COURSE-NO
00401	03 FILLER	PIC	X(19)	VALUE	SPACE
00402	03 FILLER	PIC	X(11)	VALUE	COURSE-NAME
00403	03 FILLER	PIC	X(15)	VALUE	SPACE
00404	03 FILLER	PIC	X(17)	VALUE	SECTION
00405	03 FILLER	PIC	X(15)	VALUE	SPACE
00406	03 FILLER	PIC	X(16)	VALUE	CREDIT
00407	03 FILLER	PIC	X(15)	VALUE	SPACE
00408	03 FILLER	PIC	X(15)	VALUE	AUDIT
00409	03 FILLER	PIC	XXX	VALUE	SPACE
00410	03 FILLER	PIC	X(15)	VALUE	GRADE
00411	03 FILLER	PIC	X(15)	VALUE	SPACE
00412	03 FILLER	PIC	X(18)	VALUE	SEMESTER
00413	03 FILLER	PIC	X(15)	VALUE	SPACE
00414	03 FILLER	PIC	X(15)	VALUE	SPACE
00415	03 FILLER	PIC	X(19)	VALUE	YEAR
00416	03 FILLER	PIC	X(19)	VALUE	SPACE
00417	03 FILLER	PIC	X(23)	VALUE	SPACE
00418	01 HEAD5	PIC	X(16)	VALUE	SPACE
00419	03 FILLER	PIC	X(17)	VALUE	SPACE
00420	03 PRT-COURSE-NO	PIC	X(17)	VALUE	SPACE
00421	03 FILLER	PIC	X(18)	VALUE	SPACE
00422	03 PRT-COURSE-NAME	PIC	X(14)	VALUE	SPACE
00423	03 FILLER	PIC	XX	VALUE	SPACE
00424	03 PRT-SECTION	PIC	X(9)	VALUE	SPACE
00425	03 FILLER	PIC	XX	VALUE	SPACE
00426	03 PRT-CREDIT	PIC	X(9)	VALUE	SPACE
00427	03 FILLER	PIC	X	VALUE	SPACE
00428	03 PRT-AUDIT	PIC	X(7)	VALUE	SPACE
00429	03 FILLER	PIC	X	VALUE	SPACE
00430	03 PRT-GRADE	PIC	X(10)	VALUE	SPACE
00431	03 FILLER	PIC	X	VALUE	SPACE
00432	03 PRT-SEMESTER	PIC	X(10)	VALUE	SPACE
00433	03 FILLER	PIC	XX	VALUE	SPACE
00434	03 PRT-YEAR	PIC	X(30)	VALUE	SPACE
00435	03 FILLER	PIC	X(30)	VALUE	SPACE


```

00436 01 HEAD6.
00437 03 FILLER VALUE SPACE.
00438 03 FILLER PIC X(16) VALUE *****
00439 03 FILLER PIC X(58) VALUE SPACE.
00440 01 HEAD7.
00441 03 FILLER PIC X(22) VALUE SPACE.
00442 03 S PIC X(91).
00443 03 FILLER PIC X(58) VALUE SPACE.
00444 01 WS-CRD-REC.
00445 03 CRD-ID.
00446 05 ID-ALPHA X.
00447 05 ID-NUMERIC PIC 9(6).
00448 03 CRD-YEAR PIC XX.
00449 03 CRD-SEMESTER PIC X.
00450 03 CRD-SEQUENCE PIC XX.
00451 01 SWITCH-AREA.
00452 05 WS-CRD-SWITCH-A PIC X.
00453 05 WS-CRD-SWITCH-B PIC X.
00454 05 TRANS-VALID-SWITCH-A PIC X.
00455 05 TRANS-VALID-SWITCH-B PIC X.
00456 05 EQ-SWITCH-NAME-A PIC X.
00457 05 EQ-SWITCH-NAME-B PIC X.
00458 PROCEDURE DIVISION.
00459 OPEN INPUT CRD-FILE
00460 D-NAME-FILE
00461 D-NAME-FILE
00462 D-COURSE-FILE
00463 D-COURSE-FILE
00464 OUTPUT PRT-FILE.
00465 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00466 UNTIL TRANS-VALID-SWITCH.
00467 PERFORM 020-READ-NAME.
00468 PERFORM 030-COMPARE-NAME-CRD THRU 030-COMPARE-NAME-CRD-EXT
00469 UNTIL WS-CRD-SWITCH.
00470 CONCLUSION
00471 DISPLAY 'TOTAL READ TRANS = ' TRANS-CNT.
00472 DISPLAY 'TOTAL ERRCR CARD = ' ERRCR-CRD-CNT.
00473 DISPLAY 'TOTAL WRITE REPORT = ' PRT-REC-CNT.
00474 DISPLAY 'TOTAL ERROR SEQ CARD = ' ERR-SEQ-CNT.
00475 DISPLAY 'TOTAL INVALID CRD = ' INVALID-REC-CNT.
00476 CLOSE CRD-FILE
00477 X D-NAME-FILE
00478 D-NAME-FILE
00479 D-COURSE-FILE
00480 PRT-FILE.
00481 STOP
00482 READ CRD-FILE INTO WS-CRD-REC
00483 AT END
00484 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00485 MOVE 'Y' TO WS-LRD-SWITCH-A
00486 GO TO 010-READ-TRANS-EXT.
00487 ADD 1 TO TRANS-CNT.
00488 IF (ID-ALPHA IS ALPHABETIC AND

```

```

00489 ID-NUMERIC IS NUMERIC I
00490 MOVE 'Y' TO TRANS-VALID-SWITCH-A
ELSE
00491 MOVE 'N' TO TRANS-VALID-SWITCH-A
00492 DISPLAY 'ERROR-CARD' WS-CRD-REC
00493 ADD 1 TO ERROR-CARD-CNT.
010-READ-TRANS-EXT. EXIT.
020-READ-NAME.
MOVE CRD-ID TO NAME-KEY.
MOVE D-NAME-FILE
INVALID KEY
MOVE 'Y' TO EQ-SWITCH-NAME-A.
030-COMPARE-NAME-CRD.
IF EQ-SWITCH-NAME
MOVE 'N' TO EQ-SWITCH-NAME-A
DISPLAY 'ERROR-TRANS' WS-CRD-REC
ADD 1 TO ERR-SEQ-CNT
MOVE 'N' TO TRANS-VALID-SWITCH-A
PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
TRANS-VALID-SWITCH
PERFORM 020-READ-NAME
ELSE
PERFORM 040-BEETLE-HEAD
PERFORM 040-START-GRADE THRU 040-START-GRADE-EXT
PERFORM 050-READ-GRADE THRU 050-READ-GRADE-EXT
UNTIL CRD-ID NOT = DISK-ID
MOVE 'N' TO TRANS-VALID-SWITCH-A
PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
UNTIL TRANS-VALID-SWITCH
PERFORM 020-READ-NAME.
030-COMPARE-NAME-CRD-EXT. EXIT.
MOVE WS-CRD-REC TO GRADE-KEY.
START D-GRADE-FILE KEY IS EQUAL TO CRD-ID
INVALID KEY
DISPLAY 'ERROR-ID' WS-CRD-REC
ADD 1 TO INVALID-REC-CNT
GO TO 040-START-GRADE-EXT.
READ D-GRADE-FILE AT END DISPLAY 'END OF GRADE-FILE'.
040-START-GRADE-EXT. EXIT.
050-READ-GRADE.
MOVE DISK-COURSE-ID TO COURSE-KEY.
READ D-COURSE-FILE
INVALID KEY
DISPLAY 'ERROR-COURSE-FILE' WS-CRD-REC
ADD 1 TO INVALID-REL-CNT
GO TO 050-READ-GRADE-EXT.
MOVE DISK-CS-ID TO PRI-COURSE-NO.
MOVE DISK-CS-NAME TO PRI-COURSE-NAME.
MOVE DISK-SECTION TO PRI-SECTION.
MOVE DISK-CREDIT TO PRI-CREDIT.
MOVE DISK-AUDIT TO PRI-AUDIT.
MOVE DISK-GRADE TO PRI-GRADE.
    
```

RETRIEVE 16.21.18 03/07/80

00542 MOVE DISK-SEMESTER TO PRT-SEMESTER.
 00543 MOVE DISK-YEAR TO PRT-YEAR.
 00544 WRITE PRT-REC FROM HEAD5 AFTER POSITIONING 2 LINES.
 00545 READ D-GRADE-FILE AT END DISPLAY 'END OF GRADE-FILE'.
 00546 050-READ-GRADE-EXT. EXIT.
 00547 060-WRITE-HEAD.
 00548 ADD 1 TO PRT-REC-CNT.
 00549 WRITE PRT-REC FROM HEAD1 AFTER POSITIONING 0 LINES.
 00550 WRITE PRT-REC FROM HEAD4 AFTER POSITIONING 1 LINES.
 00551 PERFORM FIND-DEP THRU FIND-D.
 00552 PERFORM FIND-MAJ THRU FIND-E.
 00553 MOVE DEPT-N TO HEAD2-1.
 00554 MOVE MAJPN TO HEAD2-2.
 00555 WRITE PRT-REC FROM HEAD2 AFTER POSITIONING 2 LINES.
 00556 MOVE DISK-STUDENT-ID TO HEAD3-1.
 00557 MOVE DISK-NAME TO HEAD3-2.
 00558 WRITE PRT-REC FROM HEAD3 AFTER POSITIONING 2 LINES.
 00559 MOVE ALL 1-1 TO 5.
 00560 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
 00561 WRITE PRT-REC FROM HEAD4 AFTER POSITIONING 2 LINES.
 00562 MOVE ALL 1-1 TO 5.
 00563 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
 00564 DEP-FF SECTION. COPY CVREFDEP.
 00565 C DEP-FIND SECTION.
 00566 C FIND-DEP.
 00567 C SEARCH ALL CTRM1
 00568 C AT END GO TO NOT-F-DEP
 00569 C WHEN DEPT = DND (D)
 00570 C GO TO FOUND-D.
 00571 C FOUND-D.
 00572 C MOVE D-NAME (D) TO DEPT-N.
 00573 C GO TO FIND-D.
 00574 C NOT-F-DEP.
 00575 C MOVE SPACE TO DEPT-N.
 00576 C FIND-D.
 00577 C EXIT.
 00578 MAJ-FF SECTION. COPY CVREFMAJ.
 00579 MAJ-FIND SECTION.
 00580 C FIND-MAJ.
 00581 C SEARCH ALL MNAME
 00582 C AT END GO TO NOT-FOUND-M
 00583 C WHEN MAJOR = MND (A)
 00584 C GO TO FOUND-M.
 00585 C FOUND-M.
 00586 C MOVE 4-NAME (A) TO MAJORN.
 00587 C GO TO FIND-E.
 00588 C NOT-FOUND-M.
 00589 C MOVE SPACE TO MAJORN.
 00590 C FIND-E.
 00591 C EXIT

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

```

000000 PRINT CN,NOGEN
000000 START 0
000000 BALR 11,0
000000 USING HERE,11,12
000000 L 12,BASE
000000 B OPENF

```

```

7 BASE DC ATHERE*(4096)
8 OPENF OPEN TPCRS,PRINT,ISCRS
18 SETFL ISCRS
24 LTOP GET TPCRS,INAREA
30 MVC WORKLD*(124),INAREA
31 WRITE ISCRS,NEWKEY
36 L 4,=A(1)ISCRS(
37 T4 0(4),X*80,
38 R0 RTN1
39 T4 0(4),X*40,
40 R0 RTN2
41 T4 0(4),X*20,
42 R0 RTN3
43 R LOOP

```

```

44 FTN1 MVC OUTAREA,OUTAREA-1
45 MVC OUTAREA*(40(12)),=C'DISK ERROR'
46 PUT PRINT,OUTAREA
52 B LOOP
53 RTN2 MVC OUTAREA,OUTAREA-1
54 MVC OUTAREA*(40(12)),=C'WRONG LENGTH'
55 PUT PRINT,OUTAREA
61 B LOOP
62 RTN3 MVC OUTAREA,OUTAREA-1
63 MVC OUTAREA*(40(9)),=C'OVERFLOW'
64 PUT PRINT,OUTAREA
70 B LOOP
71 EOFTPC MVC OUTAREA,OUTAREA-1
72 CNTRL PRINT,SK,1
78 MVC OUTAREA*(40(34)),=C'PROGRAM BY MISS RATANA BENJAPUKUL'
79 ENDFL ISCRS
91 CLOSE TPCRS,PRINT,ISCRS
101 EDJ
104 FILE
255 ISCRS DTFFIS

```

```

000054 5840 8C12
000058 9180 4000
00005C 4710 8C72
000060 9140 400C
000064 4710 8092
000068 9120 4000
00006C 4710 8082
000070 47F0 8030
000074 0283 883E 883D 00840 0083F
00007A 0209 8866 8C2A 00848 00C2C

```

```

000090 47F0 8030 00032
000094 0283 883E 883D 00840 0083F
00009A 0208 8866 8C1E 00868 00C20
000080 47F0 8030 00032
000084 0283 883E 883D 00840 0083F
00008A 0208 8866 88EE 00868 008F0
000000 47F0 8030 00032
000004 0283 883E 883D 00840 0083F
00000E 0221 8866 8C34 00868 00C36

```

```

DSKXTNT=3,
IOPUT=LOAD,
KEYLEN=006,
NRECS=040,
HINDEX=3340,
RECFORM=FLXBLK,
REC SIZE=0025,
DEVICE=3340,
IDAREAL=MKLOAD,
MODNAME=IJHLZZZ,
KEYLOC=2,

```


LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
				304	TPCRS DTFMT VERIFY=YES WORKL=WORKLD BLKSIZE=960, DEVADDR=SYS011, EOFADDR=EOFTPC, IDAREAL=TAPEIN, RECFORM=FIXBLK, RECSIZE=24, FILABL=NO, WORKA=YES
000370				338	INAREA DS
000370				339	TPCFD DS
000376				340	CNSIN DS
000388				341	TAPEIN DS
000748				342	WKLOAD DS
00083F 40				343	OUTAREA DS
000840				344	OUTAREA DS
000840				345	HEADER1 DS
000868				346	HEADER1 DS
0008C4				347	WORKLD DS
			00000	348	BEGIN
				349	=C*\$\$BOPEN *
				350	=C*\$\$BSETL *
				351	=C*OVERFLOW *
				352	=C*\$\$BEMDFL *
				353	=C*\$\$BCLOSE *
				354	=A(IISCR)
				355	=A(TPCRS)
				356	=A(INAREA)
				357	=A(IISCR SC)
				358	=A(PRINT)
				359	=A(CUTAREA)
				360	=C*WRONG LENGTH *
				361	=C*DISK ERROR *
				362	=C*PROGRAM BY MISS RATANA BERJAJA ELKUL *
0008E0	5B58C2D6D7C5D540				
0008E8	5B58C2E2C5E3C6D3				
0008F0	D6E5C5D9C6D3D6E6				
0008F8	5B58C2C5D9C4C6D3				
000C00	5B58C2C3D3D6E2C5				
000C08	00000208				
000C0C	00000310				
000C10	00000370				
000C14	00000226				
000C18	00000128				
000C20	E6D9D6D5C740D3C5				
000C2C	C4C9E2D240C5D9D9				
000C36	D709D6C7D9C1D440				

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000000				1	PRINT ON,NOGEN
000000	0580			2	START 0
000002	58C0 800A		00002	3	BALR 11,0
000006	47F0 800E		0000C	4	USING HERE,11,12
00000A	0000	00010		5	L 12,BASE
00000C	00001002			6	B OPENF
000042	D217 88C3 836E 008C5	008C5	00370	7	BASE A(HERE+4096)
000054	5840 8C12		00C14	8	OPEN TPGRD,PRINT,ISGRD
000058	9180 4000		00000	18	SETFL ISGRD
00005C	4710 8C72		00074	24	GET TPGRD,INAREA
000060	9140 4000		00000	30	MVC W0RKLD(124),INAREA
000064	4710 8092		00094	31	WRITE ISGRD,NEWKEY
000068	9120 4000		00000	36	L 4,=ALLSRDC
00006C	4710 8082		00084	37	TM 014),X*80*
000070	47F0 8030		00032	38	BO RTN1
000074	D283 883E 883D 00840	00840	0083F	39	BO 014),X*40*
00007A	D209 8866 8C2A 00868	00868	00C2C	40	TM RTN2
000090	47F0 8030		00032	41	TM 014),X*20*
000094	D283 883E 883D 00840	00840	0083F	42	BO RTN3
00009A	D208 8866 8C1E 00868	00868	00C20	43	LOOP
000080	47F0 8030		00032	44	B OUTAREA,OUTAREA-1
000084	D283 883E 883D 00840	00840	0083F	45	MVC OUTAREA+4(10),=C'DISK ERROR'
00008A	D208 8866 88EE 00868	00868	00BFO	46	PUT PRINT,OUTAREA
0000D0	47F0 8030		00032	52	LOOP
0000D4	D283 883E 883D 00840	00840	0083F	53	MVC OUTAREA,OUTAREA-1
0000E8	D221 8866 8C34 00868	00868	00C36	54	MVC OUTAREA+4(12),=C'WRONG LENGTH'
				55	PUT PRINT,OUTAREA
				61	B LOOP
				62	MVC OUTAREA,OUTAREA-1
				63	MVC OUTAREA+4(9),=C'OVEFLOW'
				64	PUT PRINT,OUTAREA
				70	B LOOP
				71	MVC OUTAREA,OUTAREA-1
				72	CTRL PRINT,SK,1
				78	MVC OUTAREA+4(34),=C'PROGRAM BY MISS RANTANA REPAJAPUKHE'
				79	ENDFL ISGRD
				91	CLOSE TPGRD,PRINT,ISGRD
				101	EOJ
				104	FILE
				255	ISGRD
					PRINT
					DSKXINT=3,
					IDROUT=LOAD,
					KEYLEN=012,
					NRECD5=C40,
					HINDEX=3340,
					RECFORM=FIXBLK,
					MODNAME=1JHZZZ,
					RECSIZE=0025,
					DEVICE=3340,
					IDAREAL=WKLOAD,
					KEYLDC=2,

LDC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

```

000370
000370
00037C
000388
000748
00083F 40
000840
000840
000868
000864
0008E0 5858C2D607C5D540
0008E8 5858C2E2C5E3C6D3
0008F0 D6E5C5D9C6D3D6E6
0008F8 5858C2C5D5C4C6D3
000C00 5858C2C3D3D6E2C5
000C08 00000208
000C0C 00000310
000C10 00000370
000C14 00000226
000C18 00000128
000C1C 00000840
000C20 E6D9D6D5C74D0D3C5
000C2C C4C9E2D240C5D9D9
000C36 D7D9D6C7D9C1D440

338 INAREA DS
339 TPGD DS
340 GDIR DS
341 TAPEIN DS
342 WKLOAD DS
343 DC
344 OUTAREA DS
345 HEADER1 DS
346 DS
347 WORKLD DS
348 END
349
350
351
352
353
354
355
356
357
358
359
360
361
362

000000

VERIFY=YES,
WORKL=WORKLD
BLKSIZE=960,
DEVADDR=SYS010,
EOFADDR=EOFPG,
IDAREAL=TAPEIN,
RECFORM=FIXBLK,
RECSIZE=24,
FILABL=NC,
WORKA=YES
OCL24
CLL2
CLL2
CL960
CL1015
CLL1
OCLL32
CL40
CL92
CL25
BEGIN
=C'$$$BOPEN'
=C'$$$SETFL'
=C'OVERFLOW'
=C'$$$ENDFL'
=C'$$$RCLOSE'
=A(ISGRD)
=A(ITPGD)
=A(INAREA)
=A(ISGRDC)
=A(PRINT)
=A(CUTAREA)
=C'WRONG LENGTH'
=C'DISK ERROR'
=C'PROGRAM BY MISS FATANA BENJAFELKHE'

```

LJC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

LJC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
				1	PRINT ON, NOGEN
020000				2	START 0
000000	0590			3	BALR 9,0
				4	USING HERE, 9, 10, 11
		00002		5	LM 10, 11, BASE
		00000		6	OPENF
000002	98A8 900A		00014	7	BASE
000006	47F0 9012			8	OPENF
00000A	0090			17	DC A(HERE+4096), HERE+8(192)
00000E	0000100200002002			18	CPEN CARD, PRINT
				19	CNTRL PRINT, SK, 1
000034	D283 A0C8 A0CA 010CD 010CC			23	MVC OUTAREA, SPACES
00003A	D20E A0F3 B83A 010F5 0283C			24	MVC OUTAREA+4(015) 1=C, RETRIEVE-ISNAME,
				25	PUT PRINT, OUTAREA
				31	OPEN ISNAME, ISCS, ISCPD
				41	CNTRL PRINT, SP, 1
				47	MVC OUTAREA, SPACES
000074	D283 A0C8 A0CA 010CD 010CC			48	MVC OUTAREA+4(017), C, SUCCESS-OPEN-1 ISM,
00007A	D210 A0F3 B849 010F5 0284B			49	PUT PRINT, OUTAREA
				55	HAL 7, CREATE
00003D	4570 916A		0016C	56	LOOP
				62	GET CARD, TEM
000044	D206 A14F A9AD 01151 019AF			63	MVC KEYRTV, TEM
				68	READ ISNAME, KEY
				73	WAITF ISNAME
				79	CNTRL PRINT, SP, 1
000000	D283 A0C8 A0CA 010CD 010CC			80	MVC OUTAREA, SPACES
000006	D223 A8D6 A162 018D8 01164			81	MVC DATA(36), NAME
00000C	D206 A8C1 A9AD 018C3 019AF			82	MVC I(DD1), TEM
00000E	D206 A800 A989 01A02 01980			83	MVC CODE, TEM+12
000008	4570 91F8		001FA	84	DAL 7, SEARCH
00000C	D208 9C6D A9AD 00C6F 019AF			85	MVC GROKEY, TEM
				90	READ ISGRD, KEY
				95	WAITF ISGRD
00010A	D205 9866 A06E 00868 01070			96	MVC CRKEY, C10
				101	PEAD ISCRS, KEY
				106	WAITF ISCRS
000128	D283 A906 A0CA 019C8 010CC			107	MVC OUTAPE1, SPACE 5
00012E	D205 A91D 9866 0191F 00868			108	MVC CRNO, CR, SKY
000134	D211 A92A 9C5B 0192C 00C5D			109	MVC CRNA, DATA
00013A	D201 A948 A074 0194D 01974			110	MVC PRCR, CCP
000140	D200 A956 A076 01958 01078			111	MVC PPAU, CAD
000146	D200 A95E A079 01960 01078			112	MVC PRGR, CGD
00014C	D200 A969 A068 01968 0106D			113	MVC PRSEM, CSM
000152	D201 A974 A069 01976 01068			114	MVC PPR3, CVF
				120	PUT PRINI, OUTABEL
000168	47F0 9092		00094	121	LOOP
00016C	4130 A022		01D24	122	LA 3, TABDEP
000170	4840 B836		02838	123	LH 4, H'0
000184	0503 A9AD B80E 019AF 0281C			129	READDEP GET CARD, TEM
00018A	478D 91A2		001A4	130	CLC TEM(4), C,
00018E	D21E 3002 A9AD 00002 019AF			131	RE MAJ

SOURCE STATEMENT

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000194	4043 0000		00000	132	4,0(3)
000198	4A40 8838		0283A	133	AH 4,=H*150*
00019C	5A30 8812		02814	134	A 3,=F*33*
0001A0	47F0 9172	00174		135	R READDED
0001A4	4130 AFL1		01F13	136	3,TARMAJ
0001B8	0202 A9FD A9AD	019FF	019AF	137	LA GET
0001BE	D210 3000 A9AD	00000	019AF	144	CARD,TEM
0001C4	1843			145	TEST,TEM
0001C6	5A30 8816		02818	146	MVC 0(30,3),TEM
0001DA	0503 A9AD	019AF	0281C	147	LR 4+3
0001E0	0787			153	A 3,=F*150*
0001E2	D502 A9AD	A9FD	019AF	154	GET CARD,TEM
0001E8	4770 9196	001R9		155	CLC TEM(4),=C*
0001EC	5A40 881A		0281C	156	RES 7
0001F0	0210 4000 A9AD	00000	019AF	157	TEM(3),TEST
0001F6	47F0 91C8	001CA		158	'MOVE
0001FA	4130 AD22		01024	159	A 4,=F*30*
0001FE	D502 AAD0	3002	01A02	160	MVC 0(30,4),TEM
000204	4780 920E	00210		161	READMAJ
000208	5A30 8812	001FE		162	3,TARMAJ
00020C	47F0 91FC	001FE		163	3,TARMAJ
000210	0218 AB41	3005	01R43	164	DEPCODE,2(3)
000216	4140 AFL1		01F13	166	MAJOR
00021A	4A43 0000		00000	167	A 3,=F*33*
00021E	0503 AA03	4000	01A05	168	CMAPREL
000224	4780 922E	00230		169	BE
000228	5A40 881A		0281C	170	A 4,=F*30*
00022C	47F0 921C		0021E	171	B COMPARE2
000230	0219 A868	4004	01R6D	172	MVC MAJOR,4(4)
000308	47F0 90EA	000EC		173	CTRL PRINT,SK,1
00031A	D283 A0C8	ADCA	010CD	179	PUT PRINT,OUT1
000320	D20E A0F3	B85A	010F5	185	CTRL PRINT,SP,1
				191	PUT PRINT,OUT2
				197	CTRL PRINT,SP,2
				203	PUT PRINT,OUT3
				209	CTRL PRINT,SP,2
				215	PUT PRINT,OUT4
				221	CTRL PRINT,SP,2
				227	PUT PRINT,OUT6
				233	CTRL PRINT,SP,2
				239	PUT PRINT,OUT5
				245	CTRL PRINT,SP,2
				251	PUT PRINT,OUT6
				257	B OUTB
				258	CTRL PRINT,SP,1
				264	FOFRTN
				265	MVC OUTAREA,SPACES
				266	MVC OUTAREA+6(15),=C*END-OF-RETRIEVE*
				272	PUT PRINT,OUTAREA
					CLOSE ISNAME,ISCF5,ISCF0,PRINT,CARD



LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT	VERIFY=YES, WORKS=YES	
000868				838	CRSKEY	DS	CL6
00086E				839	CRFTVE	DS	CL1000
000C56				840	CF SFT	DS	OCL75
000C56				841		DS	CL1
000C57				842	CKEY	DS	CL6
000C5D				843	DATAI	DS	CL18
000C6F				844	GRDKEY	DS	CL12
000C7B				845	GRDTVE	DS	CL1000
001063				846	GRDPT	DS	OCL25
001063				847		DS	CL1
001063				848	KEY	DS	OCL12
001064				849	GRLD	DS	CL7
00106B				850	CYR	DS	CL2
00106D				851	CSM	DS	CL1
00106E				852	CSQ	DS	CL2
001070				853	CIO	DS	CL6
001076				854	CCR	DS	CL2
001078				855	CAD	DS	CL1
001079				856	CSC	DS	CL2
00107B				857	CGD	DS	CL1
00107C				858	INAREA	DS	OCL80
00107C				859	IDC	DS	CL7
001083				860		DS	CL73
0010CC AC				861	SPACFS	DC	CL1
0010CD				862	PUTAPEA	DS	OCL132
0010CD				863		DS	CL40
0010F5				864	HEADER	DS	CL92
001151				865	KEYR TVE	DS	CL7
001158				866	WRKRT	DS	OCL48
001158				867		DS	CL12
001164				868	NAME	DS	CL36
001188				869	WRP TVE	DS	CL1920
001908				870	OUT ARE1	DS	OCL132
00191F				871		DS	CL23
001925				872	CRNO	DS	CL6
001925				873		DS	CL7
00193E				874	CPNA	DS	CL18
001942				875		DS	CL4
001944				876	PRSEC	DS	CL2
00194D				877		DS	CL9
00194F				878	PRCR	DS	CL2
001958				879		DS	CL9
001959				880	PRAU	DS	CL1
001960				881		DS	CL7
001961				882	PRGR	DS	CL1
00196B				883		DS	CL10
00196C				884	PP SEM	DS	CL1
001976				885		DS	CL10
001976				886	PPYR	DS	CL2

LOC	OBJECT CODE	ADDP1	ADDP2	STMT	SOURCE STATEMENT
001978				887	DS CL30
001979				888	DS CL25
0019AF				889	DS CL80
0019FF				890	DS CL3
001A02				891	DS OCL7
001A02				892	DS DEPCODE
001A05				893	DS CL3
001A09	40			894	DS CL4
001A0A				895	DC C*
001A0A	40404040404040			896	DC OCL132
001A44	E2E3E4C5D5E340			897	DC 58C*
001A53	40404040404040			898	DC C*STUDENT REPORT*
001A8E				899	DC 59C*
001A8E	40404040404040			900	DC OCL132
001AC8	5C5C5C5C5C5C5C			901	DC 58C*
001AD7	40404040404040			902	DC 15C**
001B12				903	DC 59C*
001B12	40404040404040			904	DC OCL132
001B33	C4C5D7C1D9E3D4C5			905	DC 33C*
001B43				906	DC C*DEPARTMENT OF
001B5F	40404040404040			907	DC CL28
001B6D				908	DC C* MAJOR
001B87	40404040404040			909	DC CL26
001B96				910	DC 15C*
001B96	40404040404040			911	DC OCL132
001B87	C9C440D5E404C2C5			912	DC 33C*
001B83				913	DC C*ID NUMBER
001B8A	40404040404040			914	DC C* NAME
001B88	40404040404040			915	DC CL49*
001C09	40404040404040			916	DC 17C*
001C1A				917	DC OCL132
001C1A	40404040404040			918	DC 22C*
001C33	C3D6E4D9E2C56D05			919	DC C*COURSE-NO
001C41	C3D6E4D9E2C56D05			920	DC C*COURSE-NAME
001C51	E2C5E3C9D6D540			921	DC C*SECTION
001C5D	C3D9C5C4C9E34C40			922	DC C*CREDIT
001C68	C1E4C4C9E3404040			923	DC C*AUDIT GRADE*
001C75	40404040404040			924	DC C* SEMESTER*
001C82	40404040404040			925	DC C* YEAR*
001C88	40404040404040			926	DC 19C*
001C9E				927	DC OCL132
001C9E	40404040404040			928	DC 22C*
001C9E	6C6C6C6C6C6C6C6C			929	DC 91C*
001D0E	40404040404040			930	DC 19C*
001D24				931	DC OF
001D24				932	DC 15CL33
001F13				933	DC 75CL30
0027E0	5B58C3D6D7C5D940			934	DC BEGIN
0027E8	5B58C2E3D3D6E2C5			935	DC C*\$\$ROOPEN*
0027F0	000003E0			936	DC C*\$\$BCLOSE*
				937	DC =A(PRINT)

00000

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
0027FA	C00010CD			938	=A(OUTAREA)
0027FB	C0000358			939	=A(GASDI)
0027FC	C00019AF			940	=A(ITEM)
002800	C00004C0			941	=A(ISNAME)
002804	C0000730			942	=A(ISGRD)
002808	C00005FR			943	=A(ISGRS)
00280C	C000190F			944	=A(OUTAPFI)
002810	C0404040			945	=C
002814	C0000021			946	=F'33'
002818	C0000096			947	=F'150'
00281C	C000001E			948	=F'30'
002820	C0001A0A			949	=A(OUT1)
002824	C0001A8E			950	=A(OUT2)
002828	C0001B12			951	=A(OUT3)
00282C	C0001B96			952	=A(OUT4)
002830	C0001C9E			953	=A(OUT6)
002834	C0001C1A			954	=A(OUT5)
002838	C000			955	=H'0'
00283A	C0096			956	=H'150'
00283C	D9C5E3D9C9C5E5C5			957	=C'RETRIEVE-1\$NAME'
002848	F2E4C3C3C5E2E260			958	=C'SUCCESS-CPEN-1\$AM'
00285C	C505C460D6C660D9			959	=C'END-OF-RETRIEVE'

ภาคผนวก ข

```

// ASSGN SYS004,X100C,CARD
// ASSGN SYS006,X100E,PRINTS
// ASSGN SYS007,DISK,VOL=CUDAT6,SHR
// DLBL MMSG,'D,MSGRO,Q,FILE',,VCA',CAT=IJSYSUC
// EXTENT SYS007,CUDAT6
// DLBL IJSYSUC,'UCAT.NAMF.CRUCATI',,VSAM
// EXTENT SYS007,CUDAT6
// ASSGN SYS014,DISK,VCL=CUDAT6,SHR
// DLBL MAST,'D,MSCRS,Q,FILE',,VSAM,CAT=IJSYSUC
// EXTENT SYS014,CUDAT6
// DLBL IJSYSUC,'UCAT.NAMF.CRUCATI',,VSAM
// EXTENT SYS014,CUDAT6
// ASSGN SYS013,DISK,VOL=CUDAT6,SHR
// DLBL MASN,'D,MSNAME,Q,FILE',,VSAM,CAT=IJSYSUC
// EXTENT SYS013,CUDAT6
// DLBL IJSYSUC,'UCAT.NAMF.CRUCATI',,VSAM
// EXTENT SYS013,CUDAT6
// EXEC ,SIZE=100K

```

```

00001 ID DIVISION.
00002 PROGRAM-ID. CREATE.
00003 AUTHOR. HISS /ATAHA BENJAPOL KUL.
00004 DATE-COMPILED. 2/12/80
00005 REMARKS. * * * * *
00006 * * * * * THIS PROGRAM USE TO CREATE VSAM *
00007 * * * * *
00008 ENVIRONMENT DIVISION.
00009 CONFIGURATION SECTION.
00010 OBJECT-COMPUTER. IBM-370.
00011 INPUT-OUTPUT SECTION.
00012 FILE-CONTROL.
00013 SELECT TAPE-MS-I-FILE ASSIGN TO SYSCLN-UT-2400-S.
00014 SELECT D-MSGRD-D-FILE ASSIGN TO SYS007-DM-3340-MS50
00015 ORGANIZATION IS INDEXED
00016 ACCESS IS SEQUENTIAL
00017 RECORD KEY IS DISK-KEY-GRADF
00018 PASSWORD IS MCCCSPWH
00019 FILE STATUS IS MSGRD-STATUS.
00020
00021 DATA
00022 FILE
00023 FD D-MSGRD-D-FILE
00024 BLOCK CONTAINS 40 RECORDS
00025 RECORD CONTAINS 24 CHARACTERS
00026 LABEL RECORD IS STANDARD
00027 DATA RECORD IS D-MSGRD-D-REC.
00028
00029 01 D-MSGRD-D-REC.
00030 05 DISK-KEY-GRADF.
00031 07 DISK-ED PIC X(7).
00032 07 DISK-YEAR PIC XX.
00033 07 DISK-SEMESTER PIC X.
00034 07 DISK-SEQUENCE PIC XX.
00035 05 DISK-COURSE PIC X(12).
00036
00037 FD TAPE-MS-I-FILE
00038 BLOCK CONTAINS 40 RECORDS
00039 RECORD CONTAINS 24 CHARACTERS
00040 LABEL RECORD IS OMITTED
00041 DATA RECORD IS TAPE-MS-I-REC.
00042
00043 01 TAPE-MS-I-REC.
00044 05 TAPE-KEY-GRADF.
00045 07 TO-ALPHA PIC X.
00046 07 TO-NUMERIC PIC X(6).
00047 07 TAPE-YEAR PIC XX.
00048 07 TAPE-SEMESTER PIC X.
00049 07 TAPE-SEQUENCE PIC XX.
00050 05 TAPE-COURSE PIC X(12).
00051
00052 WORKING-STORAGE SECTION.
00053 77 INPUT-REC-CNT PIC 9(5) COMP VALUE ZERO.
00054 77 OUTPUT-REC-CNT PIC 9(5) COMP VALUE ZERO.
00055 77 INPUT-EFFECT-CNT PIC 9(5) COMP VALUE ZERO.
00056 77 INVALID-REC-CNT PIC 9(5) COMP VALUE ZERO.
00057 77 MSGRD-STATUS PIC 99.

```


2 CREATE 13.33.59 27/12/80

```

00054 77 MSGC SUPW
00055 01 SWITCH-AREA.
00056 05 MS-TAPE-1-SWITCH-A PIC X(8) VALUE 'MSCCSUPW'.
00057 03 MS-TAPE-1-SWITCH PIC X(8) VALUE 'X.'.
PROCEDURE
* MAIN LEVEL STARTING HERE
* ----- INITIALIZATION -----
* OPEN FILES
OPEN TAPE-MS-1-FILE
OUTPUT D-MSGRD-0-FILE.
IF MSGRD-STATUS NOT = 0
THEN DISPLAY 'FILE-STATUS = ' MSGRD-STATUS
GO TO CLOSE-FILE.
*
* MAIN PROCESS
PERFORM 010-READ-TAPE-MS-1 THRU 010-READ-TAPE-EXT
UNTIL MS-TAPE-1-SWITCH.
*
* CONCLUSION
DISPLAY ' TOTAL INPUT RECORD = ' INPUT-REC-CNT.
DISPLAY ' TOTAL OUTPUT RECORD = ' OUTPUT-REC-CNT.
DISPLAY ' TOTAL INPUT ERROR = ' INPUT-ERR-CNT.
DISPLAY ' TOTAL INVALID RECORD = ' INVALID-REC-CNT.
CLOSE-FILE.
*
* TAPE-MS-1-FILE
D-MSGRD-0-FILE.
IF MSGRD-STATUS NOT = 0
THEN DISPLAY 'FILE-STATUS = ' MSGRD-STATUS.
*
STOP
RUN.
010-READ-TAPE-MS-1.
READ TAPE-MS-1-FILE
AT END
MOVE 'Y' TO MS-TAPE-1-SWITCH-A
GO TO 010-READ-TAPE-EXT.
AND
1 TO INPUT-REC-CNT.
IF (ID-NUMERIC IS NUMERIC AND
ID-ALPHA IS ALPHABETIC)
PERFORM 020-WRITE-D-MS-0 THRU 020-WRITE-D-MS-EXT
ELSE
DISPLAY 'ERROR-INPUT' TAPE-MS-1-REC
ADD 1 TO INPUT-ERRS-CNT.
EXIT.
010-READ-TAPE-EXT.
020-WRITE-D-MS-0.
MOVE TAPE-KEY-GRADE TO DISK-KEY-GRADE.
WRITE TAPE-COURSE TO DISK-COURSE.
D-MSGRD-0-REC
INVALID KEY
DISPLAY 'INVALID-RECORD' TAPE-KEY-GRADE
DISPLAY 'FILE-STATUS = ' MSGRD-STATUS
ADD 1 TO INVALID-REC-CNT
GO TO CLOSE-FILE.
ADD 1 TO OUTPUT-REC-CNT.
EXIT.
020-WRITE-D-MS-EXT.
EXIT.

```

```

CBL STATE
00001 ID DIVISION.
00002 PROGRAM-ID. CREATE.
00003 AUTHOR. MISS RATANA BENJAPOLKUL.
00004 DATE-COMPILED. 27/12/80
00005 REMARKS.
00006 * THIS PROGRAM USE TO CREATE VSAM *
00007 * * * * *
00008 ENVIRONMENT DIVISION.
00009 CONFIGURATION SECTION.
00010 OBJECT-COMPUTER. IBM-270.
00011 INPUT-OUTPUT SECTION.
00012 FILE-CONTROL.
00013 SELECT TAPE-MS-I-FILE ASSIGN TO SYS010-IT-2400-.
00014 SELECT D-MSNAME-O-FILE ASSIGN TO SYS013-DA-3340-P-ASN
00015 ORGANIZATION IS INDEXED
00016 ACCESS IS SEQUENTIAL
00017 RECORD KEY IS DISK-STUDENT-ID
00018 PASSWORD IS MSCRSUPW
00019 FILE STATUS IS MSNAME-STATUS.
00020
00021
00022 DATA
00023 FILE
00024 FD D-MSNAME-O-FILE
00025 BLOCK CONTAINS 40 RECORDS
00026 RECORD CONTAINS 47 CHARACTERS
00027 LABEL RECORD IS STANDARD
00028 DATA RECORD IS D-MSNAME-O-REC.
00029 01 D-MSNAME-O-REC.
00030 05 DISK-STUDENT-ID PIC X(7).
00031 05 TAPE-MS-I-FILE PIC X(40).
00032
00033 FD TAPE-MS-I-FILE
00034 BLOCK CONTAINS 40 RECORDS
00035 RECORD CONTAINS 47 CHARACTERS
00036 RECORDING MADE IS F
00037 LABEL RECORD IS OMITTED
00038 DATA RECORD IS TAPE-MS-I-REC.
00039 01 TAPE-MS-I-REC.
00040 05 TAPE-STUDENT-ID. PIC A.
00041 07 ID-ALPHA PIC X(6).
00042 07 ID-NUMERIC PIC X(40).
00043
00044 WORKING-STORAGE SECTION.
00045 77 INPUT-REC-CNT PIC 9(5) COMP VALUE ZER0.
00046 77 OUTPUT-REC-CNT PIC 9(5) COMP VALUE ZER0.
00047 77 INPUT-FROR-CNT PIC 9(5) COMP VALUE ZER0.
00048 77 INVALID-REC-CNT PIC 9(5) COMP VALUE ZER0.
00049 77 MSNAME-STATUS PIC 9(9) VALUE MSCRSUPW.
00050 77 SWITCH-ACCA. PIC X(1) VALUE MSCRSUPW.
00051 05 WS-TAPE-I-SWITCH-A PIC X.
00052 05 WS-TAPE-I-SWITCH-B PIC X.
00053
00054 PROCEDURE DIVISION.
00055 * MAIN LEVEL STAFFING HERE
00056

```

```

00053 * ----- INITIALIZATION -----
00054 * OPEN FILES INPUT TAPE-MS-I-FILE
00055 * OPEN OUTPUT D-MSNAME-O-FILE.
00056 * IF MSNAME-STATUS NOT = 0
00057 * THEN DISPLAY 'FILE-STATUS = ' MSNAME-STATUS
00058 * GC TO CLOSE-FILE.
00059 * MAIN PROCESS -----
00060 * PERFORM OIO-READ-TAPE-MS-I THRU OIO-READ-TAPE-EXT
00061 * UNTIL WS-TAPE-I-SWITCH.
00062 * ----- CONCLUSION -----
00063 * DISPLAY 'TOTAL INPUT RECORD = ' INPUT-REC-CNT.
00064 * DISPLAY 'TOTAL OUTPUT RECORD = ' OUTPUT-REC-CNT.
00065 * DISPLAY 'TOTAL INPUT ERROR = ' INPUT-ERROR-CNT.
00066 * DISPLAY 'TOTAL INVALID RECORD = ' INVALID-REC-CNT.
00067 * CLOSE-FILE.
00068 * CLOSE
00069 * TAPE-MS-I-FILE
00070 * D-MSNAME-O-FILE.
00071 * IF MSNAME-STATUS NOT = 0
00072 * THEN DISPLAY 'FILE-STATUS = ' MSNAME-STATUS.
00073 * R/JN.
00074 * STOP
00075 * OIO-READ-TAPE-MS-I.
00076 * READ TAPE-MS-I-FILE
00077 * AT END
00078 * MOVE '1' TO WS-TAPE-I-SWITCH-A
00079 * GC TO OIO-READ-TAPE-EXT.
00080 * ADD 1 TO INPUT-REC-CNT.
00081 * IF ( IO-ALPHA IS ALPHARETIC AND
00082 * IO-NUMERIC IS NUMERIC )
00083 * PERFORM O20-WRITE-D-MS-O THRU O20-WRITE-D-MS-EXT
00084 * ELSE
00085 * DISPLAY 'ERROR-INPUT TAPE-MS-I-REC
00086 * ADD 1 TO INPUT-ERROR-CNT.
00087 * OIO-READ-TAPE-EXT. EXIT.
00088 * O20-WRITE-D-MS-O.
00089 * MOVE TAPE-STUDENT-ID TO DISK-STUDENT-ID.
00090 * MOVE TAPE-STUDENT TO DISK-STUDENT.
00091 * WRITE D-MSNAME-O-REC
00092 * INVALID ID KEY
00093 * DISPLAY 'INVALID-RECORD' TAPE-STUDENT-ID
00094 * DISPLAY 'FILE-STATUS = ' MSNAME-STATUS
00095 * ADD 1 TO INVALID-REC-CNT
00096 * GO TO O20-WRITE-D-MS-EXT.
00097 * OUTPUT-REC-CNT.
00098 * EXIT.

```

STATISTICS	SOURCE RECORDS =	97	DATA ITEMS =	10	PTOC DIV =
STATISTICS	PARTITION SIZE =	524168	LINE COUNT =	56	BUFFER SIZE =
OPTIONS IN EFFECT	PHAP BELONG ADF =	NONE	SPACING =	1	FLW =
OPTIONS IN EFFECT	NOLISTY	APCST	MSYSM	VCATALS	LIST
OPTIONS IN EFFECT	NOCLIST	FLAGH	ZMR	NO SUPMAP	MOXREF
					ERRS
					ERRS
					ERRS

HPF
NOE

```

0001 TO DIVISION.
0002 CREATE-1SAP-COURSE.
0003 AUTHOR. MISS PATAMA PENJAPOLKUL.
0004 DATE-COMPILED. 04/01/81
0005 * * * * *
0006 * THIS PROGRAM USE TO CREATE VSCAM *
0007 * * * * *
0008 ENVIRONMENT DIVISION.
0009 CONFIGURATION SECTION.
0010 OBJECT-COMPUTER. IBM-370.
0011 INPUT-OUTPUT SECTION.
0012 FILE-CONTROL.
0013 SELECT TAPE-MS-I-FILE ASSIGN TO SYS011-01-2400-S.
0014 SELECT D-MSCFS-C-FILE ASSIGN TO SYS014-01-3340-1-MS1
0015 ACCESS SEQUENTIAL
0016 RECORD KEY IS DISK-COURSE-ID.
0017 DIVISION.
0018 SECTION.
0019 FD D-MSCRS-O-FILE
0020 BLOCK CONTAINS 40 RECORDS
0021 RECORD CONTAINS 25 CHARACTERS
0022 LABEL IS STANDARD
0023 DATA RECORD IS
0024 D-MSCRS-C-REC.
0025 01 D-MSCRS-C-REC. PIC X(6).
0026 05 DISK-COURSE-ID PIC X(18).
0027 FD TAPE-MS-I-FILE
0028 BLOCK CONTAINS 40 RECORDS
0029 RECORD CONTAINS 24 CHARACTERS
0030 RECORDING MODE IS F
0031 LABEL RECORD IS OMITTED
0032 DATA RECORD IS TAPE-MS-I-REC.
0033 01 TAPE-MS-I-REC. PIC X(6).
0034 05 TAPE-COURSE-ID PIC X(18).
0035 05 TAPE-COURSE-NAME PIC X(18).
0036 WORKING-STORAGE SECTION.
0037 77 INPUT-REC-CNT PIC 9(5) COMP VALUE ZERO.
0038 77 OUTPUT-REC-CNT PIC 9(5) COMP VALUE ZERO.
0039 77 INPUT-ERRR-CNT PIC 9(5) COMP VALUE ZERO.
0040 77 INVALID-REC-CNT PIC 9(5) COMP VALUE ZERO.
0041 01 SWITCH-AREA.
0042 05 MS-TAPE-I-SWITCH-A PIC X.
0043 88 MS-TAPE-I-SWITCH VALUE 'Y'.
0044 PROCEDURE DIVISION.
0045 *OPEN FILES
0046 INPUT TAPE-MS-I-FILE
0047 OUTPUT D-MSCRS-O-FILE.
0048 MAIN PROCESS
0049 PERFORM O1O-READ-TAPE-MS-I THRU O1O-READ-TAPE-EXT
0050 UNTIL MS-TAPE-I-SWITCH.
0051 ----- CONCLUSION -----
0052 DISPLAY * TOTAL INPUT RECORD * * INPUT-REC-CNT.

```



```

00053 DISPLAY * TOTAL OUTPUT RECORD = * OUTPUT-REC-CNT.
00054 DISPLAY * TOTAL INPUT ERROR = * INPUT-ERR-CNT.
00055 DISPLAY * TOTAL INVALID RECORD = * INVALID-REC-CNT.
00056 CLOSE TAPE-MS-I-FILE
      D-MSCRS-D-FILE.
00057 STOP RUN.
00058 010-READ-TAPE-MS-I.
00059 READ TAPE-MS-I-FILE
      AT END
      MOVE 'Y' TO MS-TAPE-I-SWITCH-A
      GO TO 010-READ-TAPE-EXT.
00060 ADD 1 TO INPUT-REC-CNT.
      IF TAPE-COURSE-ID IS NUMERIC
      PERFORM 020-WRITE-D-MS-D THRU 020-WRITE-D-MS-EXT
      ELSE
      DISPLAY 'ERROR-INPUT' TAPE-MS-I-REC
      ADD 1 TO INPUT-ERR-CNT.
00061 010-READ-TAPE-EXT.
      EXIT.
00062 020-WRITE-D-MS-D.
      MOVE TAPE-COURSE-ID TO DISK-COURSE-ID.
      MOVE TAPE-COURSE-NAME TO DISK-COURSE-NAME.
      WRITE D-MSCRS-D-REC
      INVALID KEY
      DISPLAY 'INVALID-RECORD' TAPE-COURSE-ID
      ADD 1 TO INVALID-REC-CNT
      GO TO 020-WRITE-D-MS-EXT.
00063 ADD 1 TO OUTPUT-REC-CNT.
      EXIT.
00064 020-WRITE-D-MS-EXT.

```

```

*STATISTICS*
*STATISTICS*
*OPTIONS IN EFFECT*
*OPTIONS IN EFFECT*
*OPTIONS IN EFFECT*
*OPTIONS IN EFFECT*
*LISTER OPTIONS*

```

```

SOURCE RECORDS = 80
PARTITION SIZE = 524168
PHAP RELLOC ADP = NONE
MOLISTX APOST
MOCLIST FLAGM
STATE TRUNC
NOCCOUNT
NONE
DATA ITEMS = 15
LINE COUNT = 56
SPACING NOCATALR
MOSYM ZWB NOSUPWAB
SEQ NOSYNOM
NOVERBSHM
NOVERBFF
PREC DIV 52 =
BUFFER SIZE =
FLUR
LINK
EARS
MUTLID
NOVERBFF

```

```

CBL LIB
00001 ID DIVISION.
00002 PROGRAM-ID. RETRIEVE-VSAM.
00003 AUTHOR. MISS RATANA BENJAPOLK'UL.
00004 DATE-COMPILED. 24/12/80.
00005 REMARKS.
00006 * * * * *
00007 * THIS PROGRAM USE TO RETRIEVE VSAM * * * * *
00008 * * * * *
00009 ENVIRONMENT DIVISION.
00010 CONFIGURATION SECTION.
00011 OBJECT-COMPUTER. IBM-370.
00012 INPUT-OUTPUT SECTION.
00013 FILE-CONTROL.
00014 SELECT CRD-FILE ASSIGN TO SYS004-UF-2540F-C.
00015 SELECT PRT-FILE ASSIGN TO SYS006-UP-1403-C.
00016 SELECT D-NAME-FILE ASSIGN TO SYS013-DA-3340-WASH
00017 ACCESS IS INDEXED
00018 RECORD KEY IS MSNAME
00019 FILE STATUS IS DISK-STUDENT-ID
00020 D-GRADE-FILE ASSIGN TO SYS007-DA-3340-WASG
00021 ORGANIZATION IS INDEXED
00022 ACCESS IS SEQUENTIAL
00023 PASSWORD IS MSGRD
00024 RECORD KEY IS DISK-GRADE
00025 FILE STATUS IS MSGRD-STATUS.
00026 D-COURSE-FILE ASSIGN TO SYS014-DA-3340-WAST
00027 ORGANIZATION IS INDEXED
00028 ACCESS IS RANDOM
00029 PASSWORD IS MSCRSUPM
00030 RECORD KEY IS DISK-CS-ID
00031 FILE STATUS IS MSCRS-STATUS.
00032
00033 DATA DIVISION.
00034 FILE SECTION.
00035 FD CRD-FILE CONTAINS 90 CHARACTERS
00036 RECORD MODE IS F
00037 RECORDING LABEL IS OMITTED
00038 LABEL DATA FILED IS CRD-REC.
00039 * * * * *
00040 01 CRD-REC PIC X(80).
00041 FD PRT-FILE CONTAINS 132 CHARACTERS
00042 RECORD MODE IS F
00043 RECORDING LABEL IS OMITTED
00044 LABEL DATA FILED IS PRT-REC.
00045 * * * * *
00046 01 PRT-REC PIC X(132).
00047 FD D-NAME-FILE CONTAINS 40 RECORDS
00048 RECORD LABEL IS STANDARD
00049 LABEL DATA FILED IS D-NAME-REC.
00050 * * * * *
00051 01 D-NAME-REC.
00052

```

Code	Description	PIC	Value
00053	03 DISK-STUDENT-ID	PIC X(7)	
00054	03 STUDENT.		
00055	05 MAJOR.		
00056	07 DEPT	PIC XXX.	
00057	07 FILLER	PIC X.	
00058	03 DISK-NAME	PIC X(36).	
00059	03 D-GRADE-FILE		
00060	BLOCK	CONTAINS 40 RECORDS	
00061	RECORD	CONTAINS 24 CHARACTERS	
00062	LABEL	RECORD IS STANDARD	
00063	DATA	RECORD IS D-GRADE-REC.	
00064	01 D-GRADE-REC.		
00065	03 D-GRADE-FILE		
00066	05 DISK-ID	PIC X(7).	
00067	05 DISK-YEAR	PIC XX.	
00068	05 DISK-SEMESTER	PIC X.	
00069	05 DISK-SEQUENCE	PIC XX.	
00070	03 DISK-COURSE-ID	PIC X(6).	
00071	03 DISK-CREDIT	PIC XX.	
00072	03 DISK-AUDIT	PIC X.	
00073	03 DISK-SECTION	PIC XX.	
00074	03 DISK-GRADE-A	PIC X.	
00075	03 D-COURSE-FILE		
00076	BLOCK	CONTAINS 40 RECORDS	
00077	RECORD	CONTAINS 24 CHARACTERS	
00078	LABEL	RECORD IS STANDARD	
00079	DATA	RECORD IS D-COURSE-REC.	
00080	01 D-COURSE-REC.		
00081	03 DISK-CS-ID	PIC X(6).	
00082	03 DISK-CS-NAME	PIC X(18).	
00083	WORKING-STORAGE SECTION.		
00084	77 TRANS-CNT	PIC 9(5)	COMP VALUE ZERO.
00085	77 ERROR-CARD-CNT	PIC 9(5)	COMP VALUE ZERO.
00086	77 PRT-REC-CNT	PIC 9(5)	COMP VALUE ZERO.
00087	77 ERR-SEQ-CNT	PIC 9(5)	COMP VALUE ZERO.
00088	77 INVALID-REC-CNT	PIC 9(5)	COMP VALUE ZERO.
00089	77 MAJORN	PIC X(31)	VALUE SPACE.
00090	77 DEPT-N	PIC X(32)	VALUE SPACE.
00091	77 MSNAME-STATUS	PIC 99.	
00092	77 MSGRD-STATUS	PIC 99.	
00093	77 MSCRS-STATUS	PIC 99.	
00094	77 MSCRS SUPW	PIC X(8)	VALUE *MSCRSUPW*.
00095	77 MSGRO	PIC X(8)	VALUE *MSCRSUPW*.
00096	77 MSNAME	PIC X(8)	VALUE *MSCRSUPW*.
00097	01 DNAME	COPY CYREDEPT SUPPRESS.	
00098	01 DNAME	REDEFINES DNAME COPY CYRECADP SUPPRESS.	
00099	01 MNAME	COPY CYREMAJR SUPPRESS.	
00100	01 MNAME	REDEFINES MNAME COPY CYRECANJ SUPPRESS.	
00101	01 FILLER		
00102	01 FILLER	PIC X(158)	VALUE SPACE.
00103	01 FILLER	PIC X(15)	VALUE *STUDENT
00104	01 FILLER	PIC X(59)	VALUE SPACE.
00105	01 FILLER		

RETRIEVE 14.11.57 26/12/80

00385	03	FILLER	VALUE	PIC	X(133)	SPACE
00386	03	FILLER	VALUE	PIC	X(113)	DEPARTMENT OF.
00387	03	FILLER	VALUE	PIC	XXX	SPACE.
00388	03	HEAD2-1	VALUE	PIC	X(132)	SPACE.
00389	03	FILLER	VALUE	PIC	X(15)	MAJRE.
00390	03	FILLER	VALUE	PIC	X(15)	SPACE.
00391	03	HEAD2-2	VALUE	PIC	X(15)	SPACE.
00392	03	FILLER	VALUE	PIC	X(126)	SPACE.
00393	01	HEAD3.	VALUE	PIC	X(33)	SPACE.
00394	03	FILLER	VALUE	PIC	X(9)	TD NUMBER.
00395	03	FILLER	VALUE	PIC	XXX	SPACE.
00396	03	FILLER	VALUE	PIC	X(12)	SPACE.
00397	03	HEAD3-1	VALUE	PIC	X(14)	NAME.
00398	03	FILLER	VALUE	PIC	X(5)	SPACE.
00399	03	FILLER	VALUE	PIC	X(36)	SPACE.
00400	03	HEAD3-2	VALUE	PIC	X(30)	SPACE.
00401	03	FILLER	VALUE	PIC	X(30)	SPACE.
00402	03	FILLER	VALUE	PIC	X(22)	SPACE.
00403	03	FILLER	VALUE	PIC	X(9)	COURSE-NO.
00404	03	FILLER	VALUE	PIC	X(8)	SPACE.
00405	03	FILLER	VALUE	PIC	X(11)	SPACE.
00406	03	FILLER	VALUE	PIC	X(5)	SPACE.
00407	03	FILLER	VALUE	PIC	X(7)	SECTION.
00408	03	FILLER	VALUE	PIC	X(5)	SPACE.
00409	03	FILLER	VALUE	PIC	X(6)	CREDIT.
00410	03	FILLER	VALUE	PIC	X(5)	SPACE.
00411	03	FILLER	VALUE	PIC	X(5)	AUDIT.
00412	03	FILLER	VALUE	PIC	XXX	SPACE.
00413	03	FILLER	VALUE	PIC	X(5)	GRADE.
00414	03	FILLER	VALUE	PIC	X(5)	SPACE.
00415	03	FILLER	VALUE	PIC	X(8)	SEMESTER.
00416	03	FILLER	VALUE	PIC	X(5)	SPACE.
00417	03	FILLER	VALUE	PIC	X(4)	YEAR.
00418	03	FILLER	VALUE	PIC	X(19)	SPACE.
00419	03	FILLER	VALUE	PIC	X(23)	SPACE.
00420	03	FILLER	VALUE	PIC	X(6)	SPACE.
00421	03	FILLER	VALUE	PIC	X(7)	SPACE.
00422	03	PRT-COURSE-NO	VALUE	PIC	X(18)	SPACE.
00423	03	FILLER	VALUE	PIC	X(4)	SPACE.
00424	03	PRT-COURSE-NAME	VALUE	PIC	XX	SPACE.
00425	03	FILLER	VALUE	PIC	X(9)	SPACE.
00426	03	PRT-SECTION	VALUE	PIC	XX	SPACE.
00427	03	FILLER	VALUE	PIC	XX	SPACE.
00428	03	PRT-CREDIT	VALUE	PIC	X(9)	SPACE.
00429	03	FILLER	VALUE	PIC	X	SPACE.
00430	03	PRT-AUDIT	VALUE	PIC	X(7)	SPACE.
00431	03	FILLER	VALUE	PIC	X	SPACE.
00432	03	PRT-GRADE	VALUE	PIC	X	SPACE.
00433	03	FILLER	VALUE	PIC	X(10)	SPACE.
00434	03	PRT-SEMESTER	VALUE	PIC	X	SPACE.
00435	03	FILLER	VALUE	PIC	X(10)	SPACE.
00436	03	PRT-YEAR	VALUE	PIC	XX	SPACE.
00437	03	FILLER	VALUE	PIC	XX	SPACE.

RETRIEVE 14.11.57 24/12/80

```

00438 01 FILLER PIC X(30) VALUE SPACE.
00439 01 HEAD6.
00440 03 FILLER PIC X(58) VALUE SPACE.
00441 03 FILLER PIC X(16) VALUE *****.
00442 03 FILLER PIC X(58) VALUE SPACE.
00443 01 HEAD7.
00444 03 FILLER PIC X(22) VALUE SPACE.
00445 03 S PIC X(91) .
00446 01 MS-CRD-REG.
00447 03 CRD-IO.
00448 05 ID-ALPHA PIC X.
00449 05 ID-NUMERIC PIC 9(6).
00450 03 CRD-YEAR PIC XX.
00451 03 CRD-SEMESTER PIC X.
00452 03 CRD-SEQUENCE PIC XX.
00453 01 SWITCH-AREA.
00454 05 MS-CRD-SWITCH-A PIC X.
00455 05 MS-CRD-SWITCH-B PIC X.
00456 05 TRANS-VALID-SWITCH-A PIC X.
00457 05 TRANS-VALID-SWITCH-B PIC X.
00458 05 EQ-SWITCH-NAME-A PIC X.
00459 05 EQ-SWITCH-NAME-B PIC X.
00460 05 EQ-SWITCH-NAME-C PIC X.
00461 PROCEDURE DIVISION.
00462 INPUT CRD-FILE.
00463 OPEN I-O D-NAME-FILE D-GRADE-FILE D-COURSE-FILE.
00464 OUTPUT PRT-FILE.
00465 IF MSNAME-STATUS NOT = '00'
00466 THEN DISPLAY 'FILE-STATUS-NAME = ' MSNAME-STATUS
00467 MSCRS-STATUS NOT = '00'
00468 THEN DISPLAY 'FILE-STATUS-CRS = ' MSCRS-STATUS
00469 GO TO CLOSE-FILE.
00470 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00471 UNTIL TRANS-VALID-SWITCH.
00472 PERFORM 020-READ-NAME.
00473 PERFORM 030-COMPARE-NAME-CFD THRU 030-COMPARE-NAME-CFD-EXT
00474 UNTIL MS-CFD-SWITCH.
00475 CONCLUSION
00476 *
00477 DISPLAY 'TOTAL READ TRANS = ' TRANS-CNT.
00478 DISPLAY 'TOTAL ERROR CARD = ' ERROR-CARD-CNT.
00479 DISPLAY 'TOTAL WRITE REPORT = ' PRT-REC-CNT.
00480 DISPLAY 'TOTAL ERROR SEQ CARD = ' ERR-SEQ-CNT.
00481 DISPLAY 'TOTAL INVALID CARD = ' INVALID-FC-CNT.
00482 CLOSE-FILE.
00483 CLOSE CRD-FILE
00484 D-NAME-FILE
00485 D-COURSE-FILE
00486 PRT-FILE.
00487 IF MSNAME-STATUS NOT = '00'
00488 THEN DISPLAY 'FILE-STATUS-NAME = ' MSNAME-STATUS
00489 STOP RUN.
00490

```

RETRIEVE 14.11.57 24/12/80

5

```

00491 IF MSCRS-STATUS NOT = '00'
00492 THEN DISPLAY 'FILE-STATUS-CRS = ', MSCRS-STATUS.
00493 STOP
00494 010-READ-TRANS.
00495 READ CRD-FILE INTO WS-CRD-REC
00496 AT END
00497 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00498 MOVE 'Y' TO WS-CRD-SWITCH-A
00499 GO TO 010-READ-TRANS-EXT.
00500 ADD 1 TO TRANS-CNT.
00501 IF (ID-ALPHA IS ALPHABETIC AND
00502 ID-NUMERIC IS NUMERIC)
00503 MOVE 'Y' TO TRANS-VALID-SWITCH-A
00504 ELSE
00505 MOVE 'N' TO TRANS-VALID-SWITCH-A
00506 DISPLAY 'ERROR-CARD', WS-CRD-REC
00507 ADD 1 TO ERROR-CARD-CNT.
00508 010-HEAD-TRANS-EXT.
00509 MOVE CRD-ID TO DISK-STUDENT-ID.
00510 READ D-NAME-FILE
00511 IF MSNAME-STATUS = '23'
00512 MOVE 'Y' TO EQ-SWITCH-NAME-A.
00513
00514 030-COMPARE-NAME-CRD.
00515 IF EQ-SWITCH-NAME
00516 MOVE 'N' TO EQ-SWITCH-NAME-A
00517 DISPLAY 'ERRPR-TRANS', WS-CRD-REC
00518 ADD 1 TO ERR-SEQ-CNT
00519 MOVE 'N' TO TRANS-VALID-SWITCH-A
00520 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00521 TRANS-VALID-SWITCH
00522 PERFORM 020-READ-NAME
00523 ELSE
00524 PERFORM 060-WRITE-HEAD
00525 PERFORM 040-START-GRADE THRU 040-START-GRADE-EXT
00526 PERFORM 050-READ-GRADE THRU 050-READ-GRADE-EXT
00527 UNTIL CRD-ID NOT = DISK-ID
00528 MOVE 'N' TO TRANS-VALID-SWITCH-A
00529 PERFORM 010-READ-TRANS THRU 010-READ-TRANS-EXT
00530 TRANS-VALID-SWITCH
00531 PERFORM 020-READ-NAME.
00532 030-COMPARE-NAME-CRD-EXT.
00533 040-START-GRADE.
00534 MOVE CRD-ID TO DISK-ID.
00535 START D-GRADE-FILE KEY IS EQUAL DISK-ID.
00536 READ D-GRADE-FILE AT END DISPLAY 'END OF GRADE-FILE'.
00537 040-START-GRADE-EXT.
00538 050-READ-GRADE.
00539 MOVE DISK-COURSE-ID TO DISK-CS-ID.
00540 READ D-COURSE-FILE
00541 IF MSCRS-STATUS = '23'
00542 DISPLAY 'ERROR-COURSE-FILE', WS-CRD-REC
00543 ADD 1 TO INVALID-REC-CNT

```

6 RETRIEVE

14.11.57

24/12/80

00544 MOVE GO TO 050-READ-GRADE-EXT.
00545 MOVE DISK-CS-ID TO PRT-COURSE-NO.
00546 MOVE DISK-CS-NAME TO PRT-COURSE-NAME.
00547 MOVE DISK-SECTION TO PRT-SECTION.
00548 MOVE DISK-CREDIT TO PRT-CREDIT.
00549 MOVE DISK-AUDIT TO PRT-AUDIT.
00550 MOVE DISK-GRADE TO PRT-GRADE.
00551 MOVE DISK-SEMESTER TO PRT-SEMESTER.
00552 MOVE DISK-YEAR TO PRT-YEAR.
00553 WRITE PRT-REC FROM HEAD5 AFTER POSITIONING 2 LINES.
00554 READ D-GRADE-FILE AT END DISPLAY 'END OF GRADE-FILE'.
00555 050-READ-GRADE-EXT. EXIT.
00556 060-WRITE-HEAD.
00557 ADD 1 TO PRT-REC-CNT.
00558 WRITE PRT-REC FROM HEAD1 AFTER POSITIONING 0 LINES.
00559 WRITE PRT-REC FROM HEAD6 AFTER POSITIONING 1 LINES.
00560 PERFORM FIND-DEP THRU FIND-D.
00561 MOVE DEPT-N TO HEAD2-1.
00562 MOVE MAJORN TO HEAD2-2.
00563 MOVE PRT-REC FROM HEAD2 AFTER POSITIONING 2 LINES.
00564 MOVE DISK-STUDENT-ID TO HEAD3-1.
00565 MOVE DISK-NAME TO HEAD3-2.
00566 WRITE PRT-REC FROM HEAD3 AFTER POSITIONING 2 LINES.
00567 MOVE ALL ' ' TO S.
00568 WRITE PRT-REC FROM HEAD4 AFTER POSITIONING 2 LINES.
00569 WRITE PRT-REC FROM HEAD4 AFTER POSITIONING 2 LINES.
00570 MOVE ALL ' ' TO S.
00571 WRITE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
00572 MOVE PRT-REC FROM HEAD7 AFTER POSITIONING 2 LINES.
00573 DEP-FF SECTION. COPY CYRFFDEP.
00574 DEP-FIND SECTION.
00575 FIND-DEP.
00576 SEARCH ALL DTNAME
00577 AT END GO TO NOT-F-DEP
00578 WHEN DEPT = DNO (D)
00579 GO TO FOUND-D.
00580 FOUND-D.
00581 MOVE D-NAME (D) TO DEPT-N.
00582 GO TO FIND-D.
00583 NOT-F-DEP.
00584 MOVE SPACE TO DEPT-N.
00585 FIND-D.
00586 EXIT.
00587 MAJ-FF SECTION. COPY CYRFFMAJ.
00588 MAJ-FIND SECTION.
00589 FIND-MAJ.
00590 SEARCH ALL MNAME
00591 AT END GO TO NOT-FOUND-M
00592 WHEN MAJOF = MNO (A)
00593 GO TO FOUND-M.
00594 FOUND-M.
00595 MOVE M-NAME (A) TO MAJORN.
00596 GO TO FIND-F.

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000000				1	PRINT CN,NGEN
000000	0580			2	START 0
000002	58C0 800E		00002	3	BEGIN BALR 11,0
000006	4100 86FE		00000	4	USING HERE,11,12
00000A	47FC 8C12		00014	5	L 12,BASE
00000E	0800			6	LA 13,SAVEX
000010	00001002			7	B OPENF
00002A	1855			8	DC ATHERE+40961
00002C	1866			9	OPEN TPNAME,VMNAME,PRINT
00002E	12FF			19	SR 5,5
000030	4770 80E4		000E6	20	SR 6,6
000044	022E 8AF6 8825 004F8 00B27			21	LTR 15,15
00004A	5A50 8C6E		00C70	22	BNZ OPENERR
00004E	58A0 8B56		00B58	23	GET TPNAME,INAREA
				29	MVC 5,F,1
				30	A 10,LENGTH
				31	L MODCO RPL=VMRPL,
				32	RECLLEN=f10)
0000C0	12FF			76	LTR 15,15
0000C2	4770 8124		00126	77	BNZ #BDERR
				78	PUT RPL=VMRFL
				85	LTR 15,15
				86	BNZ PUTERR
				87	A 6,F,F,1
				88	LCCP
				89	OPENERR MVC OUTAREA,OUTAREA-1
				90	MVC OUTAREA+4C(10),=C,UPJ1 ERRNR,
				91	PUT PRINT,CUTAREA
				97	EOFTPN B
				99	MVC OUTAREA,OUTAREA-1
				99	MVC OUTAREA+4C(11),=C,CLOSE ERRNR,
				100	PUT PRINT,OUTAREA
				106	ENDPGM B
				106	MVC OUTAREA,CUTAREA-1
				107	MVC OUTAREA+4C(10),=C,MCODE ERROR,
				108	PUT PRINT,OUTAREA
				109	EOFTPN B
				115	MVC OUTAREA,OUTAREA-1
				116	MVC OUTAREA+4C(10),=C,PUT ERRNR,
				117	PUT PRINT,CUTAREA
				118	EOFTPN B
				124	EOFTPN B
				133	LTR 15,15
				134	BNZ CLCSERR
				135	MVC OUTAREA,OUTAREA-1
				136	MVC OUTAREA+4C(10),=C,TOTAL READ TAPE =
				137	CVD 5,DEL
				138	MVC OUTAREA+6C(15),MASK
				139	ED OUTAREA+6C(10),DEL
				140	PUT PRINT,CUTAREA

LDC OBJECT CODE ADDR1 ADDR2 STAT SOURCE STATEMENT

```

000958
000718
000733
000743 00000018
00074C 40
00074J
00074J
000775
0007D4
00081C D4E2C3D9E2E4D7E6
000823 5858C2D6D7C5D540
000830 C9D2D8E5E3D4E243
000838 5858C2C3D3D6E2C5
000843 050G0280
000844 00000730
000848 00000324
00084C 000001A0
000850 0C000740
000854 D6D7C5E54DC5D9D9
00085E D4D6C4C54CC5D9D9
000868 C3D3D6E2C54CC5D3
000873 D7E4E34CC5D9D9D6
00087C C5D5C4CC5C666D5

```

```

ICARLAI=TAPPEIN,
RECFORM=FIXBLK,
RECSIZE=24,
WCPKA=YES,
DDNAME=MAST,
MACRF=(KEY,SEQ,OUT),PASSWD=PASS
ACB=VMCFS,
AM=VSAM,
AREA=WRKCPDS,
AREALEN=24,
OPTCD=(KEY,SEC,NUP,NVBE)
CL980
CL24
CL24
F'24'
CL1'
CL132
CL40
CL92
19F
CL8*MSRSLPK*
BEGIN
=C'$$$OPEN'
=CLB*IKCVTMS*
=C'$$$BCLOSE'
=A(I*PGRS)
=A(I*INAREA)
=A(I*VMRCL)
=A(I*PRINT)
=A(I*CUTAREA)
=C'UPEN ERROR'
=C'ADGE ERROR'
=C'CLOSE ERROR'
=C'DIT ERROR'
=C'END-OF-VSAM'

```

```

335 VMCRS ACB
365 VMRCL RPL
373 TAPPEIN OS
374 WRKCPDS DS
395 INAREA DS
376 LENGTH DC
397 CL1' DC
398 CUTAREA DS
399 HEADER DS
400
401 SAVEX DS
402 PASS DC
403
404
405
406
407
408
409
410
411
412
413
414
415
416

```


LOC	OBJECT CODE	ADDR1	ADDR2	STAT	SOURCE STATEMENT
000000				1	PRINT GN,NUGEN
000000	0580			2	START 0
000002	5800	800E		3	BEGIN
000006	4100	B702	00002	4	USING HERE,11,12
00000A	47F0	8012	00704	5	L2,BASE
00000E	0900			6	LA L2,SAVEX
000010	0001CC2		00014	7	B OPEN
00002A	12FF			8	BASE
00002C	4770	9C08	0000A	9	OPEN
000040	0217	B716	B72E-00718	20	OPENERR
000046	58A0	B745	00748	21	TPGRD,INAREA
000088	12FF			27	MVC
00008A	4770	8118	0011A	28	L10,LENGTH
000090	12FF			29	MDDCB
000092	4770	8138	0013A	73	LTP
000096	47F0	802E	00030	74	BNZ ACDEKR
00009A	0283	B748	B74A-00740	75	PUT KPL=VMRGL
00009E	02C9	B773	B952-00775	82	LTR L5,15
0000FA	47FC	9156	00158	83	BNZ PUTERR
0000FB	0283	B748	B74A-00740	84	LOOP
0000FD	0209	B773	B85C-00775	85	MVC
0000FE	47FC	9156	00158	86	MVC
0000FA	0283	B748	B74A-00740	87	PUT
000100	020A	B773	B866-00775	93	B
000106	47F0	8156	00158	94	CLCSEEP
00010A	0283	B748	B74A-00740	95	MVC
00010C	0209	B773	B85C-00775	96	MVC
00010E	47F0	8156	00158	102	MODERR
00011A	0283	B748	B74A-00740	103	MVC
00011C	0209	B773	B871-00775	104	MVC
00011E	47F0	8156	00158	105	B
00012A	0283	B748	B74A-00740	111	PUTERR
00012C	0208	B773	B871-00775	112	MVC
00012E	47F0	8156	00158	113	MVC
00013A	0283	B748	B74A-00740	114	PUT
00013C	0208	B773	B871-00775	120	EOFTPQ
00013E	47F0	8156	00158	129	CLOSE
000140	0283	B748	B74A-00740	130	BNZ
00014A	020A	B773	B87A-00775	131	CLOSERR
00014C	4770	8C08	000FA	132	MVC
00014E	0283	B748	B74A-00740	133	MVC
000150	020A	B773	B87A-00775	134	MVC
00015A	4770	8C08	000FA	139	PUT
00015C	0283	B748	B74A-00740	140	CLOSE
00015E	020A	B773	B87A-00775	147	EOJ
001	TPGRD			150	FILE
				151	PRINT
				152	BIKSIZE=92C,
				153	DEVAADR=SYS01C,
				154	ECFADDR=ECFTPG,
				155	IUARCAL=TAPEIN,

PRINT
BIKSIZE=92C,
DEVAADR=SYS01C,
ECFADDR=ECFTPG,
IUARCAL=TAPEIN,

LCC OBJECT CODE ADDR1 ADDR2 STAT SOURCE STATEMENT

```

000358          FILABL=ND,
000718          RECFER=FIXELK,
000730          REC SIZE=24,
          WORKA=YES,
          CCNAME=PASSG,
          MACRF=(KEY,SEC,CUT),PASSWD=PASS
          AM=VSAM,
          AREA=WRKGRD,
          AREALEN=24,
          OPTCD=(KEY,SEC,NUP,NVE)
          CL960
          CL24
          CL24
          DC F'24',
          DC GLL1',
          DC DCLL32
          DC CL40
          DC CL92
          DC LRF
          DC CL8*MSCR SUPM*
          REGIN
          ENO
          =C'5590PEN'
          =CL8*IKCVTMS*
          =C'55BCLCSE'
          =ALTPRFD
          =ALIMAF(A)
          =ALVPRGL)
          =ALPRINT)
          =ALJUTAREA)
          =C'CPEN EPRC'
          =C'MODE EPRGR'
          =C'CLOSE EPRGR'
          =C'PUT EPRGR'
          =C'END-CF-VSAM'
          ACB
          RPL
          DS
          DS
          DS
          DC
          DS
          DS
          DS
          DC
          ENO
          00000
          000828 5B58C2D6C7C5D540
          000830 C9B2D8E5E3D4E240
          000831 5B58C2C3D3D6E2C5
          000840 00000280
          000841 00000730
          000848 00000324
          00084C 000001A0
          000850 00000740
          000854 06D7C5D540C5D909
          00085E 0406646542E5D989
          000868 C3D3D6E2C54C5D9
          000873 07E4E340C5D90906
          00087C C5D5C46D06C66D05
          335 VMGRD
          365 VMRGL
          393 TAPEIN
          394 WRKGRD
          395 INAREA
          396 LENGTH
          397
          398 OUTAREA
          399 HEADER
          400
          401 SAVEX
          402 PASS
          403
          404
          405
          406
          407
          408
          409
          410
          411
          412
          413
          414
          415
          416
    
```

LDC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

LDC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000000				1	PRINT ON,NOGEN
000000	0590			2	START 0
000002	98AB 9C0E			3	BALR 9,0
000006	41D0 9796	00002		4	USING HERE,9,10,11
00000A	47F0 9016	00010		5	LM 10,11,0BASE
00000E	0000			6	LA 13,SAVEX
000010	00001C0200002002			7	B OPENF
000033	5850 978E			8	DC AHERE+4056,HERE+3192)
00003A	5860 978E			9	OPEN VNAME,CARD,PRINT,VMGRD,VMCRS
00003E	12FF			21	L 5,INCMT
000040	4770 9196			22	L 6,INCMT
000044	0283 96B7 5686 0C6B9 C06B8	00148		23	LTR 15,15
00004A	020C 96DF 47F2 306E1 C17F4			24	BNZ OPENERR
000060	4570 91A2			25	MVC OUTAREA,CUTAREA-1
000074	0206 97C9 9914 007C9 C0916			26	MVC OUTAREA+40(13),C=RETRIEVE VSAM*
00007A	5A5C 477E			27	PRINT,CUTAREA
000090	12FF			28	BAL 7,CREATE
000092	4770 4186			29	GET CARD,TEM
000096	5A60 477E			30	KEYLEN,TEM
00009A	0223 083C 9745 0099F C07A7			31	MVC 5,FILL
0000A0	0206 5667 9620 00969 C0922			32	RPL=VMRPL
0000A6	0206 5828 9914 3082A C0916			33	LTR 15,15
0000AC	4570 9230			34	BNZ PUTERR
0000C2	0206 9666 9914 02863 C0916			35	A 6,FILL
0000C8	0506 9866 984E 00364 C0853			36	MVC DATA(36),NAME
0000CE	478C 9C04			37	MVC CODE,TEM+12
0000D2	4720 9CAE			38	MVC 1001,TEM
0000D6	0205 9848 985A C085C			39	DAL 7,SEARCH
0000EE	0205 986A 9863 C0886 C086A			40	GET RPL=VMRGL
0000F4	0211 9891 9836 C0893 C083B			41	MVC 10C,TEM
0000FA	0201 9882 9860 C0864 C0862			42	CLC 10C,GRDID
000100	0200 9880 9862 C089F C0804			43	BE CUTC
000106	0200 9865 9865 C08C7 C0867			44	BH CUTR
00010C	0200 9860 9857 C08D2 C0859			45	MVC KEYCRS,CID
000112	0201 9808 9855 C08DD C0857			46	GET RPL=VMRCL
00013A	0506 9866 984E C0865 C0850			47	MVC CRNO,KEYCRS
000140	478C 9C04			48	MVC CRNA,CRS3P
000144	0200 9082			49	MVC PRGR,CCR
000148	0283 96B7 96B6 C06B9 C06B8			50	MVC PRGR,CDC
00014E	0205 56DF 47DA C06E1 C170C			51	MVC PRSEM,CSM
				52	MVC PRGR,CYR
				53	PUT PRINT,CUTAREA1
				54	RPL=VMRGL
				55	CLC 10C,GRDID
				56	BE CUTC
				57	BH LOOP
				58	MVC OUTAREA,CUTAREA-1
				59	MVC OUTAREA+40(13),C=OPEN ERROR*
				60	PRINT,CUTAREA

LCC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
000164	47F0 9342	00344		106	B EDRTRN
000165	0203 9687 9686 00689	00689	00689	107	MVC OUTAREA,OUTAREA-1
000166	D20A 96DF A7FF 006E1 01801	006E1	01801	108	MVC OUTAREA+AC(11),=C*CLOSE ERROR*
				109	PUT PRINT,OUTAREA
				115	B ENDPGM
000184	47FC 535E	00360		116	MVC OUTAREA,OUTAREA-1
000188	0203 9687 9686 00689	00689	00689	117	MVC OUTAREA+AC(10),=C*PUT ERROR*
00018E	0209 96DF A7E4 006E1 017E6	006E1	017E6	118	PUT PRINT,OUTAREA
				124	LA 3,TABDEP
0001A4	4130 9C8A	00C8C		125	LH 4,=H*J*
0001A8	4840 A7EE	017F0		126	GET CARD,TEM
				132	CLC TEM(4),=C*
0001C0	0203 9914 A792 00916	01794		133	BE MAJ
0001C2	4780 91CA	0010C		134	MVC 2(31,3),TEM
0001C5	021E 36C2 9914 00002	00916		135	SIH 4,0(3)
0001CC	4043 000C	00C00		136	AH 4,=H*15C*
0001D0	5A40 A7F0	017F2		137	A 3,=F*33*
0001D4	5A3C A796	01798		138	B READDEP
0001D8	47FC 91AA	001AC		139	LA 3,TABMAJ
0001DC	4130 9E79	00E78		140	GET CARD,TEM
0001F0	D202 9664 9914 00966	00916		146	MVC TEST,TEM
0001F6	021D 3000 9914 00000	00916		147	MVC 0(30,3),TEM
0001FC	1843			148	LR 4,3
0001FE	5A30 A19A	0179C		149	A 3,=F*15C*
				150	GET CARD,TEM
				156	CLC TEM(4),=C*
000212	0503 5414 A792 00916 01794	00916	01794	157	BER 7
000218	0787			158	CLC TEM(3),TEST
00021A	0502 9514 9664 00915 00966	00915	00966	159	BNE MOVE
000220	4770 91EE	001F0		160	A 4,=F*30*
000224	5A40 A79E	017A0		161	MVC 0(30,4),TEM
00022E	47FC 9260	00300	00916	162	B REAGMAJ
		00202		163	SEARCH LA 3,TABDEF
000232	4130 9C8A	00C8C		164	CLC DEPCODE,2(3)
000236	0502 9967 3002 00969	00C02		165	BE SMAJOR
00023C	4780 9246	00248		166	A 3,=F*33*
000240	5A30 A796	01798		167	B COMPARE1
000244	47F0 9234	00236		168	MVC DEPARTY,5(1)
000248	0210 9AA8 9095 009AA	00805		169	LA 4,TABMAJ
00024E	4140 9E79	00C78		170	AH 4,0(3)
000252	4A43 0600	00C00		171	CLC MAJCODE,0(14)
000256	0503 996A 4000 0096C	00C00		172	BE OUTPUT
00025C	4780 9260	00268		173	A 4,=F*30*
000260	5A40 A79E	017A0		174	A COMPARE2
000264	47F0 9254	00256		175	MVC MAJOR,4(14)
000268	D219 9AD2 4004 00A04	00004	00004	176	ENTRL PRINT,SK+1
				182	PUT PRINT,OUT1
				188	ENTRL PRINT,SP+1
				194	PUT PRINT,CUI2
				200	ENTRL PRINT,SP+2
				206	PUT PRINT,OUT3

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
212					CNTRL PRINT,SP,2
218					PUT PRINT,OUT6
224					CNTRL PRINT,SP,2
230					PUT PRINT,OUT6
236					CNTRL PRINT,SP,2
242					PUT PRINT,OUT5
248					CNTRL PRINT,SP,2
254					PUT PRINT,OUT6
260					OUTB
261					MVC CUTAREA,CUTAREA-1
262					MVC CUTAREA+4(C12),=C'END RETRIEVE'
263					PUT PRINT,OUTAREA
269					MVC CUTAREA,CUTAREA-1
270					MVC CUTAREA+4(C20),=C'TOTAL TRANSACTION'
271					CVD 5,DBL
272					MVC CUTAREA+6(116),MASK
273					ED CUTAREA+6(116),DBL
274					PUT PRINT,OUTAREA
280					MVC CUTAREA,CUTAREA-1
281					MVC CUTAREA+4(C24),=C'TOTAL RETRIEVE RECORD'
282					CVD 6,DBL
283					MVC CUTAREA+6(116),MASK
284					ED CUTAREA+6(116),DBL
285					PUT PRINT,OUTAREA
291					CLOSE CARD,VNAME,VVGRD,VMCRS
302					LTR 15,15
303					BNZ CLOSERR
304					CLOSE PRINT
312					ECJ
315					FILE CARD
469					FILE PRINT
620					ACB DONAME=NAME, MACRF=KEY,SEC,INI,PASSWD=PASS
650					RPL ACB=VMCRB, AM=VSAM, AREA=MRKGRD, AREALEN=24, OPTCD=(KEY,SEG,NUM),MVEJ
678					ACB DONAME=PASS, MACRF=KEY,DIR,INIT,PASSWD=PASS
708					RPL ACB=VMCRS, AM=VSAM, AREA=MRKCRS, AREALEN=24, OPTCD=(KEY,DIR,NUM,KEY,FKS,MVEJ)
736					ACB DONAME=PASS, MACRF=KEY,DIR,INIT,PASSWD=PASS
766					RPL ACB=VMNAME, AM=VSAM, AREA=MRKNAM,

LOC OBJECT CODE ADDR1 ADDR2 STMT SOURCE STATEMENT

AREALEN=47,
OPTCD=(KEY,DIR,NUP,KE,JFKS,MVEI),
MSG=KEYLEN,KEYLEN=7

000688	40			794	CUTAREA	DC
000689				795	FEADER	DS
000690				796	DBL	DS
000691				797	SAVEX	DS
000692				798	IMCNT	DC
000693				799	PASS	DC
000694				800	WRKNAM	DS
000695				801	ID	DS
000696				802	MJ	DS
000697				803	NAME	DS
000698				804	KEYLEN	DS
000699				805	INAREA	DS
000700				806	KEYCD	DS
000701				807	MASK	DC
000702				808	WRKCRS	DS
000703				809	IDCRS	DS
000704				810	CRSDA	DS
000705				811	KEYCRS	DS
000706				812	WRKGRD	DS
000707				813	GRDIO	DS
000708				814	CYR	DS
000709				815	CSM	DS
000710				816	CSQ	DS
000711				817	CR	DS
000712				818	GAD	DS
000713				819	CSC	DS
000714				820	C60	DS
000715				821	IOC	DS
000716				822	OUTARE1	DS
000717				823	CRND	DS
000718				824	CRNA	DS
000719				825	PRSEC	DS
000720				826	PRCR	DS
000721				827	PRAU	DS
000722				828	PRGR	DS
000723				829	PRSEM	DS
000724				830	PRSEM	DS
000725				831	PRSEM	DS
000726				832	PRSEM	DS
000727				833	PRSEM	DS
000728				834	PRSEM	DS
000729				835	PRSEM	DS
000730				836	PRSEM	DS
000731				837	PRSEM	DS
000732				838	PRSEM	DS
000733				839	PRSEM	DS
000734				840	PRSEM	DS
000735				841	PRSEM	DS

LOC	OBJECT CODE	ADDR 1	ADDR 2	STRT	SOURCE STATEMENT
000800				842	PRYR DS GL2
000801				843	DS GL30
000802				844	GNAM DS CL25
000803				845	TEM DS GL00
000804				846	TEST DS CL3
000805				847	CODE DS 0CL7
000806				848	DEPCODE DS CL3
000807				849	MAJGDE DS GL4
000808				850	DC C
000809				851	OUT1 DS 0GL132
000810				852	DC 58C
000811				853	DC G*STUDENT REPORT*
000812				854	DC 59C
000813				855	OUT2 DS 0GL132
000814				856	DC 58C
000815				857	DC 15C*
000816				858	DC 59C
000817				859	OUT3 DS 0CL132
000818				860	DC 33C
000819				861	DC C*DEPARTMENT OF
000820				862	DS CL28
000821				863	DC C* MAJCR
000822				864	MAJOR DS CL26
000823				865	DC 15C
000824				866	OUT4 DS 0CL132
000825				867	DC 33C
000826				868	DC C*IC NUMBER
000827				869	DS GL7
000828				870	DC C* NAME
000829				871	CATA DC GL40*
000830				872	DC 17C
000831				873	OUT5 DS 0GL132
000832				874	DC 22C
000833				875	DC C* COURSE-NO
000834				876	DC C* COURSE-NAME
000835				877	DC C* SECTION
000836				878	DC C* CREDIT
000837				879	DC C* AUDIT GRADE*
000838				880	DC C* SEMESTER*
000839				881	DC C* YEAR*
000840				882	DC 19C
000841				883	OUT6 DS 0CL132
000842				884	DC 22C
000843				885	DC 91C*
000844				886	DC 19C
000845				887	DC OF
000846				888	TABDEP DS 15CL33
000847				889	TABMAJ DS 75CL30
000848				890	END BEGIN
000849				891	DC*SDOPEN*
000850				892	DC*TOTAL RETRIEVE RECORD*

COC00

000851 5858C2D07C50580
000852 E306E3C1D34009C5

LOC	OBJECT CODE	ADDR1	ADDR2	STMT	SOURCE STATEMENT
001768	5858C2C303D6E2C5			893	=C'\$\$BCLOSE'
001770	00000479			894	=A(PRINT)
001774	00C00689			895	=A(OUTAPE#)
001778	000003E8			896	=A(CARD)
00177C	0C00916			897	=A(TEX)
001784	00000001			898	=F'1'
001784	0C000684			899	=A(VMRPL)
001784	0C000594			900	=A(VMRGL)
00178C	00C0060C			901	=A(VMRCL)
001793	0CC0086F			902	=A(GUIAFF1)
001794	40484C40			903	=C'
001798	00000021			904	=F'33'
00179C	00000056			905	=F'150'
0017A0	0000001E			906	=F'30'
0017A4	00000971			907	=A(GUI1)
0017A8	0009F5			908	=A(OUT2)
0017AC	00000A79			909	=A(OUT3)
0017B0	00C00AF0			910	=A(OUT4)
0017B4	0CC0C05			911	=A(OUT6)
0017B8	00000881			912	=A(OUT5)
0017BC	C5D5C440D9C5E3D9			913	=C'END RETRIEVE'
0017C8	E3D6E3C1D34CE3D9			914	=C'TOTAL TRANSACTION = '
0017D0	06D7C5D54CC5D9D9			915	=C'OPEN EJECT'
0017E6	07E4E34040C5D9D9			916	=C'PUT ERROR'
0017F0	0000			917	=H'0'
0017F2	0096			918	=H'150'
0017F4	09C5E3C9C9C5E5C5			919	=C'RETRIEVE VSAM'
001801	C3D3D6E2C540C5D9			920	=C'CLOSE FPROP'

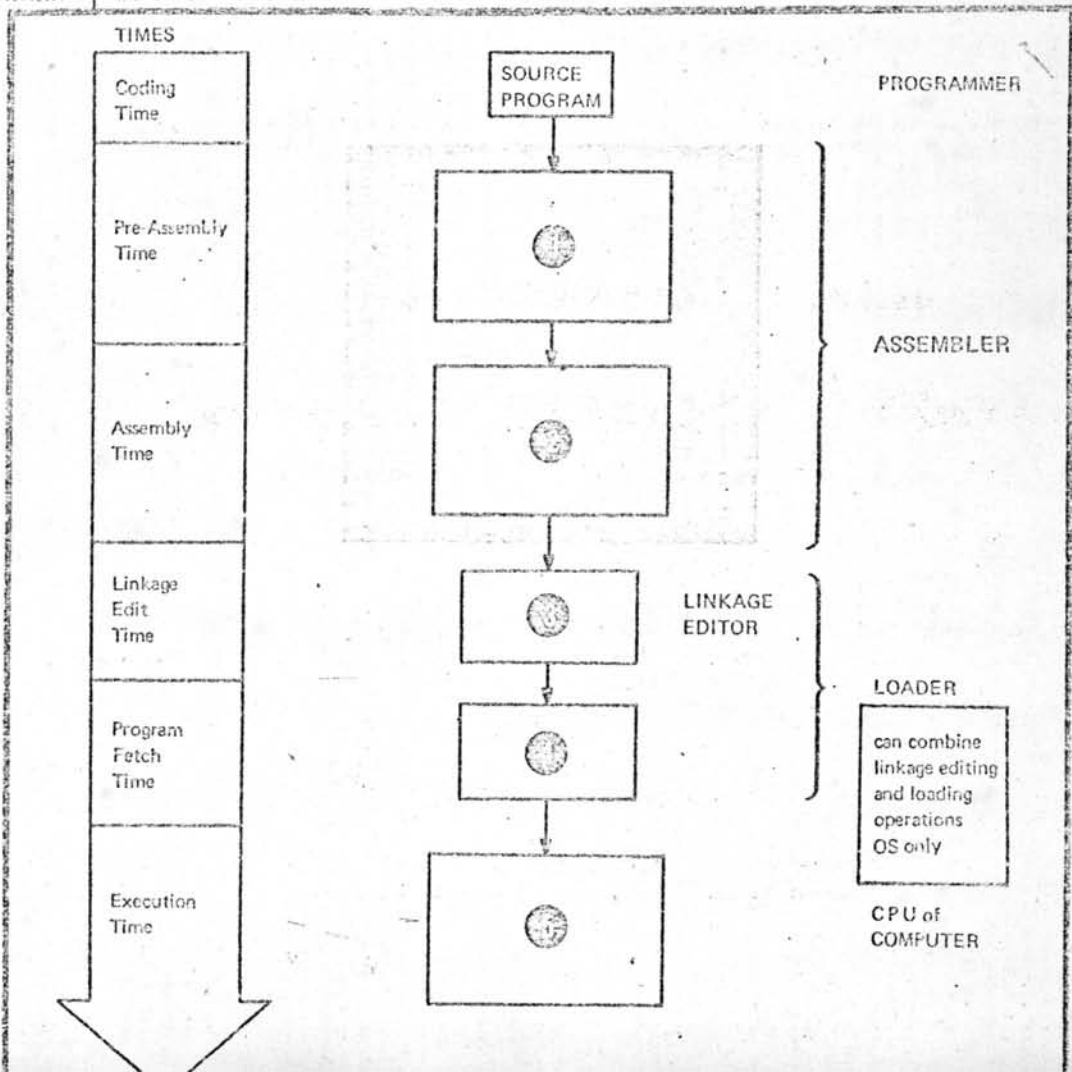
ภาคผนวก ช.

โดยปกติแล้วเครื่องคอมพิวเตอร์สามารถรู้และเข้าใจเฉพาะภาษาเครื่อง (Machine Language) เท่านั้น ซึ่งเป็นภาษาที่คนจะเข้าใจค่อนข้างยาก เพราะมีลักษณะเป็นเลขฐานสอง (Binary system) ทั้งหมด กล่าวคือจะเป็นตัวเลขเฉพาะ 0 และ 1 เท่านั้น ดังนั้นบริษัทผู้ผลิตทั้งหลาย จึงได้คิดประดิษฐ์ภาษาใหม่ขึ้น เพื่ออำนวยความสะดวกให้แก่ผู้ใช้เครื่องหรือ Programmer ให้สามารถเขียนโปรแกรมคำสั่งเครื่องได้สะดวก รวดเร็ว และมีประสิทธิภาพมากยิ่งขึ้น

ตัวแปลภาษา ASSEMBLER เป็นตัวแปลภาษาที่เรียกว่า Symbolic Language เฉพาะใช้สัญลักษณ์ (Symbolic) ช่วยในการเขียนโปรแกรมคำสั่งแทนการใช้ภาษาเครื่อง (Machine Language) ที่ยุ่งยาก การใช้ภาษา Assembler ผู้เขียนคำสั่งสามารถใช้สัญลักษณ์ใด ๆ อันใดจากการรวมตัวอักษร (Alphabetic) และตัวเลข (numeric) ซึ่งทำให้เขียนโปรแกรมคำสั่ง อ่านหรือแก้ไขได้ง่าย

อย่างไรก็ตามตัวแปลภาษา ASSEMBLER ต้องทำหน้าที่แปลคำสั่งเครื่องที่ผู้เขียนคำสั่งเขียนขึ้นเป็นภาษา Assembler ซึ่งมีลักษณะเป็นสัญลักษณ์ให้เป็นภาษาเครื่อง (Machine Language) ซึ่งมีลักษณะเป็นเลขฐานสองเสียก่อนแล้วจึงจะสามารถทำงานได้ โดยมีลักษณะการทำงานเป็นขั้นตอน สรุปได้ดังรูปข้างล่างนี้

รูปที่ 1 ขั้นตอนหลัก ๆ ของการทำงานควยตัวแปลภาษา ASSEMBLER



จากรูป สมมติว่าโปรแกรมคำสั่งที่เขียนไว้เขียนด้วยภาษา Assembler และเจาะลงในบัตรซึ่งขอเรียกว่า "Source Deck" หรือ "Source Module" ซึ่งบัตรเหล่านี้จะถือว่าเป็น input ของตัวแปลภาษา ASSEMBLER จากนั้นตัวแปลภาษา ASSEMBLER จะทำหน้าที่แปลโปรแกรมคำสั่งนั้นให้เป็นภาษาเครื่องซึ่งจะขอเรียกว่า "Object Code" หรือ "Object module" และถ้าสมมติว่าภาษาเครื่องที่ตัวแปลภาษา ASSEMBLER แปลออกมา นั้น ออกมาในรูปของบัตรอาจเรียกได้ว่า "Object Deck" แต่ก่อนที่ภาษาเครื่อง (Object deck หรือ Object module) ที่ตัวแปลภาษา ASSEMBLER ผลิตขึ้นนั้น จะสามารถทำงานได้ ยังต้องมีขั้นตอนการทำงานอีกสองขั้นตอน คือ

ขั้นตอนที่หนึ่ง ต้องนำ Object module ไปผ่านกระบวนการโดยชุดของโปรแกรมที่เรียกว่า LINKAGE EDITOR เพื่อผลิตชุดโปรแกรมใหม่อื่น เรียกว่า "Load module"

ขั้นตอนที่สอง ต้องนำ Load module เข้าไปในหน่วยความจำหลัก (Main storage) เสียก่อน เครื่องจึงจะสามารถทำงานได้

อนึ่ง ตลอดระยะเวลาทุกขั้นตอน เครื่องจะพิมพ์รายงานออกมาทางกระดาษต่อเนื่องทุกระยะ เพื่อผลในการแก้ไขหรือเพื่อเก็บไว้เป็นหลักฐาน

โครงสร้างของตัวแปลภาษา ASSEMBLER

ดังที่ได้อธิบายมาแล้วว่า ตัวแปลภาษา ASSEMBLER เป็นตัวแปลที่เป็นสัญลักษณ์ เป็นภาษาแรก ดังนั้นสัญลักษณ์ต่าง ๆ ที่เขียนขึ้นยังมีลักษณะที่ใกล้เคียงกับภาษาเครื่อง (Machine Language) มากที่สุด เมื่อเปรียบเทียบกับภาษาอื่น ๆ ด้วยเหตุนี้ การเขียนโปรแกรมคำสั่งด้วยภาษา ASSEMBLER จะมีประโยชน์อย่างยิ่งเมื่อลักษณะของงาน (Application) มีลักษณะดังนี้

- (1) ต้องการเขียนโปรแกรมคำสั่งไปประยุกต์ส่วนความจำหรือต้องการควบคุมระบบการทำงาน หรือข้อมูลใด ๆ ที่เล็กลงไปถึงระดับ Byte หรือ Bit
- (2) ต้องการสร้างชุดโปรแกรมคำสั่งย่อย (Subroutine หรือ Function) ใด ๆ ซึ่งไม่มีหรือไม่สามารถทำได้ในภาษาอื่น ๆ เช่น ในภาษา ALGOL, COBOL, FORTRAN หรือ PL/I เป็นต้น

โครงสร้างของโปรแกรมคำสั่งในภาษา Assembler นั้น ประกอบด้วยประโยคคำสั่ง (instruction) สามประเภท คือ

- (1) Machine instructions
- (2) Assembler instructions
- (3) Macro instructions

ซึ่งจะได้นำมาหารือถึงพื้นฐานของคำสั่งทั้งสามประเภทต่อไป

(1) Machine instruction (คำสั่งเครื่อง)

คำสั่งเครื่องที่จะกล่าวถึงนี้ ถือว่าเป็นคำสั่งที่ใช้แทนภาษาเครื่องได้เลย ซึ่งจัดเป็นคำสั่งที่ใช้กับเครื่องแต่ละเครื่อง เหตุที่เรียกว่า Machine instruction เพราะตัวแปลภาษา ASSEMBLER จะแปลคำสั่งเหล่านี้ให้เป็นภาษาเครื่อง (Machine Language) แล้วสามารถทำงานได้เลยทันที

(2) Assembler instruction (คำสั่งของตัวแปลแอสเซมเบอร์)

คำสั่งของตัวแปลแอสเซมเบอร์ (Assembler instruction) เป็นคำสั่งที่มีไว้เพื่อใช้สำหรับดำเนินงานบางอย่างในการเสริมสร้างและปรุงแต่ง source module ตัวอย่างเช่น การกำหนดค่าคงที่ (Constant) ของตัวแปร กำหนดจุดเริ่มต้นหรือสุดท้ายของโปรแกรมหรือแมตเทคาร์กั้นที่ในหน่วยความจำเพื่อใช้งานต่าง ๆ

(3) Macro instruction

คำสั่งของ Macro มีไว้เพื่อจัดแต่งหรือเพิ่มเติม assembler โปรแกรม ซึ่งกำหนดไว้เป็นโครงสร้างได้ในรูปที่เรียกว่า Macro definition และจาก Macro definition นี้ ตัวแปลภาษา ASSEMBLER สามารถนำมาสร้างหรือขยายออกให้เป็น machine instruction หรือ Assembler instruction เพิ่มเติม เพื่อให้สามารถทำงานได้เสมือนว่าเป็น machine instructions หรือ assembler instructions ที่เขียนขึ้นโดยผู้เขียนคำสั่งเอง โดยปกติแล้วทางบริษัท IBM ได้จัดเตรียม Macro definition ใดหลายอย่างเพื่ออำนวยความสะดวกแก่ผู้เขียนคำสั่ง โดยจะอยู่ในรูปของ Input/Output, Data Management และ Supervisor operations ซึ่งผู้ใช้สามารถเรียกใช้ได้อย่างไรก็ตาม ผู้เขียนคำสั่งสามารถสร้าง Macro definitions ขึ้นเองและเรียกมาใช้งานได้ก็ตามความต้องการ ซึ่งก่อให้เกิดประสิทธิภาพและความคล่องตัวอย่างยิ่งในการเขียนคำสั่งด้วยภาษา Assembler ซึ่งตัวแปลภาษาอื่นไม่มีคุณลักษณะเช่นนี้

ขั้นตอนต่าง ๆ ของการทำงานของตัวแปลภาษา ASSEMBLER

คำสั่งหรือโปรแกรมที่เขียนด้วยภาษาแอสเซมเบอร์ โดยปกติแล้วจะเรียกสั้น ๆ ว่า "Assembler" ซึ่งนี้หมายถึงรวม Machine, Assembler และ Macro instructions เข้าไปด้วยแล้ว ซึ่งโปรแกรมดังกล่าวจะถูกตัวแปลภาษา ASSEMBLER แปลออกมาเป็น Machine Language

ในการแปลคำสั่งที่เขียนด้วยภาษา ASSEMBLER ซึ่งประกอบด้วยคำสั่ง (instruction) 3 ประเภท ดังได้กล่าวไปแล้วนั้น การแปลคำสั่งแต่ละประเภทจะทำได้ในโอกาสกัน (ผู้เขียนโปรแกรมด้วยภาษา Assembler จำเป็นอย่างยิ่งต้องเข้าใจก่อนว่า instructions ใดถูกแปลในโอกาสใด จึงจะสามารถเขียนโปรแกรมคำสั่งด้วยภาษาที่ใดถูกต้อง) จากรูปที่ 1 จะเห็นว่าในโอกาสใดจะทำงานอะไร

จะเห็นได้ว่าคำสั่งต่าง ๆ จะถูกแปลในสองโอกาสด้วยกัน คือ

(1) Pre-assembly time และ

(2) Assembly time

ซึ่งจะได้อธิบายโดยละเอียดต่อไป และหลังจากผ่านช่วง Assembly time แล้ว ตัวแปล ASSEMBLER จะส่งข้อมูลเพื่อเป็นประโยชน์ต่อการปฏิบัติงานในขั้นตอนที่ 3 ต่อไป คือ Linkage-edit time (3) เพื่อให้ Linkage Editor สามารถนำข้อมูลเหล่านั้นเพื่อไปรวมกับ Object module ให้เป็น Load module จากนั้นเป็นขั้นตอนที่ 4 ซึ่งเรียกว่า Program Fetch time (4) คือโปรแกรมคำสั่งเครื่องชุดหนึ่ง ซึ่งเรียกว่า "Loader" จะนำ Load module นั้นบันทึกลงไปใน Virtual Storage และช่วงสุดท้ายเรียกว่า Execute time (5) เป็นช่วงที่เครื่องคอมพิวเตอร์จะทำงานตามคำสั่ง

ต่อไปนี้จะได้อธิบายถึงว่าตัวแปลภาษา ASSEMBLER ดำเนินการกับ Machine instruction, Assembler instruction และ Macro instruction อย่างไร

(I) การดำเนินการกับ Machine instruction

ตัวแปลภาษา ASSEMBLER จะดำเนินการกับ Machine instruction ตอน Assembly time (ดูรูปที่ 2 ตอนที่ 1)

(II) การดำเนินการกับ Assembler instructions

โดยปกติแล้ว Assembler instructions แบ่งออกเป็น 2 แบบใหญ่ ๆ

- Assembler instructions โดยทั่วไป
- Conditional assembly instruction และ Macro processing instruction (MACRO, MEND, MEXIT และ MNOTE)

(ก) คำแปลภาษาจะดำเนินการกับ Assembler instruction ทั่วไปใน assembly time

ซึ่งเรียกว่า Full processed ①
แต่มีข้อยกเว้นว่า

(1) การทำ absolute และ relocatable expressions จะทำในช่วง assembly time

ซึ่งอาจเรียกว่า "Assembly time expression" ②

(2) instruction บางชนิดอาจจะให้ Output ออกมาเพื่อผลประโยชน์ในการดำเนินงาน
หลัง assembly time แล้ว ③

(ข) คำแปลภาษาจะดำเนินการกับ Conditional assembly instruction และ
Macro processing instruction ในช่วง pre-assembly time ④
แต่มีข้อยกเว้นว่า

(1) คำแปลภาษาจะดำเนินการกับ Condition assembly ที่เป็น expressions
(arithmetic, logical และ character) ในโอกาส pre-assembly time ⑤ เช่นกัน

(2) คำแปลภาษาจะดำเนินการกับ Machine และ Assembler instruction ที่
generated ใ้ในขั้น pre-assembly time ในช่วงของ Assembly time

(III) การดำเนินการกับ Macro Instruction

คำแปลภาษาจะดำเนินการกับ Macro instruction ในช่วงเวลา pre-assembly
time ⑥

แต่มีข้อยกเว้นว่า คำแปลภาษาจะดำเนินการกับ Machine และ Assembler instruction ที่
generated โดย Macro definition ในช่วง Assembly time ⑦

ตอนที่ 1

ตอนที่ 2

ตอนที่ 3

ตอนที่ 4

TIMES

Machine Instructions

Ordinary Assembler Instructions and assembly time expressions

DC DS CCW

ENTRY ENTRN WXTRN Address constants

PUNCH REPRO

Control Assembler (macro processing) instructions and additional assembly expressions

MNOTE

Macro Instructions

Macro Definitions

Listing
Object code
Assembly

Coded
Assembled into object code
Executed

Fully Processed

Data or
Areas used in execution of machine inst.

Provide linkage information
Provide areas to hold addresses used in execution of machine inst.

Punched cards
Can provide linkage commands

Generated statements

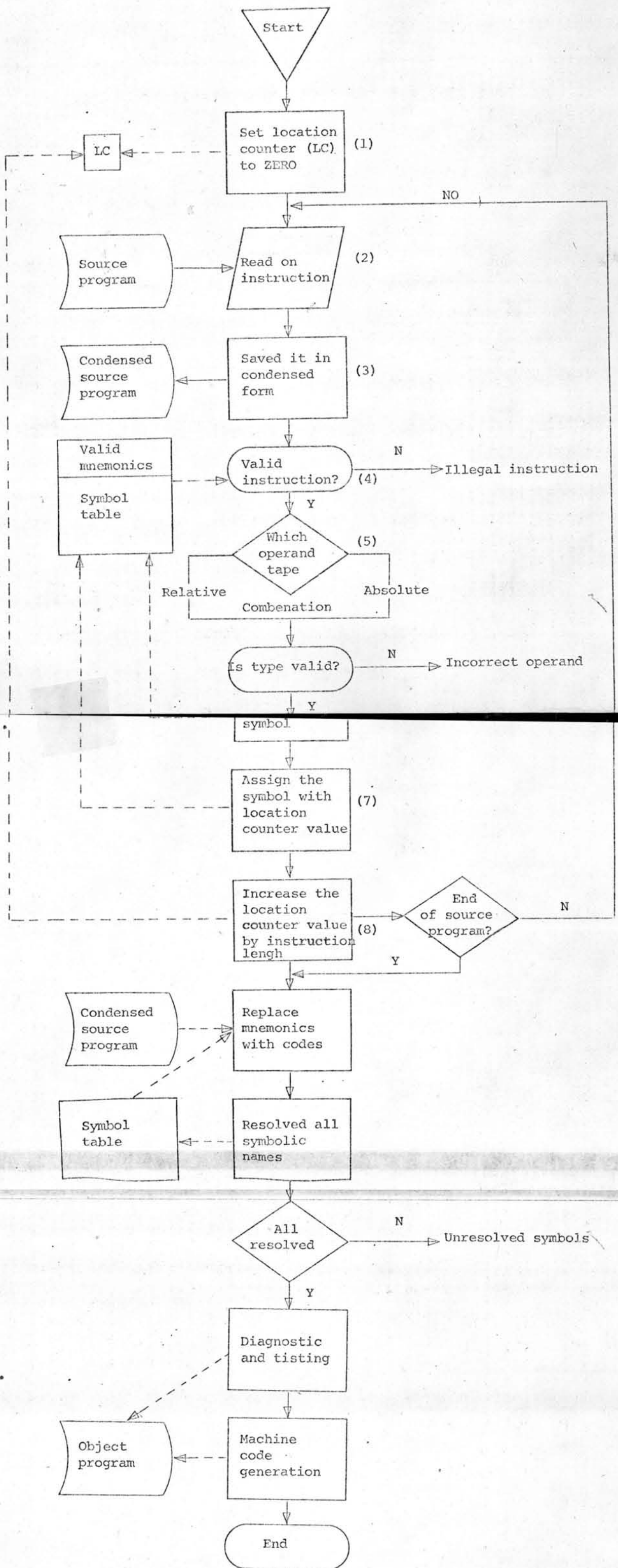
Printed message

Model Statements

Generated Statements

Linkage Edit
Program Fetch
Execution

รูปที่ 2 แสดงการทำงานของ instructions ในช่วงต่าง ๆ



ประวัติผู้เขียน

นางสาว รัตนา เบญจพหลุณ ได้รับปริญญาครุศาสตรบัณฑิต จากคณะครุศาสตร์
จุฬาลงกรณ์มหาวิทยาลัย เมื่อปีการศึกษา ๒๕๒๑

