To the Graduate Council:

I am submitting herewith a thesis written by Thomas C. Richmond entitled "A System of Programs for Running Foot Races." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Computer Science.

Charles P. Pfleeger

Charles P. Pfleeger, Major Professor

We have read this thesis
and recommend its acceptance:

Accepted for the Council:

The Graduate School

A SYSTEM OF PROGRAMS FOR RUNNING FOOT RACES

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Thomas C. Richmond

December 1984

Copyright © Thomas C. Richmond, 1985

All rights reserved

ACKNOWLEDGMENTS

This thesis owes its existence to the cooperation of the Knoxville Track Club, Dr. Gordon R. Sherman, the University of Tennessee Computing Center, and the University of Tennessee administration. Without that cooperation, the work described herein would never have been undertaken, and the Expo 10,000 might well have ceased to exist in 1983.

That I had the ability to create this system of programs is due in no small part to the knowledge of computer science imparted to me by my teachers. I wish to thank especially Professors K. C. O'Kane, C. P. Pfleeger, D. W. Straight, and M. G. Thomason for the knowledge and assistance they have given me. That I could make the time for this undertaking is the result of a great deal of flexibility allowed me by my co-workers at the computing center. And finally, that I was able to capture my thoughts on paper is due to the assistance of S. L. Pfleeger and B. J. Richmond. For these friends and the many others who have helped me, I give thanks.

ABSTRACT

Without computer assistance, the timely processing of data for a foot race with between two and three thousand finishers would be slow and tedious, if not impossible. During 1983 and 1984 a system of programs to process data for such a race was developed and refined.

Processing was divided into two phases, pre-race processing and race-day processing. Before the race, the system was used to assist in data entry and verification and to generate reports about the entrants. On the day of the race, the system enabled race personnel to rapidly enter and verify information about finish order and finish times. It was then necessary for the system to merge the finish time and finish order information and to locate and remove erroneous data. Finally, the system computed values for missing or removed data based on existing data, and generated reports supplying finish results for all finishers.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION	1
II.	DEFINITIONS	4
III.	PRE-RACE PROCESSING	13
IV.	POST-RACE REPORTING	23
V.	PROCESSING PROBLEMS AND SOLUTIONS	26
VI.	CHANGES AFTER 1983	46

LIST OF FIGURES

FIGURE		PAGE
1.	Map of Expo 10,000 race course	2
2.	Expo 10,000 entry form	5
3.	Running number	7
4.	Finish area	8
5.	Timing sheet	10
6.	Master file records	14
7.	Runner master list report	16
8.	Runner master list report showing errors . . .	20
9.	Last page of runner master list report	21
10.	Summary of cases in time processing	43

CHAPTER I

INTRODUCTION

In Knoxville, Tennessee, the Saturday morning of the last weekend in May is set aside for a 6.2 mile (10,000 meter) family foot race, the Expo 10,000. Figure 1 displays a map of the course. The race was first held in 1978 with 1,576 finishers. Over the years, that number grew to 2,190, 2,278, 2,439 and finally, in 1982 (the year of the Knoxville World's Fair) it peaked at 3,237. Now that the World's Fair, originally called Expo '82, is history, it is not expected that the number of finishers will continue growing; rather, the number decreased sharply in 1983 and then began to build slowly in 1984.

Without computer assistance, timely processing of the entry forms and finish results for a race of this size would be extremely difficult, if not impossible. This year, a runner can compete in his/her sex-age group; in the Knox County Lawyers vs. Doctors competition; as an educator for the Knox County schools, the Knoxville city schools or the University of Tennessee; in the 220 pound and over weight category and in up to four of the five family pairs competitions (e.g., mother-daughter, husband-wife). There are a number of other races with as many

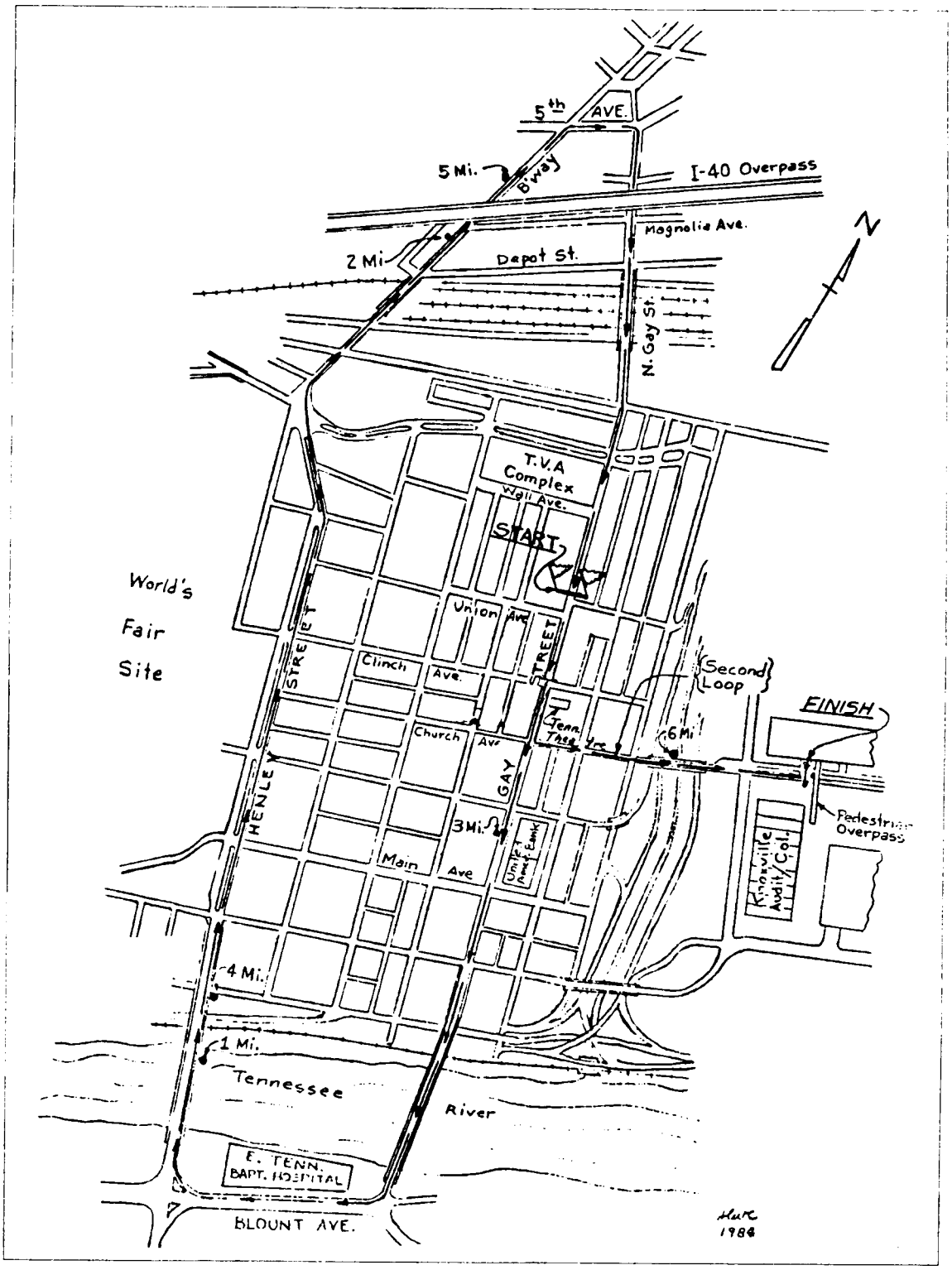


Figure 1. Map of Expo 10,000 race course.

finishers, but few if any of the others provide results to all finishers in as many categories.

In previous years, computer assistance was supplied by the United American Service Corporation; however recent events in the Knoxville banking community have made that support unavailable. The programs that had been used were offered to the Knoxville Track Club (KTC) for $6,000.00, an amount that KTC could not afford. Having exhausted all other sources, KTC approached the University of Tennessee Computing Center for help. As a member of the Knoxville Track Club and an employee of UTCC, I volunteered to "help out a little bit." This thesis describes the system of programs which grew out of that "little bit" of help, how they were developed, the problems encountered, and the solutions devised.

CHAPTER II

DEFINITIONS

Prior to the race, each person wishing to participate must complete an entry form and submit it to the KTC. A sample of this form is included as Figure 2. As each entry form is processed, a running number is assigned. This number will be worn by the runner during the race and will serve as the prime key in the computer programs for processing information about the runner. The name will, of course, be used in all reporting, and the combined name and address will appear on mailing labels to be used both before and after the race. The sex, age, team competition and specialties information will be used in generating finish results within all of these categories. T-shirt size data will be used before the race to determine the quantity to be ordered in each of the sizes.

To save storage space and facilitate processing, the computer program uses single character flags to keep track of the team competitions and the 220 pound and over categories, and runner numbers are cross referenced for the family pairs competitions. Since one parent may run with several sons or daughters, the parent's record is flagged, and the offspring's record contains the parent's runner number recorded. For husband-wife teams, the

1983 EXPO 10,000

For the benefit of The Knoxville Track Club
Sponsored By

Athletic House

Supported by Pilot Oil Corporation and Nike
with the cooperation of The City of Knoxville

Saturday, May 28, 8:00 A.M.
Downtown, Knoxville, Tennessee

RACE DAY: All competitors report to the Parkrite Parking Lot on Market Street between Union Avenue and Clinch Avenue by 7:30 a.m. The race begins promptly at 8 a.m. Please arrive early to facilitate organization.

DISTANCE: The road race is 10,000 meters (6.2 Miles). The race is sanctioned by The Athletics Congress/USA. **AWARDS:** All finishers of the race will receive a 1983 Expo 10,000 running shirt. Additional awards of trophies, Nike running shoes, Nike athletic bags and silver platters will be made in all classifications.

RACE HEADQUARTERS: The Athletic House at 522 South Gay Street will serve as race headquarters. They have additional locations: The University Shop at 1829 West Cumberland Avenue, Suburban Shopping Center West, the Midland Shopping Center in Alcoa and Brooklawn Shopping Center in Farragut. You may register in person at any of these locations until entry deadline, Tuesday, May 24. Applications will be available at all Pilot Oil stations. Mail applications will be accepted with a postmark ON OR BEFORE May 24. Additional information may be obtained at The Athletic House on Gay Street.

REGISTRATION: There will be no registration after Tuesday, May 24, 1983. The race is open to all runners regardless of age or sex. Entry fee is $5.00 per person and must be received by *entry deadline.* Everyone will receive a number to be worn and additional information through their packet. Some packets will be mailed; however, if your entry is received too late for mailing, then it may be picked up at the Gay Street Athletic House starting Thursday, May 26.

Leach and mail or bring to Athletic House.
Mail to: The Athletic House, Expo 10,000, P.O. Box 1085, Knoxville, Tennessee 37901.
Entry fee of $5.00 must accompany each entry application. Please make check payable to Knoxville Track Club.
No refunds, exchanges or transfers.

Name _____ Telephone _____
Last name first

Street _____ Apt. No. _____ City _____ State _____ Zip _____

Date of Birth ____/____/____ Age as of 5/28/83 _____ Sex—Check One ☐ Male ☐ Female

Circle your T-shirt size S M L XL

ALL APPLICATIONS MUST BE SIGNED:
In consideration of, and as a condition of my participation in the Expo 10,000 road race, I hereby for myself, my heirs, executors, and administrators, waive and release any and all rights and claims of liability for damages for personal injury and otherwise which I may have against the Knoxville Track Club, Inc., any and all participating sponsors, supporters and cooperating agencies, their agents, employees, officers, representatives, successors and assigns and all persons performing services in connection with this event. I verify and attest that I am physically fit and sufficiently trained for the completion of this event.
Further, I hereby grant full permission to any and all of the foregoing to use any photographs, videotapes, motion pictures, recordings or any other record of this event for any legitimate purpose.

Date _____

SIGNATURE _____
(IF UNDER 18, PARENT OR GUARDIAN MUST SIGN)

OFFICIAL USE ONLY

_____

_____

_____

Circle Your Classification

MALE		FEMALE	
9 and under	40-44	9 and under	40-44
10-14	45-49	10-14	45-49
15-19	50-54	15-19	50-54
20-24	55-59	20-24	55-59
25-29	60-69	20-29	60-69
30-34	70 and over	30-34	70 and over
35-39		35-39	

TEAM COMPETITIONS

Knox County Doctors
Knox County Lawyers
Knoxville City Teachers & Staff
Knox County Teachers & Staff
UTK Faculty & Staff

CIRCLE YOUR SPECIALTIES

Father-Son
Father-Daughter
Mother-Son
Mother-Daughter
Husband-Wife
Over 220 lbs.

COMPUTER PROGRAM AND RACE RESULTS BY THE UNIVERSITY OF TENNESSEE COMPUTING CENTER

Figure 2. Expo 10,000 entry form.

husband's record is flagged, and his runner number is recorded in her record.

At the finish area, information must be generated to designate those entrants who finished, in what order they finished, and how long it took them to run the race. To permit the collection of information which will show who finished and in what order, the runners are directed single file into a chute and told to maintain their finish order. As each runner reaches the far end of the chute, the runner gives finish line personnel a tear-away (see Figure 3) from his/her running number. The tear-away (which also contains the running number) is then placed in order on a finish pin for that chute. This chute pin of numbers will be used later for data entry.

The process of collecting tear-aways creates a bottleneck, since at times runners will be crossing the finish line much faster than their tear-aways can be collected. It is to address this problem that the multiple chutes are used, as shown in Figure 4. Only one chute is open at any given time. When designated finish line personnel estimate that approximately 200 runners have entered a chute, that chute is closed and the next opened. The chute pins are numbered in the same order in which the chutes are opened. Thus, after all of the physical chutes have been filled once, the first chute is

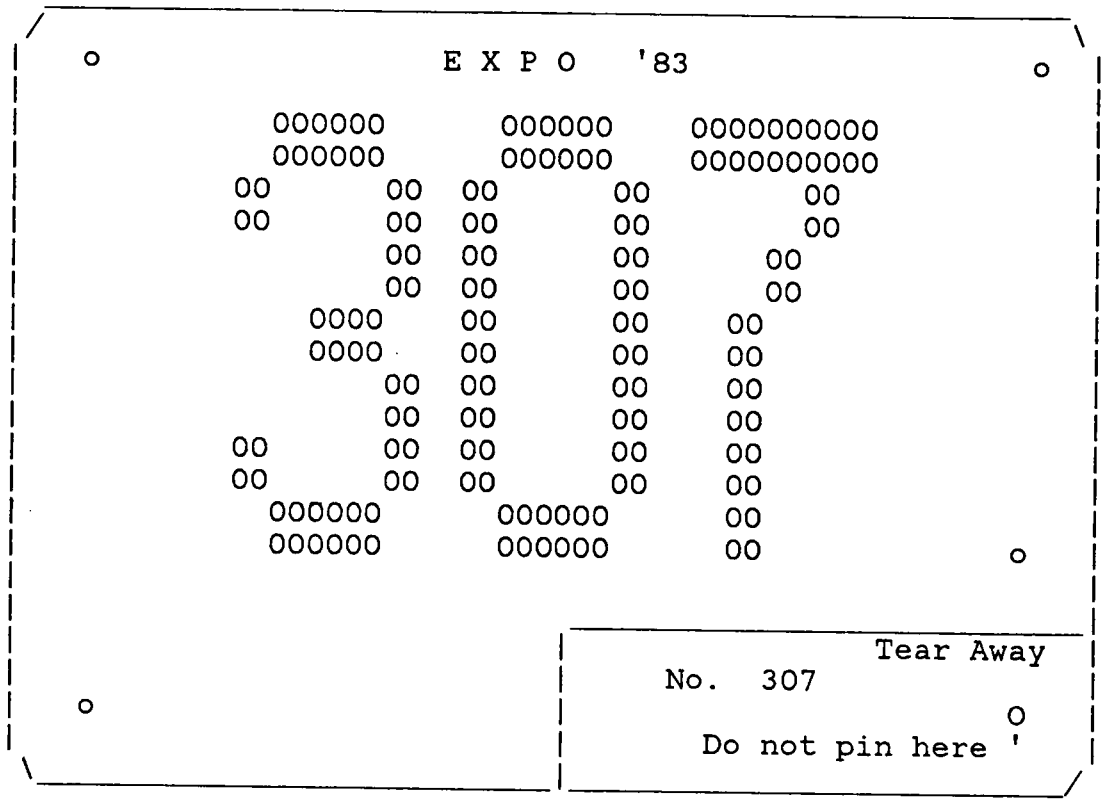


Figure 3. Runner number.

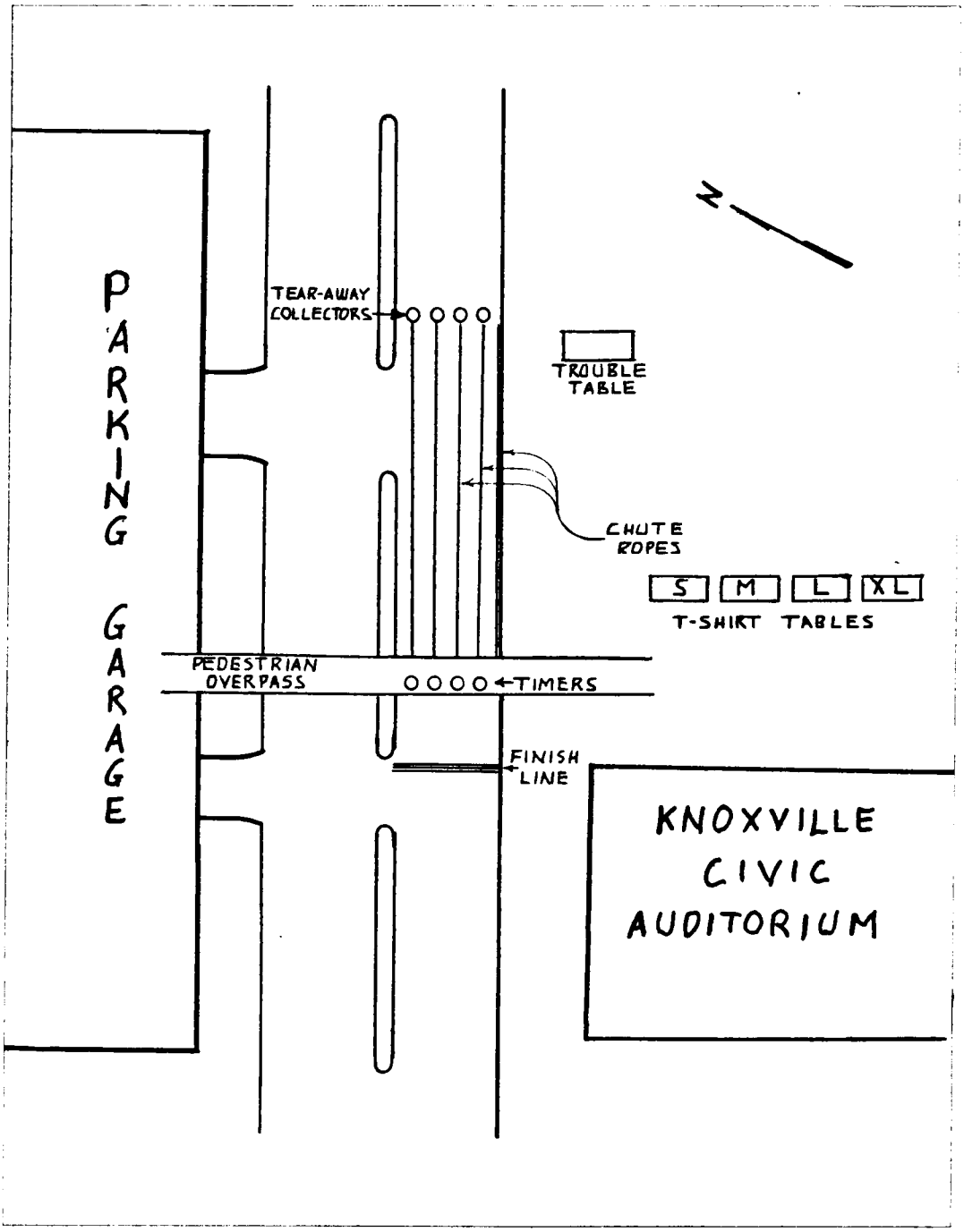


Figure 4. Finish area.

reopened and given the next number. For example, suppose there are eight physical chutes. The second time the first chute is opened, the race personnel will use the chute pin for chute nine in collecting tear-aways from the finishers in the chute. Thus, when the last chute has emptied, the chute pins will contain a record of finish order for all runners who completed the race and went through a chute.

The determination of exactly when each of the runners crossed the finish line is a problem not easily resolved. It is impossible for one person to record a piece of information or even push a button each time one of the approximately 2,500 runners crossed the finish line. Therefore, times are recorded only for some runners. By using the finish order from the chute pins, the other runners' estimated times can be computed.

A group of race personnel is assigned to record finish times; each person records as many accurate finish times as possible on timing sheets (see Figure 5). One person calls out the race elapsed time by the second while four people each try to write down the runner number of a person crossing the finish line. They write this number on the timing sheet next to the time that was called out. The result is that the finish time of any given runner can be recorded by up to four timers or not at all.

TIMER 07 SHEET 01 -EXPO 10,000 TIME SHEET- (0:24 - 0:25)

24:00/RUNNER	**24:30/RUNNER**	**25:00/RUNNER**	**25:30/RUNNER**
24:00	24:30	25:00	25:30
:01	:31	:01	:31
:02	:32	:02	:32
:03	:33	:03	:33
:04	:34	:04	:34
24:05	24:35	25:05	25:35
:06	:36	:06	:36
:07	:37	:07	:37
:08	:38	:08	:38
:09	:39	:09	:39
24:10	24:40	25:10	25:40
:11	:41	:11	:41
:12	:42	:12	:42
:13	:43	:13	:43
:14	:44	:14	:44
24:15	24:45	25:15	25:45
:16	:46	:16	:46
:17	:47	:17	:47
:18	:48	:18	:48
:19	:49	:19	:49
24:20	24:50	25:20	25:50
:21	:51	:21	:51
:22	:52	:22	:52
:23	:53	:23	:53
:24	:54	:24	:54
24:25	24:55	25:25	25:55
:26	:56	:26	:56
:27	:57	:27	:57
:28	:58	:28	:58
:29	:59	:29	:59

Figure 5. Timing sheet.

Using statistics from the 1982 Expo 10,000, there were 22 finishers with times between 35:00 and 36:00, an average of one finisher every three seconds. A woman in this group might have her number recorded by all four timers, but one timer might have written it by time 35:54 while another wrote it by time 35:55. On the other hand, 178 runners finished with times between 45:00 and 46:00, an average of three runners every second. A man in this group might not have his number recorded by any of the timers. The process by which the computer will associate these times with the numbers on the chute pins and then generate the remaining times will be described in later chapters.

Once it is known who finished and when, reports must be generated presenting this information in a number of ways. As is indicated on the entry form (Figure 2), a runner can register for an age group within his/her sex, one or both of the team competitions, up to four of the family-pairs specialties, and the over 220 pound competition. A total of forty reports must be generated, and one runner could be listed in as many as ten of them.

The remainder of this thesis describes data collection and processing before the race, data collection and processing after the race, and the solutions to

several problems that arose during development of the computing system to manage this race.

CHAPTER III

PRE-RACE PROCESSING

Before the race is run, information from the runner's entry form is entered, checked for validity, and printed for reference by race personnel. This chapter describes the data collected and processed in advance of the race.

When an entry form is received by registration personnel, it is scanned for obvious mistakes and/or omissions, a runner number is assigned, and the form is passed on to the data entry group. The information on each form is then typed into the computer and verified to avoid typographical errors. The master file entry for each runner (see Figure 6) consists of two 80-byte records containing all necessary information from the entry form. The information on these records is used for the generation of all reports and forms for the race. The information is arranged so that the first record alone is sufficient for finish result reporting. The two records are linked together by the runner number in the first five positions of each, and are differentiated by either an "A" or a "B" in position seven.

In loading the information into the "A" and "B" records, data entry personnel type the data two times. The first time the information is typed, the record is

TYPE "A" RECORD

Position	Length	Field
01-05	5	Runner Number
07-07	1	Record Type (A-Name)
09-38	30	Runner's Name (last, first mi.)
40-40	1	Sex (1-Male, 2-Female)
42-43	2	Age (in years)
45-46	2	Class (lowest age in age group)
50-54	5	Family Pairs - Father-Son
55-59	5	" " - Father-Daughter
60-64	5	" " - Mother-Son
65-69	5	" " - Mother-Daughter
70-74	5	" " - Husband-Wife
75-75	1	Prof. Teams (A-M.D., B-Law.)
77-77	1	Educ. Teams (C-City, D-Co., E-UTK)
79-79	1	Over 220 lbs. (F)

TYPE "B" RECORD

Position	Length	Field
01-05	5	Runner Number
07-07	1	Record Type (B-Address)
09-38	30	Street Address (and Apt. No.)
40-55	16	City
57-58	2	State (two letter abbreviation)
60-64	5	Zip Code
66-69	4	Zip Code Extension
71-72	2	Date of Birth - Month
73-74	2	" " " - Day
75-76	2	" " " - Year
78-78	1	T-Shirt Size (S, M, L, X)
80-80	1	Knoxville Track Club Member

Figure 6. Master file records.

created. During the second, or verification entry, the typed data is compared with the information in the record. The data entry person resolves any inconsistencies at this time. This system of entry and verification insures that the information in the records is consistent with that on the entry form (insofar as the data entry operator can read it).

Entry forms are processed in groups of fifty. With each addition of fifty new entrants to the master file, reports are generated and returned to the registration personnel. The information on this report (see Figure 7) can then be used by registration personnel for:

1. matching runner numbers for family pairs;
2. contacting race entrants to obtain additional information if necessary;
3. answering inquiries from entrants;
4. correcting errors.

To facilitate these functions, two reports are generated, the first in order by runner number, the second in order by runner name. The information for each runner on both reports is identical and reflects the information stored in the master file as entered from the entry forms.

The computer program which generates these reports tests some of the data fields for erroneous or

EXPO 10,000 RUN, MAY 28, 1983
KNOXVILLE, TENNESSEE

RUNNER MASTER LIST (IN RUNNER # SEQUENCE) 05/18/83 PAGE 14

RUNNER NUMBER	NAME & STREET ADDRESS	CITY STATE & ZIP	SEX	AGE CL B'DATE	*-----FAMILY PAIRS-----* F-S	F-D	M-S	M-D	H-W	PROF CLASS	EDUC CLASS	220 LBS	TSHIRT SIZE	KTC MEMB
00236	PETERSON, MIKE 332 CROSSFIELD DR	KNOXVILLE TN 37920	M	42 40 04/10/41	00000	-	-	-	-	LAW	-	YES	XL	-
00237	PETERSON, D M JR 332 CROSSFIELD DR	KNOXVILLE TN 37920	M	10 10 11/21/72	00236	-	-	-	-	-	-	-	S	-
00238	ADAMS, RALPH W RT 4 BOX 207	SEYMOUR TN 37865	M	37 35 04/14/46	-	-	-	-	-	-	CNTY	-	XL	-
00239	TOPPINS, JONATHAN 12922 WARRIOR TR	KNOXVILLE TN 37922	M	16 15 08/17/66	-	-	-	-	-	-	-	-	L	YES
00240	RICHMOND, HUGH ELLSWORTH 15 1/2 CRAGMERE RD	KNOXVILLE TN 37919	M	73 70 11/02/09	-	-	-	-	00000	-	-	-	XL	-
00241	RICHMOND, JEANNETTE PIERCE 15 1/2 CRAGMERE RD	KNOXVILLE TN 37919	F	74 70 03/24/09	-	-	-	-	00240	-	-	-	M	-
00242	JONES, WILLIAM R SR 301 MAPLE AVE	KNOXVILLE TN 37919	M	71 70 01/31/12	00000	-	-	-	-	DOC	-	-	XL	YES
00243	JONES, BILL 301 MAPLE AVE	KNOXVILLE TN 37919	M	42 40 10/07/40	00242	00000	-	-	00000	-	UTK	YES	XL	YES
00244	JONES, JANE 301 MAPLE AVE	KNOXVILLE TN 37919	F	43 40 09/09/39	-	-	00000	00000	00243	LAW	-	-	M	-
00245	JONES, WILLIAM R III 301 MAPLE AVE	KNOXVILLE TN 37919	M	20 20 06/25/62	00243	-	00244	-	-	-	CNTY	-	L	-
00246	JONES, JANEY 301 MAPLE AVE	KNOXVILLE TN 37919	F	20 20 06/25/62	-	00243	-	00244	-	-	CITY	-	S	-
00247	BEASON, MARGARET 450 CHEROKEE BLVD	KNOXVILLE TN 37919	F	32 30 05/22/51	-	-	-	-	-	-	UTK	-	S	-
00248	DAVIS, ROB 9500 HALLBROOK RD	KNOXVILLE TN 37918	M	45 45 06/10/37	00000	-	-	-	-	-	-	-	M	YES
00249	DAVIS, BOB 9500 HALLBROOK RD	KNOXVILLE TN 37918	M	18 15 11/17/64	00248	-	-	-	-	-	-	-	M	YES
00250	DAVIDSON, MIKE 2117 ANDY HOLT AVE APT NO 388	KNOXVILLE TN 37916	M	19 15 11/30/63	-	-	-	-	-	-	-	-	L	-
00251	SCHWINGER, GERALD R 12106 BRANDYWINE BLVD	KNOXVILLE TN 37922	M	39 35 01/25/44	-	-	-	-	-	DOC	-	-	XL	-
00252	CONRAD, DAVID O 22308 CLEAR POINT DR	KNOXVILLE TN 37922	M	34 30 12/20/48	-	-	-	-	-	-	-	-	L	-
00253	FLETCHER, JOHN 8200 SPRUCELAND DR	POWELL TN 37849	M	15 15 07/07/67	-	-	-	-	-	-	-	-	M	-

Figure 7. Runner master list report.

unreasonable data. If any of these fields is in error, the program prints an error message in the report. This message is printed just before the two lines of information about the runner and tells which field or fields are bad; the data in the bad field is then printed exactly as it appears in the record. The runner number is the unique number that is assigned by the registration personnel to the entrant when his or her form is first received. Name, street address, city, state and zip code are stored in five separate fields, allowing access to each of the five fields as necessary. A sixth field for zip code extension is included for those entrants having a nine-digit zip code. These fields are not tested and are printed on the report exactly as they are stored in the records.

The runner's sex is stored as either a "1" for a male or a "2" for a female. This field is tested, and the letter "M" or "F" is substituted for the number before generating the report. Anything other than a "1" or a "2" is flagged as an error. The master file entry includes fields for the runner's age, and the beginning year of the age group into which the runner falls, along with the runner's date of birth. The runner's age is checked against the age group, and if it is outside of the range of that group, it is flagged as an error.

The next five fields are used for people who have entered one or more of the family pairs competitions. In the first four fields, (i.e., "F-S" father-son, "F-D" father-daughter, "M-S" mother-son, and "M-D" mother-daughter), if the records being processed are those of a parent, five zeros are stored in the in the appropriate field. For the records of a child, the runner number of the parent is stored in the same field. In the fifth field (i.e., "H-W" husband-wife), the husband's record will have zeros, and the wife's record will contain her husband's runner number. If any of these fields are blank the report will contain " - " in the appropriate column; otherwise the number stored in the record will be printed.

The PROF CLASS column shows whether or not the entrant registered for the Knox County Doctors vs. Knox County Lawyers competition. A test of this field in the record must yield an "A" for a doctor or a "B" for a lawyer or a blank for someone not participating in this competition. Any other entry in this field will be flagged as an error. In the report, an "A" will be translated "DOC", a "B" will be translated "LAW", and a blank will be translated " - ". Likewise, the EDUC CLASS column shows if the entrant has registered as a member of the Knoxville City Teachers and Staff, Knox County Teachers and Staff, or UTK Faculty and Staff competition.

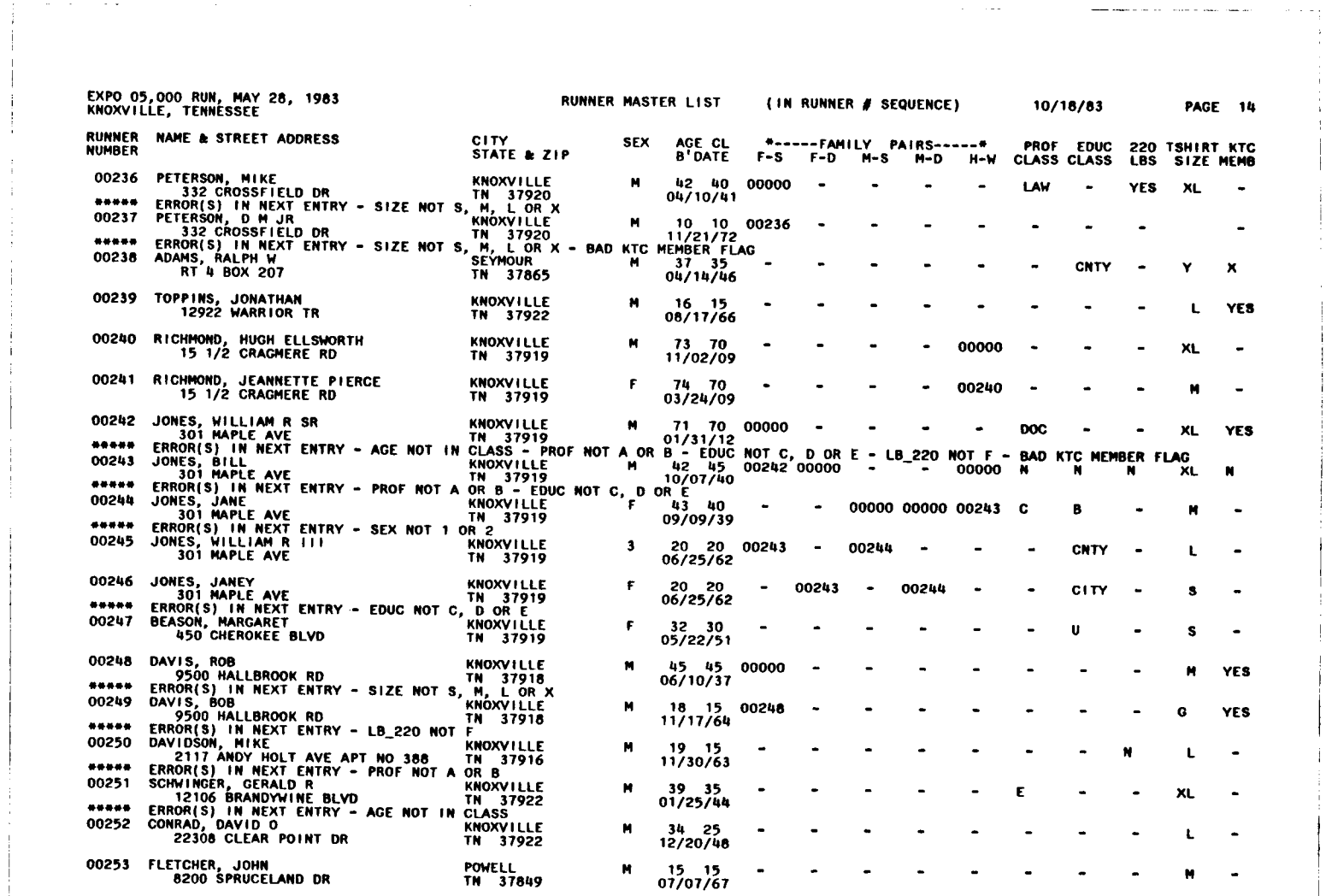
This field in the record is tested as follows: "C", "D", "E", and " " are valid entries. "C" is translated for the report as "CITY", "D" is translated as "CNTY", "E" is translated as "UTK " and a blank on the record is translated as " - ". Any other entry in that field is an error.

The next column, 220 LBS., indicates whether or not the entrant is participating in the Over 220 Pound competition. This field in the record must contain an "F" or a blank, for which the report will translate "F" as "YES", and a blank as " - ".

On the second, or "B", record, only the T-shirt size and the KTC membership flag will be checked. The T-shirt size field must contain an "S", an "M", an "L" or an "X", and the corresponding data on the report will be " S", " M", " L", "XL". The KTC membership flag must be "Y" or a blank. "Y" is translated as "YES", and a blank is translated as " - ". Figure 8 shows an example of a report containing errors.

The final page of this report (see Figure 9) contains a count of the number of people requesting each of the T-shirt sizes and a count of the number of race entrants who are members of the Knoxville Track Club. The T-shirt data are used to order T-shirts prior to the race.

Any changes or corrections to the entry data are



EXPO 05,000 RUN, MAY 28, 1983 — RUNNER MASTER LIST (IN RUNNER # SEQUENCE) — 10/18/83 — PAGE 14
KNOXVILLE, TENNESSEE

RUNNER NUMBER	NAME & STREET ADDRESS	CITY STATE & ZIP	SEX	AGE CL B'DATE	*-----FAMILY PAIRS-----* F-S	F-D	M-S	M-D	H-W	PROF CLASS	EDUC CLASS	220 LBS	TSHIRT SIZE	KTC MEMB
00236	PETERSON, MIKE 332 CROSSFIELD DR	KNOXVILLE TN 37920	M	42 40 04/10/41	00000	-	-	-	-	LAW	-	YES	XL	-
*****	ERROR(S) IN NEXT ENTRY - SIZE NOT S, M, L OR X													
00237	PETERSON, D M JR 332 CROSSFIELD DR	KNOXVILLE TN 37920	M	10 10 11/21/72	00236	-	-	-	-	-	-	-		-
*****	ERROR(S) IN NEXT ENTRY - SIZE NOT S, M, L OR X - BAD KTC MEMBER FLAG													
00238	ADAMS, RALPH W RT 4 BOX 207	SEYMOUR TN 37865	M	37 35 04/14/46	-	-	-	-	-	-	CNTY	-	Y	X
00239	TOPPINS, JONATHAN 12922 WARRIOR TR	KNOXVILLE TN 37922	M	16 15 08/17/66	-	-	-	-	-	-	-	-	L	YES
00240	RICHMOND, HUGH ELLSWORTH 15 1/2 CRAGMERE RD	KNOXVILLE TN 37919	M	73 70 11/02/09	-	-	-	-	00000	-	-	-	XL	-
00241	RICHMOND, JEANNETTE PIERCE 15 1/2 CRAGMERE RD	KNOXVILLE TN 37919	F	74 70 03/24/09	-	-	-	-	00240	-	-	-	M	-
00242	JONES, WILLIAM R SR 301 MAPLE AVE	KNOXVILLE TN 37919	M	71 70 01/31/12	00000	-	-	-	-	DOC	-	-	XL	YES
*****	ERROR(S) IN NEXT ENTRY - AGE NOT IN CLASS - PROF NOT A OR B - EDUC NOT C, D OR E - LB_220 NOT F - BAD KTC MEMBER FLAG													
00243	JONES, BILL 301 MAPLE AVE	KNOXVILLE TN 37919	M	42 45 10/07/40	00242	00000	-	-	00000	M	N	N	XL	N
*****	ERROR(S) IN NEXT ENTRY - PROF NOT A OR B - EDUC NOT C, D OR E													
00244	JONES, JANE 301 MAPLE AVE	KNOXVILLE TN 37919	F	43 40 09/09/39	-	-	00000	00000	00243	C	B	-	M	-
*****	ERROR(S) IN NEXT ENTRY - SEX NOT 1 OR 2													
00245	JONES, WILLIAM R III 301 MAPLE AVE	KNOXVILLE TN 37919	3	20 20 06/25/62	00243	-	00244	-	-	-	CNTY	-	L	-
00246	JONES, JANEY 301 MAPLE AVE	KNOXVILLE TN 37919	F	20 20 06/25/62	-	00243	-	00244	-	-	CITY	-	S	-
*****	ERROR(S) IN NEXT ENTRY - EDUC NOT C, D OR E													
00247	BEASON, MARGARET 450 CHEROKEE BLVD	KNOXVILLE TN 37919	F	32 30 05/22/51	-	-	-	-	-	-	U	-	S	-
00248	DAVIS, ROB 9500 HALLBROOK RD	KNOXVILLE TN 37918	M	45 45 06/10/37	00000	-	-	-	-	-	-	-	M	YES
*****	ERROR(S) IN NEXT ENTRY - SIZE NOT S, M, L OR X													
00249	DAVIS, BOB 9500 HALLBROOK RD	KNOXVILLE TN 37918	M	18 15 11/17/64	00248	-	-	-	-	-	-	-	G	YES
*****	ERROR(S) IN NEXT ENTRY - LB_220 NOT F													
00250	DAVIDSON, MIKE 2117 ANDY HOLT AVE APT NO 388	KNOXVILLE TN 37916	M	19 15 11/30/63	-	-	-	-	-	-	-	N	L	-
*****	ERROR(S) IN NEXT ENTRY - PROF NOT A OR B													
00251	SCHWINGER, GERALD R 12106 BRANDYWINE BLVD	KNOXVILLE TN 37922	M	39 35 01/25/44	-	-	-	-	-	E	-	-	XL	-
*****	ERROR(S) IN NEXT ENTRY - AGE NOT IN CLASS													
00252	CONRAD, DAVID O 22308 CLEAR POINT DR	KNOXVILLE TN 37922	M	34 25 12/20/48	-	-	-	-	-	-	-	-	L	-
00253	FLETCHER, JOHN 8200 SPRUCELAND DR	POWELL TN 37849	M	15 15 07/07/67	-	-	-	-	-	-	-	-	M	-

Figure 8. Runner master list report showing errors.

T SHIRTS		KTC MEMBERSHIP	
SMALL	200	YES	217
MEDIUM	865	NO	2,809
LARGE	1,376		
X-LARGE	565		

Figure 9. Last page of runner master list report.

initiated by registration personnel and processed by data entry personnel with the next batch of entry forms, so that by race day all errors have been detected and corrected. In the final days prior to the race, a separate report in this same format is generated for each of the following groups: Knox County Doctors, Knox County Lawyers, Knoxville City Teachers and Staff, Knox County Teachers and Staff, and UTK Faculty and Staff. These reports contain only the entrants participating in each group and are distributed to a representative of each group.

CHAPTER IV

POST-RACE REPORTING

The previous chapter described reporting before the race is run. After the race is run, additional reports are needed. These reports show race finish results, separated into the various runner classes.

As soon as possible after runners begin crossing the finish line, a series of reports must be generated to supply finish information to race officials, news media and the runners themselves. These reports are as follows: (1) all finishers in order by finish position, (2) all male finishers, (3) all female finishers, (4) through (16) all male finishers in age categories 0 - 9, 10 - 14, 15 - 19, 20 - 24, 25 - 29, 30 - 34, 35 - 39, 40 - 44, 45 - 49, 50 - 54, 55 - 59, 60 - 69, 70 and above, (17) through (29) all female finishers in the same age categories, (30) all Knox County Doctors, (31) all Knox County Lawyers, (32) all Knoxville City Teachers and Staff, (33) all Knox County Teachers and Staff, (34) all UTK Faculty and Staff, (35) all over 220 lbs., (36) all Father-Son pairs, (37) all Father-Daughter pairs, (38) all Mother-Son pairs, (39) all Mother-Daughter pairs, (40) all Husband-Wife pairs.

Entries in each of the first thirty-five reports are arranged by finish position. Each entry contains the

runner's finish position in that group, overall finish position, finish time in hours, minutes and seconds, and a flag to indicate whether that time was observed or computed (the recording and computing of finish times is explained in the next chapter), the finisher's runner number, his or her name and address, as well as, age, sex and age group.

The last five reports concern the family pairs and must be in a different format. Entries are arranged in increasing order by the sum of the finish times of the pair. An entry consists of two lines, the first contains information about the parent or the husband and the second contains information about the child or wife. Each line contains: (1) the finish position of the pair, and for the person their (2) finish time, (3) runner number, (4) name, (5) city, (6) state, and (7) street address, as well as their (8) age, (9) sex, and (10) age group.

Partial versions of each of the above reports are generated with initial data, and more complete versions are produced as more and more finish data becomes available. As soon as possible after all runners have finished the race, a final, complete copy of each report is generated.

Also following race conclusion, a shortened version of the report containing all finishers is generated. It

contains only the overall finish position, finish time, sex, age, finish position within the age group, name, city and state of the finishers. This report, sorted by finish position, is distributed to all entrants, and is published in the following year's race booklet. Multiple copies of this report, sorted by runner name, along with copies of the first forty reports, are distributed to the five race registration locations so that runners can obtain finish information in person or by telephone from each of these locations. Copies of all reports are also distributed to the news media.

After the race, several sets of mailing labels are also generated. Full sets are used to send out the race results to all entrants, to send out applications for the race the following year, and to send out running numbers the following year for those entrants whose addresses have not changed. Partial sets of mailing labels, including only runners having selected zip codes, are printed for distribution of applications for other races.

CHAPTER V

PROCESSING PROBLEMS AND SOLUTIONS

This chapter describes the programs that were developed to accept input data and to produce the reports just described. In developing these programs, several difficulties were uncovered, which complicated the algorithm development. This chapter also outlines these problems and the solutions that were developed.

Because the time available to develop the processing programs was minimal, and because data to test the system was generated by hand, it was likely that some sections of the system would not be fully tested until race day. In order to minimize the adverse effects of a possible flaw in the system, a decision was made to modularize the system as much as possible. In this way, if one module failed, it could be by-passed and its function performed manually, if necessary, while the remaining modules could still be used.

A further requirement of the system was that some race results had to be made available before all the runners had finished. As can be surmised from the pairs categories, this is a family-oriented race. Therefore, finish times vary from thirty to one hundred and eighty minutes or more. Thus, awards must be made to the winners

before the last entrants have even finished the course. Therefore, the system had to be capable of producing partial results before the last finishers crossed the finish line.

Problems encountered in the processing include entrants who arrived at the starting line without running numbers because the numbers were lost in the mail, or were left at home, or the runner had not registered. Added to this group are those who lose their number on the race course.

In any given Expo 10,000 race, two to three thousand runners are expected to finish the race over a two hour period. At peak times, as many as ten runners can cross the finish line in a one second interval. The strain that this places on the personnel recording runner numbers on the timer sheets results in many numbers not being recorded, the same number being recorded under different time slots by different timers, and garbled runner numbers being recorded on timer sheets.

It is important to report finishers in their correct finish order. Although a great deal of effort is made to keep the runners in their correct finish order as they move through the chutes, some people fall back or pass other runners in the chute after having crossed the finish line. Even though it is understood that there will be

some out-of-order numbers on the chute pins, the pins contain the only complete record of the runners who finished the race. Further, these numbers are pre-printed and thus not subject to transcription errors, as are the numbers on the time sheets. Therefore, the design decision was made to accept the information on these pins as the official order of finish for the race. Recorded finish times are adjusted if necessary to comply with this finish order.

The following procedures are used to solve problems during and after the race. These solutions were developed over several years of administering this race. The numberless runner problem is resolved before the race, whenever possible. A trouble table is set up where a runner can obtain a duplicate running number if his or her number was lost in the mail or left at home. A range of numbers is set aside prior to the race to be used by the chute personnel. If a runner reaches the end of the chute without a running number, a runner number from this range is assigned to the runner; the runner is then instructed to report to the finish area trouble table. A race official at that table decides whether the runner and the new runner number are to be allowed in the final race statistics. This range of numbers is flagged in the computer programs, and the numbers in this range are

eliminated if information is not received from the trouble table within a preset length of time.

Race personnel are assigned to the chutes to assist runners who cross the finish line but are physically unable to proceed through the chute. The personnel give aid to the runner and insure that the tear-away from the runner's runner number is carried to the end of the chute and placed on the chute pin in the correct order. Chute personnel and trouble table officials accept any complaints about runners passing other runners while in the chutes. On the basis of these complaints, a runner can be disqualified and his or her running number tear-away removed from the chute pin.

Having done as much as possible to insure the correctness of the data represented by the tear-aways on the chute pins, race officials next translate this data into machine-readable form. Personnel responsible for this data entry are untrained volunteers. Therefore, a friendly interactive data entry and verification procedure was designed and written for the 1983 race. The following is a description of that data entry program.

Upon activation the program identifies itself and asks the user if he or she wants instruction. As will be the case whenever information is requested from the operator, the acceptable answers are displayed within

parentheses at the end of the question. Any unacceptable input, either data or a response to a question, is signaled both on the screen and with an audible signal. If the user responds that help is desired, a subroutine is invoked to give an explanation of the general procedures for the program. All input to the program is read as character data and then is tested to determine whether it is alphabetic or numeric and whether it falls within the range of acceptable input for that section of the program.

The operator is next instructed to enter and verify the number of the chute to be processed. The program generates two file names for the chute; one file will be used for data which is being loaded for the first time, and the second file will store data which has been verified. The user is asked next which of those two activities (entry or verification) he or she wants to perform. If entry mode is chosen, the user is asked again if help is needed. If the answer is "yes" a help file on data entry is displayed. All data is tested to be sure that it is numeric. If a carriage return (<CR>) is entered without data having been previously entered, the following menu of actions is presented.

```
YOU HAVE ENTERED A BLANK - YOUR OPTIONS ARE TO:
    A- CONTINUE (ENTER NEXT NUMBER)
    B- REPLACE LAST NUMBER
    C- SAVE FILE (ALL NUMBERS HAVE BEEN ENTERED)
    D- QUIT (THROW AWAY ALL NUMBERS JUST ENTERED)
    E- PRINT INSTRUCTIONS
WHICH OPTION WOULD YOU LIKE (A/B/C/D/E)? [A]
```

Note that <CR> acts as the file terminator when used with option C.

If a bona fide runner number is entered, that number is right-justified and then echoed to the screen along with that runner's position in his or her chute. The information is stored in an array to allow the program to execute faster. When the operator enters a response of "C", the entire array is written to disk. The user is then given the option of entering or verifying additional chutes. If more data is to be entered, the user is returned to the beginning of the program.

If verification mode is chosen, a help file on this activity is offered. The program then searches the disk for the file to be verified. If no file is found, or if an empty file is found, the user is given the option of switching to entry mode. If the operator enters a blank or a <CR> without data, the following menu is displayed:

```
YOU HAVE ENTERED A BLANK - YOUR OPTIONS ARE TO:
    A- CONTINUE (ENTER NEXT NUMBER)
    B- QUIT (STOP PROCESSING THIS FILE)
    C- PRINT INSTRUCTIONS
WHICH OPTION WOULD YOU LIKE (A/B/C)? [A]
```

In verification mode, the program will right-justify each number read and compare it to the number in the same position in the file. If the number matches, it is echoed to the screen with an indication that verification was successful. If a mismatch is detected, a message similar to the following is displayed along with an audible signal.

```
THE NUMBER YOU ENTERED DOES NOT MATCH THE NUMBER
IN THE FILE.  PLEASE CHECK THE FOLLOWING AGAINST
YOUR NUMBER:
RUNNER:  56    FILE NO.: 1234    YOUR NO.  124
THESE OPTIONS ARE AVAILABLE:
    A- CHANGE MY NUMBER
    B- CHANGE NUMBER IN FILE
    C- INSERT NUMBER
    D- DELETE NUMBER
    E- SEE LONGER EXPLANATION OF THESE OPTIONS
WHICH OPTION WOULD YOU LIKE (A/B/C/D/E)? [A]
```

Choosing option "E" displays the following:

```
THESE OPTIONS ARE AVAILABLE:
     A- MY NUMBER IS WRONG, I WILL ENTER THE CORRECT
        NUMBER.
     B- THE FILE IS WRONG, I WILL RE-ENTER MY NUMBER
        FOR VERIFICATION.
     C- MY NUMBER IS CORRECT, BUT IT IS MISSING FROM
        THE FILE.  I WILL RE-ENTER MY NUMBER TO
        VERIFY THAT IT SHOULD BE INSERTED.
     D- MY NUMBER IS CORRECT, BUT THE ONE IN THE FILE
        SHOULD BE DELETED.  I WILL RE-ENTER MY NUMBER
        TO MATCH THE NEXT NUMBER IN THE FILE.
     E- REPEAT THIS MESSAGE.
 WHICH OPTION WOULD YOU LIKE (A/B/C/D/E)? [A]
```

When all numbers in the file have been verified, the operator is given the choice again of entering data for another chute or leaving the program.

At the same time that the chute data are being entered, other data entry personnel are entering the time data. Data entry personnel enter and verify each runner number and the minutes and seconds of the finish time for that number. Timing sheets can have only two minute digits and two second digits, and it reasonable to assume that the first finisher will take at least 24 minutes to complete the 6.2 mile race. Thus the times on these sheets range from 24:00 to 99:59 (1:39:59).

When a reasonable number of entries is accumulated, the file is transferred to a program that stores it on disk. Since they must be accessed by other processing programs executing concurrently and must be available for updates with additional data, these finish time files are stored in a generation data group (GDG). The data is organized as an array with one cell for each runner, referred to as a "finish time array."

As a time for a given runner is read, the cell for that runner number is checked to determine whether a time has already been recorded. This would be the case if more than one of the timers recorded the runner's number when he or she crossed the finish line or if a timer wrote down the wrong number, for example, if runner 1234 crossed the line but one of the timers wrote 124. If a time has already been recorded in the cell and the two times are

within two seconds of each other, then the faster of the two times is stored in the cell. If there is a time recorded and the difference between the two is greater than two seconds, then it is assumed that there has been an error in recording the time and the cell is set to zero. If there is no time in the cell, then the number is stored. When all the time and number combinations in this batch are processed, the file is stored as a new generation in the data group. Because of this method of handling the data, the input generation is available to other programs for processing while the file is being updated.

An inter-chute data file is established to store selected information about each chute and to allow that information to be carried over from one chute to another. Included in the inter-chute data file is the following information about each chute: (1) a count of all runners who have been processed at this point (including this chute), (2) the number of runners in this chute, (3) a count of the runners at the beginning of this chute who did not have times in the time array, (4) the finish time of the first person in this chute who had a recorded time, (5 - 7) the finish times of the last three people in this chute who had recorded times, (8) a count of the runners

at the end of this chute without times recorded in the time array.

Before the first chute is processed this file is zeroed. The chutes are processed one at a time, in order. One chute pin data file, the current generation of the finish time array, and the inter-chute data file are loaded. The chute number is extracted from the chute pin data file, and the inter-chute data file is checked to verify that this data has not been processed previously. First a check is made to ensure that the first record in the chute pin data file is in fact a valid chute number (01 - 40); then the inter-chute data file is checked to see if the previous chute has been processed. The program terminates if the previous chute has not been processed.

Each time this program is executed it reads a parameter which must be supplied through job control language. The acceptable values for this parameter are NORMAL or RERUNnn where nn is the number of the chute being processed. The data for a chute can not be processed more than once unless the parameter is set for rerun. If the previous chute has been processed and if the parameter is set for rerun, then a warning is issued to say that all higher-numbered chutes must be processed again. If the parameter is set for normal processing, then a message is issued to tell the user that the chute

has already been processed, and the program then terminates. If processing continues, a chute/runner data array is created within the program, and the individual runner numbers in the chute pin data file are processed, creating for each an entry in the chute/runner data array that contains a runner number, an absolute finish number, and a finish time (if one is available from the finish time array).

The last entry in each chute pin data file contains a count of the runner numbers in that file. After all entries in the chute pin data file have been processed, the number in the last entry is compared with the number of records processed. If there is a discrepancy the program terminates. If there are no problems, the inter-chute data file is updated to reflect the information obtained, and the file is written back to the disk. At this time the chute/runner data array is also stored as a file on the disk.

The most difficult part of this thesis was designing an algorithm that could detect and compensate for errors in the finish times. It must be assumed that finish times for runners in a race are monotonically increasing. Since it was expected that there would be errors in recording runner numbers on the finish time sheets, and since it was known that (contrary to instructions) some runners would

pass others in the finish chutes, a mechanism was required to eliminate the resulting incorrect times from the chute/runner data file. The remainder of this chapter contains a description of how incorrect times are handled.

The file processed will not have times for all of the entries, so that consecutive processing in the normal sense will not work. By reading the file from disk, storing the data in main memory in the form of an array, and using five address pointers into that array, it is possible to overlay five runner entries in the array that did have recorded times and treat them as if they were consecutive data points. It must be remembered in processing these data points that they are not truly consecutive. Thus all changes must be made through the manipulation of pointers and overlays. For purposes of this discussion, the times in the five data points will be referred to as A, B, C, D, and E, with A the earliest.

Before any other processing can take place a base of three acceptable times must be established. If in processing the first five points in the chute 01 data file, it cannot be established that A is less than or equal to B, (A <= B), and that B is less than or equal to C (B <= C), the file is rejected and manual intervention is needed to supply correct beginning data before computer processing can begin.

Once the data for the first chute is processed, the last three acceptable times in the chute are stored in the inter-chute data file. This file is read and added to by the computer during the processing of each subsequent chute. Because of this, the base for processing chute 2 and above is taken from information in the inter-chute data file for the previous chute, and the first three times in the chute being processed do not have to be in monotonically increasing order for the chute to be processed successfully. For example, to begin the processing of chute 5, A, B and C would be set to the highest accepted times in chute 4, while D and E would be set to the first two recorded times in chute 5.

Having established a monotonically increasing base of A <= B <= C, we examine D and E to determine whether they are acceptable times or must be rejected. The possible relationships between D and E and the base (A, B and C) are enumerated as a series of cases. These cases are tested sequentially, and when a case is found to be true, the pointers are shifted and consideration of the five cases begins again with new values. As a safeguard, if none of the cases is found to be true for any given set of data points, an error message is printed and the program stops. In some cases a decision can be made on a value of D alone, while other cases will require the use of E.

The first case to be tested is that of D being greater than or equal to C (D >= C). That is, the time of the D runner was either greater than that of the C runner, or both runners had the same recorded finish time. If this is true, D is accepted as a valid time, and all pointers are moved forward. That is, the A pointer is changed to point at the data currently stored in B, the B pointer is changed to point at the data currently stored in C, the C pointer is changed to point at the data currently stored in D, and the D pointer is changed to point at the data currently stored in E. The file is searched for the next runner number that has a recorded time, and the E pointer is set to point at this entry. Processing then begins again with the new set of pointers.

If the first case was not true, the second case to be tested is that of D being less than A (D < A). If this is true, the D time is considered to be in error, (i.e., the D runner could not have finished faster than three or more people who were in the chute ahead of him or her). In this case, the D time is removed from the file and that runner will be treated as if he or she had not had a recorded finish time. For this case the A, B and C pointers remain unchanged, the D pointer is changed to point at the E entry, and a new value for E is found as before.

If the second case is false, then the third case is examined. We know from the failure of the first case that D is less than C. We now test to find if D is greater than or equal to B. If case 3 is true, one of two subcases must exist. The first subcase occurs when D's finish time is within two seconds of C's finish time. In this subcase the C runner's finish time is changed to be equal to the D runner's finish time. In other words, if the finish times of two out-of-order runners differ by two seconds or less, both runners are given the faster finish time. Then all pointers are shifted as in case 1, and processing begins again. In the second subcase, the supposition is that the C runner's time is erroneously high. The C runner's time is removed from the file. In this subcase, pointers A and B remain unchanged, the C pointer is changed to point at D, the D pointer is changed to point to E, and E receives a new value.

If none of the first three cases is true, the most complicated of the four cases remains. The fourth case contains four subcases, and testing involves the use of the E entry. Because of the failure of the previous three cases, it is now known that A <= D < B <= C. The first subcase occurs when E is greater than or equal to C. If this is true, then A <= B <= C <= E (that is A, B, C, and E are monotonically increasing) and D is in error. This

subcase is handled identically to case 2. Otherwise E is less than C. The second subcase occurs when E is greater than or equal to B. If this is true, then A, B, and E are monotonically increasing, D is low and in error, and at the same time C is high and in error. In this case, both C and D are removed from the file. The pointers are then moved as necessary. That is, A and B pointers remain unchanged, the C pointer is changed to point at E, and D and E are set to point at the next two runner numbers in the file that have recorded times.

The third subcase is the case where E is greater than or equal to D. This subcase holds when A, D, and E are monotonically increasing and both B and C are erroneously high. In this subcase, B and C are removed from the file, and the D and E pointers are moved forward. That is, A remains unchanged, B pointer is changed to point at D, C pointer is changed to point at E, and D and E are handled as in the previous subcase.

The fourth subcase is the situation when E is less than D. If this subcase is true, then both D and E are low and in error. These times for D and E must therefore be removed from the file. In this subcase, the A, B and C pointers remain unchanged, and the D and E pointers are handled as in previous subcases.

Once the case at hand has been determined, the

appropriate action is taken and the resulting new set of times (A, B, C, D, E) is then subjected to the same set of tests. This process continues until all data points in the file have been tested. A summary of the cases and their subcases can be found in Figure 10.

If any of the subcases of case 4 is encountered while D is the last recorded time in a chute data file, an attempt is made to obtain a value for E from the following chute by looking at the inter-chute data file. Once all entries in the current chute file have been processed, the inter-chute data file is updated and saved on a disk for later use in processing other chutes. The corrected chute data file, with all included times in monotonically increasing order, is also saved on a disk. This chute file is now ready to be processed by the program that will supply finish times for all runners who did not have a recorded finish time, or whose finish time was removed because it was found to be in error by the program just described.

The algorithm for assigning finish times is relatively simple. Each chute file is processed from beginning to end, and each time a runner without a finish time is located, the program calculates the difference between the recorded finish times on either side of the untimed runner or runners. This difference is divided

	Situation	Action
Case 1.	A<=B<=C<=D	Advance A<--B, B<--C, C<--D, D<--E, get new E.
Case 2.	D<A<=B<=C	Leave A, B, C, advance D<--E, get new E.
Case 3.		
Subcase 1.	A<=B<=D<C	Advance A<--B, B<--D, C<--D, D<--E, get new E.
Subcase 2.	A<=B<=D<<C	Leave A, B, advance C<--D, D<--E, get new E.
Case 4.		
Subcase 1.	A<=D<B<=C<=E	Same as Case 2.
Subcase 2.	A<=D<B<=E<C	Leave A, B, advance C<--E, get new D, E.
Subcase 3.	A<=D<=E<B<C	Leave A, advance B<--D, C<--E, get new D, E.
Subcase 4.	A<=E<D<B<C	Leave A, B, C, get new D, E.

Figure 10. Summary of cases in time processing.

into the necessary number of equal intervals to approximate a finish time for each of the untimed runners who finished in that interval. If the interval is one second or less, all runners are given the faster time. In order to facilitate computations, all times are processed in seconds, and integer division is used to avoid fractional results.

Additional information is required when processing the end of the first chute file and the beginning and the end of each additional chute file. This information, which is stored in the inter-chute data file, includes:

1. The number of runners at the beginning of each file who do not have recorded finish times.

2. The first recorded finish time in the chute file.

3. The last recorded time in the chute file.

4. The number of runners at the end of the file for whom finish times are not recorded.

An obvious implication is that the times for any given chute cannot be completely filled in until data from the next chute has been stored in the inter-chute data file.

Race results can now be generated by a program which reads in one or more chute/runner data files and the runner master file which has been stored as a direct-access data set using the runner number as the key. This

program produces the necessary reports as separate output files. As each runner number in the chute data file is processed, the information in his or her master file records is used to determine what combination of the thirty-five output files will be written to. A separate program is used to generate the finish results from the family pairs competition. This program must add together the times of the paired individuals and store this combined time internally. Once times have been recorded for all finishing pairs, final results for the pairs competition can be reported. These two programs were generated by another member of the UTCC staff, and were written as Statistical Analysis System (SAS) applications programs.

CHAPTER VI

CHANGES AFTER 1983

Up until now we have been discussing work done in preparation for the 1983 race. Now we will describe problems that occurred in the processing of the 1983 race data, and changes that were made to the system of programs prior to the 1984 race.

As had been feared, one of the modules contained a programming error which had not been detected in the limited testing time before the 1983 race. This module was manually by-passed by editing the affected files. The edited files were then fed to the next module in the system, and computer processing was able to be resumed. What was not anticipated was that no results could be generated until the error had been identified and the files were manually changed. This was complicated by the fact that the inter-chute data file contained its information in binary form, and therefore, any changes to this data had to be done by computer program.

To insure that these problems did not re-occur after 1983, the way in which the chute data was processed was changed. As soon as chute data was received from the input program, it was put into a form which would allow generation of finish results which were correct as to

order, but included no finish times. This technique insures that finish results will be available even in the event of catastrophic problems in the processing of the time data. Further, the storage of finish times in binary form was abandoned in both the chute/runner files and the inter-chute data file. Although storing times in character format requires more space and processor time, it has the distinct advantage that the data can be changed manually by the use of a text editor.

With these changes in place, a test of the revised system was conducted. This test did not have to rely on hand generated test data, but instead was run by using the data generated by the 1983 race.

This testing showed that over fifty percent of the 1983 time data was in error, and investigation determined that the race personnel in charge of recording finish times had been instructed to write down as many runner numbers as possible and not to worry about accuracy. Because of the high error rate, there were occasions where a file would contain three consecutive errors which were monotonically increasing. Once these erroneous times had been accepted by the processing program as a base, all subsequent correct times in that file were rejected. Because of this, the program was changed so that for finish times of less than sixty minutes, a time which was

more than two minutes higher than the previous finish time was rejected. These times were chosen by examination of the 1983 data, as well as published race results from previous years. They were not coded as constants in the program; rather, they were fed to the program as parameters. By this means, if the times were found not to work with the 1984 data, the parameters could be changed without having to re-compile the program.

The final test of the revised system was performed on May 26, 1984. On that date a new record was set by the generation of all final race results by this system of programs within seven hours of the sounding of the starting gun.

Future plans include the generalization of the report writers in the system so that it can be used to process other races. The algorithms have now been proven to function correctly, and no further changes are anticipated.

VITA

Thomas C. Richmond was born in Milford, Delaware, on October 7, 1940. He attended Mount Pleasant High School in Wilmington, Delaware. During the eighteen years following his graduation in 1960, he pursued a bachelor's degree, first at Dakota Wesleyan University in Mitchell, South Dakota, and then at The University of Tennessee in Knoxville, where, in 1978, he received a Bachelor of Arts degree with majors in mathematics and computer science.

During that same period of time he also served four years in the United States Air Force as a Balistic Missile Analyst Technician and worked in other jobs for one and one half years as a maintenance electrician, for three years as a burglar alarm installation and maintenance technician, for two and one half years as an engineering technician, and for one year as an electronics technician.

Upon completion of a Bachelor of Arts degree, he accepted a position as a computer programming technician at the University of Tennessee Computing Center in Knoxville and began his studies toward a Master of Science degree in Computer Science.

The author is a member of the Knoxville Track Club, the Association for Computing Machinery, and continues to be employed by the University of Tennessee Computing Center as a systems programmer/analyst.