



CCS-NW Christmas & New Year quiz 2021

Last year, I said “What a year!”. This year became “Not quite so bad” but has turned to “Here we go again!”

In Manchester, we had hoped we would hold a number of face-to-face meetings in 2021. Sadly, that has not turned out as hoped. We hope on.

I didn't get out to visit other museums or collectors either, so I don't have any “interesting” photos of my own to include. However, there are so many museums and collectors to view online.

As always, I apologise for any errors. I will publish the answers sometime in January.

Good luck, Merry Christmas, and Happy (and better) New Year to all.

Bob Geatrell

(Bob@Geatrell.co.uk)

Chair, CCS NW Group.

Were you paying attention?

A few questions relating to the presentations this year. Although one or two were held face to face, they were ALL broadcast (to a gratifying large audience) and have ended up on YouTube. *I have added links to the relevant section of the videos*

1. Where did Norman Spiney answer the phone?
 - Norman Spiney (at the time of Monty Python's Spiny Norman) was used as a generic name in the offices of SIA when a phone call was received.
2. What was saved by the fridge?
 - James Bond? In the digital filming of “Spectre” on the Thames, a video storage device failed to read until it was cooled.
3. Who needs to follow OHM's law?
 - Curators care about “Objects, History and Meaning” from Historiography of the History of Computing
4. Who used a national version of logarithms?
 - Percy Ludgate proposed “Irish Logarithms” for multiplication on his Difference Engine
5. What was going on, on the site of a future Grand Prix?
 - American cryptographers, with punched card equipment, were moved to Albert Park. Melbourne



6. Who could calculate the odds on 7 horses in 15 races to 0.1%?
 - People on the shop floor in the steel industry, who could probably understand computers if necessary.
7. Who sent congratulations on the development of an Interfaith Message Processor?
 - Senator Edward Kennedy, congratulating the ARPAnet team, misheard the word "Interface"
8. What relation was Umkehrwalze?
 - "Uncle Walter" (misheard) is the turnaround ring or Reflector in the Enigma

Don't Sweat The Small Stuff

Most of us lived through the birth of the individual / home / personal computer. The CCS generally focuses on the big stuff, but here's some questions about early little stuff:

1. Which home computer of the early 1980s had a name which suggests that it had come from a popular BBC space opera?
 - The "Oric-1" computer was probably named after the all-powerful "Orac" from Blake's 7.
2. In 1984 ICL introduced a personal computer (not a "PC") that was named to suggest there would be dozens in the office. What was it called, and what was the design derived from?
 - ICL's "One Per Desk, or OPD", was based on the hardware of the Sinclair QL.
3. Which device, with a good claim to be the world's first home computer, was named after a bright star? And why did they give it that name?
 - The Altair. One story has it that they wanted an interesting name, which was suggested by the daughter of editor of Popular Electronics. Star Trek's Enterprise was visiting Altair on TV that evening.
4. Name an early mouse-controlled computer. The designer eventually admittedly it was named after his daughter (not a yellow cartoon)
 - Apple originally claimed that the Lisa was an acronym for "Locally Integrated Software Architecture" but eventually admitted that it was a "backronym" after Steve Jobs' daughter.



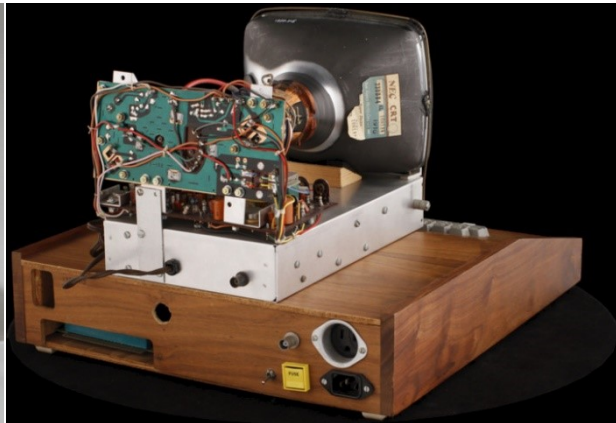
5.
Amstrad CPC-64



6.
Osborne 1 ("Luggable")



7.
Commodore PET 2001



8.
Apple 1 (photo from a rotatable image [at the Science Museum](#))



9.
Sinclair ZX-Spectrum. The "Speccky"



10.
TRS-80 (Tandy Radio Shack)



11.
Exidy Sorcerer



12.
Sinclair ZX81

Microbiography

As in "Very Short Biography". Some people are so well known that only a few words are required to identify them.

Can you identify these (mainly) computer related people? The descriptions are rather "tabloid" level and may quite validly refer to many other people. I've gone for the obvious ones.

1. Difference

- Charles Babbage. For his "Difference Engine." Also, Percy Ludgate, if you watched that lecture.

2. Leave it on

- Tommy Flowers insisted that Colossus, as a valve machine, would be reliable so long as it was not continually powered off and on.

3. Sunflowers

- Alan Mathison Turing. After 1951 he used the Manchester Mark 1 computer to further his studies into morphogenesis including why the seed heads in sunflowers display Fibonacci numbers.

4. Bombe

- Gordon Welchman. Controversial, maybe, but he often gets overlooked, and AMT appears above.

5. Differential

- Douglas Hartree, for his mechanical Differential Analyser

6. Baby Designer

- Tom Kilburn, chief designer of the SSEM, or "Baby". Later

7. EDSAC

- Maurice Wilkes. Obviously!

8. Fourth Man

- Andrew Booth of Birkbeck. So called by our own Roger Johnson. Designed & built APE(X)C

9. Forth Man

- Charles ("Chuck") Moore invented the FORTH programming language.

10. Rear Admiral

- Grace Hopper, a significant contributor to the history of computing, including the development of COBOL. She rose to the rank of Rear Admiral



11. Pres

- J. Presper ("Pres") Eckert, famous with John Mauchly for ENIAC and UNIVAC

12. First Draft

- John von Neumann, a great mathematician, wrote the seminal "[First Draft of a Report on the EDVAC](#)"

13. Structured programming

- Edsger W. Dijkstra. Also jointly developed the first ALGOL 60 compiler

14. Tube Store

- Frederic "Freddie" Williams, developer of the Williams tube store.

15. Harvard

- Howard H. Aiken, Designer behind the ASCC at Harvard, renamed the Harvard Mark 1 (followed by II, II and IV)

16. Mechanical digital

- Konrad Zuse built the mechanical Z1 between 1936 and 1938.

17. FORTRAN designer

- John Backus. Also famous for BNF, used to define (e.g.) Algol-like languages

18. Autocode

- Tony Brooker developed Mark I Autocode.

19. Census

- Herman Hollerith. Worked at the US Census Bureau and went on to develop the punched card systems which automated the census.

20. Byron

- Ada Lovelace. Augusta Ada King, Countess of Lovelace, daughter of Lord Byron.

21. 5 computers

- Thomas J. Watson allegedly(*) said "I think there is a world market for maybe five computers"

22. Information theory

- Claud Shannon

23. Linux

- Linus Torvalds

24. First Analyst

- David Caminer – hugely influential on the design of the LEO computers.

(*) Re the apocryphal "only 5 computers needed" statement. I received the following interesting communications:

We believe the statement that you attribute to Thomas Watson is a misunderstanding of remarks made at IBM's annual stockholders meeting in 1953. In referring to the [IBM 701](#) which had been introduced the year before as the company's first production computer designed for scientific calculations -- Thomas Watson, Jr., told stockholders that "IBM had developed a paper plan for such a machine and took this paper plan across the country to some 20 concerns that we thought could use such a machine. I would like to tell you that the machine rents for between \$12,000 and \$18,000 a month, so it was not the type of thing that could be sold from place to place. But, as a result of our trip, on which we expected to get orders for five machines, we came home with orders for 18."
Via, and with thanks to, [Robert Garner](#) (IBM & CHM)

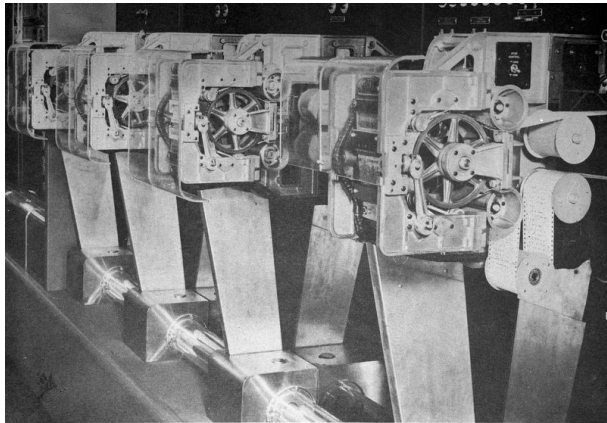
That big sales trip was led by two IBMers, Cuthbert Hurd and Jim Birkenstock. They essentially shopped around a block diagram (for what became the 701), a simple drawing approved by [John von Neumann](#) himself and based mostly on [the IAS](#) plan (JVN was an IBM consultant at the time).
[Dag Spicer](#) (CHM)

And I'm delighted to mention that the IAS and 701 used the Williams Tube store, developed in Manchester and still regularly demonstrated in use in the [Baby replica at the Science and Industry Museum](#) [Bob Geatrell](#)

A bit on the side

One of our members pointed out to me a while back, that us processor lot from West Gorton tended to ignore the peripherals. And where would we be without them?

Quite right. Here's a few things to identify. Some can clearly be labelled as peripherals, others may be devices, but they aren't processors.



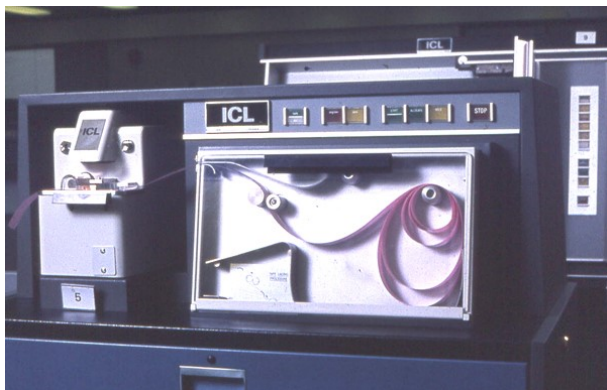
1

Sequence tape readers, Harvard Mark 1



2

Early magnetic drum from Manchester Mark 1



3

ICL 1906A Tape reader (Chilton Computer Centre)



4

Friden Flexowriter



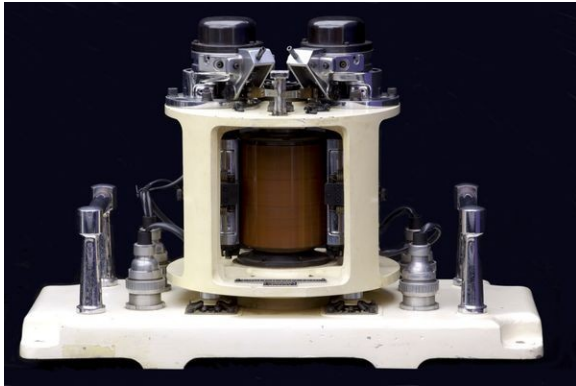
5

ENIAC Portable (sic) function table. Input via 10-position switches.



6

DECtape (Microtape)



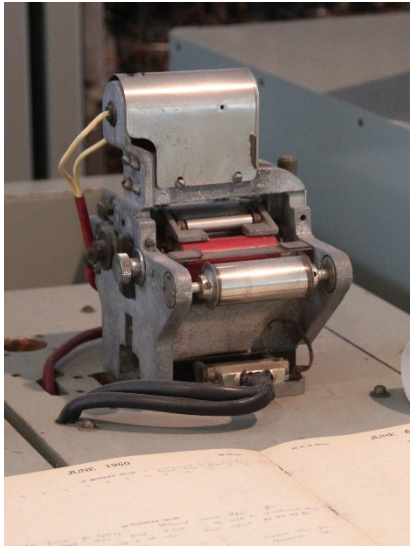
7

Deuce drum store



8

ICT 1900 card reader



9

CSIRAC 12-track paper tape reader



10

Paper tape unit on
Teletype 33 ASR



11

UNISERVO tape unit
(UNIVAC)

Gone but not (totally) forgotten

A very recent email discussion between members of the CCS committee concerned programming languages that might be considered dead. Maybe they should be titled “Merely resting.”

Since many languages and their derivatives are very similar, it's possible that these short clips may be valid for more than one language. If so, I apologise in advance.

1.

- FLOW MATIC – Originally B-0, developed by Grace Hopper

```
(0) INPUT INVENTORY FILE-A PRICE FILE-B ; OUTPUT PRICED-INV FILE-C UNPRICED-INV
FILE-D ; HSP D .
(1) COMPARE PRODUCT-NO (A) WITH PRODUCT-NO (B) ; IF GREATER GO TO OPERATION 10 ;
IF EQUAL GO TO OPERATION 5 ; OTHERWISE GO TO OPERATION 2 .
(2) TRANSFER A TO D .
(3) WRITE-ITEM D .
(4) JUMP TO OPERATION 8 .
(5) TRANSFER A TO C .
(6) MOVE UNIT-PRICE (B) TO UNIT-PRICE (C) .
(7) WRITE-ITEM C .
(8) READ-ITEM A ; IF END OF DATA GO TO OPERATION 14 .
(9) JUMP TO OPERATION 1 .
(10) READ-ITEM B ; IF END OF DATA GO TO OPERATION 12 .
(11) JUMP TO OPERATION 1 .
(12) SET OPERATION 9 TO GO TO OPERATION 2 .
(13) JUMP TO OPERATION 2 .
(14) TEST PRODUCT-NO (B) AGAINST ; IF EQUAL GO TO OPERATION 16 ;
OTHERWISE GO TO OPERATION 15 .
(15) REWIND B .
(16) CLOSE-OUT FILES C ; D .
(17) STOP . (END)
```

2.

- CMS-2 “Compiler Monitor System”

```
XMPL 1 READIN SYS-PROC $
XMPL 2 COMMENT IN THE ABOVE STATEMENT THE SYMBOL READIN IS THE NAME
XMPL 3 IDENTIFIER ASSIGNED TO THIS COMPILE-TIME SYSTEM ELEMENT
XMPL 4 WHICH IS A SYSTEM PROCEDURE. IT IS NOT A STATEMENT LABEL
XMPL 5 AND MUST NOT BE FOLLOWED BY A PERIOD $
XMPL 6 PROCEDURE READIN EXIT NOMORE $
XMPL 7 SET WORK TO 0 $
XMPL 8 INPUT READ (WORK(TRKNO), WORK(X1), WORK(Y1), WORK(T1),
XMPL 9 WORK(X2), WORK(Y2), WORK(T2),
XMPL 10 IDEN , LOCREM) READER $
XMPL 11 IF WORK(TRKNO) EQ ENDINPUT THEN RETURN NOMORE $
XMPL 12 GOTO IDENSW $
XMPL 13 IDUNK. SET WORK(ID) TO 'UNKNOWN'$
XMPL 14 GOTO LR ''PROGRAM CONTROL IS TRANSFERRED
XMPL 15 TO STATEMENT LABELLED LR. NOTE
XMPL 16 THAT LR, WHEN REFERENCED, IS NOT
XMPL 17 FOLLOWED BY A PERIOD'' $
XMPL 18 IDFOE. SET WORK(ID) TO 'HOSTILE' $ GOTO LR $
XMPL 19 IDFR. SET WORK(id) TO 'FRIEND'$
XMPL 20 LR. IF LOCREM EQ H(NO) THEN SET WORK(REMOTE) TO 1 $
XMPL 21 RETURN$
XMPL 22 END-PROC READIN$
```


3.

- PL/1

```
/* Read in a line, which contains a string,
/* and then print every subsequent line that
/* contains that string. */
```

```
find_strings: procedure options (main);
  declare pattern character (100) varying;
  declare line character (100) varying;
  declare line_no fixed binary;
```

```
  on endfile (sysin) stop;
```

```
  get edit (pattern) (L);
```

```
  line_no = 1;
```

```
  do forever;
```

```
    get edit (line) (L);
```

```
    if index(line, pattern) > 0 then
```

```
      put skip list (line_no, line);
```

```
      line_no = line_no + 1;
```

```
    end;
```

```
end find_strings;
```

4.

- B. After BCPL. Before C

```
/* The following program will calculate the constant e-2 to
   about 4000 decimal digits, and print it 50 characters to the
   line in groups of 5 characters. The method is simple output
   conversion of the expansion
   1/2! + 1/3! + ... = .111....
   where the bases of the digits are 2, 3, 4, . . . */
```

```
main() {
  extrn putchar, n, v;
  auto i, c, col, a;

  i = col = 0;
  while(i<n)
    v[i++] = 1;
  while(col<2*n) {
    a = n+1;
    c = i = 0;
    while (i<n) {
      c =+ v[i] *10;
      v[i++] = c%a;
      c =/ a--;
    }

    putchar(c+'0');
    if(!(++col%5))
      putchar(col%50?' ': '*n');
  }
  putchar('*n*n');
}
v[2000];
n 2000;
```

5.

- SNOBOL

```
OUTPUT = "This program will ask you for personal names"
OUTPUT = "until you press return without giving it one"
NameCount = 0 : (GETINPUT)
AGAIN NameCount = NameCount + 1
OUTPUT = "Name " NameCount ": " PersonalName
GETINPUT OUTPUT = "Please give me name " NameCount + 1
PersonalName = INPUT
PersonalName LEN(1) :S (AGAIN)
OUTPUT = "Finished. " NameCount " names requested."
END
```



6.

I discovered this one in a discarded manual in the early 80s. It was so different to anything I'd met previously.

- **RPG-II Report Program Generator from IBM**

```
F*****
F* THIS PROGRAM READS THE CONTENTS OF AN INVOICE HEADER FILE
F* AND PRINTS THE INVOICES PROCESSED FOR THE DATE SET IN THE
F* LOCAL DATA AREA.  THERE ARE LEVEL BREAKS AND TOTALS FOR
F* EACH STORE.
F*****
F*
FINVHDR  IP  F      62          DISK
FPRINTER O  F     132        OF  PRINTER
IINVHDR  NS  01   1NC
I
I          1   3 STORE L1
I          4  13 INVNO
I          14  20 CUSTNO
I          21  45 STNAM
I          46  53 INVDAT
I          54 622TOTINV
I
I          UDS
I          1   8 RPTDAT
C   01      INVDAT      COMP RPTDAT      11
C   01 11      ADD TOTINV      L1TOT  92
C   01 11      ADD TOTINV      LRTOT  92
OPRINTER H  101   1P
O          OR          OF
O
O          PAGE  Z  106
O          102 'PAGE'
O          59 'VERY BIG'
O          72 'STORES, INC.'
O          UDATE Y  17
O          8 'RUN DATE'
O
O          H  1      1P
O          OR          OF
O
O          H  2      1P
O          OR          OF
O          RPTDAT      83
O          63 'FOR DATE:'
O
O          H  1      1P
O          OR          OF
O
O          6 'STORE'
O          18 'INVOICE'
O          28 'CUSTOMER'
O          58 'CUSTOMER'
O          78 'INVOICE'
O
O          H  2      1P
O          OR          OF
O
O          6 'NUMBER'
O          18 'NUMBER'
O          28 'NUMBER'
O          58 'NAME'
O          78 'TOTAL'
O
O          D  0      L1
O          D  1      01 11      STORE      6
O          INVNO      18
O          CUSTNO      28
O          STNAM      58
O          TOTINV1B    78
O          T  1      L1
O          45 'STORE TOTAL...'
O          L1TOT 1B    78
O          T  1      LR
O          45 'GRAND TOTAL...'
O          LRTOT 1    78
O
```