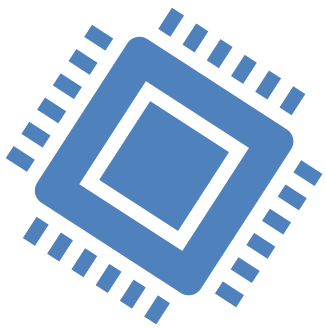


Curso de

# Arquitectura de computadores



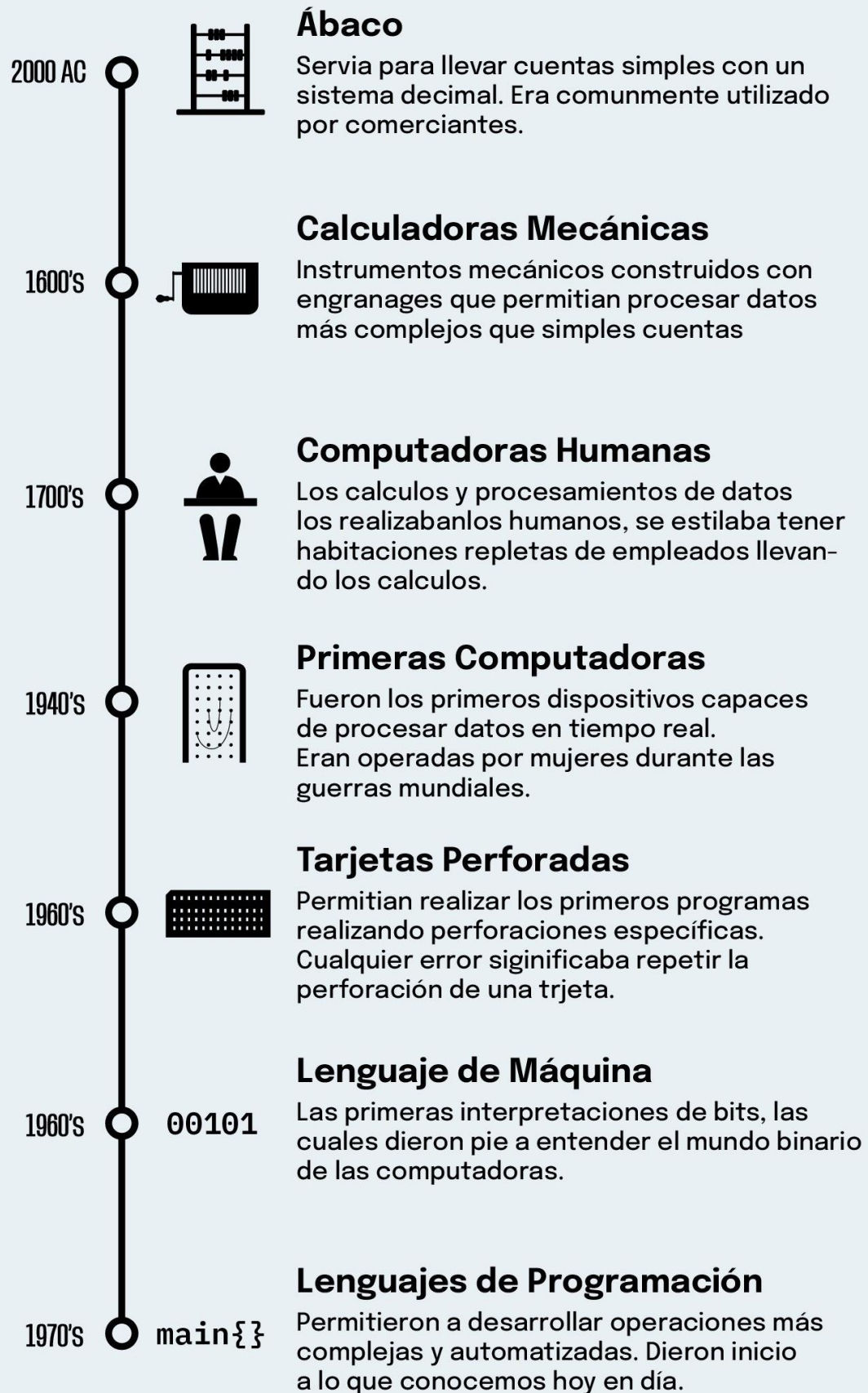
1010  
1010



Santiago Murillo Rendón

Yaneth Mejía Rendón

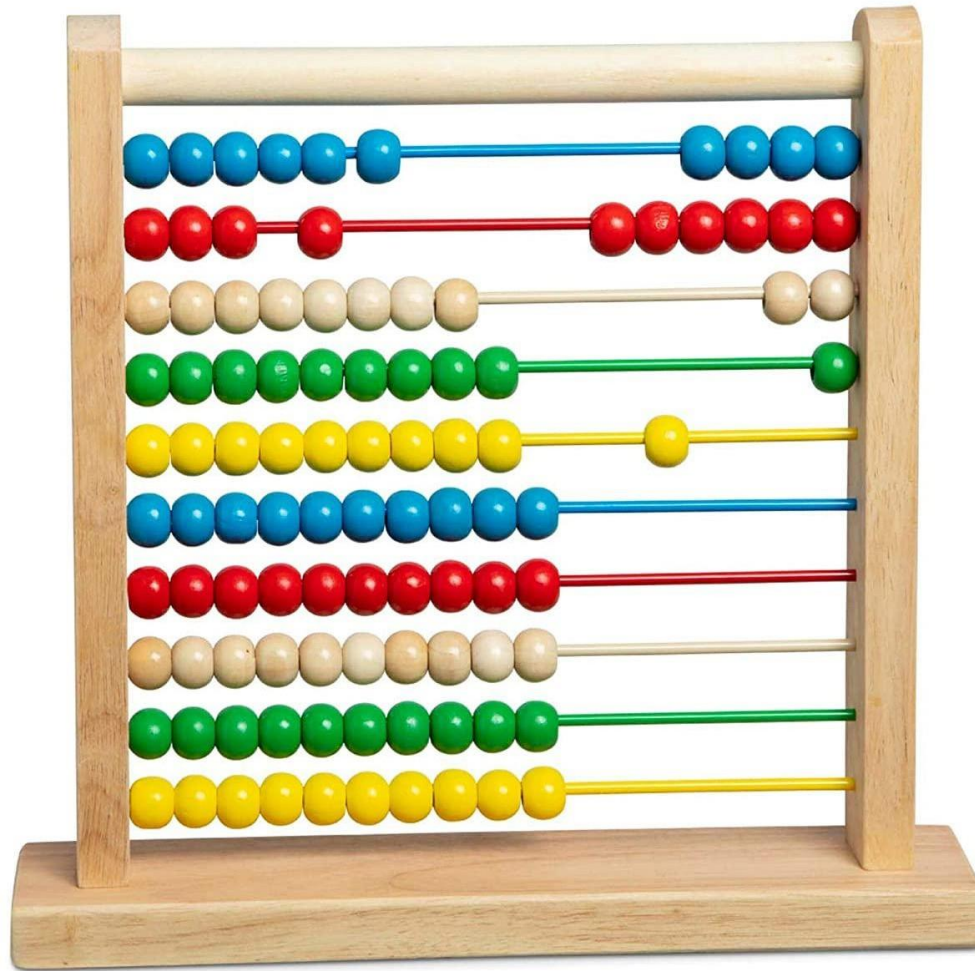
# Historia de la Computación



¿Cómo  
empezó  
todo?

---

# Ábaco 5000 A.C.



# Quipos y Yupanas



# Calculadoras mecánicas

## Pascalina 1642



# Máquina de Leibniz 1672

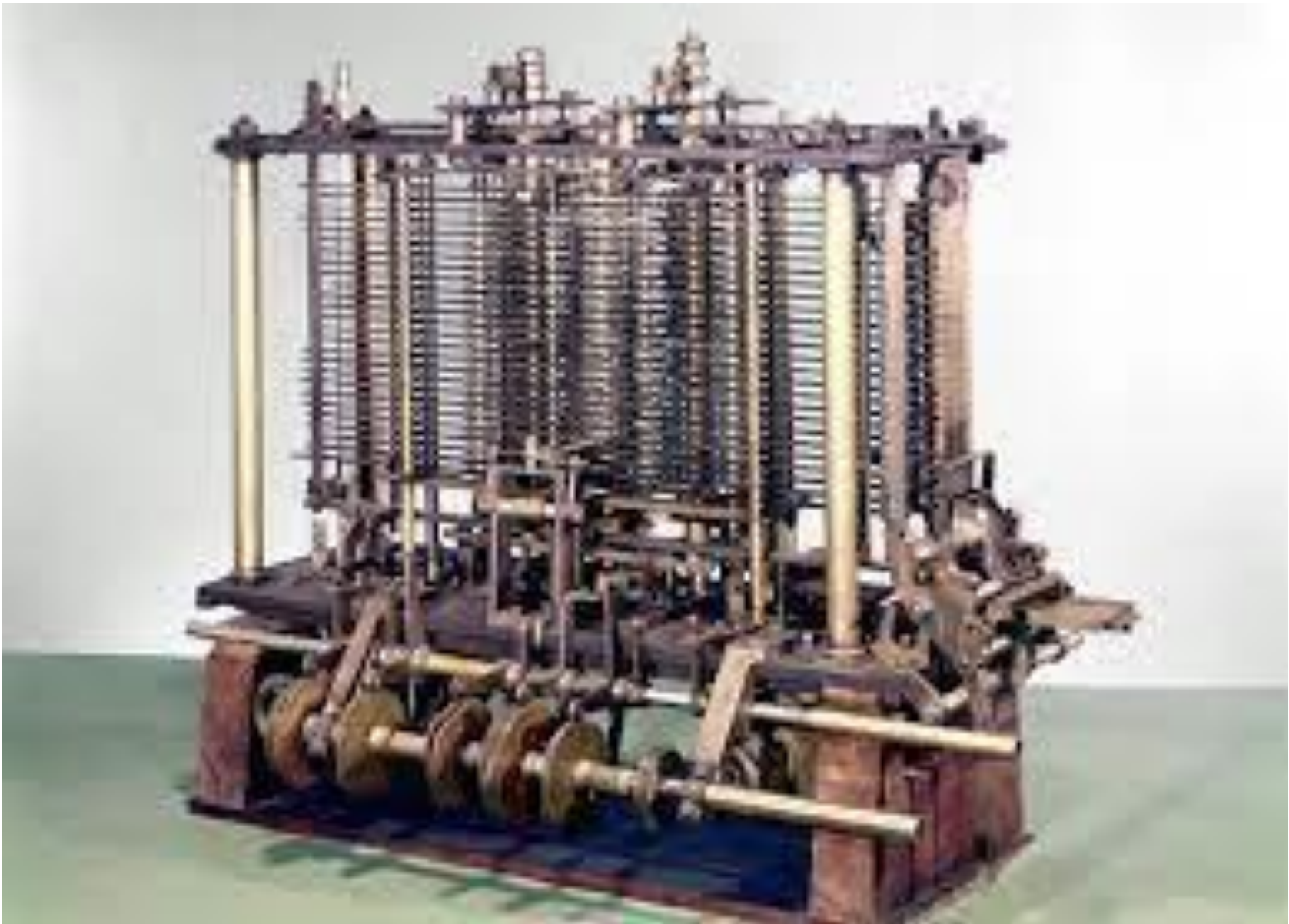


# Telares de Jacquard 1801



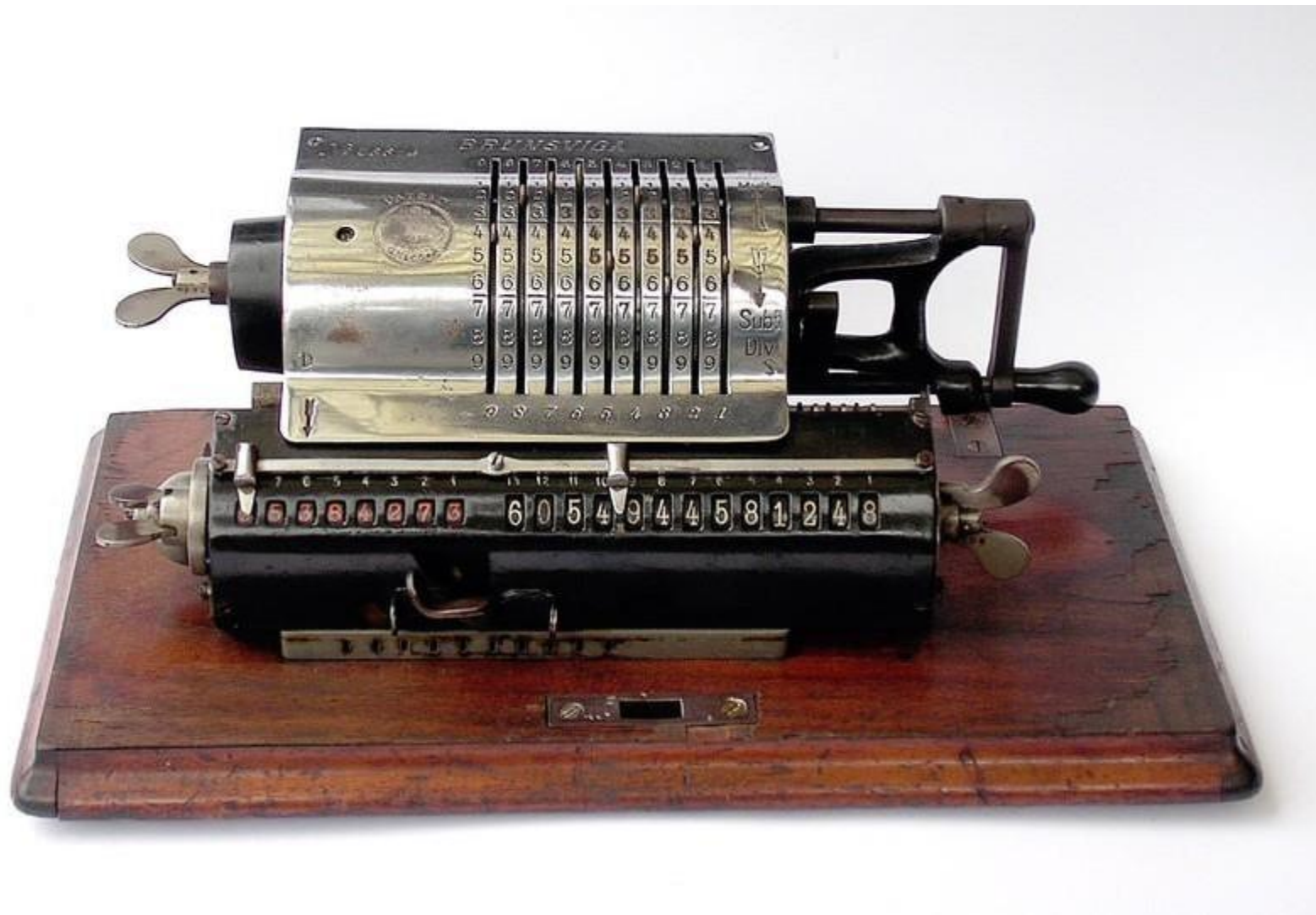
Tarjetas perforadas

# Máquina de Babbage 1837





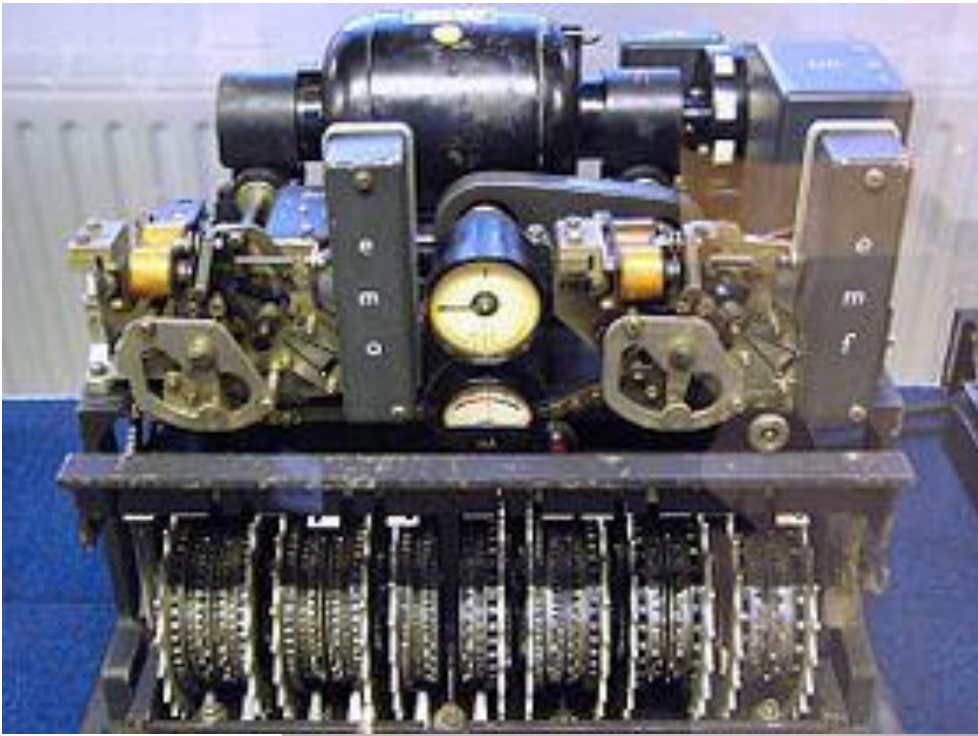
# Calculadoras mecánicas 1920



# La guerra

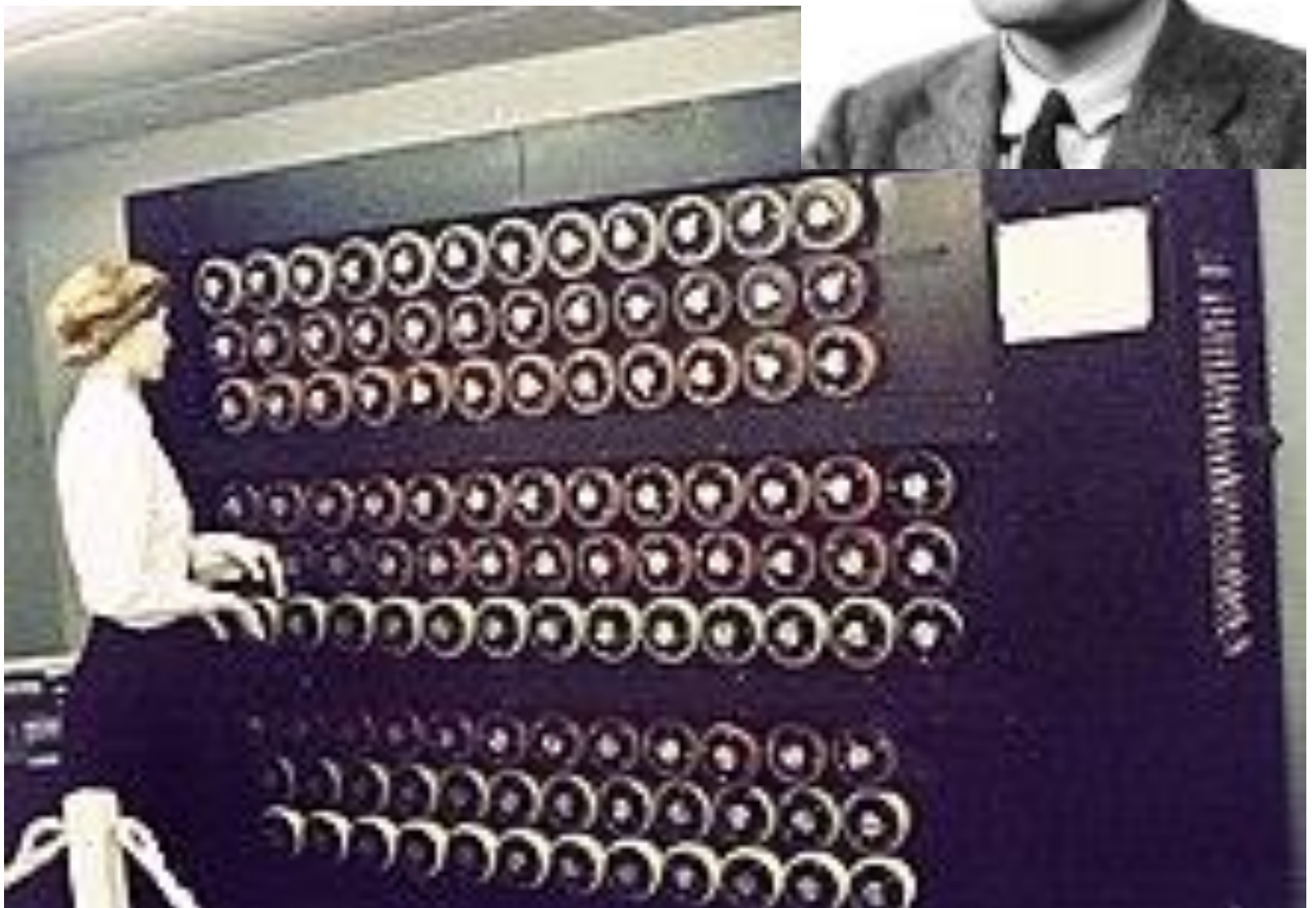


# Enigma y máquina de Lorenz 1941

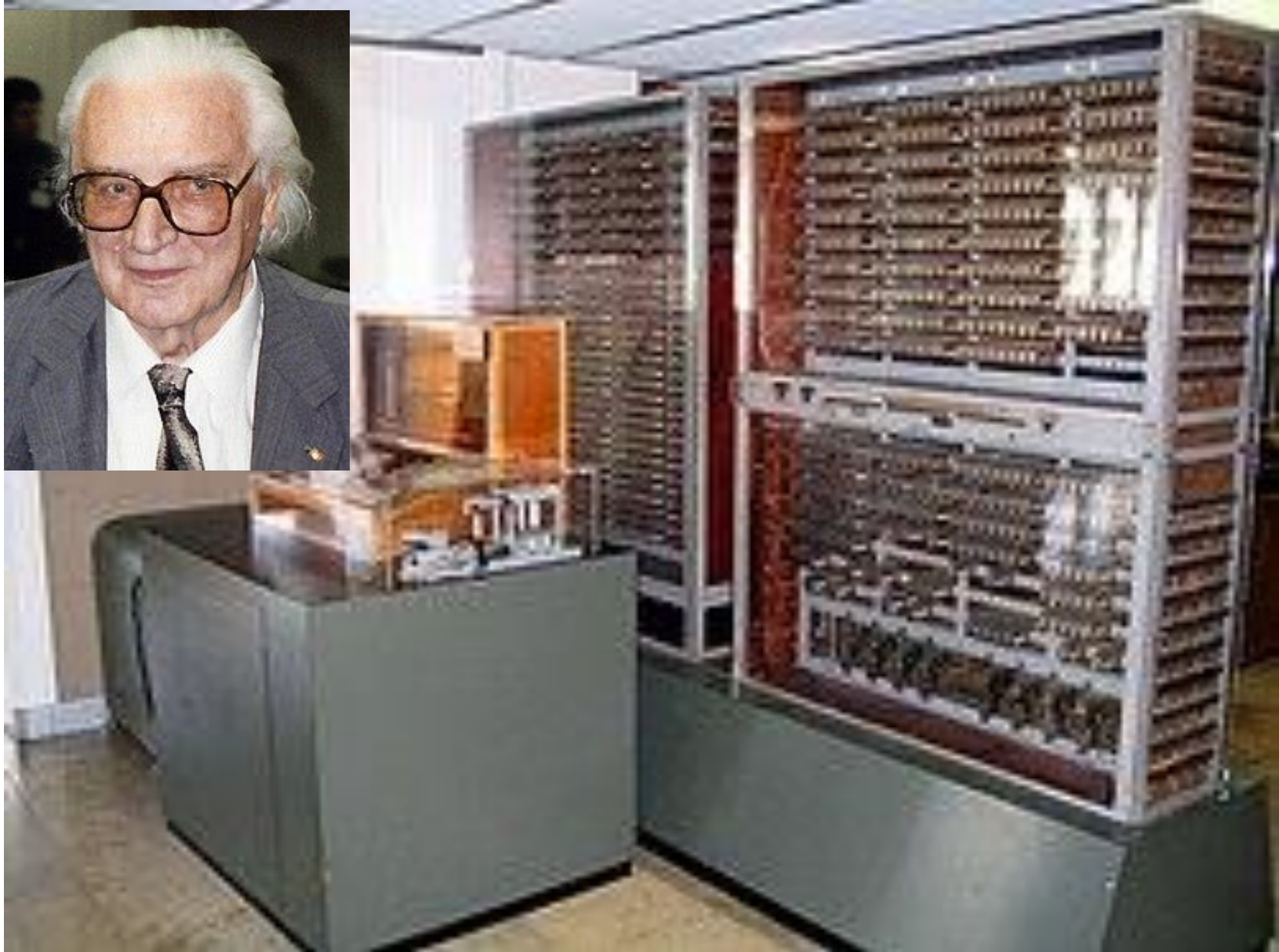


# Bombe (bomba criptológica)

1939 - 1949



# Z1, Z2, Z3, Z4 y Z22 Konrad Zuse 1937



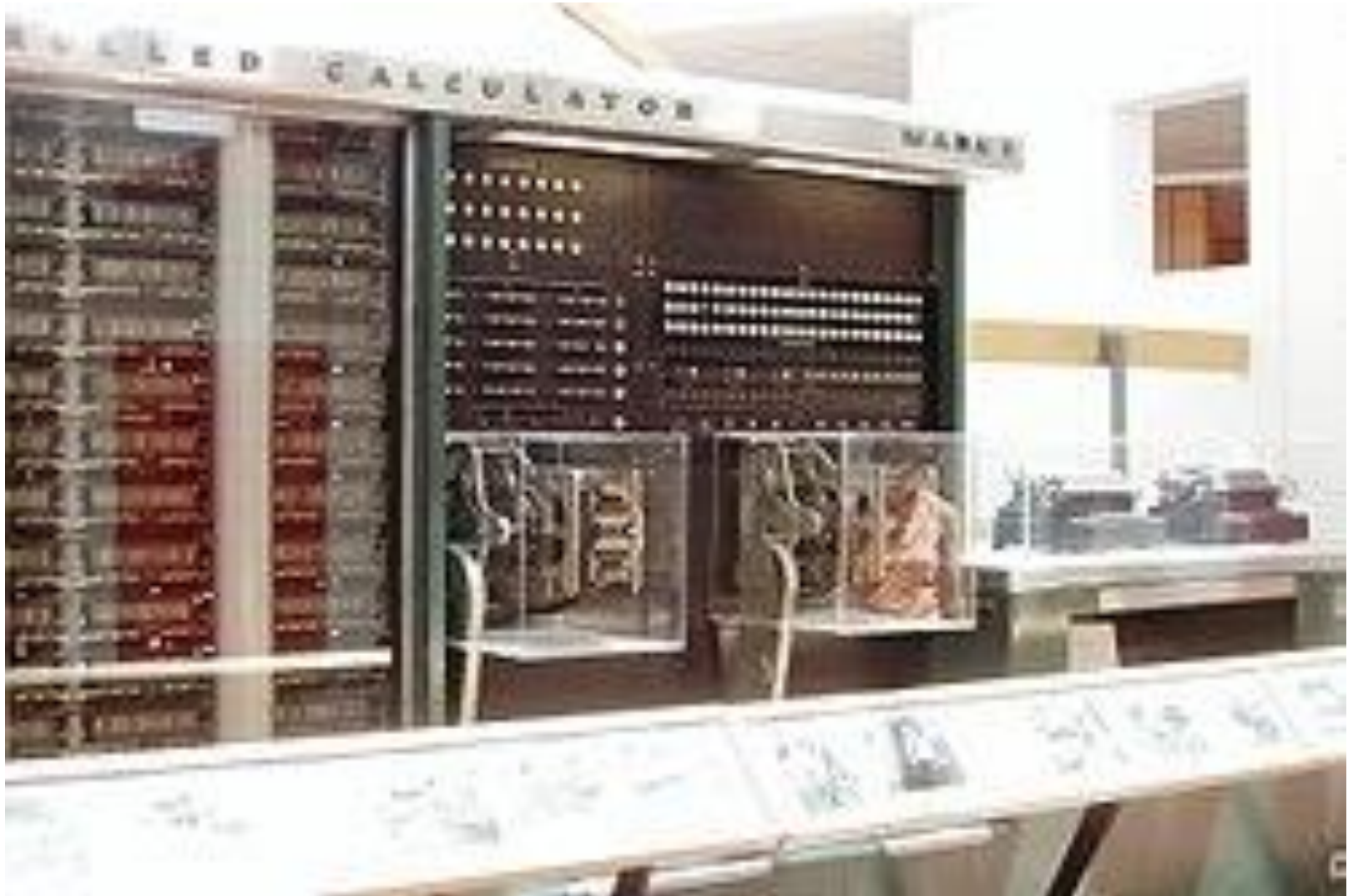
Primera computadora programable, con una frecuencia de 5Hz, construida con 2300 relés. Utilizaba palabras de 22 bits para manejar cálculos con aritmética en coma flotante binaria.

# Atanasoff Berry Computer 1942



Usaba condensadores para almacenar datos en lo que sería la primera memoria RAM.

# Hardvard Mark 1944



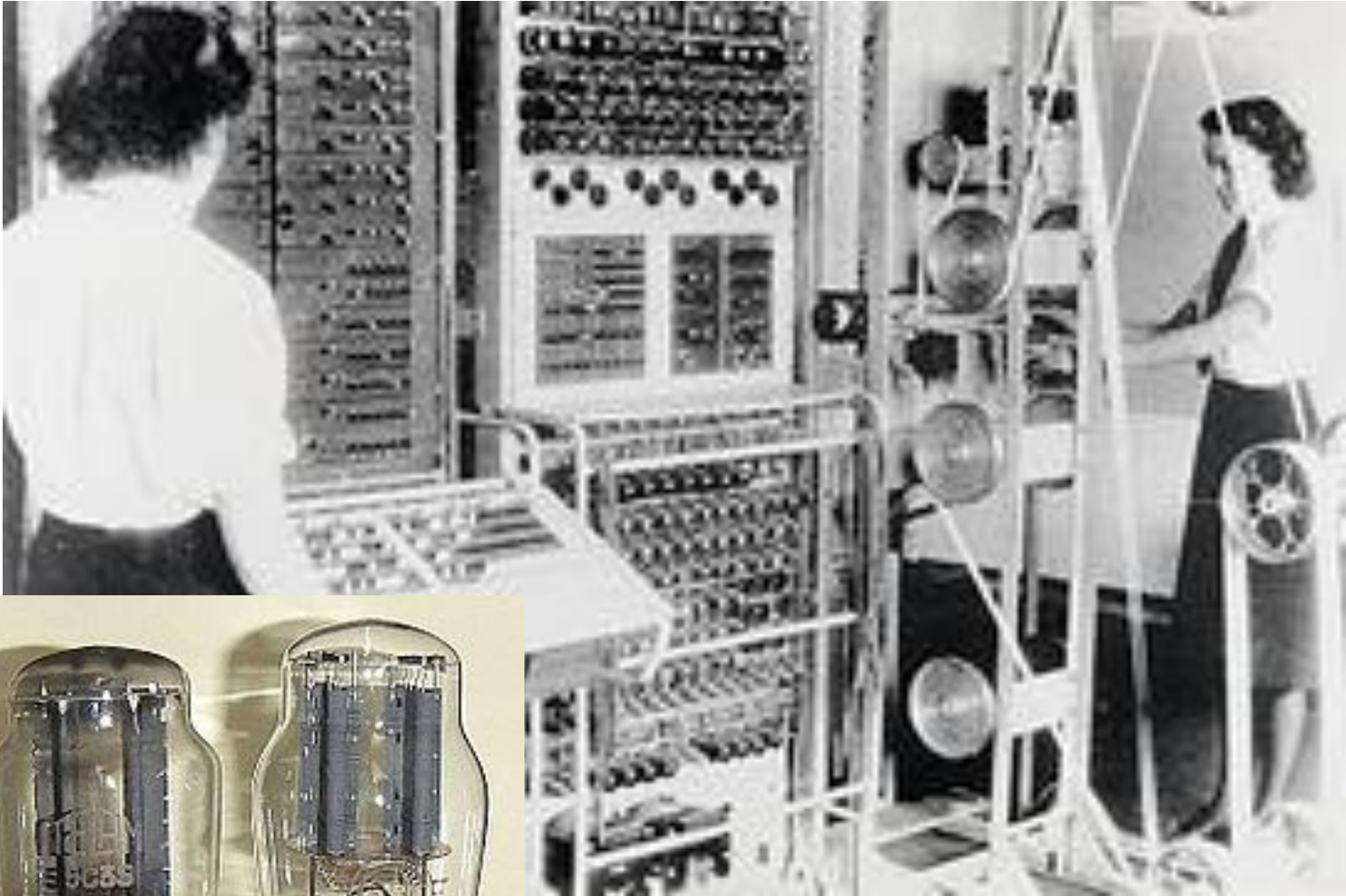
Estaba compuesta de más de 1400 interruptores. La máquina contenía 72 registros mecánicos. Cada uno de los registros mecánicos era capaz de almacenar 23 dígitos, los dígitos que se usaban para el signo era un 0 para signo positivo y un 9 para el signo negativo. Resolvía operaciones de suma y resta en 0,3 segundos, la multiplicación y la suma tardaban mas.

# Computadores de primera generación: Válvulas electrónicas

---

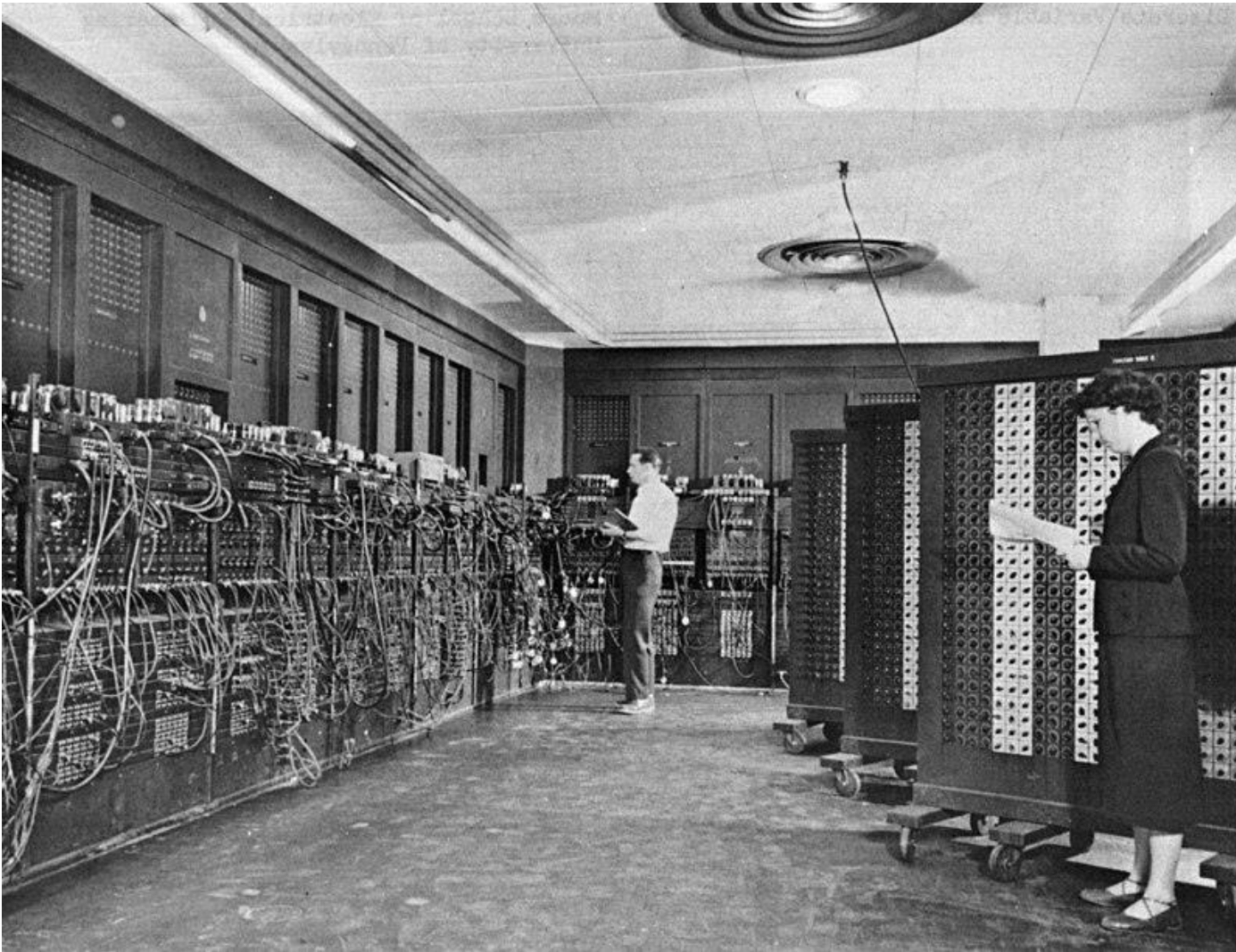


# Colosus 1944



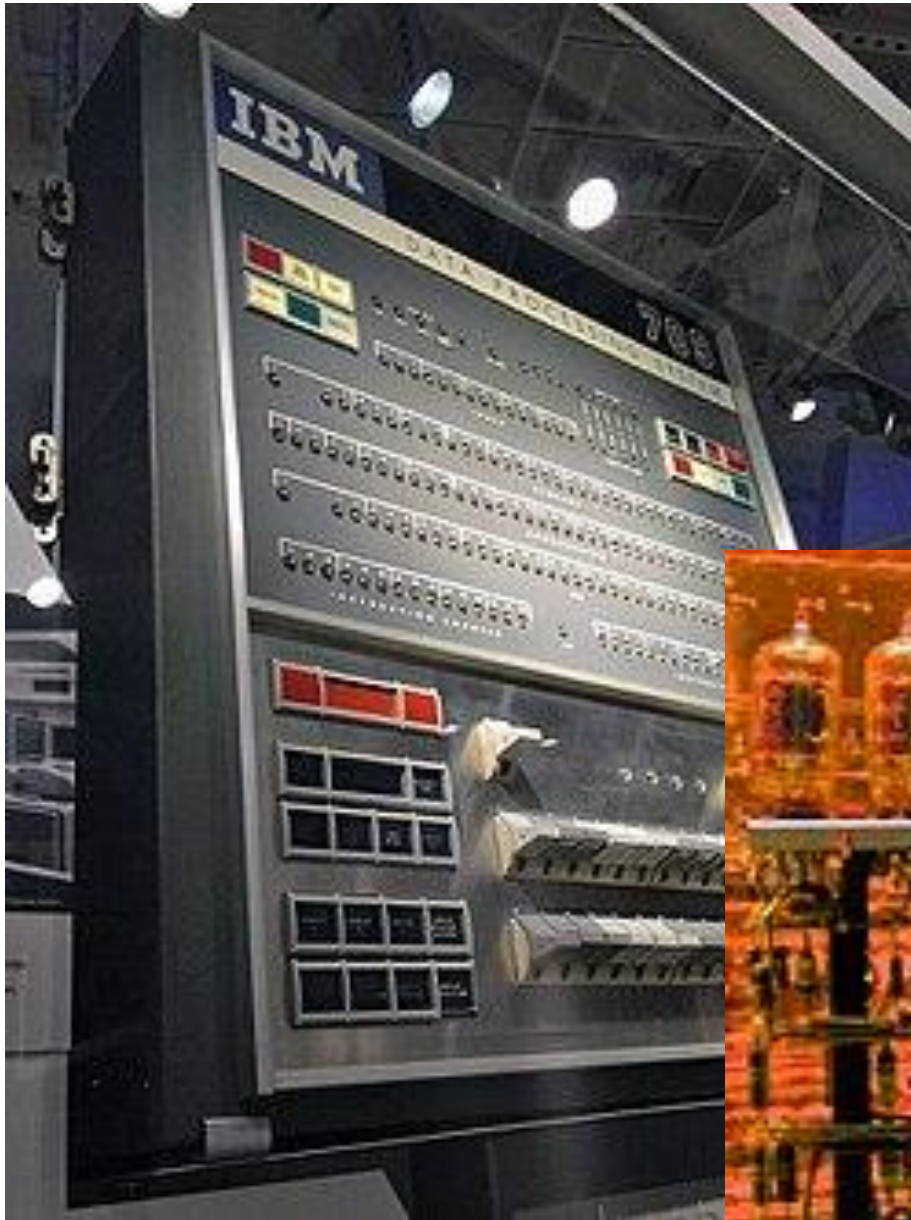
Tubos de vacío.

# ENIAC (Electronic Numerical Integrator And Computer) 1946



17 468 tubos de vacío, 7200 diodos de cristal, 1500 relés, 70 000 resistencias, 10 000 condensadores y cinco millones de soldaduras. 5000 sumas y 300 multiplicaciones por segundo

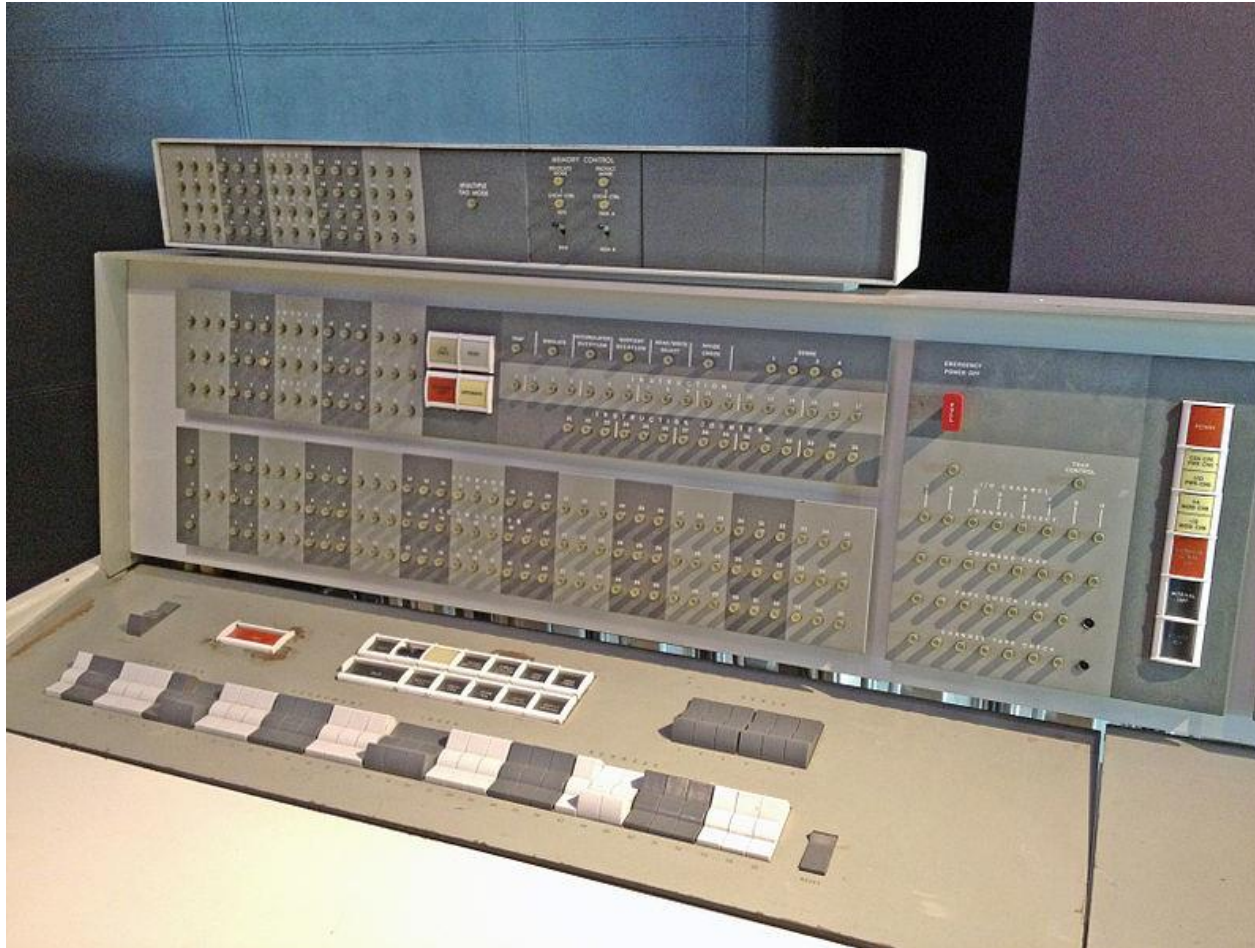
# IBM 709 1955



# Computadores de segunda generación: Los transistores

---

# IBM 7090 1958



Nombre	Longitud	Descripción
Contador de instrucciones	15 bits	Apunta a la siguiente instrucción a ejecutar
Registro de direcciones	15 bits	Contiene al dirección de las instrucciones
Registro de almacenamiento	36 bits	Registro donde se colocan los datos leídos de memoria antes de ser usados
Registro de instrucciones	18 bits	Registro donde se almacena la instrucción cuando está ejecutándose

Nombre	Longitud	Descripción
AC (Registro Acumulador)	38 bits	Se usa para realizar operaciones como la suma y la resta, donde un operando se almacena en el registro acumulador y el otro operando en otro registro distinto, quedando el resultado en el registro acumulador
MQ (Registro Multiplicador)	36 bits	Se usa principalmente para la multiplicación y la división. El resto queda almacenado en el registro AC.
XR (Registros de índice)	15 bits	Había 3 registros índice en el 7090 y siete en el 7094
SE (Indicador del sentido)	36 bits	Usado para leer y escribir

# IBM 7090 1958

primer videojuego computarizado de la historia, el Spacewar de Steve Russell.

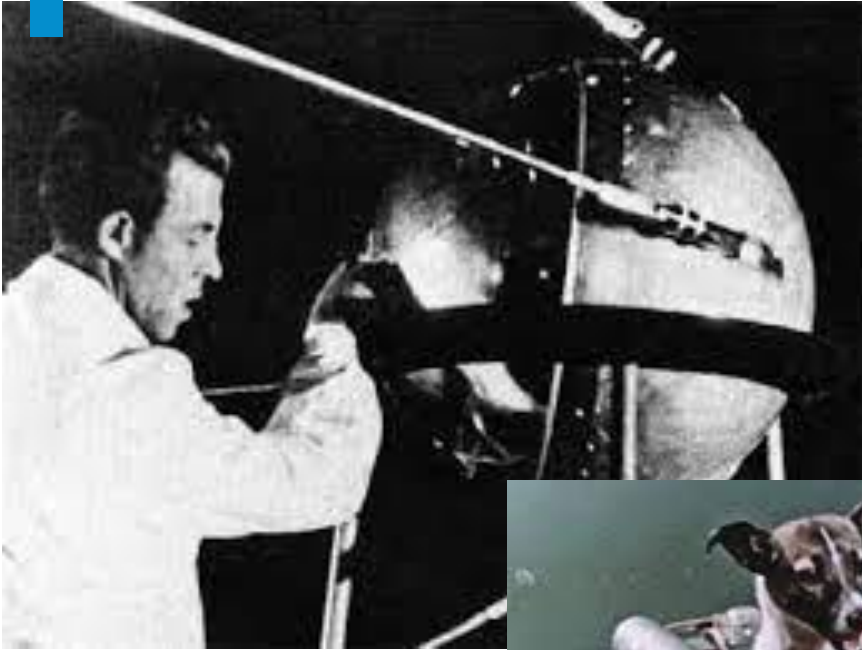


Tenía palabras de 18 bits y 4K de palabras como memoria principal estándar (equivalente a 9 kilobytes), ampliable a 64K de palabras (144 KB). 100.000 operaciones por segundo

# Computadores de tercera generación: Los circuitos integrados

---

# La era espacial





# IBM S/360 1964



Circuitos integrados



# PDP – 8 19



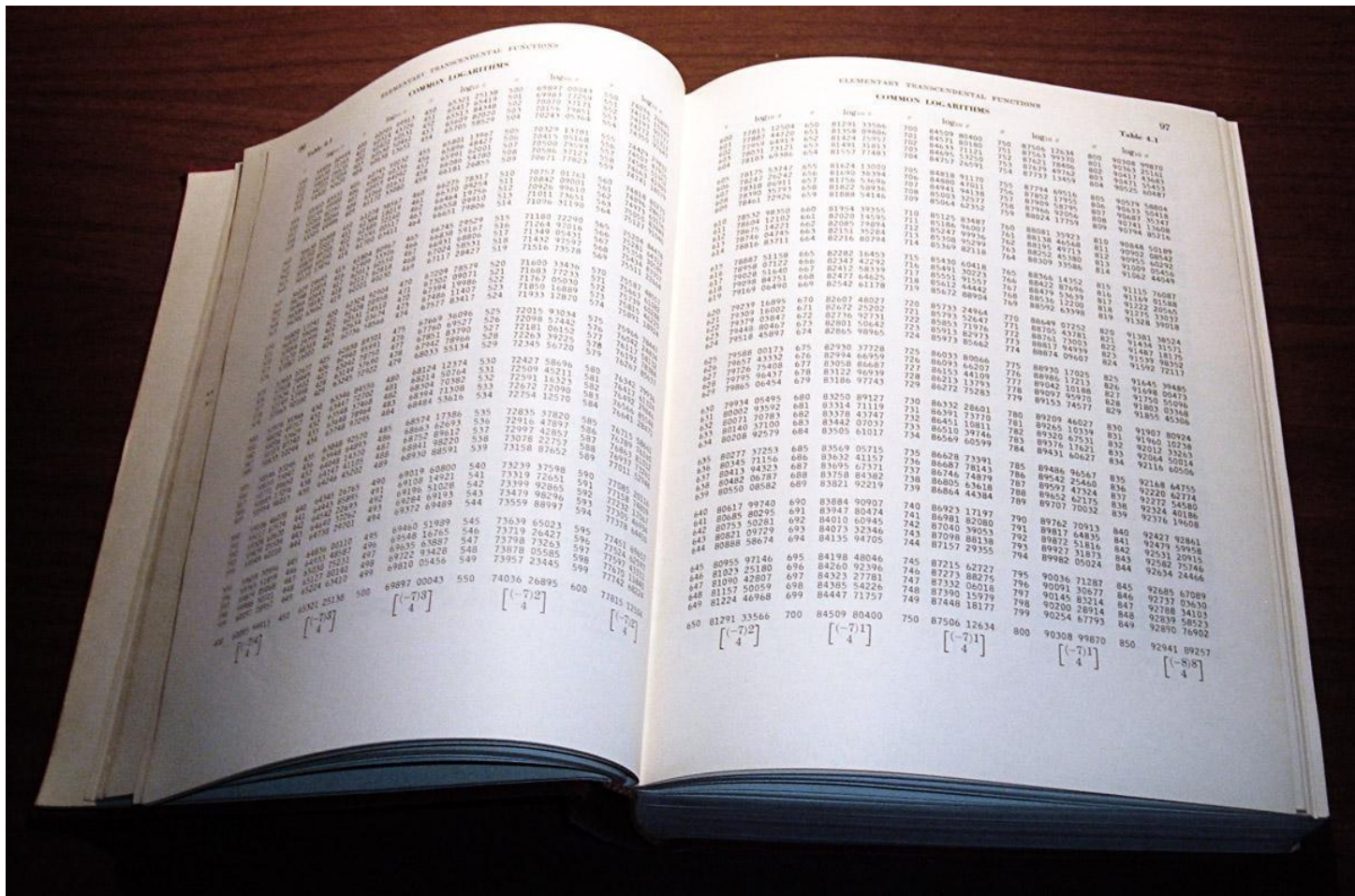
# Computadoras humanas 1969

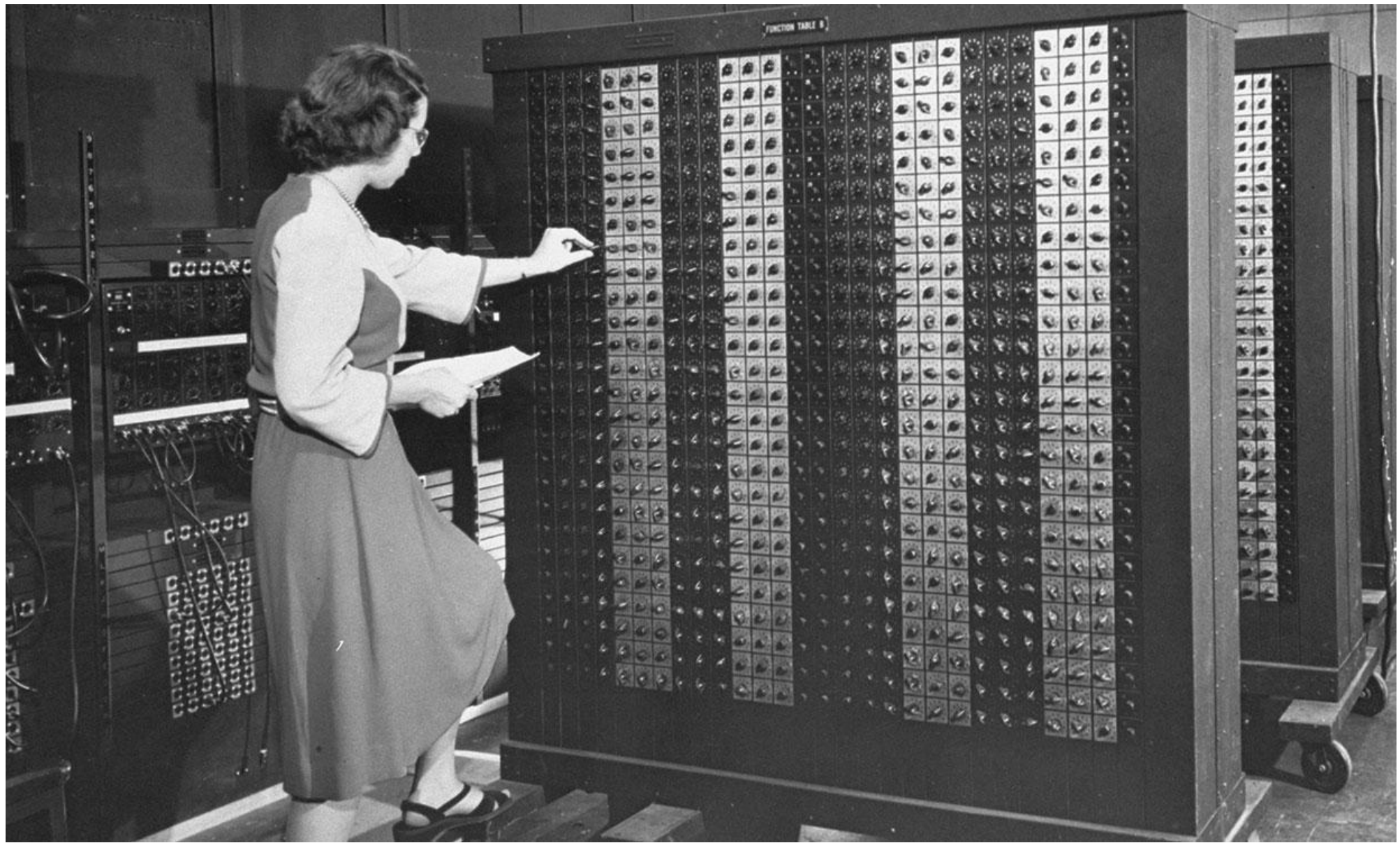


Katherine Johnson, Dorothy Vaughan y Mary Jackson, tres matemáticas afroamericanas que estuvieron a cargo de calcular las trayectorias de la misión Mercury Atlas 6, Apolo 11, Apolo 13, además de analizar el diseño de los combustibles, construir cohetes y desarrollar antenas de comunicación para los astronautas.

# Libros con cálculos

(precomputation table)





Computadores de cuarta  
generación:  
LSI Large-Scale Integration

---

# La era del microprocesador



# La era del microprocesador





# Computadores de quinta generación:



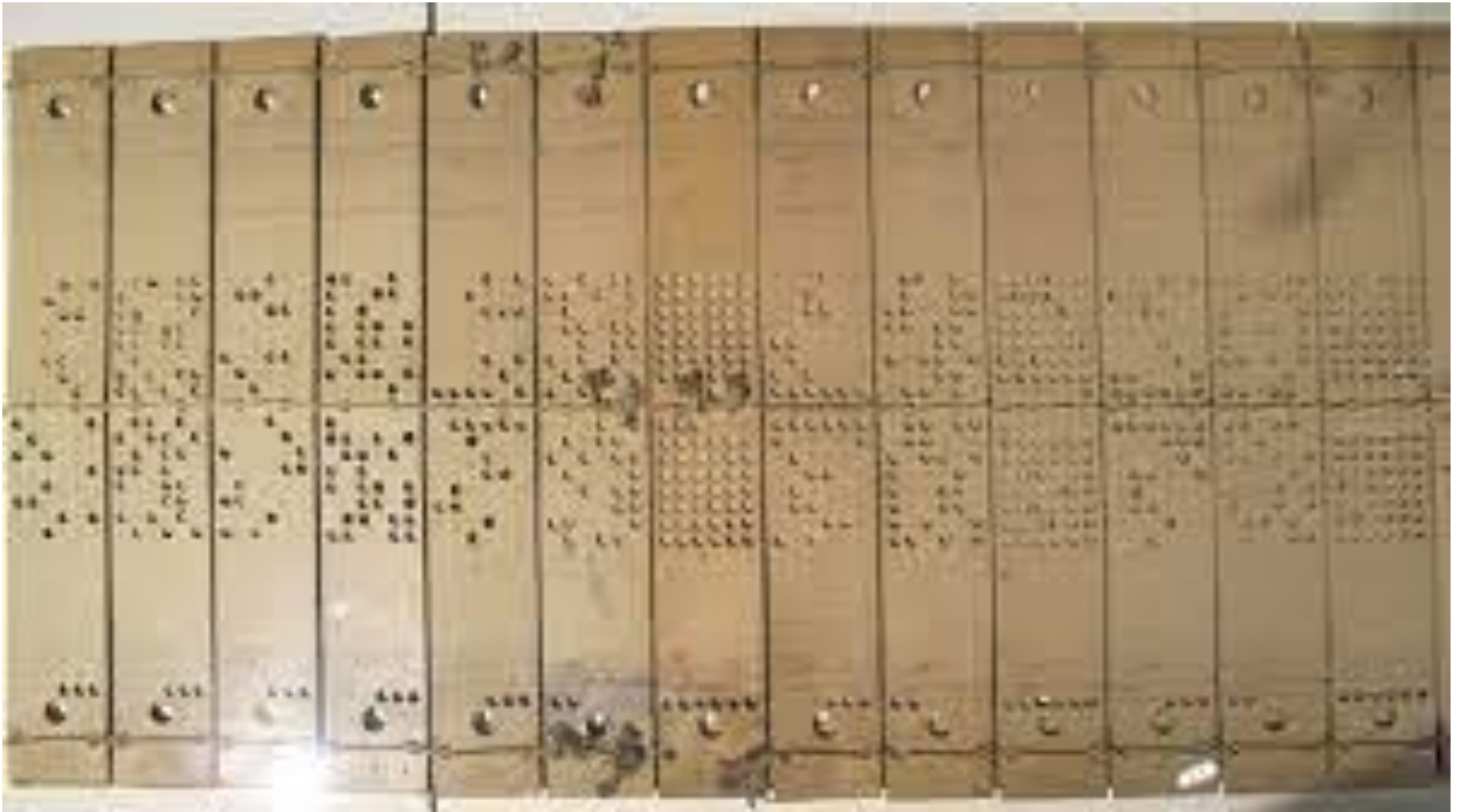
# La era cuántica



# Sobre los programas



# Tarjetas perforadas





# Machine code

## *Machine Code*

```
10011101000110100000  
01100011010001110110  
10000010111101101110  
11110110001011011000  
10000010011100011011  
10010011000111000000
```

# Primeros lenguajes de programación

0089814E	57	PUSH EDI	
0089814F	6A 00	PUSH 0	
00898151	FF95 57040000	CALL DWORD PTR SS:[EBP+457]	GlobalAlloc
00898157	8938	MOV DWORD PTR DS:[EAX],EDI	
00898159	8D78 04	LEA EDI,[EAX+4]	
0089815C	8D85 77040000	LEA EAX,[EBP+477]	
00898162	57	PUSH EDI	
00898163	50	PUSH EAX	
00898164	E8 46000000	CALL Decompress	
00898169	57	PUSH EDI	
0089816A	E8 EB000000	CALL PrepareImage	
0089816E	33C5	MOV ESI,EAX	
00898171	31C0	XOR EAX,EAX	
00898173	83EF 04	SUB EDI,4	
00898176	8B0F	MOV ECX,DWORD PTR DS:[EDI]	
00898178	57	PUSH EDI	
00898179	F3:AA	REP STOS BYTE PTR ES:[EDI]	
0089817B	5F	POP EDI	
0089817C	57	PUSH EDI	
0089817D	FF95 5B040000	CALL DWORD PTR SS:[EBP+45B]	
00898183	8BBD 67040000	MOV EDI,DWORD PTR SS:[EBP+467]	
00898189	31C0	XOR EAX,EAX	
0089818B	83EF 04	SUB EDI,4	
0089818E	8B0F	MOV ECX,DWORD PTR DS:[EDI]	
00898190	57	PUSH EDI	
00898191	F3:AA	REP STOS BYTE PTR ES:[EDI]	
00898193	5F	POP EDI	
00898194	57	PUSH EDI	
00898195	FF95 5B040000	CALL DWORD PTR SS:[EBP+45B]	
0089819B	85F6	TEST ESI,ESI	
0089819D	74 07	JE SHORT 008981A6	
0089819F	897424 1C	MOV DWORD PTR SS:[ESP+1C],ESI	
008981A3	61	POPAD	
008981A4	FFE0	JMP EAX	
008981A6	61	POPAD	
008981A7	B8 FFFFFFFF	MOV EAX,FFFFFFFF	
008981AC	C2 08	MOV EAX,8	



Fortran 1958



BASIC 1959



BASIC 1969

```
static char* get_boundary(apr_pool_t* p, const char* ctype) {
    char* ret = NULL;
    if (ctype) {
        char* lctype = lccopy(p, ctype);
        char* bdy = strstr(lctype, "boundary");
        if (bdy) {
            char* ptr = strchr(bdy, '=');
            if (ptr) {
                bdy = (char*) ctype + (ptr - lctype) + 1;
                for (ptr = bdy; *ptr; ++ptr)
                    if (*ptr == ';' || isspace(*ptr))
                        *ptr = 0;
                ret = apr_pstrdup(p, bdy);
            }
        }
    }
    return ret;
}
```

BASIC 1970

# Primeros lenguajes de programación



C 1972

```
c:\program files\win prolog 4580\examples\ladderman.pl  
% initialise data, prepare graphics objects, and create the dialog  
:-  
  !load_prolog,  
  !init_window,  
  !w = [w_explicit,w_maximizebox,w_thickframe],  
  !e = [w_child,w_visible,w_toplevel,w_pushbutton],  
  !d = [w_child,w_visible,w_w_oriented],  
  wcreate( dig, 'Travelling Salesman', 10, 10, 520, 460, !w ),  
  wcreate( (dig,3), button, '<<Recursive', 420, 8, 80, 22, !e ),  
  wcreate( (dig,4), button, '<>Recursive', 420, 38, 80, 22, !e ),  
  wcreate( (dig,5), button, 'Close', 420, 68, 80, 22, !e ),  
  wcreate( (dig,5), button, 'Quit', 420, 98, 80, 22, !e ),  
  wcreate( (dig,5), static, ' ', 10, 435, 460, 25, !d ),  
  wcreate( (dig,5), graphic, ' ', 10, 10, 400, 400, !d ),  
  set_buttons( 0, 0, 0, 1 ),  
  show_graphics,  
  window_handler( dig, window_handler ),  
  call_dialog( dig, _ ),  
  !end_prolog.
```

Prolog 1972



SQL 1978



HTML 1991



Python 1991



C++ 1980



Java 1995



JS 1995



PHP 1995



C# 2001



# Primeros lenguajes de programación

