July 2014

Some Reflections on Computing and Computing Science at the University of Alberta

Keith Smillie

History … interprets the past to understand the present and confront the future … . P. D. James, *The Children of Men*.

Introduction

A digression

- NCR 102-A
- Stantec Zebra

Before the computer at the U of A

- Calculators
- Mathematics
- Physics
- Agriculture
- Engineering

The LGP-30

- Acquisition
- Description
- Programming
- Applications

The Computing Centre

The Department of Computing Science

- Formation
- Disputes
- Consolidation
- Peregrinations
- Teaching
- Research
- Three views from the 90s

Programming languages

- Introduction
- Machine language
- Fortran and BASIC
- Some other languages

The Department today

Epilogue

References

Appendix – Roster of academic staff

Introduction

The year 1964 may be considered a noteworthy one in the history of computing. IBM announced the System/360 "third-generation" line of computers and the completion of their seven-year development of the Sabre system for airline reservations. Dartmouth College in New Hampshire released the first version of BASIC, Beginner's All-Purpose Symbolic Instruction Code, with the first program being run at about 4:00 a.m. on May 1. And Douglas Engelbart of the Stanford Research Institute developed the computer mouse. Passing unnoticed amongst such events as these was the formation on April 1 of the Department of Computing Science at the University of Alberta in Edmonton, Alberta, Canada.

In this paper we shall look back over the Department as it celebrates its fiftieth year. The viewpoint will be that of one whose principal interest has been the introduction of programming langauges to beginning students and the application of these languages to the efficient and appealing solution of a variety problems in several areas especially statistics. Of course the hardware and management aspects of computing will not be forgotten.

Much of the material in this paper has appeared in earlier papers by the author and are listed in the References. We shall mention here only the three which have been published in one format or another. The first is *The Department of Computing Sciene: The First Twenty-Five Years* which was issued as Department of Computing Science Technical Report TR 91-01 in February 1991. The second is *Computing Science at the University of Alberta 1957 – 1993* which was intended to mark in a more formal manner and with many illustrations the first twenty-five years of the Department. The third is "Early computing at the University of Alberta and the introduction of the LGP-30.", *IEEE Annals of the History of Computing*, vol. 29, no. 1, 2007, pp. 65-73.

We shall begin, however, with a short digression to discuss two computing facilities that were in operation in the Province of Alberta before the University introduced its first computer in 1957. These are the NCR 102-A at R.C.A.F. Station, Cold Lake and the Stantec Zebra at the Defence Research Station at Suffield, both of which the author made use of before he came to the University of Alberta.

A digression

NCR 102-A

One of the first, if not the first, computer in Alberta was the NCR 102-A manufactured by the National Cash Register Company of Dayton, Ohio. It was originally referred to as the NCR CRC 102-A as it had been designed by the Computer Research Corporation. Sixteen 102-A computers were produced altogether, and two were installed in Canada. The first was at A. V. Roe (Canada) Limited, an aircraft company in Malton, Ontario adjacent to what is now Pearson Airport The company will be remembered for designing, building and flying a supersonic jet fighter, the Avro Arrow, which was scrapped before it went into production. The second 102-A was in Alberta at RCAF Station Cold Lake, (now Canadian Forces Base, Cold Lake), where it was used for some of the calculations involved in the design and testing of the Velvet Glove air-to-air guided missile which like the Arrow was another ill-fated Canadian endeavour.

The 102-A consisted of the "computer proper", as it was termed in the programming manual, which was contained in a large cabinet about six feet high, five and a half feet deep and three feet wide, and a console with a Flexowriter (a modified electric typewriter) with a 10-character-per-second paper tape reader and punch and a small console. A magnetic tape unit and a card reader and punch were optional. The price was about $82,000 for the basic system and $16,000 for the tape unit. Specifications for the computer stated that it required 7.7 kilowatts of power, occupied 250 square feet, and had 800 tubes of twelve different types and 8000 crystal diodes. Air conditioning and a separate power supply were required.

The magnetic drum memory had a capacity of 1024 42-bit words and an additional "high-speed" memory of eight words. (The capacity of the main memory measured in gigabytes, the units customarily used today to measure memory size, was 0.0000054 GB.) The internal number system was binary while input and output was in octal, or in decimal if appropriate conversion routines were available. All arithmetic operations were in fixed point, and it was the responsibility of the programmer to keep track of the decimal point by appropriate "scaling" operations. Addition and subtraction times varied from 7 to 20 milliseconds depending upon drum access, and multiplication and division times were from 25 to 38.5 milliseconds.

All programming was done in machine language using a three-address code with 27 instructions with the format `op m1 m2 m3` where `op` is the operation code and `m1`, `m2` and `m3` are memory addresses. The computer came without any program library or other software. Indeed the word *software* did not enter the English language until the early 1960s. All supporting programs had to be written by the user. There does not appear to have been any NCR 102 Users' Group. The following is a listing of the first page of a three-page program to calculate the arithmetic mean written by the author and Bill Adams and dated April 8, 1956:

```
0100  ex32 0140 0142 0103   Restore pick-up command
0101  ex32 0140 0142 0105   Restore put away command
0102  ad35 0103 0141 2006   Set up comparer
0103  ad35 [f]  2100 2000   Pick up decimal number
```

```
0104  tm34 3000 2100 1731   Unconditional skip to d.-b. conversion
0105  dr23 2004 0146 [f]    Scale and put away binary number
0106  ad35 0103 0133 0103   Increase pick-up command
0107  ad35 0105 0145 0105   Increase put-away command
0110  tm34 2006 0103 0103   All numbers converted?
0111  ex32 0140 0142 0114   Restore pick-up command
0112  ad35 0114 0141 2006   Set up comparer
0113  ad35 2100 2100 2007   Clear accumulator
0114  ad35 [f]  2007 2007   2^-u x_i + 2^-u S_i = 2^-u S_i
0115  to37 2007 3000 0134   Overflow?
0116  ad35 0114 0144 0114   Increase pick-UP command
0117  tm34 2006 0114 0114   All numbers added?
```

Stantec Zebra

The Research Branch of the Department of Agriculture maintained a Research Station at Lethbridge, Alberta for research on problems peculiar to the Great Plains area in Canada. In the mid-1959s some of the research staff began to use the Stantec Zebra computer at the Defence Research Station at Suffield, Alberta which is about 130 miles east of Lethbridge. This was encouraged by the enthusiasm of the Director at Lethbridge, Dr. Thomas H. Anstey, who not only used the computer in his own work but would write programs for members of his staff.

The Stantec Zebra was manufactured by the Standard Telephones and Cables Limited at its Information Processing Division in Monmouthshire, Wales and was marketed in Canada by the company's Montreal offices. The computer cabinet was 6.5 feet by 5.5 feet by 2 feet; there was also a console for the operator. There was a drum memory with a capacity of 8192 33-bit words. Basic input and output was paper tape with optional punched card and high-speed paper tape and magnetic tape input and output. Addition and subtraction times were given as 312 microseconds, and multiplication and division times were 11 and 35 milliseconds, respectively. There were two accumulator registers and fourteen other registers.

Programming in machine language, called Normal Code, was extremely complicated as indicated by the following paragraph taken from a descriptive brochure:

> The structure of the Normal (machine) Code is based on a novel idea. Single letters specify basic operations such as add, test, store; but there are 15 such letters (called function digits) and these may be used in any combination so that the programmer may construct thousands of different instructions. It is possible to instruct the machine to add, transfer, shift, modify and test "all at the same time", thus making the effective speed of operation of the computer greater than the intrinsic electronic speed would suggest.

If the above description of Normal Code was not sufficient to discourage a prospective programmer, a look at some "typical" instructions certainly would. The following example is representative:

X1856LBC3 Take your next instruction from main store location 1856. Left shift the A accumulator. (Here, although the L digit indicates a shift of both accumulators, the BC digits indicate that the B accumulator is to be cleared. However, as register 3 is equivalent to the B accumulator, the unshifted contents of the B accumulator are replaced after shifting and clearing has taken place.)

Theoretically there was a total of 2^{15} or 32768 of these instructions. Fortunately there was Simple Code. Simple Code, described in Ord-Smith (1960), was an interpretive language using floating-point arithmetic and had facilities for relative addressing and automatic counting in loops. The memory consisted of an instruction store and a number store each consisting of 1490 locations, although numbers could be stored in the instruction store and instructions in the number store. There was an accumulator register, an accumulative multiplicative register, and six special registers for counting and order modification. Simple Code was described in a brochure as being the "simplest available to any British computer". The following is a Simple Code program to find the sum of 20 numbers stored in locations 50, 52, …, 68 and to store the sum in location 1:

```
T     Clear accumulator
+020  Set for 20 cycles
AR50  Add number to sum
T1    Store sum in location 1
```

There were only two models of the Stantec Zebra in Canada, the one at Suffield and the other at the Company's offices in Montreal. Stantec's computing operations in Canada came to an abrupt end when there was a short circuit in the transformer in the Montreal computer. The Stantec Zebra at Suffield was eventually replaced by an IBM 1130. When the Zebra at Suffield was decommissioned, Tom Anstey wanted to arrange a wake. Unfortunately no one was interested.

Before the computer at the U of A

Calculators

Before the stored program computer became available, calculations were performed either by slide rule or by some make of mechanical or electromechanical desk calculator, often with the aid of mathematical tables.. Starting in the early 1970s these calculators were gradually replaced by the electronic pocket calculator, some of which were programmable, and soon they disappeared almost completely from offices and laboratories. Most of them were discarded and now only a few remain on display in various places. One small collection may be seen in a display case in the ground floor hallway of the Biological Sciences Building and may be viewed online at http://www.cs.ualberta.ca/~smillie/Calculators/Calculators.html .

We shall mention only two of the exhibits, one a bright yellow slide rule forty-two inches in length which formerly was displayed in the University Bookstore over the counter where slide rules were sold. The second is the Millionaire calculator which deserves its own paragraph.

The Millionaire was designed in the 1890s in Munich and manufactured in Zurich. It was a large machine measuring 25 inches by 12 inches by 6 inches and weighed more than 60 pounds. Between 1894 and 1935 a total of 4655 Millionaires were sold in Europe and the United States with government agencies being the largest customers. The first page of the instruction manual contained the following advertisement:

> Calculating machines of superior workmanship, embracing Expedition and Accuracy in reading results in the Four Rules of Arithmetic, <u>Economy in Time</u> and <u>Energy of the Operator</u>." On the following page we read that "The MILLIONAIRE Calculating machine Is the most efficient Calculating Machine in the world. Requires only one turn of the crank for each figure in the Multiplier.

Below a picture of the machine is shown the product

$$18{,}769{,}423 \times 23{,}769{,}814 = 446{,}145{,}693{,}597{,}322$$

followed by "Result obtained in 6 or 7 seconds, by only eight turns of the crank." Addition was performed by setting an indicator to "A" and then successively entering the numbers to be added on a keyboard and pressing the crank after each summand had been entered. The three other arithmetic operations were performed in a similar manner. The Millionaire became one of the most important machines for scientific calculations for about thirty years. It was this machine that was used by the astronomer Percival Lowell for the calculations that resulted eventually in the discovery of Pluto.

In the remainder of this section we shall describe the use of calculators in several departments of the University of Alberta with which the author was familiar.

Mathematics

An excellent account of mathematical and statistical computing in the Department of Mathematics is given in a short departmental history written by E. S. Keeping. Professor Keeping joined the Department of Mathematics in 1929 and retired in 1961, having been Head for the previous seven years. He continued to lecture for the following nine years, and maintained his association with the University for many more years. He died in 1984 in his eighty-ninth year.

An early set of mathematical tables used in the Department of Mathematics was Campbell's *Numerical Tables* prepared by Professor J. W. Campbell who came to the University in 1920 and was well-known for his work in astronomy and classical mechanics. The tables were divided into two parts, the first giving tables for the common logarithm, square, cube and reciprocal, the circular functions, and a short table of exponential and hyperbolic functions, and the second giving an extensive table of hyperbolic functions which, according to Professor Keeping, Professor Campbell "himself calculated on a hand machine". It was printed locally, first appearing in 1929, and was reprinted in 1946. At some subsequent date *Knott's Four-Figure Mathematical Tables*, first published in 1900, was introduced into the Department and was used until the mid-1960s. These tables were loaned to students during examinations, and had the cover overprinted in red with the following caveat:

THIS BOOK IS THE PROPERTY OF

THE UNIVERSITY OF ALBERTA

It must not be defaced or mutilated

Private possession may be cause for action
by the University authorities

Eventually, these tables were remaindered in the University Bookstore for twenty-five cents each.

Professor Keeping was well aware of the advantages of the use of mechanical desk calculators in statistical calculations, and has the following remarks about their use in the Department:

> Equipment is another item which has increased significantly in cost in recent years. For a long time the laboratories in elementary statistics used small Monroe calculators [and also the Multo calculator] which were cranked by hand, and there were only one or two electric desk-calculators in the whole department, of rather old-fashioned type. By 1962 many of the old calculators were almost worn out, and some newer types of hand calculators were purchased. [These were probably the Swedish-made Odhner calculators which weighed thirteen pounds and which accomplished multiplication and division by addition and subtraction with repeated shifting.] A little later some improved electric calculators, such as the Friden square-root type, came along and then some desk electronic models were purchased.

The only other reference that Professor Keeping makes to equipment prior to the late 1950s is the following: “In 1955 the department purchased an electric kettle to be used in preparing tea or coffee before the [Mathematics] colloquium, and the staff were assessed one dollar each to pay for it.”

Professor Keeping was the co-author of a two-volume text on mathematical statistics in which a large number of carefully worked numerical examples were a prominent feature. In the introductory chapter of the first volume there is a short section entitled "Calculating Machines" in which we read the following: "A calculating machine is constructed to add and subtract. By means of continued addition or subtraction, operations involving multiplication, division, and square root can also be performed with great speed."

Later in the same section he gives the details of performing repetitive calculations using as an example finding the values of "12 + 6*x* for *x* = 5, 7, 15, 12, etc."

Finally, it is interesting to note that Professor Keeping's lectures in the Faculty of Engineering had an influence on the design of the pocket calculator. Two of his former students who were subsequently employed by Hewlett-Packard remembered his lectures on reverse Polish notation and incorporated this feature into the company's calculators.

Physics

The first use of an electronic computer on the campus was in the Department of Physics which in May 1957 established a link with the FERUT (Ferranti University of Toronto) computer at the University of Toronto. An account of this work was reported in the Summer 1957 issue of *The New Trail*, the University of Alberta Alumni magazine, in a short article entitled "Electronic brain aids University research". It is of interest to quote from this article as an example of how the computer was presented to the public:

> Any problem that can be reduced to a numerical analysis and requires a numerical answer can now be answered in minutes on the U. of A. campus through the medium of teletype and FERUT, a high-speed digital electronic computer housed in the computation centre at the University of Toronto.
>
> A direct-line, teletype communication system with FERUT has been placed with the University through the courtesy of the Canadian National Telegraphs, on an experimental basis. Equipment is located in the Physics Department laboratory, basement floor, Arts Building.
>
> This new type of "correspondence course" has students and physics professors at the U. of A. preparing problems on teletype ticker tape every week. Then, on Thursdays at 5:00 p.m., a direct line with the computation centre in Toronto is cleared for University use. Transmission time is made available by the Canadian National Telegraphs and time on FERUT is paid for by the National Research Council.
>
> The mechanics of getting a solution to any mechanical problem are comparatively simple. Previously prepared tapes of problems are fed through the teletype machine on the campus and identical tapes are punched instantaneously in the computation centre in Toronto. FERUT is fed the problem tapes which are then processed and FERUT feeds back an answer tape in typed numbers in tabular form. The answer tape is fed through a Toronto teletype which activates the keys of the machine on the campus. There are many particular advantages of the new hook-up. Ambitious problems in the fields of physics, mathematics, engineering, statistics, etc., can now be solved on the campus in a matter of minutes, that otherwise would have taken months, perhaps

years, of hard labour on a desk calculator. Problems of a numerical nature can be solved much more accurately than previously, and through the teletype medium, operators at both ends can converse.

It is interesting to compare the above account with some recollections made many years later by Don Betts, a theoretical physicist at the University of Alberta who was very much involved in using FERUT and later the LGP-30:

> During the 1956/57 academic year [the theoretical physicists] decided to try to gain access to FERUT. Don Scott, who was then Assistant Head of the Department, was enthusiastically supportive of our goal and was instrumental in completing the necessary arrangements.
>
> The director of FERUT, Dr. C. C. Gotlieb, agreed to let us have access to it, Canadian National Telegraphs agreed to lend us a teletype machine and to provide free use of a telegraph line from Edmonton to Toronto one evening a week, and the National Research Council provided for the cost of time on FERUT. By April 1957 all arrangements were in place, and I was sent to Toronto for two weeks to learn to write programs and prepare input tapes for the machine. I was helped by Dr. B. H. "Trixie" Worsley and Miss Dorothy Goulding of the Computation Centre in learning Transcode, a high-level language for FERUT which was written at the University of Toronto. I did not learn the much more difficult machine-language programming. ...
>
> The teletype machine was established in a glorified closet in the basement of the Arts Building and the teletype link was first used on May 9, 1957. The official opening of the facility, which included a ribbon cutting, was attended by the President of the National Research Council, Dr. E. W. R. Stacie, the Director of the Western Region of Canadian National Railways, the President of the University of Alberta, Dr. Andrew Stewart, representatives of the media, and by all directly involved.
>
> Throughout the summer we and our students would prepare programs and code teletype tapes through the week for transmission on Tuesday evenings. [The article in the *New Trail* cited previously gave the day as Thursday.] The two copies would then be run through a mechanical comparator at the Toronto end. Any discrepancies were fixed up by the exchange of teletype messages. The system worked tolerably well except when there was a thunderstorm anywhere between Edmonton and Toronto!

Agriculture

The Faculty of Agriculture has always required calculating machines for the analysis of experimental data. In Canada an excellent example of the importance of proper statistical designs and efficient calculating procedures was given by Cyril H. Goulden who worked for the then Canada Department of

Agriculture, first at the Dominion Rust Research Laboratory in Winnipeg, Manitoba and later at the Central Experimental Farm in Ottawa. In the 1952 edition of his *Methods of Statistical Analysis*, first published in 1939, he makes the following remarks: "In the development of each procedure an attempt has been made to form a uniform method. After general statements the algebraic development is given, and then is followed by a completely worked-out example." The book contains a very large number of numerical examples for everything from the simplest statistical procedures to the very complex lattice designs used in agricultural work. From the late 1950s Goulden took a keen interest in electronic computers, encouraging their use in the Department of Agriculture and even speculating if the Department should build its own computer.

For many years the biometrics course in the Faculty of Agriculture at the University of Alberta used a statistics manual prepared by Professor L. P. V. Johnson. This carefully written document with its numerous worked examples clearly follows the format adopted by Goulden. Indeed the first edition of Goulden's text is one of the 16 papers and books cited in the References.

The Department of Soil Science has carried out fertilizer trials since the early 1920s. However, it was not until 1956 that statistical designs such as randomized blocks and balanced lattices were introduced. For the first two years desk calculators were used for the calculations. One of the machines used was a Monromatic which allowed the operator to accumulate at one time the sums of squares of two sets of observations and their cross products. This was accomplished by entering pairs of observations, one pair at a time with one datum on the left side of the keyboard and the other on the right. The cumulative sums of squares would be displayed on the left and right sides of the accumulator dial with the sums of cross products appearing between them. Also a counter mounted on one side of the calculator would show the number of pairs of observations involved in these sums.

One final computational example from the Faculty of Agriculture is of interest because it forms a link between the use of hand calculators and the LGP-30. It is a report entitled "On the meaning of marks" written by Brian Hocking, Head of the Department of Entomology, and dated June 4, 1958. It gives the results of an analysis of the variations in the distribution of the final marks in some selected courses at the University of Alberta, and makes recommendations for removing inequities caused by these variations. In the Procedures section of the report we read that "The first calculations were done on a desk calculator; when a digital computer [the LGP-30] became available, this was used to complete the work." This report is written in a style evocative of a gentler age at the University, and opens delightfully with the words "For my personal guidance in trying to deal justly with my own classes, I set out to determine, during a few of the less crowded hours last summer, ...".

Engineering

The experiences of other departments and faculties would appear to be similar to those of Mathematics and Agriculture with calculations, many of them statistical, being performed on Marchant, Monroe and Friden calculators. It is unfortunate that few records appear to exist of this work. However, a few remarks on engineering calculations are given in a history of the Faculty of Engineering:

> Their education was highly structured, free of options, very basic and without computers. True, there were the standard six-place log tables and one volume of "Vega" eight-place tables and ten-inch slide rules. The Faculty owned the computer of that age, a 30-foot spiral slide rule which gave five figures at the low end and four at the upper end but it took a hefty pair of arms to muscle it about.

The LGP-30

Acquisition

In May 1957 the University President, Dr. Andrew Stewart, appointed a "Committee on Electronic Equipment" to make an assessment of computing needs at the University. The Committee's mandate was fourfold: To determine what electronic equipment the University should acquire, to determine the most efficient method of servicing this equipment, to arrange for interdepartmental cooperation in its use, and to investigate the extent to which the University should become the provincial centre for the installation of large-scale electronic equipment. The Committee was chaired by Professor Donald B. Scott who had joined the Department of Physics as a Sessional Lecturer in 1940 and included Professor Keeping who was mentioned in the previous section. In July the Committee recommended unanimously that the University purchase an LGP-30 computer from the Royal McBee Corporation of Port Chester, New York at a price of forty thousand dollars, which is about 300,000 dollars today. In September the Committee's recommendations were approved by the Board of Governors, the order for the computer was placed, and the computer was installed the following month in the Department of Physics on the main floor of the Arts Building. The University of Alberta was the third Canadian university, after the University of Toronto and the University of British Columbia, to acquire a computer.

Description

The LGP-30 was termed a "desk computer" not because it could be placed on a desk but because it was the size of a desk. It was 26 inches deep, 33 inches high and 44 inches long, and weighed about 800 pounds. One user described it as "a great hunk of a machine – about the size of a deep freeze". The original installation consisted of the computer with a Flexowriter console with a mechanical paper tape reader and punch), a photoelectric paper tape reader and a mechanical paper tape punch, and an additional Flexowriter for the preparation of program and data tapes. The main memory was a magnetic drum with a capacity of 4096 32-bit words. The internal operation was binary so that all data, assumed to be less than unity in absolute value, had to be converted from decimal to binary on input and from binary to decimal on output. The clock speed was 120 kilocycles giving addition and multiplication times of 8750 and 24,000 microseconds, respectively.

There were very few reference manuals for the LGP-30. Indeed one user remarked some years later that he couldn't recall ever having seen a manual. There was a glossy illustrated printed *Operations Manual* describing the LGP-30 logical structure, the console switches and auxiliary equipment. There were also a *Programming Manual*, *Programming Class Notes*, and a *Subroutine Manual Coding Sheets* manual giving programs for input and other utility routines, and a set of floating-point arithmetic routines. Published somewhat later than these manuals was one for the ACT algebraic compiler which became available too late to have received much use on campus.

Programming

Programming for the LGP-30 was done in machine language using an order code which had sixteen instructions. Each instruction had a one-address format giving the operation to be performed and the address, i.e., location in memory, of the single operand. Arithmetic in the LGP-30 was performed in the one-word Accumulator Register with one of the operands stored in the Accumulator and the other in memory. As a very simple example we shall consider a program to find the sum of two arbitrary numbers assumed to be stored in memory as binary numbers with the computed binary sum to be also stored in memory. A program for finding the sum may be stated in words as follows:

> *Bring* the first number to the Accumulator; *Add* the second number, and retain the sum in the Accumulator replacing the value of the operand which was previously there; store the sum in a specified word in memory and *Hold* the sum in the Accumulator; *Stop*.

The corresponding annotated machine-language program is as follows:

```
0100  b3750 Bring first number to Accumulator
0101  a3751 Add second number
0102  h0104 Store sum
0103  z0000 Stop
0104        Sum
```

The program is stored in memory locations `0100`, `0101`, `0102` and `0103`, the two numbers to be summed in locations `3750` and `3751`, and the sum in location `0104`.

Suppose now as an example that the numbers to be added are 5 and 7.5, which have the sum 12.5. To ensure that the numbers as well as their sum when represented in memory are less than unity in absolute value we shall introduce a "scale factor" of 2^4 or 16 so that the two numbers to be summed become $5 \div 2^4$ or 0.3125 and $7.5 \div 2^4$ or 0.46875 whose sum is 0.78125 which when multiplied by the 2^4 gives the required decimal sum of 12.5. Of course the computations within the LGP-30 would be done on the binary equivalents of these numbers. (Conversion of data from decimal to binary on input and from binary to decimal on output was handled by subroutines from the small program library supplied with the LGP-30.) It might be noted that the appropriate scaling of initial data and intermediate and final results in a computation was often one of the most difficult, and frustrating, problems in machine-language programming.

Applications

The use of the LGP-30 by Brian Hocking of the Department Entompplogy to complete the calculations required in his study of students' marks has already been referred to. The Department of Soil Science was another very early user of the computer in the Faculty of Agriculture. Their use of desk calculators for the analysis of data has been described earlier. However, the 1958 data were analyzed with an LGP-30 program for a two-way randomized block giving the treatment totals, the error mean square and the F-ratio for treatments. Another study on the effect of cropping systems and fertilizers on yields acknowledged the usefulness of the LGP-30 for the analysis of the experimental results.

Another example of the use of the LGP-30 is provided by a paper given by Fenton MacHardy of the Department of Mechanical Engineering at Congrès International Technique du Machinisme Agricole held in Paris in 1961. One of the examples given in the paper involved finding the optimal allocation of tractor resources for preparing and planting a given area of land. Computationally the problem involved the solution of a linear programming problem with ten variables including slack and surplus variables and four constraints and required about two minutes on the LGP-30.

Other departments using the LGP-30 included Animal Science, Chemical Engineering, Chemistry, Educational Psychology, Electrical Engineering, Geology, Mathematics, Physiology, Plant Science, Political Economy, and Psychology, as well as the Alberta Research Council which was then located on the campus. An article appearing in *The Edmonton Journal* on November 22, 1960 said that the University's computer, which was referred to as a "30 computer", was being used twenty-four hours a day and seven days a week. Furthermore, it stated that "the University expects shortly to launch an extensive computer training program for students, and that arrangements are being made to obtain a faster unit for the newly established centre".

The Computing Centre

Initially the administration and use of the LGP-30 was under the Committee on Electronic Equipment, but on November 1, 1960 a separate Computing Centre with Don Scott as Director was formed with a mandate to offer computing services to the entire University. The Centre was run on an open-shop basis with users doing their own programming with some help from one or another of the members of the Committee. Soon however several students were hired to provide temporary support during the summer. The first full-time employee of the Computing Centre was Ursula Bielenstein – later Ursula Maydell - who was hired in 1960. She resigned a year later to begin an MSc degree in Mathematics, and in 1965 became a member of the academic staff of the Department of Computing Science.

In late 1960 the Committee on Electronic Equipment began to consider ways of meeting the increasing computational needs on campus, and recommended that the IBM 1620 Data Processing System be rented at a cost of 3126 dollars a month with the National Research Council providing partial support. The Board of Governors approved the Committee's recommendation, and the IBM 1620 arrived in May 1961. The recommendation to acquire this computer was the last official action of the Committee on Electronic Equipment which then ceased to function.

The IBM 1620 Data Processing System had 20 000 characters of core storage. This machine was initially intended primarily for research while the LGP-30 was to be kept for teaching. Later in the year it was replaced by a new 1620 with a card reader and punch and the following July three IBM 7330 magnetic tape units were added. The original 1620 system was sent to the Alberta Research Council which had been using about forty percent of the time available. The LGP-30 remained in use until the middle of 1963.

After the arrival of the 1620 the Centre continued to be operated on an open-shop basis with users running their own programs. However, as the demand increased this form of operation was seen to be unsatisfactory and a change to Computing Centre staff operating the equipment was begun. On April 1, 1963 a completely closed-shop operation was implemented.

Before the formation of the Department of Computing Science in 1964 several academic courses were given through the Department of Mathematics. Courses in numerical analysis and in computing had been given for several years by John McNamee. Mathematics 460, "Numerical analysis", was listed in the 1959/60 Calendar and covered topics such as vector and matrix equations and solution of eigenvalues, finite differences, interpolation, differential and integral operators, initial value and boundary value problems. Mathematics 640, "Advanced Numerical Analysis", was listed the following year and included such topics as eigenvalues and eigenvectors of large-order matrices, quadrature, partial differential equations, linear programming and game theory. The first course devoted to computers rather than numerical analysis was Mathematics 641, "Automatic digital computers and programming", which was also given by John McNamee and which appeared for the first time in the 1961/62 Calendar. The course description was in part as follows: “Scientific applications of accounting machines, and development of automatic computers. Storage systems, arithmetic units, input and output mechanisms. Arithmetic in general radix. Simple algebraic logic. Logical systems of control (one-address, three-address, etc.). Flow diagrams. Construction and use of library routines. Codes of principal types of machines, with detailed knowledge of one machine ... .” From 1962 to 1964 seven students working on computational topics were awarded MSc degrees in Numerical Analysis through the Department of Mathematics. Also during the 1963/64 academic year several courses were given by the staff of the Computing Centre to students in Mathematics, Engineering, and Commerce. Finally in the 1963/64 academic year Statistics 256 was introduced; initially it was an equal blend of probability, statistics and numerical analysis.

With the formation of the Department of Computing Science in 1964 there was confusion about the roles played by the Department and the Computing Centre, especially as Don Scott was Head of the first and Director of the second.. To make a long story short in 1970 the Computing Centre became a separate organization with a Director reporting to the Vice-President (Academic) with the Department of Computing Science being only one of the many users of its facilities.

The Department of Computing Science

Formation

The Department of Computing Science at the University of Alberta became a department in the Faculty of Science on April 1, 1964. In discussions leading up to the Department’s formation some members of established departments in the Faculty asked if computing needs could be met within the Department of Mathematics and others asked if reputable universities such as Harvard had their own computing departments. However, the skeptics were finally won over - or decided to not voice further objections - possibly taking some satisfaction in noting that April 1 is April Fools’ Day.

It might be noted that the choice of the name “Computing Science” rather than “Computer Science” was deliberate to emphasize *computing* rather than *computers* even though exactly what constituted “computing” was probably not appreciated then, and may not be even today. Another explanation attributed the name to a typographical error. Alhough both names appear in the correspondence regarding the formation of the Department, the first explanation is the more reasonable one.

In addition to Don Scott who was the first Head of the Department of Computing Science there were four additional academic staff. These were Bob Julius who had just completed a PhD in Physics at the

University of Alberta and who was made Assistant Head, Bill McMinn who had an engineering degree from the University of Toronto and who had worked for IBM, Bill Adams who had just completed an MSc in the Department of Mathematics, and Keith Smillie who had several years of government and industrial experience in Ottawa and who had joined the Computing Centre and the Department of Mathematics the previous year.

The first few years were spent in developing both an undergraduate and a graduate program in Computing Science, attracting new faculty, and for the existing faculty establishing their own teaching and research careers. The first introductory course for Computing Science students was Computing Science 310, or CMPUT 310 to give the Calendar designation, with the following description: “Logical structure of computers; instructions; algorithms and programs; language of computer programming; conventional computers; assemblers and compilers.” The 1966/67 Calendar listed eight Computing Science courses, and the following year there were fifteen courses listed. The 1970/71 Calendar gave forty courses, of which fourteen were undergraduate, six were graduate, and the remaining twenty were courses that could be taken by either graduate or senior undergraduate students. The first MSc degree in the Department was awarded in 1964, the first BSc degree in 1968, and the first four PhD dgrees in 1973.

We shall close this section by mentioning again the five founding members of the Department. Don Scott stepped down as Head in 1971 and shortly afterwards was appointed University Ombudsman, a position which he held until shortly before his death in June 1975 in his sixty-second year. Bob Julius left in 1967 to accept a position at the University of Toronto where he stayed for a few years before moving to Israel. Bill McMinn resigned in 1966 to devote his time to business interests; he died quite unexpectedly in 1983. Bill Adams, apart from one year in the private sector in the United States, and Keith Smillie stayed in the Department until their retirement on August 31, 1992.

Disputes

During the early years of the Department there were many lively debates about the content of courses, the suitability of the growing number of programming languages for instructional purposes, and various administrative matters. We shall mention a few of them here if only to illustrate both how little some debates change with time and how trivial some of them can be.

One problem that emerged very soon after the formation of the Department was concerned with the language to be used in an introductory programming course. There were those who believed strongly that one should start with a high-level language such as Fortran while others believed with equal fervour that a simple machine language should be the first language. Indeed, a machine-language and assembly-language simulator MENTOR/MENTORSAP had been developed in the Department for precisely this purpose. This topic was debated at departmental meetings, faculty offices and undoubtedly in the hallways for some months. Differences over this problem between two of the principal protagonists were finally settled very amicably over lunch one noon in the Faculty Club. A very similar situation emerged many years later when students in one course given to students in the Faculty of Engineering were first taught Pascal so that they would understand "programming" before being introduced to MATLAB. Indeed the question of the most appropriate first programming language is still being discussed in the Department and undoubtedly will continue to be for years to come.

A somewhat similar discussion arose over the APL language and its place in a computing science curriculum. Since Kenneth Iverson, the originator of APL, was raised in Alberta, it was relatively easy for the Department to have him visit and give a lecture on his work. Although APL was used enthusiastically by a few members of the Department, the language was largely ignored by most faculty who regarded it, if they thought of it at all, as "that language with all the funny symbols".

A more serious problem arose over the PL/I language. PL/I was developed within IBM in the 1960s as a general-purpose language for scientific, commercial and special-purpose applications on their System/360 computers. As far as can be recalled, its use in any course in the Department was never seriously considered. However in the mid-1970s the Dean of Science received a complaint from the local offices of a multinational cooperation forwarded to him by the Chancellor of the University that we were being negligent in not teaching PL/I which they were using in their work. One polite, but firm, letter from the Chairman of Computing Science to the Dean of Science saying that it was not our purpose to teach what could be immediately useful in the marketplace terminated our involvement in the discussion.

We shall conclude by mentioning a couple of arguments which are quite trivial, even ludicrous, in retrospect and probably should have been considered so when they arose. One concerned the shortage of space for IBM keypunches, and during one lively discussion the suggestion was made to move some of the keypunches into Don Scott's large office. Fortunately, reason prevailed and Don Scott retained the use of his entire office. Methods for the distribution and collection of the fragile and expensive type elements, the so-called "golf-ball elements", for the IBM 2741 terminals produced another spirited discussion, the details of which have been long forgotten.

Consolidation

The separation of the Department of Computing Science from the Computing Centre in 1970 resulted in the transfer of all of the computing hardware and most of the support staff to the Computing Centre. Although the Department had access to the Centre's equipment for teaching and research purposes, it was imperative to recruit staff, both academic and non-academic, to accommodate the increasing number of students enrolling in computing courses and to acquire equipment for handling the specialized research needs of the Department. However, before considering both of these topics, we shall make a few remarks about how the Computing Centre handled the processing of programs in the 1970s.

In the 1960s and 1970s input of programs and data was by means of punched cards. In order that student programs be processed efficiently and quickly the Computing Centre developed a "Student-oriented Batch Facility", with the acronym "SOBF" which was abbreviated of course to "SOB", Students in a computing course were issued a number of "SOB tickets", each valid for one run. Those who used up their allotment of tickets could purchase "SOB balls" for a nickel each made of translucent plastic and dispensed by a gum-ball machine which had been bought at an auction. The SOB Facility was in a large room on the third floor of the General Services Building where there was sufficient space to accommodate a number of keypunches. The SOB Facility was used very heavily with up to one hundred persons out of the twenty-seven hundred possible users waiting to submit program decks or pick up output at any one time. The continued upgrading of the facilities in the Computing Centre with time-sharing capabilities and the use of terminals for remote of programs and data entry resulted in the SOB Facility being discontinued at the end of the decade.

During 1970s the Department of Computing Science acquired over a dozen minicomputers ranging from very small machines such as the PDP-9 to the PDP-11/45 and the Nanodata QM-1. The PDP-11/45 was operated with the UNIX operating system, believed to be the first use of UNIX outside of AT&T. For many persons the PDP-11/45 was the most significant machine in the Department, and when it was finally switched off the Hour Meter registered 100,000 hours of use. In the 1980s the PDP computers were replaced with the VAX computer, four VAX 1/780s and one VAX 11/730. Other computers obtained at this time were two Sun-1 workstations and a MIPS M/1000.

Another computer which was used for teaching and research was the IBM 5100 which supported both APL and BASIC. The 5100 was a small computer by the standards of the mid-1970s, measuring 17.5 inches by 24 inches by 8 inches and weighing 48 pounds. It could be accommodated on a small wheeled cart, with a printer on the lower shelf, for convenient movement between classroom and faculty office. (We might note the first IBM personal computers, the IBM PC and IBM XT, had model numbers 5150 and 5160.)

Very few faculty were recruited in the 1970s to handle the dramatic increase in the number of students, both those working towards degrees in the Department and those from other departments taking service courses. Fortunately in the early 1980s the Department was able to convince the University administration of the dire need for more resources and substantially increased funding for new faculty and support staff. This was indeed fortunate as the mid-1980s saw the development of the microcomputer and the resultant computer literacy movement – one of several less polite terms might be more accurate - which put demands on the Department met only by a hastily assembled group of Sessional Lecturers almost all of whom performed in an exemplary manner.

The continued increase in staff and equipment in the Department resulted in the almost continual need for more space which we shall discuss briefly in the next section.

Peregrinations

Electronic computing at the University of Alberta had an inauspicious beginning, as was noted earlier, with a terminal giving a connection to the University of Toronto FERUT located in a closet, albeit a "glorified" one, in the basement of the Arts Building. The University's first computer, the LGP-30, was also installed initially on the ground floor of the Arts Building, presumably in a room larger than a closet. It is of interest to discuss briefly the various locations where the Department of Computing Science and its predecessors have been located on the campus.

The LGP-30 was soon moved for a short time to the Arts Building Annex, a small two-story brick building immediately to the north and west of the Arts Building. This building, which no longer exists, is worth a short digression if only to help preserve some memory of it. It was built during the Second World War for testing aviation gasoline, and was known informally as the "Gas Lab". In 1954 the testing work was moved to the newly constructed Alberta Research Council laboratories on the campus, and the building was then occupied by the Faculty of Arts and named the Arts Building Annex. As one of its new uses was for housing mice used in experimental work by the Department of Psychology, it was known to some as the "Mouse House". It was demolished during the summer of 1986 or 1987. All that remains are a few bricks which were taken as souvenirs from the hole left by the demolition crew before it was filled. There appear to be no photographs in existence, even in the University Archives.

In 1961 the Computing Centre and its equipment moved to the newly opened Physical Sciences Centre which housed the Departments of Physics, Chemistry, and Mathematics. (This building wass demolished in about 2010 to make way for the Centennial Centre for Interdisciplinary Science) In the summer of 1968 the Department moved to the newly opened General Services Building, a move which at the time was considered temporary until a more permanent home could be found. It soon outgrew its accommodation on the sixth floor and some of the Support Staff moved to the Central Academic Building. Also a few of the graduate students were accommodated in the Printing Services Building. In 1983 a few of the academic staff were given offices in the newly renovated Assiniboia Hall, one of the oldest buildings on the campus. This space was given up in 1988 when more space became available on the seventh floor of the General Services Building.

In May 1996 approval was given to move the Department to Athabasca Hall as space became available dependent on the relocation of the departments then occupying the building and for the construction of a modern Computing Science Centre to replace the Annex located behind the building. The move of faculty, staff and equipment was done over a period of a year or so and its completion was marked with an official opening of the Centre on June 13, 2001. Thus finally after almost forty years the Department of Computing Science had a permanent home in one most historic and beautiful buildings on the campus.

Teaching

The undergraduate curriculum changed very little during the 1970s, although a number of new courses were added, particularly at the beginning of the decade. However, there were fundamental changes made during the 1980s. Not only did the new faculty who came during this period introduce courses in their own specialties, but the Curriculum Committee was influenced by computer science programs in Canadian and American universities and by the curricula developed by the Association for Computing Machinery and the Institute for Electrical and Electronics Engineers. By the end of the decade the curriculum covered analysis of algorithms, artificial intelligence, compiler construction, computer graphics, computer organization, data base management, data structures, discrete mathematics and logic, file management, image processing, introductory programming, logical design, numerical methods, programming languages, simulation, switching theory, systems programming, and telecommunications.

For many years the Specialization Program consisted of four streams: Computer Design, Business Applications, Scientific Applications, and Software Design. Students graduating in the program had the area of specialization shown on their diplomas. However, in 1987 the accreditation study of the Honors and Specialization programs by the Canadian Information Processing Society suggested that these streams imposed too great a constraint on students' programs. On the recommendation of the Curriculum Committee the Department decided to abolish the streams and relax some of the requirements to give the students more flexibility in the selection of courses.

Research

In the early years of the Department there was informal cooperation between faculty with similar research interests but there were few formal working arrangements and faculty worked independently with their own students. Moreover there were few visible signs of the types of research being undetaken. Organization of the faculty into formal research groups began in the 1980s.

Many of the first MSc degrees awarded were on topics in numerical analysis. The first seven were designated “Master of Science in Numerical Analysis”, and of these six of the seven theses were in this subject while the seventh was in programming languages. However in the late 1960s and early 1970s there was an increasing number of degrees awarded for topics in the implementation and use of programming languages, computer graphics and operating systems. The first four PhD degrees were awarded in 1973. In this section we shall mention some of the research that was done in the first twenty or twenty-five years of the Department's existence.

The Department worked for several years with the Department of Computer and Information Science of the Harbin Shipbuilding Engineering Institute. In 1987 the two Departments signed a formal agreement to provide better cooperation and for the exchange of information, and several faculty made visits to Harbin.

In 1988 an agreement was reached between the University and the All-Union Research Institute for Systems Studies of the USSR Academy of Sciences for a four-year cooperative in artificial intelligence and data bases. The main areas of research were heuristic search methods, hierarchical data bases, and knowledge representation. There were several exchange visits between the Department and the Institute.

In the late 1960s and early 1970s Keith Smillie developed an APL statistical package called *STATPACK2* which in several incarnations enjoyed a modest international reputation for a surprising number of years. Indeed its author often remarked in the 1980s, and occasionally in the 1990s, that *STATPACK2* was "dead but refused to lie down".

In the early 1980s Wayne Davis together with Terry Caelli, then a member of the Department of Pschology and later a member of Computing Science, organized a series of monthly seminars in image processing and computer vision. From these modest beginnings evolved the Alberta Centre for Machine Intelligence and Robotics (ACMIR) to promote research in computer vision, intelligent systems, robotics and control, and integrated manufacturing. The Centre prepared a number of technical reports and organized seminars until it became inactive in 1988.

The most widely acclaimed early research in the Department was undoubtedly the work in computer chess and checkers led by Jonathan Schaeffer and Tony Marsland. At the World Computer Chess Championship, held in Edmonton during the 1980 National Conference of the Canadian Information Processing Society, Jonathan Schaeffer's Phoenix program ranked amongst the top ten programs competing. In addition his checkers program Chinook came second in the United States National Open Tournament in 1990 and thus became eligible to compete in the World Checkers Championship in London, England in 1992 where it lost to the reigning world champion by only two points.

Three views from the 90s

In the 1990s three well-illustrated and professionally printed brochures were prepared which give an excellent picture of the organization and work of the Department in its first twenty-five years. The first is *Computing Science at the University of Alberta 1957 – 1993* and has a text prepared by Keith Smillie. The second and third, published in 1992 and 1995, have identical titles, *Computing Science at the University of Alberta*, and have text contributed by the academic staff and sybsequently compiled and edited by other members of the Department. We shall have a brief look at each of these publications.

The first is a slim publication 8½-by-11 inches in format and 26 pages in length giving a summary of the origins and growth of the Department, the acquisition of hardware and software, and highlights in teaching and research. The many illustrations include pictures of Department Chairs (designated by the archaic term “Chairmen”), many group pictures of faculty and students, four pictures illustrating the SOB Facility, and one page of childhood photographs of a few of the faculty and staff. Programming languages are illustrated by programs in various languages to calculate the sum of the first N positive integers by adding 1 and 2 and … rather than by using the well-known expression N(N+1)/2. The brochure ends with the following five Appendices: Chronology 1957 – 1992; Full-time academic staff; Current full-time Support staff; Some statistics [number of faculty, MSc, MSc, and PhD degrees fpr each year]; Courses 1992/1993.

The second and third brochures are slightly smaller in format but are 88 and 105 pages, respectively, in length. The cover of the 1992 publication is intended to illustrate “a web of computers” and “emphasizes the difference between functionality within the Web, as opposed to ensnarement by the Web. The goal is to become navigators rather than fall prey to i.t.” The cover of the 1995 publication shows a human figure produced by the Ballroom Dance Animation System which was developed in the Department. The text in each brochure gives very brief accounts of faculty and staff accompanied by a recent photograph, followed by a description of areas of research, undergraduate and graduate programs, colloquia speakers, research grants, and a brief historical overview..

Programming languages

Introduction

Programming languages have always played an important role in the curriculum. Several languages have been used in succession as a first programming language to be used in introductory courses and in later courses. In this section we shall discuss briefly some of these languages. In spite of .the MENTOR/MENTORSAP dispute mentioned earlier, we shall begin by considering a very simple machine langauge intended to illustrate some of the difficulties that may arise with machine-langage programming. We shall then discuss some of the langauges that have been used in the Department starting of course with Fortran.

We shall use one very simple numerical example throughout the discussion to help illustrate the language features.The example is to find the total amount spent on a shopping trip for groceries in which the following items were purchased: Tea $2.50; 2% milk $3.69; Skim milk $3.69; Bread $3.49; Bananas $1.27; Cob of corn $0.89; Broccoli $1.16; Carrots $0.84; Mandarin ora nges $7.99. The total amount spent was $25.52. These data may be represented by the following list of positive real numbers:

2.50 3.69 3.69 3.49 1,27 0.89 1.16 0.84 7.99

followed when necessary when considered as input to a program by an end-of-data marker of either 0.0 or an arbitrary negative number as appropriate.

The algorithm used to illustrate most, but not all, of the programming languages is the same as that used when doing the calculations with a pocket calculator: Initialize the appropriate register, and then add successivly each number to the sum until all numbers have been considered, and finally display the sum.

Machine langage

As there appears to be no extant documentation for MENTOR/MENTORSAP, we shall illustrate machine-language programming with a very simple language implented in a simulator, the internal details of whivh need not concern us. As an indication of the simple nature of the simulator, it was given the name “Simon” by its creator. (“Simon” was also the name of a large orange cat who sadly is no longer with us.) Simon is a one-address decimal computer with an order code of ten instructions numbered `0`, `1`, `2`, …,`9` and a memory of 100 words with addresses `00`, `01`, ...,, `99`. Input is by paper tape and output is by printer, both of which are of course simulated. Instructions have the format `ixx`, where `i` is the order code and `xx` is the address. In addition to the main memory there is a one-word accumulator for performing arithmetic operations. Programs are stored in consecutive memory locations with the instruction in location `00` being executed first. The instructions are as follows:

`000` Halt	Halt computation (Address portion irrelevant)
`1xx` Add	Add number in location `xx` to accumulator
`2xx` Subtract	Subtract number in location `xx` from accumulator
`3xx` Multiply	Multiply contents of accumulator by number in location `xx`
`4xx` Copy	Copy contents of address `xx` to accumulator
`5xx` Store	Store contents of accumulator in location `xx`
`6xx` Transfer	Take next instruction from location `xx`
`7xx` Trans. neg.	Take next instruction from location `xx` if contents of accumulator neg.
`8xx` Read	Read number from Tape to location `xx`
`9xx` Print	Print number in location `xx`

The program to find the total amount spent on the shopping trip is

```
412 212 512 813 413 709 112 512 603 912 000
```

which may be made comprehensible if presented in the following format with the address in memory in the first column, the instruction in the second column, and a comment in the remaining columns:

```
00 412      )
01 212      )   Set Total to 0
02 512      )
03 813      Read Price to locn. 13
04 413      Copy Price to Accumulator
05 709      Transfer to locn. 09 if Price < 0
06 112    )
07 512      )   Add Price to Total
08 603      Transfer to locn. 03
09 912      Print Total
10 000      Halt
11
12          Total
13
```

Fortran and BASIC

In 1954 IBM formed a small group headed by John Backus to develop an automatic programming system for the IBM 704 to produce efficient object code which woyld execute almost as fast as hand-generated object code. The language was called Fortran for “Formula Translating Language”. Emphasis was on efficiency of compilation and execution rather than on the design of the language. Indeed Backus later remarked that “we simply made up the language as we went along”. Fortran first became available for the IBM 705 in 1956 and later versions were released for other IBM computers. Fortran was adopted as a first programming language in introductory computing courses in the Department for several years in the mid-1960s. The following is a Fortran program for the sample problem:

```
 100  FORMAT (F8.2)
      TOTAL = 0.0
   3  READ(5,100) COST
      IF(COST) 1,1,2
   2 TOTAL = TOTAL + COST
      GO TO 3
   1 WRITE (6,100) TOTAL
     STOP
    END
```

BASIC, for Beginner's All-purpose Symboloic Instruction Code, was deveveloped at Dartmouth College, a small liberal arts college in Hanover, New Jersey. It was intended as an altrnative to Fortran for non-science students which would be "friendly", easy to learn and use, and convenient to access. As was mentioned in the Introduction the first BASIC program was run at about 4:00 a.m. on May 1, 1964. The following is a BASIC program for the sample problem:

```
100 BASIC PROGRAM
110 DATA 2.50 3.69 3.69 3.49 1,27 0.89 1.16 0.84 7.99 0.0
120 TOTAL = 0.0
130 READ COST
140 IF COST = 0.0 THEN 170
150 TOTAL = TOTAL + COST
160 GOTO 130
170 PRINT TOTAL
180 STOP
190 END
```

The BASIC language has been used on occasion in some intoductory courses for students who are not specializing in Computing S cience.

Some other languages

Fashions in programming languages tend to come and go and this has been reflected in the first langages used in introductory computing courses in the Department. For several years Algol W, derived from the European Algol language and implemented at Stanford University, was used as was Pascal originally developed by Niklaus Wirth of the Swiss Federal Institute of Technology in Zurich. Other languages were C, Java and Perl. Presently the first language for Computing Science students is Python. The following is a Python program for the sample problem:

```
count = 0
userinput = None
while (userinput != 0):
    userinput = input()
    count += userinput
 print count
```

As has been mentioned previously, the array language APL was used by only a very few members of the Department and is not used at all today. Incidentally a program in **J**, which Ken Iverson termed a "modern dialect" of APL, for the sample problem is given by the APL or **J** expression

```
+/2.50 3.69 3.69 3.49 1.27 0.89 1.16 0.84 7.99
```

which has the value `25.52`.

Although spreadsheets have played little, if any, part in the curriculum of the Department, we cannot resist noting in conclusion that a spreadsheet program for the sample problem may be written as `=SUM(A1:A9)`, where the data are assumed to be in cells A1, A2,…, A9

The Department Today

As has been discussed earlier, the Department of Computing Science now occupies the completely renovated Athabasca Hall, one of the oldest and most attractive buildings on the main campus, and the recently constructed and adjoining Computing Science Centre. The Department now numbers 41 academic staff and a suuport staff of 9. At this year's Spring Convocation there were 48 BSc degrees, 8 MSc degrees and 6 PhD degrees in Compiting Science awarded.

The 2013-2014 University Calendar lists the following undergraduate programs in Computing Science: Honors in Computing Science, for highly-motivated students with exceptional ability; Specialization in Computing Science, designed for students to pursue the concentrated study of Computing Science, or to combine the study of Computing Science with another discipline; Specialization in Computing Science – Minor in Business, for students interested in a career that combines Computing Science and Business; Computing Science Specialization in Software Practice, for students interested in a career as a software professional; Computing Science Honors Stream in Bioinformatics, for students who wish to work with large sets of data arising in genome and DNA sequencing; and Computing Science Specialization Stream in Bioinformatics. The Calendar lists 57 undergraduate courses in Compiting Science, almost all of which appear to be specialized courses rather than those for a more general audience. Forty-six graduate courses, many of which are Topics courses, are given in the Calendar along with the following list (with short annotations) of areas of resrarch pursued by faculty members: virtualized reality, abvanced man-machine interfaces, algorithmics, natural language processig, artificial intelligence, bioinformatics, cluster-based routers, communication networks, computer games. computer graphics, online education, computer vision and multimedia communications, wireless sensors, database systems, brain tumour analysis, machine learning, reinforcement learning, robot navigation, software engineering, grid computing.

Epilogue

"… the more we preoccupy ourselves with the bewildering complexity of external apparatus which science places in our hands, the less vigour have we left for the development of the holier purposes of humanity within us." E. C. Bentley, *Trent's Last Case* (1913).

References

Anstey, T. H. and K. W. Smillie, 1963. "Use of a remote digital computer on an open-shop basis in agricultural research." *The Computer Journal*, vol. 6, no. 2, pp. 118 – 120.

Bell, C. Gordon and Alan Newell, 1971. *Computer Structures, Readings and Examples*. McGraw-Hill Book Company, Inc., New York.

Charlton, Bruce, 1991. "When science should be a humanity." *New Scientist*, May 25, 1991, pp. 55 – 56.

Keeping, E. S., 1971. *A Short History of the Department of Mathematics*.

Kenney, John F. and E. S. Keeping, 1954. *Mathematics of Statistics, Part I*, 3rd edition. D. Van Nostrand, Princeton, N.J.

MacHardy, F. V., 1961. "Application of mathematical techniques to the problems of agricultural mechanization of areas undergoing development." *Congrès International Technique du Machinisme Agricole*, 1961, pp. 3 – 24.

Ord-Smith, R. J., 1960. "The STANTEC-ZEBRA" Simple Code and its Interpretation. " *Annual Reciew in Automatic Programming*, vol. 1, pp. 146- 168.

Smillie, Keith, 1991. *The Department of Computing Science: The First Twenty-five Years*. 88pp. Department of Computing Science Technical Report TR-91-01.

Smillie, Keith, 1993. *Computing Science at the University of Alberta 1957 – 1992*. 26 pp.

Smillie, Keith, 1996. "The history of computing science at the University of Alberta." *IEEE Annals of the History of Computing*, vol. 18, no. 1, pp. 70 – 76.

Smillie, Keith, 2002, "Remembering some computers and languages." *IEEE Annals of the History of Computing*, vol. 24, no. 4, pp. 43-44.

Smillie, Keith, 2003. *The LGP-30: The University of Alberta's First Computer*. 21 pp.

Smillie, Keith, 2005. The LGP-30 and its Effect on Computing at the University of Alberta. 21 pp.

Smillie, Keith. 2007. "Early computing at the University of Alberta and the introduction of the LGP-30." *IEEE Annals of the History of Computing*, vol. 29, no. 1, pp. 65-73.

Smillie, Keith, 2012. “An early statistical calculation.” *IEEE Annals of theHistory of Computing*, vol. 34, no. 3, pp. 74 – 76.

van der Poel, W. L., 1956. *The Logical Principles of Some Simple Computers*. University of Amsterdam.

van der Poel, W. L., 1971. “ZEBRA: A simple binary computer.” *Computer Structures: Readings and Examples*, Bell, C. Gordon and Alan Newall (ed.), McGraw-Hill Book Company, Inc., New York.

Weik, M., 1957. *A Second Survey of Electronic Digital Computing Systems*. Office of Technical Services, United States Department of Commerce, Washington, D. C.

Computing Science at the University of Alberta. (1991), 88 pp.

Computing Science at the University of Alberta. (1995), 104 pp.

Timeline of Computing History. The Computer Society, 2012. www.computer.org/cms/Computer.org/cms/Computer.org/Publications/timeline/pdf.

Appendix. Roster of Academic Staff

1962 – 1971	Donald B. Scott
1962 – 1967	Robert S. Julius
1962 – 1992	William S. Adams
1962 - 1966	William B. McMinn
1963 – 1992	Keith W. Smillie
1965 – 1999	Ursula M. Maydell
1965 – 1988	Kellogg V. Wilson
1965 – 1966	Frank Stenger
1966 – 1970	Srisakdi Charmonman
1967 – 1973	Doreen M. Heaps
1967 – 1973	H. Stanley Heaps
1967 - 1972	John P. Penny
1967 – 1970	Gordon H. Syms
1968 – 1981	I - Ngo Chen
1968 – 1977	John F. Hauer
1968 – 1982	Barry J. Mailloux
1968 – 1970	William J. Meyers
1968 – 1969	Richard Peddicord
1968 – 1985	Jeffrey R. Sampson
1969 – 1991	Wayne A. Davis
1969 – 1983	L. Wayne Jackson
1969 – 1993	John Tartar
1970 – 1973	Kon - Vinh Leung
1970 – 2002	T. Anthony Marsland
1970 – 1972	Eric N. West
1971 – 1999	Stanley Cabay
1972 – 1983	Arthur Wouk
1973 – 1989	Lenhart K. Schubert
1973 – 1980	Clement T. Yu
1976 – 1985	Francis Y. Chin
1980 – 1982	Subrata Dasgupta
1980 – 1982	Richard W. Heuft
1980 – 1982	Mary McLeish
1980 – 1982	Pavel J. Voda
1982 – 2003	F. Jeffry Pelletier
1982 – 2000	William W. Armstrong
1983 – 1985	Clifford A. Addison
1983	Duane A. Szafron
1983 – 1988	Lee J. White
1983 – 1984	Karl Winklmann
1984 – 1987	Edward P. F. Chan
1984 – 2002	Mark W. Green
1984 – 2000	M. Tamer Őzsu
1984 – 1989	Robert A. Reckhow
1984	Jonathan Schaeffer
1985	René E. Elio
1985 - 2011	Joseph C. Culberson
1985 – 1994	Wlodzimierz Dobosiewicz
1986 - 2010	Pawel Gburzynski
1986 – 1996	Barry Joe
1986 – 1994	Ahmed E. Kamal
1986 – 2009	Xiaobo Li
1986 - 2012	Piotr Rudnicki
1986	Lorna K. Stewart
1986	Jia - Huai You
1987	Randolph G. Goebel
1987 - 2014	H. James Hoover
1988	Ehab S. Elmallah
1988	Li - Yan Yuan
1988	Hong Zhang
1988 – 1989	Robert J. Crawford
1989 – 2009	Paul G. Sorenson

1990	Anup Basu
1990 – 2000	Peter van Beek
1991	Janelle J. Harms
1993 – 1999	John W. Buchanan
1994 – 1998	Ling Liu
1995 – 1997	Ronald C. Unrau
1997	Russell Greiner
1997	Ioanis Nikolaidis
1997	Eleni Stroulia
1997 – 2000	Benjamin A. Watson
1999	Ryan Hayward
1999	Paul Lu
1999	Michael H. MacGregor
1999	Mario Nascimento
1999	Kenneth Wong
1999	Osmar R. Zaïane
2000	Nelson Amaral
2000 - 2014	Walter Bischof
2000 – 2005	Terry Caelli
2000	Martin Jägersand
2000 – 2008	Dekang Lin
2000	Martin Müller
2000	Davood Rafiei
2001	Pierre Boulanger
2001	Vadim Bulitko
2001 – 2007	Sherif Ghali
2001	Robert Holte
2001	Guo - Hui Lin
2001	Joerg Sander
2001	Herbert Yang
2002	Michael Buro
2002	Greg Kondrak
2003	Michael Bowling
2003	Dale Schuurmans
2003	Richard S. Sutton
2003	David Wishart
2004	Mohammed Salavatipour
2006	Nilanjan Ray
2006	Csaba Szepesvari
2008	Denilson Barbosa
2011	Abram Hindle
2014	Zachary Friggstad

Faculty Service Officers

1976 - 2014	Steven F. Sutphen
1983 – 1983	Daniel C. Wilson
1986 – 1992	Alynn B. Klassen
1987 - 2010	Catherine Descheneau
1993 - 2014	Carol J. Smith

Administrative Professional Officers

1972 – 1973	Graham Toope
1973 – 1975	Gary Albright
1982 – 1984	Arnold Adam
1984 – 1987	Ivan E. Bering
1987 – 1997	Brian R. Pinchbeck
1997 – 2006	June MacLellan
2006 – 2009	Jill Bagwe
2006 - 2012	Fran Moore
2011	Louise White

Lecturer

1982 – 1985	Daniel A. Boulet

Faculty Lecturers

1999 – 2000	Mario Costa Sousa
2000 – 2001	Jonathan Sillito
2000 – 2006	Cerian Jones

2000 – 2001	Yngvi Bjornsson
2001 – 2002	William Elazmeh
2001 – 2002	Thomas Harke
2003 – 2004	John Anvik
2003 – 2003	Ed Leonard
2003 – 2004	Andrew Macdonell
2004 – 2005	Sadaf Ahmed